An Adaptive Self-Regular Proximity Based Large-Update IPM for LO

Maziar Salahi[*†] and Tamás Terlaky[†]

Revised version,

November 28, 2003

Abstract

Primal-Dual Interior-Point Methods (IPMs) have shown their power in solving large classes of optimization problems. However, there is still a gap between the practical behavior of these algorithms and their theoretical worst-case complexity results with respect to the update strategies of the duality gap parameter in the algorithm. The so-called small-update IPMs enjoy the best known theoretical worst-case iteration bound but their performance in computational practice is poor, while the so-called large-update IPMs have superior practical performance but with relatively weaker theoretical results. This gap was reduced by Peng, Roos, and Terlaky [7] who introduced a new family of Self-Regular (SR) proximity functions based IPMs. In this paper, by restricting us to linear optimization, we propose an adaptive single step large-update IPM for a class of SR-proximities. At each step our algorithm chooses the target value adaptively and the update is a large update. The new algorithm does not do any inner iterations, unlike other large update methods. An $\mathcal{O}\left(qn^{\frac{q+1}{2q}}\log\frac{n}{\varepsilon}\right)$ worst-case iteration bound of the algorithm is established, where q is the barrier degree of the SR-proximity. For a special choice of q the best complexity for large-update IPMs is established.

Keywords: Linear Optimization, Primal-Dual Interior-Point Method, Self-Regular Proximity Function, Polynomial Complexity.

1 Introduction

Since Karmarkar's seminal paper [4], many researchers have proposed and analyzed various interior-point methods (IPMs) for solving large classes of optimization problems. For a survey of these results we refer to the recent books [12, 15, 16] on the subject. In this paper, we deal with primal-dual IPMs for solving the following standard Linear Optimization (LO) problem:

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

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

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

[*]Department of Mathematical Sciences, Sharif University of Technology, P.O. Box 11365-9415, Tehran, Iran, email: salahi@mehr.sharif.edu

[†]Advanced Optimization Lab, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada, L8S 4K1. email: terlaky@mcmaster.ca, msalahi@optlab.cas.mcmaster.ca

Without loss of generality, 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 $Ax^0 = b$, $x^0 > 0$, and $A^T y^0 + s^0 = c$, $s^0 > 0$. For detailed discussions on the IPC and some other properties mentioned in the sequel the reader is referred to the literature, e.g., [12]. If the IPC holds, then finding optimal solutions of (P) and (D) is equivalent to solving the following system:

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

where $xs \in R^n$ denotes the componentwise product of the vectors x and s. The basic idea of primal-dual IPMs is to replace the third equation in (1) by the parameterized equation $xs = \mu e$ with $e = (1, \ldots, 1)^T$. 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 unique 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), respectively [5, 14]. 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) [12].

Primal-dual IPMs follow the central path $(x(\mu), y(\mu), s(\mu))$ approximately, and approach the optimal solution sets of the underlying LO problems as μ goes to zero. Let us briefly indicate how this works [12, 15, 16]. Without loss of generality we assume that the present point (x, y, s) is in a certain neighborhood of the central path for some positive μ. We first update μ to $\mu_+ := (1-\theta)\mu$, for some $\theta \in (0, 1)$. Note that θ can be constant, it may depend on the dimension n, or may also be dependent on the actual iterate. Then we solve the Newton system

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

to obtain the unique search direction $(\Delta x, \Delta y, \Delta s)$. By taking a step along the search direction, where the step size is chosen so that the new iterate has significantly smaller proximity value w.r.t. $(x(\mu_+), y(\mu_+), s(\mu_+))$. We repeat this procedure until the present iterate is 'close enough' to $(x(\mu_+), y(\mu_+), s(\mu_+))$, and thus we can set $\mu := \mu_+$. Then μ is reduced again by the factor $1 - \theta$ and we apply Newton's method again with targeting the new μ_+-center, and so on. This process is repeated until μ is small enough.

Note that our primary goal is to reduce the duality gap as fast as possible. This is done by subsequently decreasing the parameter μ with a ratio $1 - \theta$ at each outer iteration of the algorithm. As a consequence, the choice of the parameter θ has an important role in the design and analysis of IPMs. Usually, if θ is a constant independent of n, the dimension of the problem, for instance $\theta = \frac{1}{2}$, then we call the algorithm a large-update (or long-step) method. If θ depends on the problem dimension, such as $\theta = \frac{1}{\sqrt{n}}$, then the algorithm is called a small-update (or short-step) method. Theoretical worst-case complexity results with respect to small-update methods have so far the best $\mathcal{O}(\sqrt{n} \log \frac{n}{\varepsilon})$ iteration bound, while large-update methods based on the classical Newton direction have a worse $\mathcal{O}(n \log \frac{n}{\varepsilon})$ iteration complexity. In spite of weaker theoretical complexity results, large-update IPMs perform much better in practice than small-update methods [1, 2, 6]. To resolve this discrepancy, Peng, Roos, and Terlaky have recently introduced recently the family of SR-proximity functions for IPMs and for a special member of the SR family established an $\mathcal{O}(\sqrt{n} \log n \log \frac{n}{\varepsilon})$ iteration bound for large-update IPMs [7].

In this paper, we focus on the approach suggested in [7, 8] where new IPMs are indrouced based on SR proximity functions. The basic idea in [7, 8] is to employ a specific class of SR-proximity functions and consequently its induced search directions in the algorithm. To describe briefly the algorithm in [7, 8], we need to introduce some notations and define SR functions. For any strictly feasible primal-dual pair (x, s) and $\mu > 0$, we define the vectors

$$v = \sqrt{\frac{xs}{\mu}}, \quad \text{and} \quad v^{-1} = \sqrt{\frac{\mu e}{xs}},$$

whose i^{th} components are $\sqrt{\frac{x_i s_i}{\mu}}$ and $\sqrt{\frac{\mu}{x_i s_i}}$, respectively. Observe that $v = e$ holds if and only if $x = x(\mu)$ and $s = s(\mu)$, i.e., the vectors x and s are on the central path. For ease of reference, we also scale the search directions Δx and Δs in the scaled v-space as

$$d_x := \frac{v\Delta x}{x} \quad \text{and} \quad d_s := \frac{v\Delta s}{s}. \tag{4}$$

