McMaster University

Advanced Optimization Laboratory

Title:

On Self–Regular IPMs

Authors:

Maziar Salahi, Renata Sotirov, and Tamás Terlaky

AdvOl-Report No. 2004/4

April 2004, Hamilton, Ontario, Canada

On Self–Regular IPMs

M. Salahi*†, R. Sotirov†, and T. Terlaky†

April 30, 2004

Abstract

Primal-dual interior-point methods (IPMs) have shown their power in solving large classes of optimization problems. However, at present there is still a gap between the practical behavior of these algorithms and their theoretical worst-case complexity results, with respect to the strategies of updating the duality gap parameter in the algorithm. The so-called small-update IPMs enjoy the best known theoretical worst-case iteration bound, but work very poorly in practice. To the contrary, the so-called large-update IPMs have superior practical performance but with relatively weaker theoretical results. In this paper we discuss the new algorithmic variants and improved complexity results with respect to the new family of Self-Regular proximity based IPMs for Linear Optimization problems, and their generalizations to Conic and Semidefinite Optimization.

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

Contents

*Department of Mathematics and Statistics, McMaster University, Hamilton, ON, L8S 4K1, Canada.

†Advanced Optimization Laboratory, Department of Computing & Software, McMaster University, Hamilton, ON, L8S 4K1, Canada. Email: msalahi@optlab.mcmaster.ca, rsotirov@optlab.mcmaster.ca, terlaky@mcmaster.ca

1 Introduction

In 1984, Karamarkar [28] proposed a Linear Optimization (LO) algorithm with a polynomial complexity that was able to solve large–scale LO problems more efficiently than the Simplex method. This was a breakthrough in solving large–scale LO problems and the beginning of the era of modern interior-point methods (IPMs). In 1994, Nesterov and Nemirovski [43] invented the theory of self–concordant functions, allowing polynomial time IPMs to be extended to more complex problems such as Semidefinite Optimization (SDO) and Second-Order Conic Optimization (SOCO). Nowadays IPMs offer the most efficient algorithms for SDO of reasonable size. For a survey on IPMs see the recent books [54, 55, 68, 71].

In this paper, we give an overview of primal-dual IPMs based on the so–called *Self–Regular (SR) proximity functions* [48]. We explain the SR approach for LO problems briefly, and only the basic concept for SDO and SOCO problems. The extensions of the IPMs based on SR-proximities for linear and nonlinear $P_*(\kappa)$ complementarity problems (CP) are presented in [48]. This generalization is not addressed here since we limit the scope of this paper to Conic Linear Optimization (CLO).

We consider LO problems in the standard form

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

where $A \in I\!R^{m\times n}$ satisfies $\text{rank}(A) = m$, $b \in I\!R^m$, $c, x \in I\!R^n$. The dual problem of (P) is

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

We may assume without loss of generality (see [29, 55]) that there exists an (x^0, y^0, s^0) such that

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

i.e., (P) and (D) satisfy the *interior point condition* (IPC). If the IPC holds, then finding an optimal solution of (P) and (D) is equivalent (see [68]) to solving the system of *optimality conditions*

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

where xs denotes the coordinatewise product of the vectors x and s. The last equation in (1) is known as the *complementarity condition*. A basic step toward primal-dual IPMs is to perturb the complementarity condition with some $\mu > 0$. This leads to the following system

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

where $e = (1, \ldots, 1)^T$. If the IPC holds, then for each $\mu > 0$, system (2) has a *unique solution* $(x(\mu), y(\mu), s(\mu))$, see [23, 32, 55]. The set

$$\{(x(\mu), y(\mu), s(\mu)) : \mu > 0\},$$

defines a smooth analytic curve parameterized by μ, called *the primal–dual central path* [55]. The limit of the central path as μ goes to zero exists, see [24, 38, 59]. Because the limit point satisfies the complementarity condition, it naturally yields optimal solutions for both (P) and (D). Primal-dual IPMs follow the central path approximately, and generate points in a certain neighborhood of the central path. A neighborhood of the central path can be defined in the following way

$$\mathcal{N}(n, \tau) := \{(x, y, s) : (x, s) > 0,\ Ax = b, A^T y + s = c,\ \Phi(x, s, \mu) \leq \eta(n, \tau),\ \mu > 0\}, \tag{3}$$

where $\Phi(x, s, \mu)$ is a *proximity function* to measure the distance from the present point to the central path, and $\eta(n, \tau)$ is a function of a parameter $\tau > 0$ and the dimension n of the underlying problem. For different choices of the proximity measure for classical IPMs see [55, 68].

Now we describe how classical primal–dual IPMs work (see also Algorithm 1). We start with a point (x, y, s) that satisfies the IPC. Without loss of generality [55] we may assume that $(x, y, s) \in \mathcal{N}(n, \tau)$ on the central path with $\mu = 1$. The aim is to move this triple toward the optimal set. In order to do that we target a point on the central path given by the target parameter $\mu := (1 - \theta)\mu$, for some $\theta \in [0, 1]$. Then we solve the Newton system

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

to obtain the search direction. Since matrix A has a full row rank, the Newton system (4) has a unique solution for all $\mu > 0$. Then we take a step along the search direction with a step size $\alpha \in (0, 1]$ which is defined by some line search rule. The search direction and line search rule ensure that the new triple $(x+\alpha\Delta x, y+\alpha\Delta y, s+\alpha\Delta s)$ is closer to the μ–center $(x(\mu), y(\mu), s(\mu))$. This step is repeated as long as the actual iterate is sufficiently close to the μ center. Then μ is reduced again by the factor $(1 - \theta)$ and the process is repeated until an approximate solution to the problem is obtained, e.g., unitil μ gets small enough.

Algorithm 1

Input:
a proximity parameter τ;
an accuracy parameter $\varepsilon > 0$;
(x^0, s^0) and $\mu^0 = 1$ such that $\Phi(x^0, s^0, \mu^0) \leq \eta(n, \tau)$;
begin
$x := x^0$; $s := s^0$; $\mu := \mu^0$;
while $n\mu \geq \varepsilon$ **do**,
begin
choose[a] μ;
while $\Phi(x, s, \mu) \geq \eta(n, \tau)$ **do**,
begin
solve the system (8) 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

[a]There are different ways to choose μ. In classical IPMs μ is decreased by the factor $1 - \theta$ at each iteration, where θ may depend on n or may be independent of n, see [55]. In our adaptive algorithm, μ is chosen as it is described in Section 3 (see also [57]).

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 fixed ratio $1 - \theta$ at each iteration of the algorithm. As a consequence, the choice of the parameter θ has an important role in the design and analysis of IPMs. If θ is a constant, for instance $\theta = 1/2$, then we call the algorithm a *large-update* (or long-step) method. If θ depends on the problem dimension, such as $\theta = 1/\sqrt{n}$, then the algorithm is named a *small-update* (or short-step) method. In the classical primal–dual IPMs there is a gap between the practical performance of IPMs and their theoretical worst-case complexity results with respect to different choices of θ. The small-update method has the best known $\mathcal{O}(\sqrt{n}\log\frac{n}{\varepsilon})$ iteration bound, while the large-update method has a worse $\mathcal{O}(n\log\frac{n}{\varepsilon})$ iteration bound [55, 68, 71]. However, large-update IPMs perform much better in practice than small-update methods [5].

Several strategies have been proposed to decrease the gap, i.e., to improve the complexity of large-update IPMs. Hung et al. [26], Jansen et al. [27], and Monteiro et al. [40] use higher order methods to reduce the complexity of large-update IPMs. However, there is a price to pay for the reduced complexity; higher order methods are computationally more expensive per iteration than first order methods, since some additional equation systems need to be solved with the same coefficient matrix at each iteration. Recently Peng et al. [48, 49] have proposed a new strategy for improving the theoretical complexity of large-update IPMs. The choice of the proximity measure and search direction turned out to be the crucial factor for the performance,

as well as for the quality of the analysis of IPMs. In this paper, we focus on the novel approach proposed in [48, 49], where new primal–dual IPMs are induced based on so-called *Self-Regular proximity measures* that are used to define new search directions and to control the iterative process. We discuss several algorithm variants for large-update path–following methods and report their polynomial iteration complexity.

To describe the new family of algorithms, we introduce the following notation. Let $\mathbb{R}^n_{++} = \{x \in \mathbb{R}^n | x > 0\}$ denote the positive orthant and $\mathbb{R}^n_+$ denote the nonnegative orthant in $\mathbb{R}^n$. For any strictly feasible primal-dual pair (x, s) and any $\mu > 0$, we define the vectors

$$v := \sqrt{\frac{xs}{\mu}}, \quad \text{and} \quad v^{-1} := \sqrt{\frac{\mu e}{xs}}, \tag{5}$$

whose i^{th} components are $\sqrt{\frac{x_i s_i}{\mu}}$ and $\sqrt{\frac{\mu}{x_i s_i}}$, respectively. Using the above notation, one can state the *centrality condition* in (2) as

$$v = v^{-1} = e,$$

and rewrite the last equation in (4) as

$$d_x + d_s = v^{-1} - v, \tag{6}$$

where

$$d_x := \frac{v\Delta x}{x}, \quad \text{and} \quad d_s := \frac{v\Delta s}{s}. \tag{7}$$

The vectors d_x, d_s give the search directions in the *scaled v–space* and system (4) in the scaled space is given as

$$\begin{aligned} \bar{A}d_x &= 0, \\ \bar{A}^T\Delta y + d_s &= 0, \\ d_x + d_s &= v^{-1} - v, \end{aligned} \tag{8}$$

where

$$\bar{A} = \frac{1}{\mu} AV^{-1}X, \tag{9}$$

for $V = \text{diag}\,(v)$ and $X = \text{diag}\,(x)$.

Remark 1.1 *The right hand side of the third equation in (8) is the negative gradient of the primal-dual logarithmic barrier function that is defined as follows*

$$\Psi_{\ell b}(v) = \sum_{i=1}^{n} \frac{v_i^2 - 1}{2} - \log v_i.$$

This shows that the scaled version of the classical search direction in primal–dual methods for LO can be interpreted as the steepest descent direction for the scaled logarithmic barrier function.

The approach described in this paper is based on the observation that in principle any twice continuously differentiable strictly convex function $\psi(t)$ which satisfies the following conditions

$$\psi'(1) = \psi(1) = 0, \tag{10}$$

$$\psi''(t) > 0, \quad \forall\, t > 0, \tag{11}$$

$$\lim_{t\downarrow 0} \psi(t) = \lim_{t\to\infty} \psi(t) = \infty, \tag{12}$$

is a good candidate to replace the logarithmic barrier function $(t - 1 - \log t)$ in defining the proximity measure and the search direction in primal-dual IPMs. Then a *proximity measure* $\Psi(v) : I\!R^n_{++} \to I\!R_+$, based on the *kernel function* ψ, in the scaled space can be defined as the sum of the componentwise deviations

$$\Psi(v) := \sum_{i=1}^{n} \psi(v_i). \tag{13}$$

The motivation of the new search direction is to increase the small components $v_i < 1$ and to decrease the large components $v_i > 1$, *more* than the classical one. It is reasonable to expect that such approach leads to approaching the μ–center faster. Hence, the crucial step in the new approach is to define appropriate functions that measure the discrepancy between the vectors $e = (1, \ldots, 1)^T$ and v and satisfies properties (10)–(12). Then (13) can be used to define a new *proximity function* and the negative gradient of $\Psi(v)$ can be used to define a new (scaled) search direction, i.e., the third equation in (8) may be written as

$$d_x + d_s = -\nabla\Psi(v). \tag{14}$$

In [48, 49] Peng et al. introduce the class of SR functions which provides a rich source of kernel functions and satisfies all desired properties. They modify system (8) according to (14), i.e., the solution of the modified system gives the projected steepest descent direction for the proximity measure $\Psi(v)$. By this modification of the third equation in (8), the following Newton system is obtained:

$$\begin{aligned} \bar{A}d_x &= 0, \\ \bar{A}^T\Delta y + d_s &= 0, \\ d_x + d_s &= -\nabla\Psi(v). \end{aligned} \tag{15}$$

System (15) can be equivalently presented in terms of the search directions in the original space in the following way

$$\begin{aligned} A\Delta x &= 0, \\ A^T\Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= -\mu v\nabla\Psi(v). \end{aligned} \tag{16}$$

By using a SR function as a kernel function, Peng et al. [48, 49] show that the worst case iteration complexity of large-update methods can be improved so that the gap between the theoretical results for small- and large-update IPMs is significantly reduced. This surprising result follows from a careful analysis of the new method, which depends on a number of technical results. Here we present some of the main results, and for details we refer to the papers [48, 49, 50, 57, 58].

The paper is organized as follows. In Section 2 we define univariate SR functions and give their basic properties. We also give the properties of SR-proximity functions for LO. In Section 3 we present an adaptive large-update SR–IPM for LO, and in Section 4 a SR-proximity based predictor-corrector IPM for LO. In Section 5 we explain the concept of an infeasible SR-IPM. Some closely related kernel functions that are not SR, but still allow to prove polynomial complexity of IPMs, are described in Section 6. In Sections 7 and 8, we explain how the concept of SR proximity based IPMs for LO can be extended to more complex problems, such as SDO and SOCO, respectively.

Notation. Throughout the paper $\|\cdot\|$ denotes the 2-norm of vectors and $\|\cdot\|_\infty$ denotes the infinity norm. 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 $x = (x_1, x_2, \ldots, x_n)^T \in I\!R^n$, $x_{\min} = \min\{x_1, x_2, \ldots, x_n\}$ and $x_{\max} = \max\{x_1, x_2, \ldots, x_n\}$ denote the smallest and the largest components of x, respectively.

2 Self–Regular Proximity Functions

2.1 Univariate SR Functions

Here we give the definition and the basic properties of a univariate SR kernel functions [48, 49]. In Sections 7 and 8 we show how the concept of SR functions can be transparently extended to various cones, such as the cone of positive semidefinite matrices and the second–order cone, respectively.