Using this notation the Newton system (3) can be written as

$$\begin{aligned} \bar{A} d_x &= 0, \\ \bar{A}^T \Delta y + d_s &= 0, \\ d_x + d_s &= v^{-1} - v, \end{aligned} \tag{5}$$

where $\bar{A} = \frac{1}{\mu} A V^{-1} X$ with $X = \operatorname{diag}(x)$ and $V^{-1} = \operatorname{diag}(v^{-1})$. Observe that the right-hand-side of the third equation in (5) is the projected steepest descent direction for the primal-dual logarithmic barrier function

$$\frac{e^T v^2 - n}{2} - \sum_{i=1}^{n} \log v_i.$$

The new SR search directions introduced in [7, 8] are slight modifications of the standard Newton direction. They are defined as solutions of the following system:

$$\begin{aligned} \bar{A} d_x &= 0, \\ \bar{A}^T \Delta y + d_s &= 0, \\ d_x + d_s &= v^{-q} - v, \end{aligned} \tag{6}$$

where $q \geq 1$ is a parameter.

This search direction is defined also as the projected steepest descent direction of a proximity function in the scaled space, where the proximity function is a SR proximity function introduced in [7, 8]. SR proximity functions are induced by one dimensional self-regular kernel functions that are defined as follows.

Definition 1.1 *A function $\psi(t) \in \mathcal{C}^2 : (0, \infty) \to R$ is self-regular if it satisfies the following conditions:*

SR.1 $\psi(t)$ *is strictly convex with respect to* $t > 0$ *and vanishes at its global minimal point* $t = 1$, *i.e.,* $\psi(1) = \psi'(1) = 0$. *Further, there exist positive constants* $\nu_2 \geq \nu_1 > 0$ *and* $p \geq 1,\ q \geq 1$ *such that*

$$\nu_1(t^{p-1} + t^{-1-q}) \leq \psi''(t) \leq \nu_2(t^{p-1} + t^{-1-q}), \quad \forall t \in (0, \infty); \tag{7}$$

SR.2 *For any* $t_1, t_2 > 0$,

$$\psi(t_1^r t_2^{1-r}) \leq r\psi(t_1) + (1-r)\psi(t_2), \quad \forall r \in [0, 1]. \tag{8}$$

If $\psi(t)$ is SR, then parameter q is called the *barrier degree* and p the *growth degree* of the SR function $\psi(t)$.

There are two popular families of SR functions. The first family is given by

$$\Upsilon_{p,q}(t) = \frac{t^{p+1} - 1}{p(p+1)} + \frac{t^{1-q} - 1}{q(q-1)} + \frac{p-q}{pq}(t-1), \quad p, q \geq 1, \tag{9}$$

with $\nu_1 = \nu_2 = 1$. The second family is defined as

$$\Gamma_{p,q}(t) = \frac{t^{p+1} - 1}{p+1} + \frac{t^{1-q} - 1}{q-1}, \quad p \geq 1,\ q > 1, \tag{10}$$

with $\nu_1 = 1$ and $\nu_2 = q$.

Let[1] $v \in R^n_{++}$. Then an SR-proximity $\Psi : R^n_{++} \to R_+$ measures the discrepancy between v and $e = (1, 1, \cdots, 1)^T$, and is defined as $\Psi(v) = \sum_{i=1}^{n} \psi(v_i)$, where $\psi(t)$ is a univariate SR function, called the kernel function of the SR-proximity. Using the notation introduced by (4), the authors in [7, 8] have designed the Newton system (11) in the scaled space that gives the projected steepest descent direction w.r.t. a SR-proximity. This direction is the unique solution of the new system,

$$\begin{aligned} \bar{A}d_x &= 0, \\ \bar{A}^T \Delta y + d_s &= 0, \\ d_x + d_s &= -\nabla\Psi(v). \end{aligned} \tag{11}$$

One can easily verify that the search direction given by (6) is the projected steepest descent direction w.r.t. the $\Gamma_{1,q}(t)$ family of SR functions.

Using the notation $\Phi(x, s, \mu) := \Psi(v)$, the algorithm in [8] is outlined as Algorithm 1.

[1] Throughout R^n_{++} denotes the positive orthant and R^n_+ denotes the nonnegative orthant in R^n.

Algorithm 1: A Large-update SR-IPM

Input:
A proximity parameter τ;
an accuracy parameter $\varepsilon > 0$;
a fixed update parameter θ, $0 < \theta < 1$;
(x^0, s^0) and $\mu^0 = 1$ such that $\Phi(x^0, s^0, \mu^0) \leq \tau n$.
begin
$x := x^0$; $s := s^0$; $\mu := \mu^0$;
while $n\mu \geq \varepsilon$ **do**
begin
$\mu := (1-\theta)\mu$;
while $\Phi(x, s, \mu) \geq n\tau$ **do**
begin
Solve the system (11) for $\Delta x, \Delta y, \Delta s$;
determine a step size α;
$x := x + \alpha\Delta x$;
$s := s + \alpha\Delta s$;
$y := y + \alpha\Delta y$.
end
end
end

It should be noted that in the algorithmic scheme presented in Algorithm 1, μ is treated as an independent parameter, not related to the present duality gap. This is different from what is implemented in most IPM solvers [1, 18], where μ is always chosen as a fraction of $\frac{x^T s}{n}$, i.e., μ is directly linked to the current duality gap $x^T s$. Further, when search directions induced by SR proximity functions are used, it is not so easy to estimate how the duality gap changes along the Newton step [10]. It is possible that the duality gap increases after a step in the inner process of the algorithm, even if the target μ value is smaller than $\frac{x^T s}{n}$. We will elaborate on this issue in Section 2. We also point out that in IPM solvers, one always employ an adaptive large-update at each iteration and only one or very few inner iterations are needed to recenter. Although such a heuristic is very efficient in practice, there is no theoretical explanation for this phenomenon. We note that the proximity function $\Psi(v)$ has a dominant role in Algorithm 1, and the role of the proximity and its interplay with the μ updating strategy needs further investigation. In this paper we focus on one of the specific classes of SR functions with kernel functions

$$\psi(t) := \Gamma_{1q}(t) = \left(\frac{t^2-1}{2} + \frac{t^{1-q}-1}{q-1}\right),$$

where $p = 1,\ q > 1$. At each iteration our algorithm chooses the target μ value adaptively and does not do any inner iterations.

The paper is organized as follows. First, in Section 2, we explore the role of the parameter μ w.r.t. one of the SR classes of proximity functions with kernel function $\psi(t)$. Then we discuss how the duality gap changes along the search direction. In Section 3, we propose a family of new IPMs and establish its complexity. Finally, we close this paper by some concluding remarks in Section 4.

A few words about our notation. Throughout the paper $\|\cdot\|$ denotes the 2-norm of vectors. We denote by $\mathcal{I}$ the index set $\mathcal{I} = \{1, 2, \ldots, n\}$ and $x^{-T}s^{-1} = \sum_{i\in\mathcal{I}} x_i^{-1}s_i^{-1}$. For any $x = (x_1, x_2, \ldots, x_n)^T \in \Re^n$, $x_{\min} = \min\{x_1, x_2, \ldots, x_n\}$ is the smallest component of x and $x_{\max}$ is defined similarly to denote the largest component. We also denote $\sigma = \|v - v^{-q}\|$.

2 Properties of the Proximity Function

In this section we investigate some properties of the family of SR proximity functions

$$\Phi_q(x, s, \mu) := \Psi_q(v) = \frac{e^T v^2 - n}{2} + \frac{e^T v^{1-q} - n}{q-1} \tag{12}$$

with respect to the argument μ, where (x, s) are fixed and $q > 1$. This family of SR-proximity functions is induced by the kernel functions $\Gamma_{1q}(t)$ given by (10). Let the current iterate be (x, s) and let $\mu_g := \frac{x^T s}{n}$ denote the parameter value associated with the current duality gap. Next, we consider the behavior of the function $\Phi_q(x, s, \mu)$ w.r.t. μ. The following result gives the global minimum of the proximity function.

Proposition 2.1 *For any fixed $(x, s) > 0$, the proximity function $\Phi_q(x, s, \mu)$, as a function of μ, has a global minimizer at*

$$\mu^* = \left(\frac{x^T s}{(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}}}\right)^{\frac{2}{q+1}}.$$

Proof: It can be easily proved that for any $q \geq 3$ the proximity function is strictly convex and for any $1 < q < 3$ the proximity function is quasi-convex. Using the strict and quasi-convexity of the proximity function and the optimality conditions, one can easily show that μ^* is the global minimizer of the proximity function. □

One can easily prove the following proposition.

Proposition 2.2 *The proximity function $\Phi_q(x, s, \mu)$ is a decreasing function w.r.t. μ when $\mu \leq \mu^*$, and it is an increasing function of μ if $\mu > \mu^*$.*

In [10], for the special case $q = 3$, the authors used the property $\Phi_q(x, s, \mu_g) = \Phi_q(x, s, \mu_h)$, where μ_h is the harmonic mean of the components of the vector xs. In this paper, for the general case, instead of μ_h we use

$$\hat{\mu}_h = \left(\frac{n}{(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}}}\right)^{\frac{2}{q-1}},$$

as a generalized harmonic mean. Unfortunately, in general the relation $\Phi_q(x, s, \mu_g) = \Phi_q(x, s, \hat{\mu}_h)$ does not hold.

The following lemma [13] plays an important role in the definition of the SR neighborhood.

Lemma 2.3 *Let $\tau \geq 2$ be a constant. Then, the following statements are equivalent.*
1) $\frac{\mu_g}{\hat{\mu}_h} \leq \tau$,
2) $\Phi_q\left(x, s, \frac{\mu_g}{\tau}\right) \leq \frac{(\tau-1)n}{2}$,
3) $\Phi_q(x, s, \mu_g) \leq \frac{\left(\tau^{\frac{q-1}{2}} - 1\right)n}{q-1}$.

Proof: By the assumption of the lemma we can write $\mu_g = \bar{\tau}\hat{\mu}_h$ for some $\bar{\tau} \leq \tau$. It follows that

$$\begin{aligned}\Phi_q\left(x, s, \frac{\mu_g}{\tau}\right) &= \frac{(\tau-1)n}{2} + \frac{(\frac{\mu_g}{\tau})^{\frac{q-1}{2}}(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} - n}{q-1} \\ &\leq \frac{(\tau-1)n}{2} + \frac{\left((\frac{\bar{\tau}}{\tau})^{\frac{q-1}{2}} - 1\right)n}{q-1} \leq \frac{(\tau-1)n}{2}.\end{aligned}$$

This completes the proof that 1) implies 2). One can easily prove that 2) implies 1). For 1) implies 3) we have

$$\Phi_q(x, s, \mu_g) = \frac{\mu_g^{\frac{q-1}{2}}(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} - n}{q-1} \leq \frac{\left(\tau^{\frac{q-1}{2}} - 1\right)n}{q-1}.$$

One can prove analogously that 3) implies 1) and thus the proof is complete. □

Using Lemma 2.3, an SR neighborhood of the central path is defined by

$$\mathcal{N} := \{(x, y, s) | (x, s) > 0,\ Ax = b,\ A^T y + s = c,\ \Phi_q(x, s, \mu_g) \leq \eta(n, \tau)\}, \tag{13}$$

where $\eta(n,\tau)\} = \frac{\left(\tau^{\frac{q-1}{2}}-1\right)n}{q-1}$ with $\tau \geq 2$.

Now we proceed to discuss the properties of SR search directions for different updates of μ. Note that, due to the specific choice of the kernel function $\psi(t)$, we can rewrite system (11) in the original space as:

$$\begin{aligned} A\Delta x &= 0, \\ A^T\Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu^{\frac{q+1}{2}}(x^{\frac{1-q}{2}})s^{\frac{1-q}{2}} - xs. \end{aligned} \tag{14}$$

Let us denote the solution of system (14) by $(\Delta x(\mu), \Delta y(\mu), \Delta s(\mu))$. The following two lemmas discuss the change of the duality gap along the search direction $(\Delta x(\mu), \Delta y(\mu), \Delta s(\mu))$ for $\mu = \mu^*$, and for $\mu = \hat{\mu}_h$.

Lemma 2.4 *Let $(\Delta x(\mu^*), \Delta y(\mu^*), \Delta s(\mu^*))$ be the solution of system (14) with $\mu = \mu^*$. Then the relation*

$$x^T\Delta s(\mu^*) + s^T\Delta x(\mu^*) = 0$$

holds.