As mentioned before, the centrality condition in the v–space (i.e., $v = e$) motivates us to construct functions in $I\!R^n_+$ that attains the global minimum at e and can be used to measure the distance from any point in $I\!R^n_+$ to e. However, it is also desirable for the function to enjoy certain barrier properties that prevents the argument from moving to the boundary of $I\!R^n_+$. Motivated with the previously described function–properties Peng et al. [49] define the family of univariate SR functions in the following way.

Definition 2.1 *A function $\psi : (0, \infty) \to I\!R$, $\psi \in \mathcal{C}^2$ 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

$$\nu_1(t^{p-1} + t^{-1-q}) \leq \psi''(t) \leq \nu_2(t^{p-1} + t^{-1-q}), \quad \forall t \in (0, \infty); \tag{17}$$

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

$$\psi(t_1^r t_2^{1-r}) \leq r\psi(t_1) + (1 - r)\psi(t_2), \quad \forall r \in [0, 1]. \tag{18}$$

If $\psi(t)$ is SR, then parameter q is called the *barrier degree*, and parameter p is called the *growth degree* of the SR function $\psi(t)$. There are two popular families of SR functions that are used in the design of new SR-IPMs and play a crucial role in the analysis of the new search directions.

Example 2.2 *The first family is given by*

$$\Upsilon_{p,q}(t) = \frac{t^{p+1} - 1}{p(p + 1)} + \frac{t^{1-q} - 1}{q(q - 1)} + \frac{p - q}{pq}(t - 1), \quad p \geq 1,\ q > 1, \tag{19}$$

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

$$\Gamma_{p,q}(t) = \frac{t^{p+1} - 1}{p + 1} + \frac{t^{1-q} - 1}{q - 1}, \quad p \geq 1,\ q > 1, \tag{20}$$

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

Equivalent conditions to condition SR.2 are given in the following lemma.

Lemma 2.3 [48] *Let $\psi : (0,\infty) \to [0,\infty), \psi \in \mathcal{C}^2$. Function $\psi(\cdot)$ satisfies SR.2 if and only if the following equivalent statements hold.*
1) *$\psi(e^t)$ is convex.*
2) *$\psi'(t) + t\psi''(t) \geq 0$ when $t > 0$.*
3) *$\psi(\sqrt{t_1 t_2}) \leq \frac{1}{2}\psi(t_1) + \frac{1}{2}\psi(t_2)$, for $t_1, t_2 > 0$.*

The following property of SR functions can be easily proved.

Proposition 2.4 [48] *If the functions $\psi_1(t), \psi_2(t)$ are SR, then so is any convex conic combination $\beta_1\psi_1 + \beta_2\psi_2$ with $\beta_1, \beta_2 \geq 0,\ \beta_1 + \beta_2 > 0$.*

Since any nontrivial conic combination of two SR functions ψ_1 and ψ_2 is SR, the set of SR functions is a pointed convex cone. The following proposition collects several properties of the SR functions that are proved in [48].

Proposition 2.5 *Let Ω_1, Ω_2 respectively, be the sets of functions whose elements satisfy conditions SR.1 and SR.2, respectively. For $t > 0$ the following statements hold.*
1. *Suppose that $\psi(t) \in \Omega_1$. Then $\left|\frac{1}{t}\psi'(t)\right| \leq \frac{\nu_2}{\nu_1}\psi''(t)$.*
2. *If a function $\psi(t) \in \Omega_1$ with $\nu_1 = \nu_2$, then it is SR.*
3. *Suppose that $\psi(t) \in \Omega_1$. Then $2\nu_1\psi(t) \leq \psi(t)'(t)^2$.*
4. *If $\psi(t) = \psi(t^{-1})$ and $\psi(t) \in \Omega_1$, then $\psi(t)$ is SR.*
5. *If $\psi(t) \in \Omega_2$, then also $\psi\left(\frac{1}{t}\right) \in \Omega_2$.*
6. *Let N be any positive integer and*
   $\psi(t) = \beta_0 \log t + \sum_{i=1}^{N} \beta_i(t^{\rho_i} - 1), \quad \beta_0 \in I\!R,\ \beta_i \geq 0,\ \rho_i \in I\!R,\ i = 1, 2, \ldots, N.$
   Then $\psi(t) \in \Omega_2$.

2.2 Properties of SR-Proximity Functions for LO

Here we present some properties of SR-proximity functions that will be used in the analysis of SR-IPMs. First we study common properties of the proximity functions that are based on the $\Gamma_{1q}(t)$ family for $q \in I\!R,\ q > 1$, and then separately the properties that apply only for $q = 3$ and $q = \log(n) + 1$. SR-proximity functions induced by the $\Gamma_{1q}(t),\ q \in I\!R,\ q > 1$ family of kernel functions allow to design various large-update algorithms, with the best known iteration complexity [51, 56, 57].

Let us define the family of proximity functions based on the $\Gamma_{1q}(t)$ family as

$$\Phi_q(x, s, \mu) := \Psi_q(v) = \frac{e^T v^2 - n}{2} + \frac{e^T v^{1-q} - n}{q - 1}. \tag{21}$$

Proposition 2.6 [57] *For any fixed $(x, s) > 0$, the proximity function $\Phi_q(x, s, \mu)$ as a function of μ, attains its global minimum at*

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

The following result is an immediate consequence of Proposition 2.6.

Corollary 2.7 [57] *For any fixed $(x,s) > 0$, the proximity function $\Phi_q(x,s,\mu)$ is a decreasing function with respect to μ when $\mu \leq \mu_q^*$, and it is an increasing function of μ if $\mu > \mu_q^*$.*

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

Lemma 2.8 [57] *Let $\tau \in I\!R,\ \tau \geq 2$. For the generalized harmonic mean*

$$\mu_q^h = \left(\frac{n}{(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}}} \right)^{\frac{2}{q-1}}$$

of the components of vector xs and the current duality gap

$$\mu_g := \frac{x^T s}{n}, \tag{22}$$

the following statements are equivalent:
1) $\frac{\mu_g}{\mu_q^h} \leq \tau$,
2) $\Phi_q\left(x, s, \frac{\mu_g}{\tau}\right) \leq \frac{(\tau-1)n}{2}$,
3) $\Phi_q(x,s,\mu_g) \leq \frac{\left(\tau^{\frac{q-1}{2}}-1\right)n}{q-1}$.

Due to the choice of the kernel function $\psi(t) = \Gamma_{1q}(t)$, system (16) has the following form

$$\begin{aligned} A\Delta x &= 0, \\ A^T\Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu^{\frac{q+1}{2}}(x^{\frac{1-q}{2}})s^{\frac{1-q}{2}} - xs. \end{aligned} \tag{23}$$

Let us denote the solution of system (23) 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 $\mu = \mu_q^*$ and $\mu = \mu_q^h$.

Lemma 2.9 [57] *Let $(\Delta x(\mu_q^*), \Delta y(\mu_q^*), \Delta s(\mu_q^*))$ be the solution of system (23) with $\mu = \mu_q^*$. Then the relation*

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

holds.

Proof: Directly from the definition of μ_q^*. □

Corollary 2.10 [57] *If $\mu = \mu_q^*$, then the duality gap will not change for any feasible step size α, i.e.,*

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

Analogous to Lemma 2.9, the following result is obtained for $\mu = \mu_q^h$.

Lemma 2.11 [57] *Let $(\Delta x(\mu_q^h), \Delta y(\mu_q^h), \Delta s(\mu_q^h))$ be the solution of system (23) with $\mu = \mu_q^h$. Then the relation*

$$x^T\Delta s(\mu_q^h) + s^T\Delta x(\mu_q^h) = n\mu_q^h - x^T s$$

holds.

Corollary 2.12 *If the targeted parameter is μ_q^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, i.e.,*

$$(x + \alpha\Delta x(\mu_q^h))^T(s + \alpha\Delta s(\mu_q^h)) = x^T s\left(1 - \alpha + \frac{\mu_q^h \alpha}{\mu_g}\right).$$

In some of the algorithms described in the subsequent sections, we explore properties of some proximity functions based on particular members of the $\Gamma_{1q}(t)$ family i.e., for $q = 3$ and $q = \log n + 1$. Since these proximity functions have specific properties that do not apply for arbitrary $q > 1$, we present them separately.

• $q = 3$
The SR function $\Gamma_{13}(t)$ is used as a kernel function in the infeasible algorithm (see Section 5). The proximity function based on the kernel function $\Gamma_{13}(t)$ is defined as follows

$$\Phi_3(x, s, \mu) := \Psi_3(v) = \frac{1}{2}\left\|v - v^{-1}\right\|^2. \tag{24}$$

The following proposition shows that for fixed $(x, s) > 0$, the function $\Phi_3(x, s, \mu)$ attains its global minimum at a point $\mu < \mu_g$, where μ_g is the current duality gap defined by (22).

Proposition 2.13 [51] *For any fixed $(x, s) > 0$, the proximity function $\Phi_3(x, s, \mu)$ as a function of μ, has the global minimizer μ_3^* that is the geometric mean of μ_g and μ_3^h, i.e.,*

$$\mu_3^* = \sqrt{\frac{x^T s}{x^{-T} s^{-1}}} = \sqrt{\mu_g \mu_3^h}.$$

Note that Proposition 2.13 directly follows from Proposition 2.6 and Lemma 2.8.

It is easy to verify the following interesting relations that play a crucial role in the design of algorithmic schemes based on this special proximity function.

Proposition 2.14 [51] *Suppose that $(x, s) > 0$ is fixed. Then we have*

$$\Phi(x, s, \mu_g) = \Phi(x, s, \mu_3^h),$$

and

$$\Phi(x, s, \mu_g) = \Phi(x, s, \mu_3^*) + \frac{\Phi(x, s, \mu_3^*)^2}{2n}.$$

Based on the $\Gamma_{13}(t)$ kernel function, the authors of [51] propose also a dynamic large-update IPM which improve the complexity of large-update methods significantly.

• $q = \log n + 1$
In the SR predictor–corrector approach, that is presented in Section 4, the following SR-proximity function is used

$$\Phi_\ell(x, s, \mu) := \Psi_\ell(v) = \frac{e^T v^2 - n}{2} + \frac{e^T v^{-\log n} - n}{\log n}. \tag{25}$$

For fixed $(x, s) > 0$, the global minimum of the proximity function (25) with respect to μ is $\mu_l^* := \mu_{\log n+1}^*$ (see Proposition 2.6). We provide some specific properties of this proximity function in Section 4.

3 Adaptive Large-update IPM for LO

In this section we present a family of adaptive large-update IPMs for LO based on the $\Gamma_{1q}(t), q > 1$ family of kernel functions [57]. The motivation for our adaptive algorithm is to develop a variant, that is more flexible in updating μ than classical polynomial IPMs [55], and is closer to what is implemented in IPM solvers (LIPSOL [72], McIPM [74]). The algorithm described here use large-update at each iteration and does not make any recentering iteration.

For $\tau \geq 2$, we set $\frac{(\tau-1)n}{2}$ as the maximum allowed value of the proximity function. At each iteration we choose the target value μ_q^t so that $\Phi_q(x, s, \mu_q^t) = \frac{(\tau-1)n}{2}$. One can see that $\Phi_q(x, s, \mu_q^t) = \frac{(\tau-1)n}{2}$ if and only if μ_q^t solves the equation

$$2(x^{\frac{1-q}{2}})^T s^{\frac{1-q}{2}} \mu^{\frac{q+1}{2}} - (2n + \tau(q-1)n)\,\mu + (q-1)x^T s = 0. \tag{26}$$

This equation has two positive roots; one is less than or equal to μ_q^*, and the other is larger than or equal to μ_q^*. At each iteration of the algorithm, we use the smaller positive root μ_q^t as the target value. One can easily prove that $\mu_q^t \leq \mu_q^h$ holds when $\mu_g \leq \tau\mu_q^h$ with $\mu_q^t = \mu_q^h$ if and only if $\mu_g = \tau\mu_q^h$.

The following lemma gives a lower bound for the norm of $(v - v^{-q})$, when the target value is μ_q^t.

Lemma 3.1 *Let $\tau \geq 2$. Then $\sigma = \|v - v^{-q}\| \geq 1$.*

Proof: By using Proposition 3.1.5 of [48], we get

$$\sigma^2 \geq 2\Phi(x, s, \mu_q^t) = (\tau - 1)n \geq 1, \quad \forall n \geq 1.$$

□

Lemma 3.2 [57] *Let μ_q^t be the smaller positive root of equation (26). Then the inequality*

$$\mu_q^h \leq 2\tau\mu_q^t$$

holds.

Proof: The function on the left hand side of equation (26) is a convex function of μ. If we substitute $\mu = \frac{\mu_q^h}{2\tau}$ in that function, and use the fact that $\mu_g \geq \mu_q^h$, then one can see that the function value is larger than

$$\frac{2}{(2\tau)^{\frac{q+1}{2}}} - \frac{1}{\tau} + \frac{q-1}{2}. \tag{27}$$

It is sufficient to prove that (27) is nonnegative. Since $\tau \geq 2$, the expression in (27) is a strictly increasing function of q for $q \geq 1$. For $q = 1$ its value is zero and thus for $q > 1$ it is positive. This completes the proof of the lemma. □

When μ_q^t is the target value, we have the following result.

Lemma 3.3 [57] *Let $(\Delta x(\mu_q^t), \Delta y(\mu_q^t), \Delta s(\mu_q^t))$ be the solution of system (23), for $\mu = \mu_q^t$ where μ_q^t is the smallest positive root of equation (26). Then the relations*

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

and

$$(x + \alpha \Delta x(\mu_q^t))^T (s + \alpha \Delta s(\mu_q^t)) = (x + \alpha \Delta x)^T (s + \alpha \Delta s) = x^T s \left(1 - \alpha + \frac{{\mu_q^t}^{\frac{q+1}{2}} \alpha}{\mu_g {\mu_q^h}^{\frac{q-1}{2}}}\right) \quad (28)$$

hold.