Proof: The lemma follows immediately from the choice of μ^*. □

Corollary 2.5 *If the targeted μ parameter is μ^*, then the duality gap will not change for any feasible step size α, i.e.,*

$$(x + \alpha\Delta x(\mu^*))^T(s + \alpha\Delta s(\mu^*)) = x^T s.$$

Lemma 2.6 *Let $(\Delta x(\hat{\mu}_h), \Delta y(\hat{\mu}_h), \Delta s(\hat{\mu}_h))$ be the solution of system (14) with $\mu = \hat{\mu}_h$. Then the relation*

$$x^T\Delta s(\hat{\mu}_h) + s^T\Delta x(\hat{\mu}_h) = n\hat{\mu}_h - x^T s$$

holds.

Proof: Using the third equation in (14) with $\mu = \hat{\mu}_h$ we have

$$s^T \Delta x + x^T \Delta s = \hat{\mu}_h^{\frac{q+1}{2}} (x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} - x^T s = n\hat{\mu}_h - x^T s.$$

The proof is completed. □

Corollary 2.7 *If the targeted μ parameter is $\hat{\mu}_h$, then the search direction based on our specific SR-proximity function and the standard Newton direction will predict the change of the duality gap in the same way.*

Since we are working with a large neighborhood, we define $\frac{(\tau-1)n}{2}$ as the maximum allowed value of the proximity function w.r.t. μ_t the target μ value. One can see that $\Phi_q(x, s, \mu_t) = \frac{(\tau-1)n}{2}$ if and only if μ_t satisfies the equation

$$2(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} \mu^{\frac{q+1}{2}} - (2n + \tau(q-1)n)\, \mu + (q-1)x^T s = 0. \tag{15}$$

This equation has two positive roots. One is less than or equal to μ^* and the other is larger than equal to μ^*. In the algorithm we will use the smaller positive root μ_t as the target value at each iteration. One can easily prove that $\mu_t \leq \hat{\mu}_h$, holds when $\mu_g \leq \tau\hat{\mu}_h$ with $\mu_t = \hat{\mu}_h$ if and only if $\mu_g = \tau\hat{\mu}_h$.

Lemma 2.8 *Let μ_t be defined by equation (15). Then the inequality*

$$\hat{\mu}_h \leq 2\tau\mu_t$$

holds.

Proof: The function in (15) is a convex function w.r.t. μ. If we replace μ by $\frac{\hat{\mu}_h}{2\tau}$ in that function and use the fact that $\mu_g \geq \hat{\mu}_h$ then the function value is larger than

$$\frac{2}{(2\tau)^{\frac{q+1}{2}}} - \frac{1}{\tau} + \frac{q-1}{2}. \tag{16}$$

It is sufficient to prove that (16) is nonnegative. Since $\tau \geq 2$ is a positive constant, one can easily show that (16) is a strictly increasing function of q for $q \geq 1$. For $q = 1$ the value of this function is zero and thus for $q > 1$ the function is positive. This completes the proof of the lemma. □

When μ_t is the target value, analogue to Lemma 2.4, we can get the following result.

Lemma 2.9 *Let $(\Delta x(\mu_t), \Delta y(\mu_t), \Delta s(\mu_t))$ be the solution of system (14), where $\mu = \mu_t$ is defined by equation (15). Then the relation*

$$x^T \Delta s(\mu_t) + s^T \Delta x(\mu_t) = \mu_t^{\frac{q+1}{2}} (x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} - x^T s$$

holds.

Recall that in traditional IPMs, based on the standard Newton direction, we need to solve system (3) at each iteration. In this case, if we set the target $\mu_+ = \mu_t$, then the solution of system (3) will satisfy

$$x^T \Delta s + s^T \Delta x = \mu_t^{\frac{q+1}{2}} (x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} - x^T s.$$

This implies that if the targeted parameter is μ_t, then

$$(x + \alpha\Delta x(\mu_t))^T(s + \alpha\Delta s(\mu_t)) = (x + \alpha\Delta x)^T(s + \alpha\Delta s) = x^T s \left(1 - \alpha + \frac{\mu_t^{\frac{q+1}{2}}\alpha}{\mu_g \hat{\mu}_h^{\frac{q-1}{2}}}\right). \quad (17)$$

Remark 2.10 *If $\mu_t \sim \hat{\mu}_h$, then (17) implies that the search direction based on our specific SR-proximity function and the standard Newton direction will predict the change of the duality gap almost in the same way (If $\mu_t = \hat{\mu}_h$, see Corollary 2.7). But if $\mu_t \ll \hat{\mu}_h$, then the ratio $\frac{\mu_t^{\frac{q+1}{2}}}{\mu_g \hat{\mu}_h^{\frac{q-1}{2}}}$ is very small and for the SR search direction the duality gap reduction is much larger than it would be when using the standard Newton direction.*

3 An Adaptive Single Step Large-Update IPM

In this section we consider a specific variant of Algorithm 1. This variant is more flexible in updating μ and closer to what is implemented in IPM solvers than Algorithm 1, mainly because we use large-update at each iteration and we do not employ any inner iterations to recenter. In our algorithm we use the family of SR-proximity functions given by (12).

To motivate our design, let us start by considering an implementational issue in the algorithm. Suppose that the present point (x, s) is in a certain neighborhood of the central path. Then we solve the linear system (14) for the search direction $(\Delta x, \Delta y, \Delta s)$, from which we can estimate the maximal feasible step size $\alpha_{\max}$. A popular heuristic for choosing the step size in IPM solvers is to use a damping factor to $\alpha_{\max}$, say $0.995\alpha_{\max}$, as a step size. Of course, if the value of the corresponding proximity function is too large for this step size, then we can reduce the step size appropriately so that the value of the proximity function at the new iterate is below a prescribed threshold. Note that it is also possible that the proximity function has a relatively small value for the step size $0.995\alpha_{\max}$. In this case, theoretically we can still increase the step size so that the value of the proximity function remains below a prescribed bound. However, in practice this might not be a good idea. Since the step size is already quite close to the maximal feasible step size, even a small increase of the step size may cause numerical problems or drive the iterate too close to the boundary of the feasible region. In this situation, it is better to use the default value $0.995\alpha_{\max}$ as the step size. Note that after such a step, we have an iterate (x, s) with a small proximity function value. Motivated by the above observation, we change the procedure of Algorithm 1. Further, we utilize a parameter $\tau \geq 2$ to keep control on the distance of the iterate to the central path. Recall the results of Lemma 2.3 that quantify the relation between μ_g, $\hat{\mu}_h$ and the corresponding proximity values. As specified in (13), the definition of the neighborhood $\mathcal{N}$, we force the value of the proximity function (12) to satisfy the relation:

$$\Phi_q(x, s, \mu_g) \leq \frac{(\tau^{\frac{q-1}{2}} - 1)n}{q - 1} = \eta(n, \tau). \quad (18)$$

In our algorithm, regardless of the iterate is close to, or is far away from the central path, we always make a large update of the central path parameter μ.

Remark 3.1 *The assumption $\tau \geq 2$ is necessary to keep $\sigma = \|v - v^{-q}\| \geq 1$, what is a helpful property in the complexity analysis of the algorithm.*

For simplicity we use the notation $x(\alpha) := x + \alpha\Delta x,\ y(\alpha) := y + \alpha\Delta y$ and $s(\alpha) := s + \alpha\Delta s$. Correspondingly we also define

$$\mu_g(\alpha) = \frac{x(\alpha)^T s(\alpha)}{n} \quad \text{and} \quad \mu^*(\alpha) = \left(\frac{x(\alpha)^T s(\alpha)}{(x(\alpha)^{\frac{1-q}{2}})^T s(\alpha)^{\frac{1-q}{2}}} \right)^{\frac{2}{q+1}}.$$

At each step, we stipulate that the step size should be chosen such that the proximity function $\Phi_q(x(\alpha), s(\alpha), \mu_+)$ has a sufficient decrease while the proximity function w.r.t. $\mu_g(\alpha)$ still satisfies (18) at the new iterate. The new algorithm is presented in the scheme Algorithm 2.

Algorithm 2: An Adaptive Large-Update SR-IPM

Input:
  A proximity parameter $\tau \geq 2$;
  an accuracy parameter $\epsilon > 0$;
  $(x, s) = (x^0, s^0)$ such that $\frac{\mu_g}{\hat{\mu}_h} \leq \tau$.
begin
  while $x^T s \geq \epsilon$ **do**
  begin
    $\mu := \mu_t$ computed from (15);
    Solve system (14) for $\Delta x, \Delta y, \Delta s$.
    begin
      Determine a step size α such that
      $\Phi_q(x(\alpha), s(\alpha), \mu_t) \leq \Phi_q(x, s, \mu_t) - \frac{2^{\frac{q-1}{2q}} \Phi_q(x,s,\mu_t)^{\frac{q-1}{2q}}}{24q}$,
      and $\mu_g(\alpha) \leq \tau\hat{\mu}_h(\alpha)$;
      $x = x(\alpha);\ y = y(\alpha);\ s = s(\alpha)$.
    end
  end
end

Remark 3.2 *At each iteration, the step size α has to be chosen such that the proximity function $\Phi(x, s, \mu)$ decreases sufficiently. In the sequel we present a default value for α, based on the actual value of the proximity function.*

Remark 3.3 *For a practical implementation, it would be advantageous to choose the step size α so that it minimizes the proximity function $\Phi_q(x(\alpha), s(\alpha), \mu_t)$, while the constraint $\mu_g(\alpha) \leq \tau\hat{\mu}_h(\alpha)$ is satisfied. However, this would require an exact line search. Instead of this, in the analysis of Algorithm 2 we solve the line search problem approximately so that the value of the function $\Phi_q(x(\alpha), s(\alpha), \mu_t)$ is decreased sufficiently. Theorem 3.5 gives a default value for such a step size.*

We proceed to analyzing the complexity of the algorithm. The key element of the analysis is to estimate the value of the step size α used in Algorithm 2 that imply sufficient reduction of μ_g. For this we need to explore the changing behavior of the functions $\Phi_q(x(\alpha), s(\alpha), \mu_g(\alpha))$ and $\Phi_q(x(\alpha), s(\alpha), \mu_t)$. To simplify the analysis, we use the notation v, d_x, d_s for the system (11) when $\mu = \mu_t$ is chosen as described in Algorithm 2. First we give a lower bound for the maximal feasible step size and the decreasing behavior of the proximity function $\Phi_q(x(\alpha), s(\alpha), \mu_t)$.

Lemma 3.4 *Let $(\Delta x, \Delta y, \Delta s)$ be the solution of system (14) where $\mu = \mu_t$ is defined by equation (15). Then the maximal feasible step size, $\alpha_{\max}$, satisfies*

$$\alpha_{\max} \geq \bar{\alpha} = \frac{1}{\sigma(1+\sigma)^{\frac{1}{q}}}.$$

Proof: We know

$$v(\alpha_{\max}) = (v + \alpha_{\max} d_x)^{\frac{1}{2}}(v + \alpha_{\max} d_s)^{\frac{1}{2}} = v(e + \alpha_{\max} v^{-1} d_x)^{\frac{1}{2}}(e + \alpha_{\max} v^{-1} d_s)^{\frac{1}{2}},$$

that is nonnegative if

$$e + \alpha_{\max} v^{-1} d_x \geq 0, \text{ and } \quad e + \alpha_{\max} v^{-1} d_x \geq 0.$$

These inequalities imply

$$\alpha_{\max} \geq \frac{1}{\|(v^{-1}d_x, v^{-1}d_s)\|}.$$

We also know that

$$\left\|(v^{-1}d_x, v^{-1}d_s)\right\| \leq \frac{\|(d_x, d_s)\|}{v_{\min}} \leq \sigma(1+\sigma)^{\frac{1}{q}},$$

where the last inequality follows from Proposition 3.11 of [7]. This completes the proof of the lemma. □

Theorem 3.5 *Let $(\Delta x, \Delta y, \Delta s)$ be the solution of system (14), where $\mu = \mu_t$ is defined by equation (15). Then for any step size $\alpha \leq \alpha^* = \frac{\bar{\alpha}}{3q}$, the relation*

$$\Phi_q(x(\alpha), s(\alpha), \mu_t) \leq \Phi_q(x, s, \mu_t) - \frac{2^{\frac{q-1}{2q}} \Phi_q(x, s, \mu)^{\frac{q-1}{q}}}{24q}$$

holds.