The proof of Lemma 3.3 is analogous to the proofs of Lemma 2.9 and Lemma 2.11.

Remark 3.4 *If $\mu_q^t \sim \mu_q^h$, then (28) 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 $\mu_q^t = \mu_q^h$, see Lemma 2.11.) But if $\mu_q^t \ll \mu_q^h$, then the ratio $\frac{{\mu_q^t}^{\frac{q+1}{2}}}{\mu_g {\mu_q^h}^{\frac{q-1}{2}}}$ 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.*

Further motivation is available in [57]. The new adaptive algorithm (see Algorithm 2) is the special case of Algorithm 1 on page 3. Algorithm 2, regardless if the iterate is close to or far away from the central path, always make a large-update of the central path parameter μ. We choose $\mu = \mu_q^t$ at each iteration, and we just solve one Newton system at each iteration.

Let us define the SR-proximity based neighborhood (see (3)) as it follows:

$$\mathcal{N}_q(n, \tau) := \{(x, y, s) : (x, s) > 0,\ Ax = b, A^T y + s = c,\ \Phi_q(x, s, \mu) \leq \eta_q(n, \tau),\ \mu > 0\}, \quad (29)$$

where $\Phi_q(x, s, \mu)$ is defined by (21), $\tau \geq 2$ and we define

$$\eta_q(n, \tau) := \frac{(\tau^{\frac{q-1}{2}} - 1)n}{q - 1}.$$

Recall the results of Lemma 2.8 that quantify the relation between μ_g, μ_q^h and the corresponding proximity values. These relations provide useful tools for the complexity analysis. The key element of the analysis is to give a bound for the step size α that imply sufficient reduction of μ_g. For this, one need to explore the changing behavior of the functions $\Phi_q(x(\alpha), s(\alpha), \mu_g(\alpha))$ and $\Phi_q(x(\alpha), s(\alpha), \mu_t)$.

Algorithm 2: An Adaptive Large-Update SR-IPM

Input:
a proximity parameter $\tau \geq 2$;
an accuracy parameter $\varepsilon > 0$;
$(x, s) = (x^0, s^0)$ such that $\frac{\mu_g}{\hat{\mu}_q^h} \leq \tau$;

begin
while $x^T s \geq \varepsilon$ **do**
begin
$\mu := \mu_q^t$ computed from (26);
solve system (23) for $\Delta x, \Delta y, \Delta s$;
begin
determine a step size α such that
$\Phi_q(x(\alpha), s(\alpha), \mu_q^t) \leq \Phi_q(x, s, \mu_q^t) - \frac{2^{\frac{q-1}{2q}} \Phi_q(x,s,\mu_q^t)^{\frac{q-1}{2q}}}{24q}$
and $\mu_g(\alpha) \leq \tau \mu_q^h(\alpha)$;
$x := x + \alpha \Delta x$;
$y := y + \alpha \Delta y$;
$s := s + \alpha \Delta s$,
end
end
end

First we prove the following technical lemma.

Lemma 3.5 *Let* $\sigma = \|\nabla \Psi_q(v)\|$. *Then*

$$v_{\min} \geq (1 + \sigma)^{-\frac{1}{q}}.$$

Proof: The lemma is trivial if $v_{\min} \geq 1$. Now consider the case when $v_{\min} < 1$. Then $\sigma = \|v - v^{-q}\| \geq v_{\min}^{-q} - v_{\min} \geq v_{\min}^{-q} - 1$. This completes the proof of the lemma. □

Now we give a lower bound for the maximal feasible step size.

Lemma 3.6 [57] *Let* $(\Delta x, \Delta y, \Delta s)$ *be the solution of system (23) where* $\mu = \mu_q^t$ *is the smallest positive root of equation (26), and* $\sigma = \|\nabla \Psi_q(v)\|$. *Then the maximal feasible step size,* $\alpha_{\max}$, *satisfies*

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

Proof: By using the definition of $v(\alpha)$ (see (5) for the definition of v), we have

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

One has $v(\alpha_{\max}) \geq 0$ if

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

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 Lemma 3.5. This completes the proof of the lemma. □

The following theorem estimates the decrease of the proximity function after an iterate.

Theorem 3.7 [57] *Let $(\Delta x, \Delta y, \Delta s)$ be the solution of system (23), where $\mu = \mu_q^t$ is the smaller positive root of equation (26). Then for any step size $\alpha \leq \alpha^* := \frac{\bar{\alpha}}{3q}$, where $\bar{\alpha}$ is defined in Lemma 3.6, the relation*

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

holds.

Proof: Let

$$\begin{aligned} h(\alpha) : &= \Phi_q\left(x(\alpha), s(\alpha), \mu_q^t\right) - \Phi_q(x, s, \mu_q^t) \qquad (30)\\ &= \frac{\|v(\alpha)\|^2 - n}{2} + \frac{\left\|v(\alpha)^{\frac{q-1}{2}}\right\|^2 - n}{q-1} - \frac{\|v\|^2 - n}{2} - \frac{\left\|v^{\frac{q-1}{2}}\right\|^2 - n}{q-1}, \end{aligned}$$

where v is defined by (5). Using condition SR.2 of Definition 2.1 we have

$$\begin{aligned} h(\alpha) &\leq \frac{1}{2} v^T (d_x + d_s)\alpha + \frac{1}{2(q-1)} \sum_{i=1}^{n} (v_i + \alpha(d_x)_i)^{1-q} \qquad (31)\\ &\quad + \frac{1}{2(q-1)} \sum_{i=1}^{n} (v_i + \alpha(d_s)_i)^{1-q} - \frac{1}{q-1} \left\|v^{\frac{1-q}{2}}\right\|^2 := h_1(\alpha). \end{aligned}$$

It can be 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

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

Function $h_2(\alpha)$ is twice differentiable convex function on the interval $[0, \bar{\alpha})$. Let us denote by α_1^* the global minimum of $h_2(\alpha)$ on 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 a proof analogous to the proof of Lemma 3.3.3 in [48], one can show that $\alpha_1^* \geq \alpha^* = \frac{1}{3q\sigma(1+\sigma)^{\frac{1}{q}}}$. Furthermore, from Lemma 1.3.3 in [48] and for any $\alpha \leq \alpha^*$ it follows that

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

We also have $\sigma^2 \geq 2\Phi_q(x, s, \mu_q^t)$ (see Proposition 3.1.5, [48]), that completes the proof. □

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

Theorem 3.8 [57] *Let $(\Delta x, \Delta y, \Delta s)$ be the solution of system (23), where $\mu = \mu_q^t$ is the smaller positive root of equation (26). 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 3.7, we know that any $\alpha \leq \alpha^*$ is strictly feasible. We also have

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

Using the fact that $\mu_q^t(\alpha) \leq \mu_q^h(\alpha)$, for any $\alpha \leq \alpha^*$ we have

$$\Phi_q(x(\alpha), s(\alpha), \mu_q^h(\alpha)) \leq \frac{(\tau-1)n}{2},$$

that is by Lemma 2.8 equivalent to

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

that completes the proof of the theorem. □

To obtain an upper bound for the total number of iterations of the algorithm, one need to estimate the change of the parameter μ_q^t before and after an iterate. The following technical lemma is needed for the complexity analysis.

Lemma 3.9 [57] *Let $v_+ = \frac{v}{\sqrt{1-\theta}}$ for some $\theta \in (0,1)$. Then we have:*

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

By applying Lemma 3.9 to Theorem 3.7, the following theorem can be proved.

Theorem 3.10 [57] *Let $\tau \geq 2$, and let $(\Delta x, \Delta y, \Delta s)$ be the solution of system (23) with $\mu = \mu_q^t$ where μ_q^t is the smallest positive root of equation (26), and let α^* be the default step size as defined in Theorem 3.7. Then*

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

where

$$\theta = \frac{(\tau-1)^{\frac{q-1}{2q}}}{12q\left(\tau + 2 + \log\tau\right) n^{\frac{q+1}{2q}}}.$$

Now we proceed to present the complexity of our adaptive algorithm. By the choice of μ_q^t we know that the proximity function $\Phi_q(x, s, \mu_q^t)$ keeps invariant for all the iterates. Let us denote by $\mu_q^t(\alpha^*)$ the target parameter value after one step with step size α^*. Then we have

$$\Phi_q(x, s, \mu_q^t) = \Phi_q(x(\alpha^*), s(\alpha^*), \mu_q^t(\alpha^*)).$$

On the other hand, from Theorem 3.10 we have

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

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

$$\mu_q^t(\alpha^*) \leq \left(1 - \frac{(\tau-1)^{\frac{q-1}{2q}} n^{\frac{-q-1}{2q}}}{12q\,(\tau + 2 + \log\tau)}\right) \mu_q^t. \tag{32}$$

Now we are ready to present the complexity of our adaptive algorithm.

Theorem 3.11 [57] *Let $\tau \geq 2$. Then after at most*

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

iterations the adaptive algorithm will terminate with a feasible solution satisfying $x^T s \leq \epsilon$.

Proof: In light of inequality (32) we know that after at most

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

iterations we have $\mu_q^t \leq \frac{\epsilon}{2n\tau^2}$. By using Lemma 2.8 and Lemma 3.2, we have that $\mu_g \leq 2\tau^2\mu_q^t \leq \frac{\epsilon}{n}$, or equivalently $x^T s \leq \epsilon$. □

The following corollary gives, so far, the best complexity for large-update IPMs [48].

Corollary 3.12 [57] *For $q = \log(n)$, Theorem 3.11 provides the following upper bound for the total number of iterations:*

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

4 SR-Proximity Based Predictor-Corrector IPM (SR-PC)

In this section we present a SR-proximity based predictor-corrector IPM for LO [58]. The proximity function that we use in this section is the specific proximity function $\Phi_\ell(x, s, \mu)$ introduced by (25). Note that for this case $q = \log n + 1$.

The particular interest of this algorithm is when the iterates are far away from the central path. Predictor-corrector algorithms traditionally use an infinity neighborhood, which is defined by

$$\mathcal{N}_\infty^-(\rho) := \{(x, y, s) : (x, s) > 0,\ Ax = b,\ A^T y + s = c,\ \left\|(v^2 - e)^-\right\|_\infty \leq \rho\}, \tag{33}$$

where $a^- = \min(a, 0)$, and $\rho \in (0, 1)$ is a constant independent of n and τ. We use SR-neighborhoods, and in order to get comparable results we change $\eta(n, \tau)$ in the definition (29) of the neighborhood $\mathcal{N}_q(n, \tau)$ so that the resulting SR-neighborhood contains the $\mathcal{N}_\infty^-(\rho)$ neighborhood [53].

Let $(x, s) \in \mathcal{N}_\infty^-(\rho)$, and $\mu = \mu_g$. Then for $\Phi_\ell(x, s, \mu)$ as introduced by (25) the following inequality is satisfied,

$$\Phi_\ell(x, s, \mu_g) = \frac{e^T v^{-\log n} - n}{\log n} \leq \frac{n(1-\rho)^{\frac{-\log n}{2}} - n}{\log n} = \frac{n^\tau - n}{\log n} = \eta_\ell(n, \tau),$$

where $\tau = 1 - \log(1-\rho)$. With this choice of $\eta_\ell(n, \tau)$ the neighborhood $\mathcal{N}_\ell(n, \tau) := \mathcal{N}_{\log n+1}(n, \tau)$ contains the neighborhood $\mathcal{N}_\infty^-(\rho)$, and the inclusion $\mathcal{N}_\infty^-(\rho) \subseteq \mathcal{N}_\ell(n, \frac{1}{1-\rho})$ holds too. This demonstrates that these two neighborhood almost match each other. The following result specifies the relation between μ_g and μ_l^*.

Lemma 4.1 [58] *Let $\mu_\ell^* = \mu_{\log n+1}^*$, as defined in Proposition 2.6. Then $\mu_\ell^* = \theta_1(\tau)\mu_g$, where $\theta_1(\tau) = n^{\frac{2(1-\tau)}{\log n+2}}$.*

Proof: Let $\mu_\ell^* = \theta_1(\tau)\mu_g$ and we want to derive a lower bound for $\theta_1(\tau)$. The global minimum of the proximity function is the solution of the following equation

$$-\frac{x^T s}{2(\mu_\ell^*)^2} + \frac{(\mu_\ell^*)^{\frac{\log n-2}{2}} \left(x^{-\frac{\log n}{2}}\right)^T s^{-\frac{\log n}{2}}}{2} = 0.$$

This is equivalent to

$$\frac{n}{\theta_1(\tau)^2 \mu_g} = \theta_1(\tau)^{\frac{\log n-2}{2}} \mu_g^{\frac{\log n-2}{2}} \left(x^{-\frac{\log n}{2}}\right)^T s^{-\frac{\log n}{2}},$$

and that implies

$$n\theta_1(\tau)^{\frac{-\log n-2}{2}} = \left\| v^{\frac{-\log n}{2}} \right\|^2 = n^\tau.$$

Finally,

$$\theta_1(\tau) = n^{\frac{2(1-\tau)}{\log n+2}}.$$

□

In the original predictor-corrector algorithm [39], *the primal-dual affine scaling direction* is used in the predictor step. The so-called primal-dual affine scaling search direction is the solution of the system

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= -xs. \end{aligned} \tag{34}$$

It is well known that close to optimality, the step size of the affine scaling direction is converging to one. This fact implies that close to optimality the affine scaling direction is the best choice.

In this section we present an SR-PC algorithm that in the first iterations chooses SR directions not only in the corrector, but also in the predictor step [58]. The duality gap reduction of the affine scaling direction is comparable with the duality gap reduction of the SR direction, and the new iterate moves in the direction that gives larger duality gap reduction. Our next task is to define the new target value that the SR-PC algorithm use in the predictor step (when the predictor direction is a SR-direction).