Proof: Let

$$\begin{aligned} h(\alpha) :&= \Phi_q\left(x(\alpha), s(\alpha), \mu_t\right) - \Phi_q(x, s, \mu_t) && (19)\\ &= \frac{\|v(\alpha)\|^2 - n}{2} + \frac{\left\|v(\alpha)^{\frac{q-1}{2}}\right\|^2 - n}{q-1} - \frac{\|v\|^2 - n}{2} - \frac{\left\|v^{\frac{q-1}{2}}\right\|^2 - n}{q-1}, \end{aligned}$$

where $v(\alpha) = \sqrt{\frac{x(\alpha)s(\alpha)}{\mu_t}} = (v + \alpha d_x)^{\frac{1}{2}}(v + \alpha d_s)^{\frac{1}{2}}$. Using condition SR.2 of Definition 1.1 we have

$$\begin{aligned} h(\alpha) &\leq \frac{1}{2} v^T(d_x + d_s)\alpha + \frac{1}{2(q-1)} \sum_{i=1}^{n} (v_i + \alpha(d_x)_i)^{1-q} && (20)\\ &+ \frac{1}{2(q-1)} \sum_{i=1}^{n} (v_i + \alpha(d_s)_i)^{1-q} - \frac{1}{q-1}\left\|v^{\frac{1-q}{2}}\right\|^2 := h_1(\alpha). \end{aligned}$$

It can be easily shown that

$$h_1^{'}(0) = -\frac{\sigma^2}{2}, \quad \text{and} \quad h_1^{''}(\alpha) \leq \frac{q\sigma^2}{2}(v_{\min} - \alpha\sigma)^{-1-q}.$$

Then we have

$$h(\alpha) \leq -\frac{\alpha\sigma^2}{2} + \frac{q\sigma^2}{2}\int_0^{\alpha}\int_0^{\zeta}(v_{\min} - \alpha\sigma)^{-1-q} := h_2(\alpha).$$

Clearly $h_2(\alpha)$ is a convex function and twice differentiable in the interval $[0,\bar{\alpha})$. Let us denote by α_1^* the global minimum of $h_2(\alpha)$ in the interval $[0,\bar{\alpha})$. Then α_1^* is the unique solution of the equation

$$-\sigma^2 + \sigma\left((v_{\min} - \alpha\sigma)^{-q} - (v_{\min})^{-q}\right) = 0.$$

By using Lemma 1.3.1 of [7], it is easy to show that $\alpha_1^* \geq \alpha^* = \frac{1}{3q\sigma(1+\sigma)^{\frac{1}{q}}}$. Then by Lemma 1.3.3 of [7], for any $\alpha \leq \alpha^*$ we have

$$h(\alpha) \leq -\frac{\sigma^{\frac{q-1}{q}}}{24q}.$$

We also have $\sigma^2 \geq 2\Phi_q(x,s,\mu)$ (see Proposition 1.3.5 [7]), that completes the proof. $\square$

We proceed to estimate the proximity function $\Phi_q(x(\alpha), s(\alpha), \mu_g(\alpha))$ or, equivalently, the function $\Phi_q(x(\alpha), s(\alpha), \mu^*(\alpha))$ for a feasible step size when μ_t is used in Algorithm 2 as the targeted parameter. For this it suffices to consider the function $\Phi_q(x(\alpha), s(\alpha), \mu^*)$, because the inequality $\Phi_q(x(\alpha), s(\alpha), \mu^*(\alpha)) \leq \Phi_q(x(\alpha), s(\alpha), \mu^*)$ holds.

Theorem 3.6 *Let $(\Delta x, \Delta y, \Delta s)$ be the solution of system (14) where $\mu = \mu_t$ is defined by equation (15). Then for any step size $\alpha \leq \alpha^*$, the relation*

$$\Phi_q(x(\alpha), s(\alpha), \mu_g(\alpha)) \leq \frac{\left(\tau^{\frac{q-1}{2}} - 1\right)n}{q-1}$$

holds.

Proof: By using Theorem 3.5, we know that any $\alpha \leq \alpha^*$ is strictly feasible. We also know

$$\Phi_q(x(\alpha), s(\alpha), \mu_t(\alpha)) = \frac{(\tau - 1)n}{2}.$$

Using Proposition 2.2 and that $\mu_t(\alpha) \leq \hat{\mu}_h(\alpha)$, for any $\alpha \leq \alpha^*$ we have

$$\Phi_q(x(\alpha), s(\alpha), \hat{\mu}_h(\alpha)) \leq \frac{(\tau - 1)n}{2},$$

that by Lemma 2.3 is equivalent to

$$\Phi_q(x(\alpha), s(\alpha), \mu_g(\alpha)) \leq \frac{\left(\tau^{\frac{q-1}{2}} - 1\right)n}{q-1}.$$

This completes the proof of the theorem. $\square$

To obtain an upper bound for the total number of iterations of the algorithm, we need to estimate the value of the step size α^* or the change of the parameter μ_t before and after an iterate. The following technical lemma is needed for the complexity analysis.

Lemma 3.7 *Let $v_+ = \frac{v}{\sqrt{1-\theta}}$ for some $\theta \in (0,1)$. Then we have:*

$$\Psi_q(v_+) \leq \frac{\Psi_q(v)}{1-\theta} + \frac{n\theta}{2(1-\theta)} + \frac{\theta n}{1-\theta}\left(\frac{1 - (\frac{1}{2\tau})^{\frac{q-1}{2}}}{q-1}\right).$$

Proof: From the definition of the proximity function we have

$$
\begin{aligned}
\Psi_q(v_+) &= \frac{\|v_+\|^2 - n}{2} + \frac{\left\|v_+^{\frac{1-q}{2}}\right\|^2 - n}{q-1} \\
&= \frac{\frac{1}{1-\theta}\|v\|^2 - n}{2} + \frac{(1-\theta)^{\frac{q-1}{2}}\left\|v^{\frac{1-q}{2}}\right\|^2 - n}{q-1} \\
&= \frac{1}{1-\theta}\left(\frac{\|v\|^2 - n}{2} + \frac{\left\|v^{\frac{1-q}{2}}\right\|^2 - n}{q-1}\right) + \frac{n\theta}{2(1-\theta)} + \frac{n\theta}{(q-1)(1-\theta)} \\
&\quad + \left((1-\theta)^{\frac{q-1}{2}} - \frac{1}{1-\theta}\right)\frac{\left\|v^{\frac{1-q}{2}}\right\|^2}{q-1} \qquad (21) \\
&\leq \frac{\Psi_q(v)}{1-\theta} + \frac{n\theta}{2(1-\theta)} + \frac{\theta}{1-\theta}\left(\frac{n - \left\|v^{\frac{1-q}{2}}\right\|^2}{q-1}\right) \\
&\leq \frac{\Psi_q(v)}{1-\theta} + \frac{n\theta}{2(1-\theta)} + \frac{\theta n}{1-\theta}\left(\frac{1-\left(\frac{1}{2\tau}\right)^{\frac{q-1}{2}}}{q-1}\right), \qquad (22)
\end{aligned}
$$

where the last inequality follows from Lemma 2.8. This completes the proof of the lemma. □

By applying Lemma 3.7 to Theorem 3.5, we can prove the following theorem.

Theorem 3.8 *Let $\tau \geq 2$ and $(\Delta x, \Delta y, \Delta s)$ be the solution of system (14) as defined in Algorithm 2, and let α^* be the default step size as defined in Theorem 3.5. Then*

$$\Phi_q\left(x(\alpha^*), s(\alpha^*), (1-\theta)\mu_t\right) \leq \Phi_q(x, s, \mu_t),$$

where

$$\theta = \frac{(\tau-1)^{\frac{q-1}{2q}}}{24q\left(\frac{\tau}{2} + 1 + \frac{\log\tau}{2}\right) n^{\frac{q+1}{2q}}}.$$

Proof: From Lemma 3.7, it can be seen that to prove the theorem it suffices to choose θ satisfying the inequality

$$\Phi_q(x(\alpha^*), s(\alpha^*), \mu_t) + \frac{n\theta}{2} + \frac{n\theta\left(1-(\frac{1}{2\tau})^{\frac{q-1}{2}}\right)}{q-1} \leq (1-\theta)\Phi_q(x, s, \mu_t).$$

Using Theorem 3.5 we conclude that the above inequality will certainly be satisfied if

$$\theta\Phi_q(x, s, \mu_t) + \frac{n\theta}{2} + \frac{n\theta\left(1-(\frac{1}{2\tau})^{\frac{q-1}{2}}\right)}{q-1} \leq \frac{2^{\frac{q-1}{2q}}\,\Phi_q(x, s, \mu_t)^{\frac{q-1}{2q}}}{24q}. \qquad (23)$$

Recalling the fact that $\Phi_q(x, s, \mu_t) = \frac{(\tau-1)n}{2}$, we can rewrite inequality (23) as:

$$\theta\left(\frac{(\tau-1)n}{2} + \frac{n}{2} + \frac{n\left(1-(\frac{1}{2\tau})^{\frac{q-1}{2}}\right)}{q-1}\right) \leq \frac{(\tau-1)^{\frac{q-1}{2q}}\,n^{\frac{q-1}{2q}}}{24q}.$$

This relation implies that if we choose

$$\theta = \frac{(\tau-1)^{\frac{q-1}{2q}}}{24q\left(\frac{\tau}{2}+1+\frac{\log\tau}{2}\right)n^{\frac{q+1}{2q}}},$$

then

$$\Phi_q\left(x(\alpha^*), s(\alpha^*), (1-\theta)\mu_t\right) \ \leq \Phi_q(x, s, \mu_t),$$

that completes the proof. □

Now we can proceed to discuss the complexity of Algorithm 2. By the choice of μ_t we know that the proximity function $\Phi_q(x, s, \mu_t)$ keeps invariant for all the iterates. Let us denote by μ_t^+ the target parameter value after one step. Then we have

$$\Phi_q(x, s, \mu_t) = \Phi_q(x(\alpha^*), s(\alpha^*), \mu_t^+).$$

On the other hand, form Theorem 3.8 we have

$$\Phi_q(x(\alpha^*), s(\alpha^*), (1-\theta)\mu_t) \ \leq \Phi_q(x(\alpha^*), s(\alpha^*), \mu_t^+).$$

Since the proximity function is a convex function w.r.t. μ, we have

$$\mu_t^+ \leq \left(1 - \frac{(\tau-1)^{\frac{q-1}{2q}}n^{\frac{-q-1}{2q}}}{24q\left(\frac{\tau}{2}+1+\frac{\log\tau}{2}\right)}\right)\mu_t. \tag{24}$$

Now we are ready to give the complexity of Algorithm 2.

Theorem 3.9 *Let $\tau \geq 2$. Then after at most*

$$\left\lceil \frac{12q\left(\tau+2+\log\tau\right)n^{\frac{q+1}{2q}}}{(\tau-1)^{\frac{q-1}{2q}}}\log\frac{2n\tau^2}{\epsilon}\right\rceil$$

iterations Algorithm 2 will terminate with a feasible solution satisfying $x^Ts \leq \epsilon$.

Proof: In light of inequality (24) we know that after at most

$$\left\lceil \frac{12q\left(\tau+2+\log\tau\right)n^{\frac{q+1}{2q}}}{(\tau-1)^{\frac{q-1}{2q}}}\log\frac{2n\tau^2}{\epsilon}\right\rceil$$

iterations we have $\mu_t \leq \frac{\epsilon}{2n\tau^2}$. By using (15), we can see that $\mu_g \leq 2\tau^2\mu_t \leq \frac{\epsilon}{n}$, or equivalently $x^Ts \leq \epsilon$. □

The following corollary gives the best complexity for large-update IPMs [7, 8].