Since the SR-PC algorithm operates in a large neighborhood, the maximum allowed value of the proximity function w.r.t. the target μ is $\frac{(\tau_1-1)n}{2}$, where $\tau_1 = (1-\rho)^{-1}$, see [58]. Analogous to the previous section, $\Phi_\ell(x,s,\mu_\ell^t) = \frac{(\tau_1-1)n}{2}$ if and only if μ_ℓ^t is a root of

$$2(x^{\frac{-\log n}{2}})^T s^{\frac{-\log n}{2}} \mu^{\frac{\log n+2}{2}} - (2n + \tau_1 n \log n)\,\mu + x^T s \log n = 0. \tag{35}$$

This equation has two positive roots; one is less than or equal to μ_g, and the other one is larger than or equal to μ_g. In the SR-PC algorithm the smaller positive root μ_ℓ^t is used as the target value in the predictor step when the SR direction is the predictor direction. One can easily prove that $\mu_\ell^t \le \mu_\ell^h$, holds for $\mu_g \le \tau_1 \mu_\ell^h$ with $\mu_\ell^t = \mu_\ell^h$ if and only if $\mu_g = \tau_1 \mu_\ell^h$, where $\mu_\ell^h = \left(\frac{n}{\left(x^{\frac{-\log n}{2}}\right)^T s^{\frac{-\log n}{2}}} \right)^{\frac{2}{\log n}}$ is the generalized harmonic mean of the components of the vector xs, see [57].

Although in our SR-PC algorithm the neighborhood is defined by $\Phi_l(x,s,\mu)$, we develop a family of SR-PC-IPMs, where

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

is used to define the SR search directions. Due to the specific choice of the SR-proximity function, system (16) has the form given in (23). The duality gap prediction for different search directions and different target μ values are discussed in Section 2.2.

In the sequel we present our SR-PC algorithm. At each iteration the algorithm has a predictor step and a corrector step. In the predictor step it makes either an adaptive SR step, or an affine scaling step in order to reduce the duality gap as much as possible while staying in the given large neighborhood. In the corrector step it recenters to a smaller neighborhood. In the predictor step the decrease of the duality gap for the SR and the affine scaling steps is compared. If the reduction of the duality gap for the affine scaling step is bigger than the one theory guarantees for the SR step, then it makes an affine scaling step, otherwise it makes an adaptive SR step. With this adaptive choice of the predictor step the best known polynomial iteration complexity of large-update SR-IPMs is preserved, and the SR-PC algorithm is enhanced with the superlinear convergence of the MTY predictor-corrector IPM, see [39]. The SR-PC algorithm is outlined as Algorithm 3.

4.1 The Corrector Step

In this subsection we present the estimation of the decreasing of the proximity function in the corrector step when μ_ℓ^* (see Lemma 4.1) is the target value. Here we also give an upper bound for the step size that guarantees a sufficient reduction of the proximity function. Due to space limitations the proofs are omitted.

Theorem 4.2 [58] *Suppose that the present iterate is in the neighborhood* $\mathcal{N}_\ell(n,\tau)$ *and* $(\Delta x, \Delta y, \Delta s)$ *is the solution of (23) with* $\mu = \mu_\ell^*$. *Then for any*

$$\alpha \leq \alpha_1^* := \frac{\exp(-1)\exp(\frac{2(1-\tau)}{\log n+2})}{4\theta_2(\tau)\sigma_\ell(\log n)\exp(\frac{\log n+1-q}{2})}$$

we have

$$\Phi_\ell(x(\alpha), s(\alpha), \mu_\ell^*) \leq \Phi_\ell(x, s, \mu_\ell^*) - \max\{K1, K2\},$$

where

$$K_1 = \frac{\exp(-1)\exp(\frac{2(1-\tau)}{\log n+2})\sigma_\ell}{4\theta_2(\tau)\log n\exp(\frac{\log n+1-q}{2})}, \qquad K_2 = \frac{\exp(-1)\exp(\frac{2(1-\tau)}{\log n+2})v(\alpha)_{\min}^{-\log n-1}}{8(\log n)\exp(\log n+1-q)}$$

and $\theta_2(\tau) = \min\left\{\exp(-\frac{1}{4})\left(\frac{1}{\theta_1(\tau)}-1\right)^{\frac{1}{2}}, \exp(1-\tau)(1-\exp(-\frac{1}{2}))\right\}$.

4.2 The Predictor Step

In this subsection we present the behavior of the search direction for the different μ values that SR-PC algorithm chooses in the predictor step. It uses $\mu = 0$ (affine scaling) whenever the reduction of the duality gap is at least as much as theory predicts for the SR step with $\mu = \mu_\ell^t$. If the reduction is not satisfactory, the algorithm makes an SR step with $\mu = \mu_\ell^t$. In what follows we present the result that specify how the step size for the affine scaling step and for the SR step is computed for $\mu = \mu_\ell^t$. The step size selection rule also guarantees that after each iteration the proximity function is bounded by a prescribed value that is related to the definition of the neighborhood.

Theorem 4.3 [58] *Let* $(\Delta x, \Delta y, \Delta s)$ *be the solution of system (23) where* $\mu = \mu_\ell^t$ *is the smaller positive root of equation (35), and* $1 \leq q \leq \log n + 1$. *Then for any step size*

$$\alpha \leq \alpha_2^* := \frac{\exp(-1)\exp(\frac{2(1-\tau)}{\log n+2})}{4\theta_2(\tau)\sigma_\ell(\log n)\exp(\frac{\log n+1-q}{2})},$$

the relations

$$\Phi_\ell(x(\alpha), s(\alpha), \mu_g(\alpha)) \leq \eta(n,\tau),$$

and

$$\Phi_\ell(x(\alpha), s(\alpha), \mu_t) \leq \Phi_\ell(x, s, \mu_\ell^t) - \frac{\exp(-1)\exp(\frac{2(1-\tau)}{\log n+2}\sigma_\ell)}{4\theta_2(\tau)(\log n)\exp(\frac{\log n+1-q}{2})}$$

hold.

Algorithm 3: Adaptive Large Neighborhood SR-PC-IPM

Input:
a proximity parameter $\tau > 0$ and $\eta_\ell(n,\tau) = \frac{n^\tau - n}{\log n}$;
an accuracy parameter $\varepsilon > 0$;
$(x,s) = (x^0, s^0)$ such that $\Phi_\ell(x,s,\mu_g) \le \eta_\ell(n,\tau)$;

begin
while $x^T s \ge \varepsilon$ **do**
begin
Corrector step
solve (23) with $\mu = \mu_\ell^*$ and choose a step size $\alpha = \alpha_1^*$ such that[a]

$$\Phi_\ell(x(\alpha), s(\alpha), \mu_\ell^*) \le \Phi_\ell(x,s,\mu_\ell^*) - \max\{K_1, K_2\},$$

end
begin
Predictor step
solve (34) and choose a step size α_3 such that[b]
$(x(\alpha_3), y(\alpha_3), s(\alpha_3)) \in \mathcal{N}_\ell(n,\tau)$;
if $(1-\alpha_3) \le 1 - \alpha_2^* + \alpha_2^* \frac{(\mu_\ell^t)^{\frac{q+1}{2}}}{\mu_g (\mu_q^h)^{\frac{q-1}{2}}}$,
accept the affine scaling step,
else
solve (23) with $1 \le q \le \log n + 1$
and $\mu = \mu_\ell^t$ derived from (35),
end;
determine a step size α_2 such that
$(x(\alpha_2), y(\alpha_2), s(\alpha_2)) \in \mathcal{N}_\ell(n,\tau)$,
end
$x = x + \alpha \Delta x$;
$y = y + \alpha \Delta y$;
$s = s + \alpha \Delta s$;
where $\alpha = \alpha_2$ or α_3,
end

[a] A default step size $\alpha = \alpha_1^*$ is given in Theorem 4.2, where the values of K_1 and K_2 are specified as well.
[b] The value of α_2^* is given in Theorem 4.3.

Remark 4.4 *In Algorithm 3 the step sizes α_2^* and α_3^* are the minimum step sizes warranted by theory, while α_2 and α_3 are the actual calculated step sizes at the given iterate.*

Now we consider the case when the affine scaling direction is used for the predictor step of the algorithm. We first give the following lemma.

Lemma 4.5 [58] *Let (x^+, y^+, s^+) be the iterate after a corrector step and $\mu = \mu_\ell^*$. Then we have*

$$\|v\| \leq e^{(\tau-1)}\sqrt{n}.$$

Proof:

$$\|v\|^2 = \frac{x^T s}{\mu_\ell^*} \leq \frac{\mu_g^+}{(\mu_\ell^*)^+} n = \frac{n}{\theta_1(\tau)} \leq n^{\frac{2(\tau-1)}{\log n+2}} n,$$

where $\theta_1(\tau)$ is specified in Lemma 4.1. This completes the proof. □

The following theorem specifies a "sufficiently big" step size which guarantees that after the affine scaling step we are still in the neighborhood.

Theorem 4.6 [58] *Let (x^+, y^+, s^+) be the iterate after a corrector step, and $(\Delta x, \Delta y, \Delta s)$ be the solution of system (34) with $(x, y, s) = (x^+, y^+, s^+)$. Then the step size*

$$\alpha_3^* = \frac{\rho_0 \exp(\frac{2(1-\tau)}{\log n+2})}{\exp(\tau)\sqrt{n}(\log n + 1)\exp(\frac{\log n+1-q}{2})}$$

is strictly feasible, and the relation

$$\Phi_\ell(x(\alpha_3^*), s(\alpha_3^*), \mu_g(\alpha_3^*)) \leq \eta_l(n, \tau) \tag{36}$$

holds, i.e., after the predictor step $(x(\alpha_3^*), y(\alpha_3^*), s(\alpha_3^*)) \in \mathcal{N}_\ell(n, \tau)$ holds.

Corollary 4.7 [58] *If $q = \log n + 1$ or $q = 1$, then*

$$\alpha_3^* \geq \frac{\rho_0 \exp(\frac{2(1-\tau)}{\log n+2})}{2\exp(\tau)\sqrt{n}\log n} \quad or \quad \alpha_3^* \geq \frac{\rho_0 \exp(\frac{2(1-\tau)}{\log n+2})}{2\exp(\tau) n \log n},$$

respectively.

Remark 4.8 *If in the predictor step we make an affine scaling step, then the duality gap reduction is at least as large as the duality gap reduction of the SR step. Hence, for the worst case iteration complexity it suffices to consider that in all iterations we are making a SR step.*

Finally, using the previous results and some other technical results the following theorem can be proved, see [58] for more details.

Theorem 4.9 [58] *The SR-PC IPM algorithm terminates after at most*

$$\left\lceil \frac{1}{\theta_3(\tau)} \log \frac{2n \exp(2(\tau-1))}{\varepsilon} \right\rceil$$

iterations with a solution for which $x^T s \leq \varepsilon$, where

$$\theta_3(n, \tau) = \frac{2\theta_2(\tau)\tau \log n \exp(\frac{3-q+\log n}{2}) n^{\frac{1}{2}}}{\exp\left(\frac{2(1-\tau)}{2+\log n}\right)(\tau-1)^{\frac{1}{2}}}.$$

Proof: At each predictor step the duality gap reduces by $(1-\alpha_3^*)$, i.e.,

$$\mu_g(\alpha) = \frac{x^+(\alpha)s^+(\alpha)}{n} \leq (1-\alpha_3^*)\mu_g,$$

this completes the proof of the theorem. □

Corollary 4.10 [58] *If $q = \log n + 1$ or $q = 1$, then the maximum number of iterations are*

$$O\left(\sqrt{n}\log n \log \frac{(x^0)^T s^0}{\varepsilon}\right) \quad or \quad O\left(n \log n \log \frac{(x^0)^T s^0}{\varepsilon}\right),$$

respectively.

4.3 Superlinear Convergence

In this section we prove the superlinear convergence of Algorithm 3 (see [58]). For monotone Linear Complementarity Problems (LCP), Ye and Anstreicher [70] prove that in a predictor-corrector algorithms one has

$$|\Delta x_i \Delta s_i| = \mathcal{O}(\mu_g^2), \ i = 1, \ldots, n, \tag{37}$$

when μ_g is sufficiently small. Since LO is a special case of monotone LCP, relation (37) is valid as well for LO when μ_g is sufficiently small. We have

$$|d_{x_i} d_{s_i}| = \frac{\Delta x_i \Delta s_i}{\mu_g} = \mathcal{O}(\mu_g).$$

Since SR-PC algorithm has multiple choices in the predictor step and it has been proved that close to the optimality affine scaling is dominant to SR step, then it suffices to prove the superlinear convergence for the affine scaling direction. The following lemma gives a lower bound for the step size in the predictor step.

Lemma 4.11 [58] *Let (x^k, y^k, s^k) be an iterate in the SR-PC algorithm. If the present duality gap is so small that (37) holds, then there exist an $r \in (0,1)$ such that the step size α used in the predictor step satisfies $\alpha \geq 1 - \mathcal{O}\left((\mu_g^k)^r\right)$.*

¿From the proof of Lemma 4.11 (see [58]), for the next iterate $(x^{k+1}, y^{k+1}, s^{k+1})$ it follows that

$$\mu_g^{k+1} = (1-\alpha)\mu_g^k = \mathcal{O}\left((\mu_g^k)^{r+1}\right).$$

Our superlinear convergence result is the following.

Theorem 4.12 [58] *Let the iterate (x^k, y^k, s^k) be generated by the SR-PC algorithm. When μ_g is sufficiently small, the algorithm is superlinearly convergent, and any accumulation point of the iterate is a strictly complementary optimal solution of the problem.*

5 SR-Proximity Based Infeasible IPMs

In this section we present a SR-infeasible IPM (IIPM) based on $\Gamma_{13}(t)$ as a kernel function, see [56]. The dynamic algorithm idea of Peng and Terlaky [51] has been used in [56] to develop a SR-proximity based IIPM.