Corollary 3.10 *When $q = \log(n)$, Theorem 3.9 provides the following upper bound for the total number of iterations:*

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

4 Concluding Remarks

Some interesting properties of the proximity function induced by the family of kernel functions

$$\psi(t) = \frac{t^2 - 1}{2} + \frac{t^{1-q} - 1}{q - 1}$$

have been explored. In particular, the properties of the proximity function indicate that if the present iterate is far from the central path, then a large-update algorithm appears to be a natural choice for finding a good search direction and keeping control on the value of the proximity function. Furthermore, in some important cases, this self-regularity based search direction can predict the change of the duality gap along the search direction in the same way as the standard Newton direction does. Based on these observations, an adaptive single step primal-dual large-update SR-IPM for solving LO problems has been proposed. We showed that the complexity of the algorithm matches the one for its analogues presented in [8, 10]. It is worth mentioning that the adaptive SR-IPM does not use any inner process to get recentered. This is a common feature shared by most algorithms implemented in IPM solvers and different from the algorithmic scheme used in [8]. It is worth mentioning that we already have some preliminary numerical tests for our new algorithm based on a variant of LIPSOL [17, 18]. Compared with the standard large-update IPM, the number of iterations of the adaptive algorithm is usually less than or equal to that of the large-update IPM based on the standard Newton direction. There is another difference between these two algorithms. We utilize a line search routine to find a suitable step size for our search direction. It is interesting to note that in most cases, we can really use the default step size $0.995\alpha_{\max}$. For this step size, the proximity function at the new iterate still satisfies $\Phi_q(x, s, \mu_g) \leq \frac{(\tau^{\frac{q-1}{2}} - 1)n}{q - 1}$. Nevertheless, extensive numerical tests are needed to explore the efficiency of the adaptive algorithm. For odd $q \geq 3$ we also have the following relation:

$$\Phi_q(x, s, \mu_g) = \Phi_q(x, s, \mu^*) + \frac{\Phi_q(x, s, \mu^*)^2}{\frac{(q+1)n}{2(q-1)}} + \cdots + \frac{\Phi_q(x, s, \mu^*)^{\frac{q+1}{2}}}{\left(\frac{(q+1)}{2(q-1)}\right)^{\frac{q+1}{2}} (q-1)n^{\frac{q+1}{2}}}.$$

It is an interesting question if there exist such an equality for even q. We are interested in such an equation because such a relation allows to present a simpler complexity analysis.

There are several ways to extend our results. The first is to generalize our adaptive large-update IPM for semi-definite optimization and second-order conic optimization. For this we need to investigate the properties of the proximity function on the cone of semi-definite matrices and on the second-order cone. In [13] we proposed a new infeasible interior-point algorithm based on a SR function. Building on the results of this paper one can propose new infeasible IPMs for a large class of SR functions.

As the authors suggested in [10], we confirm that another extension of the results of this paper is to consider adaptive large-update IPMs based on general SR functions. However, it seems to be very hard to design adaptive IPMs for the whole family of SR functions. Recall our results from Section 2. Some properties of the proximity function rely on the strict and quasi-convexity property of $\Phi_q(x, s, \mu)$ with respect to μ. For the $\Gamma_{1q}(t)$ family one can easily verify strict and quasi-convexity of $\psi(\frac{1}{\sqrt{t}})$. However, in general we can not obtain the strict and quasi-convexity of $\psi(\frac{1}{\sqrt{t}})$ from the self-regularity of $\psi(t)$ (see Proposition 2.1). It is worthwhile to explore for which subclasses of the SR functions such properties holds for $\psi(\frac{1}{\sqrt{t}})$.

Acknowledgement: The first author would like to thanks the Iranian Ministry of Science, Research and Technology for supporting his research. The authors also would like to thank

R. Sotirov for carefully reading the paper and for her helpful comments that improved the paper. The research is also supported by an NSERC discovery grant, the CRC program and by MITACS.

References

[1] E.D. Andersen, J. Gondzio, Cs. Mészáros, and X. Xu (1996). Implementation of interior point methods for large scale linear programming. In: T. Terlaky, editor, *Interior Point Methods of Mathematical Programming*, pp. 189-252. Kluwer Academic Publishers, Dordrecht, The Netherlands.

[2] J. Gondzio (1996). Multiple centrality corrections in a primal-dual method for linear programming, *Computational Optimization and Applications*, **6**, 137-156.

[3] P. Hung and Y. Ye (1996). An asymptotically $O(\sqrt{n}L)$-iteration path-following linear programming algorithm that uses long steps. *SIAM J. on Optimization*, **6**, 570-586.

[4] N.K. Karmarkar (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, **4**, 373-395.

[5] N. Megiddo (1989). Pathways to the optimal set in linear programming. In: N. Megiddo, editor, *Progress in Mathematical Programming: Interior Point and Related Methods*, 131-158. Springer Verlag, New York. Identical version in: *Proceedings of the 6th Mathematical Programming Symposium of Japan, Nagoya, Japan*, 1-35, 1986.

[6] S. Mehrotra (1992). On the implementation of a (primal-dual) interior point method. *SIAM J. on Optimization*, **2**, 575-601.

[7] J. Peng, C. Roos, and T. Terlaky (2002). *Self-Regularity: A New Paradigm for Primal-Dual Interior-Point Algorithms.* Princeton University Press, Princeton, NJ.

[8] J. Peng, C. Roos, and T. Terlaky (2002). Self-regular functions and new search directions for l inear and semidefinite optimization. *Mathematical Programming*, **93**, 129-171.

[9] J. Peng, C. Roos, and T. Terlaky (2002). New polynomial primal-dual methods for second-order conic optimization based on self-regular proximities. *SIAM J. on Optimization*, **13**, 179-203.

[10] J. Peng and T. Terlaky (2002). A dynamic large-update primal-dual interior-point method for linear optimization. *Optimization Methods and Software*, **17**, 1077-1104.

[11] J. Peng, T. Terlaky and Y.B. Zhao (2003). A predictore-corrector algorithm for linear optimization based on a specific self-regular proximity function. *Submitted to SIAM J. on Optimization.*

[12] C. Roos, T. Terlaky, and J.-Ph. Vial (1997). *Theory and Algorithms for Linear Optimization: An Interior Point Approach.* John Wiley & Sons, Chichester, UK.

[13] M. Salahi, T. Terlaky, and G. Zhang (2003). The complexity of self-regular proximity based infeasible IPMs. *Submitted to Computational Optimization and Applications.*

[14] G. Sonnevend (1986). An "analytic center" for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming. In: A. Prékopa, J. Szelezsán, 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.

[15] S.J. Wright (1997). *Primal-Dual Interior-Point Methods.* SIAM, Philadelphia, USA.

[16] Y. Ye (1997). *Interior Point Algorithms, Theory and Analysis.* John Wiley & Sons, Chichester, UK.

[17] Y. Zhang (1999). User's guide to LIPSOL: linear-programming interior point solvers V0.4. *Optimization Methods and Software*, **11/12**, 385-396.

[18] 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.*