For given $x, s > 0$, the Newton direction for (2) in our IIPM is determined by the following linear system of equations:

$$\begin{aligned} A\Delta x &= -r_b, \\ A^T\Delta y + \Delta s &= -r_c, \\ s\Delta x + x\Delta s &= \mu e - xs, \end{aligned} \tag{38}$$

where r_b and r_c are residuals defined by

$$\begin{aligned} r_b &:= Ax - b, \\ r_c &:= A^T y + s - c. \end{aligned}$$

Note that, due to the specific choice of the kernel function $\psi(t) = \Gamma_{13}(t)$, we can rewrite the Newton system as

$$\begin{aligned} A\Delta x &= -r_b, \\ A^T\Delta y + \Delta s &= -r_c, \\ s\Delta x + x\Delta s &= \mu^2 x^{-1}s^{-1} - xs. \end{aligned} \tag{39}$$

Let us denote by $(\Delta x(\mu), \Delta y(\mu), \Delta s(\mu))$ the solution of system (39). By using Corollary 2.12 one can show that if the targeted parameter is μ_3^h (see Lemma 2.8), 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, i.e.,

$$(x + \alpha\Delta x(\mu^h))^T(s + \alpha\Delta s(\mu^h)) = x^T s\left(1 - \alpha + \frac{\mu^h\alpha}{\mu_g} + \alpha^2\frac{\Delta x^T\Delta s}{x^T s}\right).$$

Another important factor in designing the algorithm is how to define the neighborhood. In the sequel we introduce the SR-proximity based neighborhood that is different from the one used in feasible IPMs. The definition of the SR-infeasible neighborhood is as follows

$$\mathcal{N}_{\mathcal{I}}(\tau,\beta) := \left\{(x,y,s) \ : \ \Psi_3(v) \le \frac{(\tau-1)n}{2}, \ \|r_b\| \le \left\|r_b^0\right\|\frac{\mu_g}{\mu^0}\beta, \ \|r_c\| \le \left\|r_c^0\right\|\frac{\mu_g}{\mu^0}\beta\right\}, \tag{40}$$

where (x^0, y^0, s^0) is an arbitrary triple with $x^0, s^0 > 0$, $\mu^0 = \mu_g^0$, and $\beta \ge 1$ so that the initial point (x^0, y^0, s^0) is in the neighborhood $\mathcal{N}_{\mathcal{I}}(\tau,\beta)$. Here we impose that $\tau \ge 10$, that is necessary for deriving some estimations in the complexity analysis. Peng et al. [56] show that if $(x, y, s) \in \mathcal{N}_{\mathcal{I}}(\tau,\beta)$, then infeasibility is bounded by a multiple of μ and by the initial infeasibility.

The IIPMs that are based on the infeasible neighborhood $\mathcal{N}_{\mathcal{I}}(\tau,\beta)$ and that explore the idea of dynamic IPMs given in [51], are presented in Algorithm 4. Since the complexity analysis of the algorithm is analogue to the algorithm for feasible IPMs, we omit the details here. The following theorem give the iteration complexity for the SR-IIPM algorithm.

Theorem 5.1 *Let* $\tau \geq 10$, *and* $t_0 = \max\left(1, \frac{\|(r_b^0, r_c^0)\|}{\mu_0}\right)$. *Then the SR-IIPM algorithm will terminate after at most*

$$\mathcal{O}\left(n^2 \log \frac{n(\tau+1)t_0}{\epsilon}\right)$$

iterations with a solution satisfying $x^T s \leq \epsilon$ *and* $\|(r_b, r_c)\| \leq \epsilon$.

Although the new algorithm does not improve the iteration complexity, it is comparable to some other softwares i.e., LIPSOL and OSL. For the computational results see [56].

Algorithm 4: SR-IIPM Algorithm

Input:
proximity parameters $\tau \geq 10$ and $\beta \geq 1$;
neighborhood $\mathcal{N}_{\mathcal{I}}(\tau, \beta)$;
an accuracy parameter $\varepsilon > 0$;
$(x^0, y^0, s^0) \in \mathcal{N}_{\mathcal{I}}(\tau, \beta)$;
begin
while $\max\left\{x^T s, \|r_b\|, \|r_c\|\right\} \geq \epsilon$ **do**
begin
if $\frac{\mu_g}{\mu^h} \geq \frac{\tau}{2}$ then $\mu := \mu^h$;
else
$\mu := \mu_t$ is a root of (26) where $q = 3$,
end;
solve system (39) for $(\Delta x, \Delta y, \Delta s)$;
begin
determine a step size α such that[a]
$\Phi(x(\alpha), s(\alpha), \mu_t) \leq \Phi(x, s, \mu_t) - \frac{\alpha^*}{2}\Phi(x, s, \mu_t)$
and $(x(\alpha), y(\alpha), s(\alpha)) \in \mathcal{N}_{\mathcal{I}}(\tau, \beta)$;
$x := x + \alpha\Delta x$;
$y := y + \alpha\Delta y$;
$s := s + \alpha\Delta s$,
end
end
end

---

[a]The value of α^* is given in Corollary 6.4 [56].

6 Non-SR Functions

SR-proximity based IPMs give us almost $\sqrt{n}$ reduction of the worst case iteration complexity of large neighborhood IPMs. Motivated by these results Bai et al. [8, 9, 10] identify some new kernel functions that are not SR, but share some common properties with them, and provide similar polynomial complexity bounds as given for SR-IPMs. They modify the SR conditions

from Definition 2.1 in order to make the analysis of the algorithm easier. They also show that large classes of SR kernel functions $\Gamma_{p,q}(t)$ satisfy the modified conditions.

Although the analysis based on the new kernel functions proposed by Bai et al. [8, 9, 10] is much simpler than the original analysis for general SR functions, it is not applicable for such general class of kernel functions as SR proximity based complexity analysis does. Here we list the non-SR kernel functions proposed in [8, 9, 10], and some of their properties. The interested reader can find the detailed analysis of IPMs based on non-SR kernel functions in [9, 10].

In [9], Bai et al. replace condition (17) from Definition 2.1 by the following set of conditions

$$\psi^{'''}(t) < 0, \tag{41}$$

$$2\psi^{''}(t)^2 - \psi^{'}(t)\psi^{'''}(t) > 0, \qquad t < 1, \tag{42}$$

$$\psi^{''}(t)\psi^{'}(\beta t) - \beta\psi^{'}(t)\psi^{''}(\beta t) > 0, \qquad t > 1,\ \beta > 1. \tag{43}$$

Note that condition (42) is also satisfied for $t \geq 1$ since $\psi^{'}(t) \geq 0$ and $\psi^{'''}(t) \leq 0$ when $t \geq 1$, and that SR.2 and (42) are conditions on the barrier behavior of $\psi(t)$. Condition (43) only deals with $t > 1$. Thus, it concerns the growth behavior of $\psi(t)$. The connection between conditions (41) and (43) is given by the following lemma.

Lemma 6.1 [9] *If $\psi(t)$ satisfies (41) and*

$$t\psi^{''}(t) - \psi^{'}(t) \geq 0,$$

then $\psi(t)$ satisfies (43).

The non-SR kernel functions introduced in [8, 9, 10], are listed in the first column of Table 1. In the second column of Table 1 are given the iteration complexities of IPMs based on these functions. The first three listed functions in Table 1 satisfy conditions (10)–(12) and (41)–(43), and permit good iteration bounds.

Table 1. Non-SR kernel functions ($q > 1$, $b > 0$)

$\psi_i(t)$	Iteration Bound
$t - 1 + \frac{t^{1-q}-1}{q-1}$	$\mathcal{O}(qn \log \frac{n}{\varepsilon})$
$\frac{t^2-1}{2} + e^{\frac{1}{t}-1} - 1$	$\mathcal{O}(\sqrt{n} \log^2 n \log \frac{n}{\varepsilon})$
$\frac{t^2-1}{2} - \int_1^t e^{\frac{1}{\zeta}-1} d\zeta$	$\mathcal{O}(\sqrt{n} \log^2 n \log \frac{n}{\varepsilon})$
$\frac{t^2-1}{2} + \frac{e^{\gamma(1-t)}-1}{\gamma}$	$\mathcal{O}(\sqrt{n} \log n \log \frac{n}{\varepsilon})$

The first kernel function given in Table 1 differs from all other kernel functions since its growth term (i.e., $t - 1$) is linear in t. This function was first introduced and analyzed in [8]. The iteration complexity of large-update IPMs based on that function for specific q, is increasing with the increase of q, i.e., the complexity is $O(qn \log \frac{n}{\epsilon})$. Note that if $q = O(1)$, then this iteration bound is the same as the bound for the logarithmic barrier function.

The second function in Table 1 is the limit of the following sequence of functions

$$\psi_k(t) = \frac{t^2-1}{2} + (1 + \frac{1}{k})^{1-k} \left((1 + \frac{1}{kt})^k - (1 + \frac{1}{k})^k \right), \quad k = 1, 2, \ldots.$$

By using Lemma 2.1.2 from [48], one can show that $\psi_k(t)$ is a SR function for every $k \geq 1$. Furthermore, for any fixed $t > 0$, one has

$$\lim_{k \longrightarrow \infty} \psi_k(t) = \frac{t^2 - 1}{2} + e^{\frac{1}{t} - 1} - 1.$$

This result implies that this non-SR function is the limit point of a sequence of SR functions. Hence, the cone of SR functions is not a closed cone. Note that the large-update algorithm based on the second non-SR function from Table 1, has the iteration bound $\mathcal{O}(\sqrt{n} \log^2 n \log \frac{n}{\varepsilon})$.

The third and the second functions from Table 1, are closely related. Namely, if we integrate the approximation for the exponential part of the second function, we have a sequence of SR functions that converge to the exponential part of the third function. Therefore, this function is on the boundary of the cone of SR functions. The iteration complexity of large-update IPMs based on that function is $\mathcal{O}(\sqrt{n} \log^2 n \log \frac{n}{\varepsilon})$, that is a factor $\log n$ weaker than the best complexity bound of large-update IPMs based on the $\Gamma_{1 \log n}(t)$ SR-kernel function.

In [10], Bai et al. define a family of kernel functions having finite values at the boundary of the feasible region, i.e., these are not barrier functions. This property distinguishes them from all other kernel functions to date. For $\gamma > 0$, the last function in Table 1 belongs to that family. Here we list the properties of that function:

$$1.\ \lim_{t \to 0} \psi(t) = \psi(0) < \infty, \qquad 2.\ \psi^{''}(t) > 1 \ \text{ for } \ t > 0, \qquad 3.\ \psi'(t) + t\psi''(t) > 0 \ \text{ for } \ t \geq \frac{1}{\gamma}.$$

The first property, specify the finite value of the function at the boundary of the feasible region. Because of the second condition, we say that function $\psi(t)$ is uniformly strictly convex, see [46]. The third condition guarantees that $\psi(t)$ satisfies the SR.2 condition on the interval $[\frac{1}{\gamma}, \infty)$, see Lemma 2.3.

We point out that the forth function in Table 1, for $\gamma > 0$, is in a way a first order approximation of the following SR function:

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

where $q = 1 + \gamma$. Namely,

$$\Gamma_{1(1+\gamma)}(t) = \frac{t^2 - 1}{2} + \frac{t^{-\gamma} - 1}{\gamma} = \frac{t^2 - 1}{2} + \frac{e^{(-\gamma \log t)} - 1}{\gamma} \approx \frac{t^2 - 1}{2} + \frac{e^{\gamma(1-t)} - 1}{\gamma},$$

where the approximation in the exponent is due to the first order Taylor expansion

$$\log t \approx t - 1,$$

of the logarithm. Following this idea, we believe that other finite non-SR kernel functions can be derived as an approximation of SR functions.

7 Primal-dual IPMs for SDO Based on SR–Proximities

Semidefinite Optimization (SDO) is an extension of LO where the nonnegativity constraints are replaced by positive semidefiniteness on the matrix variables. SDO has a wide range of

applications in combinatorial optimization [22, 33, 73] and in control theory [14, 18, 66], as well as in engineering fields [7, 34, 36, 66]. Practice shows that semidefinite models for combinatorial optimization problems are sometimes significantly stronger than purely linear ones. Nowadays, IPMs provide a most powerful approach for solving SDO problems. Most IPMs for SDO can be viewed as natural extensions of the IPMs for LO, resulting with similar polynomial complexity results. Here we generalize the approach of the previous sections to the case of SDO, see [48].

We consider the SDO problem in the form

$$\begin{aligned} \text{(PSDO)} \qquad & \min \ \mathrm{Tr}(CX) \\ & \mathrm{Tr}(A_j X) = b_j, \quad j = 1, \ldots, m, \\ & X \succeq 0, \end{aligned}$$

where C and A_j $(1 \leq j \leq m)$ are symmetric $n \times n$ matrices, $b := (b_1, \ldots, b_m)^T \in I\!R^m$, and $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. The notation '$X \succeq 0$' (resp. '$X \succ 0$') means that X is symmetric positive semidefinite (resp. positive definite). The matrices A_i are assumed to be linearly independent. Let S be a dual slack matrix. Then, the dual problem of (PSDO) is

$$\begin{aligned} \text{(DSDO)} \qquad & \max \ b^T y \\ & \sum_{j=1}^{m} y_j A_j + S = C, \\ & S \succeq 0, \end{aligned}$$

where $y \in I\!R^m$.

The concept of the central path can also be generalized from LO to SDO. This was first done by Nesterov and Nemirovski [43] by introducing the extended logarithmic barrier function $\log\det(X)$ for the positive semidefinite constraint. Independently, Alizadeh [3] applied IPMs to solving SDO problems arising from combinatorics. Followed by their approach, the *perturbed optimality conditions* are

$$\begin{aligned} \mathrm{Tr}(A_j X) = \ & b_j, \ X \succeq 0, \quad j = 1, \ldots, m, \\ \sum_{j=1}^{m} y_j A_j + S = \ & C, \ \ S \succeq 0, \\ & XS = \mu I, \end{aligned} \tag{44}$$

where $\mu > 0$, and I denotes the $n \times n$ identity matrix. We may assume without loss of generality (see [29, 30]) that strict feasibility holds for both (PSDO) and its dual (DSDO). Under this assumption for every $\mu > 0$ there exists a unique solution $(X(\mu), y(\mu), S(\mu))$ of system (44), see [43]. The central path for SDO is defined by the set $\{(X(\mu), y(\mu), S(\mu)) : \mu > 0\}$. The basic idea of IPMs is to follow this central path and approach the optimal set of SDO by letting μ go to zero. Suppose that the point (X, y, S) is strictly feasible. Newton's method amounts to linearizing system (44), thus yielding the following equation:

$$\begin{aligned} \mathrm{Tr}(A_j \Delta X) = 0, & \qquad j = 1, \ldots, m, \\ \sum_{j=1}^{m} \Delta y_j A_j + \Delta S = 0, & \\ X\Delta S + \Delta X S = \mu I - XS. & \end{aligned} \tag{45}$$

A *crucial observation* for SDO is that the Newton system (45) might not have a symmetric solution ΔX. Many authors have suggested several ways for symmetrizing the third equation in (45) so that the resulting new system has a unique symmetric solution, see e.g., [25, 31, 41, 62]. In [62] Todd analyzes more than twenty different search directions for SDO. Among others, the most popular directions are: the Alizadeh, Haeberly, Overton (**AHO**) direction introduced in [4], the search direction independently proposed by Helmberg, Rendl, Vanderbei and Wolkowicz [25], and Kojima, Shindoh and Hara [31], and later rediscovered by Monteiro [41], which we refer to as the **H..K..M** direction, and the Nesterov and Todd (**NT**) direction [44, 45].

Peng et al. [48] consider the symmetrization scheme from which the **NT** direction [44, 45, 63] is derived. The important reason for this is that the **NT** scaling technique transfers the primal variable X and the dual S into the same space; the so-called *V–space*. Let

$$P = X^{\frac{1}{2}}(X^{\frac{1}{2}}SX^{\frac{1}{2}})^{-\frac{1}{2}}X^{\frac{1}{2}} = S^{-\frac{1}{2}}(S^{\frac{1}{2}}XS^{\frac{1}{2}})^{\frac{1}{2}}S^{-\frac{1}{2}}, \tag{46}$$

and $D = P^{\frac{1}{2}}$, where for any symmetric positive definite matrix G, the exponent $G^{\frac{1}{2}}$ denotes its symmetric square root. Now, the matrix D can be used to rescale X and S to the same matrix V defined by (see [29, 61])

$$V := \frac{1}{\sqrt{\mu}}D^{-1}XD^{-1} = \frac{1}{\sqrt{\mu}}DSD. \tag{47}$$

Note that $D, V \succ 0$. Using the above notation one can state the *centrality condition* as $V = E$, and the *duality gap* as $\mu\mathrm{Tr}(V^2)$. Let us further define

$$\begin{aligned} \bar{A}_i &:= \frac{1}{\sqrt{\mu}}DA_iD, \qquad\qquad i = 1, \ldots, m, \\ D_X &:= \frac{1}{\sqrt{\mu}}D^{-1}\Delta XD^{-1}, \quad D_S := \frac{1}{\sqrt{\mu}}D\Delta SD. \end{aligned} \tag{48}$$

Then the *scaled* **NT** *search direction* $(D_X, \Delta y, D_S)$ is obtained from the system (see also [44, 61])

$$\begin{aligned} \mathrm{Tr}(\bar{A}_jD_X) &= 0, \qquad j = 1, \ldots, m, \\ \sum_{j=1}^{m}\Delta y_i\bar{A}_j + D_S &= 0, \\ D_X + D_S &= V^{-1} - V. \end{aligned} \tag{49}$$

The solution of system (49) is unique, and the unscaled direction can be easily derived from (48).

Here, analogous to the LO case, new search directions can be derived from SR kernel functions. Peng et al. [48, 47] show that the **NT**-direction is induced by the kernel function of the logarithmic barrier function. Now we explain their approach and derive SR search directions based on SR-kernel functions.

First we provide some basic results of matrix analysis [11]. Let ψ be any kernel function. The definition of ψ can be extended to any diagonalizable matrix with positive eigenvalues, see [11]. In particular, given an eigen-decomposition

$$V = Q_V^{-1}\mathrm{diag}\,(\lambda_1(V), \lambda_2(V), \ldots, \lambda_n(V))\,Q_V,$$

of V, with Q_V nonsingular[1], the matrix function $\psi(V)$ is defined by

$$\psi(V) = Q_V^{-1}\text{diag}\,(\psi(\lambda_1(V)), \psi(\lambda_2(V)), \ldots, \psi(\lambda_n(V)))\, Q_V. \tag{50}$$

Since $\psi(t)$ is twice continuously differentiable, the derivatives $\psi'(t)$ and $\psi''(t)$ are well-defined for $t > 0$. Hence, replacing $\psi(\lambda_i(V))$ in (50) by $\psi'(\lambda_i(V))$ and $\psi''(\lambda_i(V))$ respectively, for each i, we obtain the matrix functions $\psi'(V)$ and $\psi''(V)$.

Remark 7.1 *Further on, when we use function $\psi(\cdot)$ and its derivatives $\psi'(\cdot)$ and $\psi''(\cdot)$, these denote matrix functions if the argument is a matrix and a univariate function if the argument is in $\mathbb{R}$.*

We give now the notation of SR functions [48] on the cone of positive definite matrices $\mathcal{S}_{++}^{n\times n}$.

Definition 7.2 *A matrix function $\psi(V)$ given by (50) is SR on $\mathcal{S}_{++}^{n\times n}$ if the kernel function $\psi(t)$ is SR.*

Peng et al. [47, 48] define the *proximity measure* for SDO in the following way:

$$\Phi(X, S, \mu) := \Psi(V) := \text{Tr}(\psi(V)) = \sum_{i=1}^{n} \psi(\lambda_i(V)), \tag{51}$$

where $\psi(V)$ is given by (50). Note that $\Psi(V) = 0$ if and only if V is the identity matrix, i.e., $XS = \mu I$, and otherwise $\Psi(V) > 0$, due to the properties of an SR-kernel function.

Remark 7.3 *From now on, we assume that $\psi(\cdot)$ is SR. Hence, $\Phi(X, S, \mu)$ is a SR-proximity function.*

Analogous to the LO case, using the NT scaling, the *new search direction* is given by the following system (see [47, 48])

$$\begin{aligned} \text{Tr}(\bar{A}_j D_X) &= 0, \qquad\qquad j = 1, \ldots, m, \\ \sum_{j=1}^{m} \Delta y_j \bar{A}_j + D_S &= 0, \\ D_X + D_S &= -\psi'(V). \end{aligned} \tag{52}$$

Having D_X and D_S, ΔX and ΔS can be calculated from (48). Note that the new search direction is a slight modification of the NT direction. The orthogonality of ΔX and ΔS follows from the orthogonality of D_X and D_S, i.e.,

$$\text{Tr}(\Delta X \Delta S) = \text{Tr}(D_X D_S) = \text{Tr}(D_S D_X) = 0.$$

One may easily verify that if $\psi(t)$ is the kernel function of the logarithmic barrier function, then $\Phi(X, S, \mu)$ coincides with the classical primal-dual logarithmic barrier function for SDO, and the search direction is then precisely the **NT**-direction. In the next subsection we deal with properties of SR-proximity functions.

[1]The matrix Q_V is not unique, but $\psi(V)$ is well defined whenever $\psi(t)$ is well defined on the eigenvalues of V [11, page 90]. Note that since V is symmetric, we can choose Q_V to be orthogonal, i.e., $Q_V^{-1} = Q_V^T$.

7.1 Properties of SR–Proximity Functions for SDO and the Algorithm

The following proposition gives properties of SR–proximity functions that are crucial for the analysis of SR–IPMs for SDO.

Proposition 7.4 [48] *Let the functions $\psi : \mathcal{S}_{++}^{n\times n} \to \mathcal{S}_{++}^{n\times n}$ and $\Psi : \mathcal{S}_{++}^{n\times n} \to I\!R$ be defined by (50) and (51), respectively. If $\psi(t)$ is a SR function, then*

(i) $\Psi(X)$ is strictly convex with respect to $X \succ 0$ and vanishes at its global minimal point $X = E$, i.e., $\Psi(E) = 0, \psi'(E) = 0_{n\times n}$. Further, there exist two constants $\nu_1, \nu_2 > 0$ such that

$$\nu_1(X^{p-1} + X^{-1-q}) \preceq \psi''(X) \preceq \nu_2(X^{p-1} + X^{-1-q}), \quad p, q \geq 1.$$

(ii) For any $X_1, X_2 \succ 0$,

$$\Psi\left([X_1^{\frac{1}{2}} X_2 X_1^{\frac{1}{2}}]^{\frac{1}{2}}\right) \leq \frac{1}{2}\left(\Psi(X_1) + \Psi(X_2)\right).$$

Statements *(i)* and *(ii)* of Proposition 7.4 can be viewed as transparent extensions of conditions SR1 and SR2 introduced in Section 2, respectively.

The following proposition provides some more properties of SR proximities for SDO.

Proposition 7.5 [48] *Let the kernel function $\psi(t)$ be SR, the proximity $\Psi(V)$ defined by (51), and*

$$\sigma := \sqrt{\mathrm{Tr}(\psi'(V)^2)} = \|\psi'(V)\|. \tag{53}$$

If the kernel function $\psi(t)$ satisfies condition SR1 (Definition 2.1, page 6), then

$$\Psi(V) \leq \frac{\sigma^2}{2\nu_1}, \tag{54}$$

$$\lambda_{\min}(V) \geq \left(1 + \frac{q\sigma}{\nu_1}\right)^{-\frac{1}{q}}, \tag{55}$$

$$\lambda_{\max}(V) \leq \left(1 + \frac{p\sigma}{\nu_1}\right)^{\frac{1}{p}}. \tag{56}$$

Statements (54)–(56) of Proposition 7.5 can be viewed as extensions of conditions derived for the case of LO (see Proposition 3.1.5, page 50 in [48]), where $v_i, v_{\max},$ and $v_{\min}$ are replaced by $\lambda_i(V), \lambda_{\max}(V)$, and $\lambda_{\min}(V)$, respectively.

The following corollary gives the relationships between the scaled duality gap $\|V\|^2$ and the proximity $\Psi(V)$.

Corollary 7.6 *Let the kernel function $\psi(t)$ be SR, and let the proximity $\Psi(V)$ be defined by (51). If the kernel function $\psi(t)$ satisfies condition SR1 (Definition 2.1, page 6), then*

$$\mathrm{Tr}(XS) = \mu \|V\|^2 \leq \mu \left(\sqrt{n} + \sqrt{\frac{2\Psi(V)}{\nu_1}}\right)^2. \tag{57}$$

Inequality (57) shows that the proximity yields an upper bound for the duality gap. Hence, it can be used as a potential function for minimizing the duality gap. The new algorithm for SDO (see Algorithm 5 and algorithm on page 118 in [48]) has the same structure as the algorithm for LO (see Algorithm 1 on page 3).

Remark 7.7 Algorithm 5 *terminates with a point satisfying* $n\mu < \varepsilon$. *By using (57), we obtain that*

$$\mathrm{Tr}(XS) < \mu \left(\sqrt{n} + \sqrt{\frac{2\tau}{\nu_1}} \right)^2 .$$

Hence if $\tau = \mathcal{O}(n)$, *which means that the algorithm works indeed in a large neighborhood of the central path, then the algorithm finally reports a feasible solution such that* $\mathrm{Tr}(XS) = \mathcal{O}(\varepsilon)$.

7.2 Complexity of the Algorithm

As in the LO case, the key issue in the analysis of the algorithm is to estimate the decrease of the proximity during one step. In order to do that we introduce the following notation. Let $X_+ = X + \alpha\Delta X$, $S_+ = S + \alpha\Delta S$, and

$$V_+ = \frac{1}{\sqrt{\mu}} D^{-1} X_+ D^{-1} = \frac{1}{\sqrt{\mu}} D S_+ D,$$

for $D = P^{\frac{1}{2}}$ and P as it is given in (46). It is trivial to verify that V_+^2 is unitary similar to the matrix $X_+^{\frac{1}{2}} S_+ X_+^{\frac{1}{2}}$ and thus to $(V + \alpha D_X)^{\frac{1}{2}} (V + \alpha D_S) (V + \alpha D_X)^{\frac{1}{2}}$. This further implies that the eigenvalues of V_+ are precisely the same as those of the matrix

$$\tilde{V}_+ := \left((V + \alpha D_X)^{\frac{1}{2}} (V + \alpha D_S) (V + \alpha D_X)^{\frac{1}{2}} \right)^{\frac{1}{2}} . \tag{58}$$

Algorithm 5: Large-Update Primal-Dual Algorithm for SDO

Input:
a proximity parameter $\tau \geq \nu_1^{-1}$;
an accuracy parameter $\varepsilon > 0$;
a fixed barrier update parameter $\theta \in (0, 1)$;
$(X \succ 0,\ S \succ 0)$ and $\mu = 1$ with $\Phi(X, S, \mu) \leq \tau$;

begin
while $n\mu \geq \varepsilon$ **do**
begin
$\mu := (1 - \theta)\mu$;
while $\Phi(X, S, \mu) \geq \tau$ **do**
begin
solve system (52) for $\Delta X, \Delta y, \Delta S$;
determine a step size α;
$X := X + \alpha\Delta X$;
$y := y + \alpha\Delta y$;
$S := S + \alpha\Delta S$,
end
end
end

Since the proximity after one step is defined by $\Psi(V_+)$, it follows immediately from the definition of the proximity measure (51) that

$$\Psi(V_+) = \Psi\left(\tilde{V}_+\right). \tag{59}$$

The decrease in the proximity during one step is considered as a function of the step size α, i.e.,

$$f(\alpha) := \Psi(V_+) - \Psi(V) = \Psi(\tilde{V}_+) - \Psi(V). \tag{60}$$

First that we want to know is how far we can go along the search direction, i.e., the maximal value $\alpha_{\max}$ of the step size. Let

$$\bar{D}_X = V^{-\frac{1}{2}} D_X V^{-\frac{1}{2}}, \quad \bar{D}_S = V^{-\frac{1}{2}} D_S V^{-\frac{1}{2}}. \tag{61}$$

Since

$$V + \alpha D_X = V^{\frac{1}{2}}(I + \alpha\bar{D_X})V^{\frac{1}{2}}, \quad V + \alpha D_S = V^{\frac{1}{2}}(I + \alpha\bar{D_S})V^{\frac{1}{2}},$$

it follows that

$$V + \alpha D_X \succ 0 \ \Leftrightarrow\ E + \alpha\bar{D}_X \succ 0 \quad \& \quad V + \alpha D_S \succ 0 \ \Leftrightarrow\ E + \alpha\bar{D}_S \succ 0.$$

Thus, the maximal feasible step size is dependent on the eigenvalues of the matrices $\bar{D}_X$ and $\bar{D}_S$. The next result gives some estimates of the norms of the matrices $\bar{D}_X$ and $\bar{D}_S$, and thus the maximal feasible step size $\alpha_{\max}$ as well.

Lemma 7.8 [48] *Let the matrices $\bar{D}_X$ and $\bar{D}_S$ be defined by (61) and σ by (53). Let $\alpha_{\max}$ be the maximal feasible step size. Then*

$$||\bar{D}_X||^2 + ||\bar{D}_S||^2 \leq \bar{\alpha}^{-2}$$

and

$$\alpha_{\max} \geq \bar{\alpha},$$

where

$$\bar{\alpha} = \sigma^{-1} \left(1 + \frac{q\sigma}{\nu_1}\right)^{-\frac{1}{q}}.$$

Now we evaluate the function $f(\alpha)$ defined by (60). From part (ii) of Proposition 7.4 one gets

$$f(\alpha) = \Psi(\tilde{V}_+) - \Psi(V) \leq \frac{1}{2}\left(\Psi\left(V + \alpha D_X\right) + \Psi(V + \alpha D_S)\right) - \Psi(V) := f_1(\alpha).$$

Hence it suffices for us to estimate the decrease of the value of the function $f_1(\alpha)$ after one step. The main difficulty in the estimation of the function $f_1(\alpha)$ is to evaluate its first and second derivatives. In [48] Peng et al. compute

$$f_1'(\alpha) = \frac{1}{2}\mathrm{Tr}(\psi'(V + \alpha D_X)D_X + \psi'(V + \alpha D_S)D_S),$$
$$f_1''(\alpha) = \frac{1}{2}\frac{d^2}{d\alpha^2}\mathrm{Tr}(\psi(V + \alpha D_X) + \psi(V + \alpha D_S)),$$

and report the following result.

Lemma 7.9 [48] *If $\alpha < \bar{\alpha}$, then*

$$f_1''(\alpha) \leq \frac{\nu_2\sigma^2}{2}\left((\lambda_{\max}(V) + \alpha\sigma)^{p-1} + (\lambda_{\min}(V) - \alpha\sigma)^{-q-1}\right).$$

The following corollary is a direct consequence of Lemma 7.9, and

$$f(0) = f_1(0) = 0, \quad f'(0) = f_1'(0) = -\frac{\sigma^2}{2}.$$

Corollary 7.10 *If $\alpha < \bar{\alpha}$, then*

$$f(\alpha) \leq f_1(\alpha) \leq -\frac{\sigma^2\alpha}{2} + \frac{\nu_2\sigma^2}{2}\int_0^\alpha \int_0^\xi \left((\lambda_{\max}(V) + \zeta\sigma)^{p-1} + (\lambda_{\min}(V) - \zeta\sigma)^{-q-1}\right) d\zeta d\xi. \quad (62)$$

We define now the function $f_2(\alpha)$ as the right hand side of the inequality in (62). Obviously, $f_2(\alpha)$ is convex and twice differentiable for all $\alpha \in [0, \bar{\alpha})$. It is also easy to see that $f_2(\alpha)$ is decreasing at zero and that it goes to infinity as $\alpha \to \bar{\alpha}$. Let α^* be the point at which $f_2(\alpha)$ attains its global minimal value, i.e.,

$$\alpha^* = \arg\min_{\alpha \in [0,\bar{\alpha})} f_2(\alpha). \quad (63)$$

We can now state the following result.

Theorem 7.11 [48] *Let the function $f(\alpha)$ be defined by (60) with $\Psi(V) \geq \nu_1^{-1}$. Then the step-size α given by $\alpha = \alpha^*$ (63) or $\alpha = \nu_5 \sigma^{-\frac{q+1}{q}}$ for*

$$\nu_5 := \min\left\{\frac{\nu_1}{2\nu_1\nu_2 + p(\nu_1+2\nu_2)}, \frac{\nu_1^2}{(1+\nu_1)\left(2\nu_1\nu_2 + q(\nu_1+2\nu_2)\right)}\right\},$$

is strictly feasible. Moreover,

$$f(\alpha) \leq \frac{1}{2} f'(0)\alpha \leq -\frac{\nu_5}{4}\left(\nu_1 \Psi(V)\right)^{\frac{q-1}{2q}}.$$

In the special case where $\psi(t) = \Upsilon_{p,q}(t)$, this bound (with $\nu_1 = \nu_2 = 1$) simplifies to

$$f(\alpha) \leq -\min\left(\frac{1}{12p+8}, \frac{1}{24q+16}\right)\Psi(V)^{\frac{q-1}{2q}}.$$

Since the proximity $\Psi(V)$ is determined by the eigenvalues of the matrix V, the growth behavior of the proximity $\Psi(V)$ is precisely the same as its LO counterpart $\Psi(v)$, see [48]. If the current point enters the neighborhood again, then μ is updated to $(1-\theta)\mu$ for some $\theta \in (0,1)$. Proceeding as in the LO case, one can show that after the update of μ, the proximity is still bounded above by the number

$$\psi_0(\theta,\tau,n) := \frac{\nu_2\tau}{\nu_1(1-\theta)^{\frac{p+1}{2}}} + \nu_2 \Upsilon'_{p,q}\left((1-\theta)^{-\frac{1}{2}}\right)\sqrt{\frac{2n\tau}{\nu_1(1-\theta)}} + n\nu_2\Upsilon_{p,q}\left((1-\theta)^{-\frac{1}{2}}\right). \quad (64)$$

Peng et al. [48] obtain the following bound.

Lemma 7.12 [48] *Let $\Psi(X,S,\mu) \leq \tau$ and $\tau \geq \nu_1^{-1}$. Then, after an update of the barrier parameter, no more than*

$$\left\lceil \frac{8q\nu_1^{-\frac{q-1}{2q}}}{\nu_5(q+1)}\left(\psi_0(\theta,\tau,n)\right)^{\frac{q+1}{2q}} \right\rceil$$

iterations are needed to recenter. In the special case when $\psi(t) = \Upsilon_{p,q}(t)$, this bound (with $\nu_1 = \nu_2 = 1$) simplifies to

$$\left\lceil \frac{8q\max(3p+2, 6q+4)}{q+1}\left(\psi_0(\theta,\tau,n)\right)^{\frac{q+1}{2q}} \right\rceil.$$

Finally, the complexity of the algorithm that is precisely the same as for the LO case, is stated by the following theorem.

Theorem 7.13 [48] *If $\tau \geq \nu_1^{-1}$, the total number of iterations required by the algorithm is not more than*

$$\left\lceil \frac{8q\nu_1^{-\frac{q-1}{2q}}}{\nu_5(q+1)}\left(\psi_0(\theta,\tau,n)\right)^{\frac{1+q}{2q}} \right\rceil \left\lceil \frac{1}{\theta}\log\frac{n}{\varepsilon} \right\rceil.$$

In the special case where $\psi(t) = \Upsilon_{p,q}(t)$, this bound (with $\nu_1 = \nu_2 = 1$) simplifies to

$$\left\lceil \frac{8q\max(3p+2, 6q+4)}{q+1}\left(\psi_0(\theta,\tau,n)\right)^{\frac{q+1}{2q}} \right\rceil \left\lceil \frac{1}{\theta}\log\frac{n}{\varepsilon} \right\rceil.$$

Hence, just as for LO, omitting the round off brackets in Theorem 7.13, by choosing $\theta \in (0,1)$, the iteration bound of the algorithm for SDO with large-updates is $\mathcal{O}(n^{\frac{q+1}{2q}} \log \frac{n}{\varepsilon})$, while the algorithm with small-update ($\theta = \mathcal{O}(\frac{1}{\sqrt{n}})$) has an $\mathcal{O}(\sqrt{n} \log \frac{n}{\varepsilon})$ iterations bound. Moreover, using Theorem 7.13 one can readily verify that if p is a small constant and $q = \log n$, then the new large-update algorithm has an $\mathcal{O}(\sqrt{n} \log n \log \frac{n}{\varepsilon})$ iteration bound, the currently best bound for large-update methods.

8 Primal-Dual IPMs for SOCO Based on Self-Regular Proximities

Second-Order Conic Optimization (SOCO) is a generalization of LO and a specific case of SDO, and it holds the "sandwich relation" $LO \subset SOCO \subset SDO$. More precisely, SOCO is the problem of minimizing a linear objective function subject to the intersection of an affine set and the direct product of several second-order cones. A second-order cone in $\mathbb{R}^n$ can be defined by

$$K = \{(x_1, x_2, \ldots, x_n) \in \mathbb{R}^n : x_1^2 - \sum_{i=2}^{n} x_i^2 \geq 0, x_1 \geq 0\}.$$

This cone is often referred to as the Lorentz cone in physics, but it is also known as "the ice-cream cone"[2]. Some general classes of problems, such as problems involving sums and maxima of norms, problems with hyperbolic constraints, matrix-fractional problems, and robust LO can be formulated as SOCO as well, see [37]. Moreover, SOCO has a wide range of applications in areas like antenna array weight design [35], grasping force optimization [15], FIR filter design [16, 69], truss design [2, 12], etc. For details about different applications of SOCO see papers [13, 37]. A SOCO problem can be solved by applying IPMs to the semidefinite formulation of a SOCO problem, or by applying IPMs *directly* which is showed to be a more efficient approach both in theory and practice [6, 43]. Because of the inherent relations among LO, SOCO and SDO, most theoretical results for SOCO can be viewed as a transparent extension of LO, or a specialization of the results for SDO. Here we expand the approach of the previous sections to the case of SOCO, see [48, 52].

We consider the SOCO problem in the following form:

$$\begin{aligned} \text{(PSOCO)} \qquad \min \ & c^T x \\ & Ax = b, \\ & x \succeq_K 0, \end{aligned}$$

where K is the product of several second-order cones, i.e., $K = K^1 \times K^2 \ldots \times K^N$ with

$$K^j = \left\{ (x_1^j, \ldots, x_{n_j}^j)^T \in \mathbb{R}^{n_j} : \left(x_1^j\right)^2 \geq \sum_{i=2}^{n_j} \left(x_i^j\right)^2, \ x_1^j \geq 0 \right\},$$

$b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$ for $n = \sum_{j=1}^{N} n_j$, and $c, x \in \mathbb{R}^n$ where $x^T = \left((x^1)^T, (x^2)^T, \ldots, (x^N)^T \right)$, $x^j \in \mathbb{R}^{n_j}$, $j \in \mathcal{J} := \{1, 2, \ldots, N\}$. The notation $x \succeq_K s$ (respectively $x \succ_K s$) means that

[2]This name comes from the similarity between the shape of a general second-order cone in the space $\mathbb{R}^3$ and the well known summer refreshment food.

$x - s \in K$ (respectively $x - s \in K_+$, where K_+ denotes the interior of K). Here we assume that the matrix A is of full row rank, i.e., rank $(A) = m$. Let $s \in I\!R^n$ be a dual slack variable. Then, the dual problem of (PSOCO) is

$$\begin{aligned} \text{(DSOCO)} \qquad & \max \ b^T y \\ & A^T y + s \ = \ c, \\ & s \succeq_K 0, \end{aligned}$$

where $y \in I\!R^m$.

The concept of the central path for SOCO is very similar to the concept of the central path for SDO, and we refer to [29, 60, 67] for details. Let the operator mat$(\cdot)$ be defined on $I\!R^{n_j}$, $j \in \mathcal{J}$ (see [48]) in the following way

$$\text{mat}(x^j) := \begin{pmatrix} x_1^j & x_{2:n_j}^j \\ (x_{2:n_j}^j)^T & x_1^j I_{n_j - 1} \end{pmatrix}, \quad x_{2:n_j}^j = (x_2^j, x_3^j, \ldots, x_{n_j}^j), \quad x^j \in I\!R^{n_j}, \tag{65}$$

and the *generalized* operator mat$(\cdot)$ on K as

$$\text{mat}(x) := \begin{pmatrix} \text{mat}(x^1) & \ldots & 0 \\ 0 & \ddots & 0 \\ 0 & \ldots & \text{mat}(x^N) \end{pmatrix}, \quad x \in I\!R^n.$$

Then, the *perturbed optimality conditions* for SOCO are

$$\begin{aligned} Ax = b, \quad & x \succeq_K 0, \\ A^T y + s = c, \quad & s \succeq_K 0, \\ \text{mat}(x) s = \mu \tilde{e}, & \end{aligned} \tag{66}$$

where $\mu > 0$ and

$$\tilde{e}^T = \Big((\tilde{e}^1)^T, \ldots, (\tilde{e}^N)^T \Big), \quad (\tilde{e}^j)^T = \Big(1, 0, \ldots, 0 \Big) \in I\!R^{n_j}, \ j \in \mathcal{J}.$$

Here we assume that both (PSOCO) and (DSOCO) satisfy the interior point condition, i.e., $x \succ_K 0$, $s \succ_K 0$. The central path for SOCO is defined by the solution sets $\{(x(\mu), y(\mu), s(\mu)) : \mu > 0\}$ of the system (66). It is easy to see that the linearized Newton system for (66) might not be well defined. To obtain a Newton–type system that has a *unique solution*, one usually refer to some scaling schemes. They were first proposed and studied by Tsuchiya [64, 65]. However, there are several popular choices for the scaling matrices and here we list some of them: the **AHO** search direction [1]; the primal (or dual) **H..K..M** direction, and the **NT** search direction [64, 65]. It can be shown that the linearized Newton system in the scaled space induced by **NT** scaling is *always* well-defined if both $x \succ_K 0$ and $s \succ_K 0$ are feasible for SOCO.

Peng et al. [48] consider **NT** scaling for SOCO. One of the reasons for this is that for SOCO problems a large-update IPMs based on the **NT** search direction, always has a theoretically

lower iteration bound than large-update IPMs relying on other search directions [65]. Let

$$\bar{A} := \frac{1}{\sqrt{\mu}} A(U_{NT}W_{NT})^{-1}, \qquad v := \frac{1}{\sqrt{\mu}} U_{NT}W_{NT}x;$$
$$d_x := \frac{1}{\sqrt{\mu}} U_{NT}W_{NT}\Delta x, \qquad d_s := \frac{1}{\sqrt{\mu}} \left(U_{NT}W_{NT}\right)^{-1} \Delta s,$$

where W_{NT} is the scaling matrix, and U_{NT} is the diagonal matrix chosen such that (see [48, 64, 65]) $U_{NT}W_{NT}x = (U_{NT}W_{NT})^{-1}s$. The existence of matrices W_{NT} and U_{NT} is proved in [48, 64, 65]. The important property of these matrices is that the gap between the primal and dual potential function in the scaled space, with respect to W_{NT} and U_{NT}, attains its global minimum value.

Note that $v \succ_K 0$. Using the above notation one can state the *centrality condition* as $v = \tilde{e}$. The **NT** search direction for SOCO is defined (see [6, 65]) as the unique solution of the system

$$\begin{aligned} \bar{A}d_x &= 0, \\ \bar{A}^T\Delta y + d_s &= 0, \\ d_x + d_s &= v^{-1} - v. \end{aligned} \tag{67}$$

In view of the orthogonality of Δx and Δs, one can easily verify that $d_x^T d_s = 0$. In the SOCO case, analogously to the SDO case, new search directions can be derived from SR kernel functions [48, 52]. Before we explain the approach of [48, 52] and derive SR search directions based on SR-kernel functions, we provide some basic results about functions associated with a second order cone [17, 19, 20].

Let $x^j \in I\!R^{n_j}$, $j \in \mathcal{J}$. Let $\lambda_{\max}(x^j)$ and $\lambda_{\min}(x^j)$ be the maximal and minimal eigenvalues respectively, of matrix $\mathrm{mat}(x^j)$ defined in (65). Let $\psi(t)$ be any kernel function. Then the function $\psi : I\!R^{n_j} \to I\!R^{n_j}$ associated with the second-order cone K^j is defined (see [48, 21]) by

$$\psi(x^j) := \begin{cases} \left(\frac{1}{2}(\psi(\lambda_{\max}(x^j)) + \psi(\lambda_{\min}(x^j))), \frac{\psi(\lambda_{\max}(x^j)) - \psi(\lambda_{\min}(x^j))}{2\|x^j_{2:n}\|}(x^j)^T_{2:n}\right)^T, & \text{if } x^j_{2:n} \neq 0, \\ (\psi(\lambda_{\max}(x^j), 0, \ldots, 0))^T, & \text{if } x^j_{2:n} = 0, \end{cases} \tag{68}$$

and the *generalized* function $\psi : I\!R^n \to I\!R^n$ associated with the product of second-order cones $K = K^1 \times \ldots \times K^N$, is defined by

$$\psi(x) := ((\psi(x^1))^T, \ldots, (\psi(x^N))^T)^T, \tag{69}$$

where $x = ((x^1)^T, \ldots, (x^N)^T)^T$. It can be easily verified that if $\psi(t) \geq 0$, $t \geq 0$ and $x \in K$ then $\psi(x) \in K$ follows. Thus, it becomes clear that every nonnegative (positive) function on the nonnegative (positive) axis naturally extends to a function that maps (the interior of) K into itself.

Remark 8.1 *Further on, when we use the function $\psi(\cdot)$ and its derivatives $\psi'(\cdot)$ and $\psi''(\cdot)$, these denote vector functions if the argument is a vector and a univariate function if the argument is in $I\!R$.*

We give now the notation of SR functions on the product of second–order cones K.

Definition 8.2 [48] *A function $\psi(x^j)$, associated with the second-order cone K^j, given by (68) is said to be SR if its kernel function $\psi(t)$ is SR. Analogous definition holds for the generalized function $\psi(x)$, that is given by (69).*

¿From now on, we assume that $\psi(\cdot)$ is SR. The trace of a vector $x \in I\!R^n$ associated with the second-order cone K is defined as follows.

Definition 8.3 *For any $x^j \in I\!R^{n_j}$, the* trace *of x^j associated with the second-order cone K^j is defined by*

$$\mathrm{Tr}(x^j) = \lambda_{\max}(x^j) + \lambda_{\min}(x^j), \tag{70}$$

and for any $x \in I\!R^n$, $n = \sum_{j=1}^{N} n_j$, $x^T = ((x^1)^T, \ldots, (x^N)^T)$, $x^j \in I\!R^{n_j}$ the generalized trace *of x associated with the product of second-order cones $K = K^1 \times \ldots \times K^N$ is defined by*

$$\mathrm{Tr}(x) = \sum_{j=1}^{N} (\lambda_{\max}(x^j) + \lambda_{\min}(x^j)).$$

Before describing the new search direction for SOCO, we need to define the SR–proximity measure used in the new IPMs for SOCO. Let $x^j, s^j \in I\!R^{n_j}$. Similar to the cases of LO and SDO, the SR–proximity measure for SOCO [48] is given by

$$\Psi(x^j, s^j, \mu) := \Psi(v^j) = \mathrm{Tr}(\psi(v^j)) = \psi(\lambda_{\max}(v^j)) + \psi(\lambda_{\min}(v^j)), \tag{71}$$

where $\psi(\cdot)$ is a univariate SR function, and for $x, s \in I\!R^n$, $n = \sum_{j=1}^{N} n_j$, $x^j \in I\!R^{n_j}$

$$\Psi(x, s, \mu) := \Psi(v) = \mathrm{Tr}(\psi(v)) = \sum_{j=1}^{N} \left(\psi(\lambda_{\max}(v^j)) + \psi(\lambda_{\min}(v^j)) \right). \tag{72}$$

The last equality in (71) is derived from Definition 8.3 and from the properties of the kernel functions defined by (68) (see Lemma 6.2.6. in [48]). The *new search direction* proposed for SOCO is a slight modification of the **NT** direction and is defined, (see [48]) by the solution of the system

$$\begin{aligned} \bar{A} d_x &= 0, \\ \bar{A}^T \Delta y + d_s &= 0, \\ d_x + d_s &= -\psi'(v). \end{aligned} \tag{73}$$

Having d_x and d_s, Δx and Δs can be derived from (67). In the next subsection we discuss various properties of SR–proximities for SOCO.

8.1 Properties of SR–Proximity Functions for SOCO and the Algorithm

In order to provide basic properties of SR–proximity functions on second-order cones, we first define an algebra for the second–order cone.

Definition 8.4 *The Euclidean Jordan algebra for second–order cone K^j is defined by the bilinear operator*

$$x^j \circ s^j = ((x^j)^T s^j, x_1^j (s_{2:n}^j)^T + s_1^j (x_{2:n}^j)^T)^T, \quad x^j, s^j \in I\!R^{n_j},$$

and for second–order cone $K = K^1 \times \dots K^N$ the Euclidean Jordan algebra is defined by

$$x \circ s = \left((x^1 \circ s^1)^T, \dots, (x^N \circ s^N)^T\right)^T, \quad x, s \in I\!R^n, n = \sum_{j=1}^{N} n_j,\ x^j, s^j \in I\!R^{n_j}.$$

Note that the Jordan product $\circ$ is commutative; for any $z \in I\!R^n$ one has $z \circ z \in K$, and for every $z \in K$ the equation $z \circ z = x$ has a unique solution z in K. It is easy to verify that

$$x \circ s = \mathrm{mat}(x)s = \mathrm{mat}(s)x = s \circ x,$$

and for any $x \in I\!R^n$

$$\mathrm{Tr}(x \circ x) = \sum_{j=1}^{N} 2||x^j||^2.$$

The following proposition characterizes several important properties of a SR functions associated with the second-order cone K.

Proposition 8.5 [48] *Let the functions $\psi(x) : K_+ \to K_+$ and $\Psi(x) : K_+ \to I\!R$ be defined by (69) and (72) respectively. If the function $\psi(x)$ is SR, then the following statements hold.*

(i) $\Psi(x)$ is strictly convex with respect to $x \in K_+$ and vanishes at its global minimal point $x = \tilde{e}$, i.e., $\Psi(\tilde{e}) = 0, \psi(\tilde{e}) = \psi'(\tilde{e}) = 0$. Further, there exist positive constants $\nu_1, \nu_2 > 0$ and $p, q \geq 1$ such that

$$\nu_1(x^{p-1} + x^{-1-q}) \preceq_K \psi''(x) \preceq_K \nu_2(x^{p-1} + x^{-1-q}); \tag{74}$$

(ii) Suppose x and s are two vectors in K_+. If $v \in K_+$ satisfies

$$\mathrm{Tr}(v^2) = \mathrm{Tr}(x \circ s)$$

and

$$\sum_{j=1}^{N} \lambda_{\max}((v^j)^2)\lambda_{\min}((v^j)^2) = \sum_{j=1}^{N} \lambda_{\max}(x^j)\lambda_{\max}(s^j)\lambda_{\min}(x^j)\lambda_{\min}(s^j),$$

then

$$\Psi(v) \leq \frac{1}{2}(\Psi(x) + \Psi(s)). \tag{75}$$

Comparing Proposition 8.5 with its SDO analogue Proposition 7.4 on page 29, we find that statements (ii) in these two propositions are slightly different. Actually, one can easily see that the matrix used in the second claim of Proposition 7.4 satisfies certain conditions similar to those posed in Proposition 8.5. However, the choice of the vector v allowing such conditions in second-order cones is much more strict. The following proposition gives some fundamental properties of SR–proximity functions that are crucial for the analysis of SR–IMPs for SOCO.

Proposition 8.6 *Let the kernel function $\psi(t)$ be SR, the proximity $\Psi(v)$ defined by (72), and*

$$\sigma^2 = \mathrm{Tr}(\psi'(v) \circ \psi'(v)) = 2\left\|\psi'(v)\right\|^2. \tag{76}$$

If the kernel function $\psi(\cdot)$ satisfies condition SR1 (Definition 2.1, page 6), then

$$\Psi(v) \leq \frac{\sigma^2}{2\nu_1}, \tag{77}$$

$$\lambda_{\min}(v) \geq \left(1 + \frac{q\sigma}{\nu_1}\right)^{-\frac{1}{q}}, \tag{78}$$

$$\lambda_{\max}(v) \leq \left(1 + \frac{p\sigma}{\nu_1}\right)^{\frac{1}{p}}. \tag{79}$$

Note the similarities of statements (77)–(79) in Proposition 8.6 and statements (54)–(56) in Proposition 7.5 on page 29, which are extensions of conditions derived for the case of LO (see Proposition 3.1.5 on page 50 in [48]). The following corollary gives the relationship between the duality gap and the proximity.

Corollary 8.7 *Let the kernel function $\psi(t)$ be SR, and let the proximity $\Psi(v)$ be defined by (72). If the kernel function $\psi(\cdot)$ satisfies condition SR1 (Definition 2.1, page 6), then*

$$\mathrm{Tr}(x \circ s) = 2\mu||v||^2 \leq 2N\mu + 4\mu\sqrt{\frac{N\Psi(v)}{\nu_1}} + \frac{2\Psi(v)}{\nu_1}\mu. \tag{80}$$

¿From Corollary 8.7 it follows that $\mathrm{Tr}(x \circ s) = \mathcal{O}(N\mu)$, whenever $\Psi(v) = \mathcal{O}(N)$. Hence, as in both the LO and the SDO case, the proximity is appropriate to use for minimizing the duality gap as μ goes to zero.

With respect to the exchange of the variables (e.g., semidefinite matrices with vectors associated with the second order cone), the new algorithm for SOCO has the same structure as the new algorithm for SDO (see Algorithm 5, page 30) and LO (see Algorithm 1, page 3). Analogues to the SDO case (see Remark 7.7), the following conclusion holds for the SOCO case.

Remark 8.8 *The algorithm will stop when an iterate satisfies $N\mu < \varepsilon$. By using (80), we obtain*

$$x^T s = \frac{1}{2}\mathrm{Tr}(x \circ s) \leq N\mu + 2\mu\sqrt{\frac{N\tau}{\nu_1}} + \mu\frac{\tau}{\nu_1},$$

where $\tau \geq \nu_1^{-1}$ is a proximity parameter. For instance, for $\tau = N$ and the proximity satisfying condition SR1 with $\nu_1 = 1$, the algorithm works in a large neighborhood of the central path. One can easily verify that the algorithm will finally report a solution satisfying $x^T s \leq 4\varepsilon$.

8.2 Complexity of the Algorithm

Like in the LO and the SDO cases, a crucial step in the estimate of the algorithm's complexity is to evaluate how fast one can reduce the value of the proximity for a feasible step along the search direction. Note that once the search direction $(\Delta x, \Delta s)$ is obtained, we need to decide

how far we can go along this direction while staying in the feasible region. This amounts to estimating the maximal feasible step size. It should be noticed that for any step size α,

$$(x + \alpha\Delta x, s + \alpha\Delta s) \;\text{ is feasible }\; \Longleftrightarrow \;\; (v + \alpha d_x, v + \alpha d_s) \;\text{ is feasible.}$$

In the sequel we give a certain sufficient condition for a step size to be strictly feasible, and thus provide a *lower bound* for the maximal step size. The following lemma is proved in [48].

Lemma 8.9 [48] *Let $\alpha_{\max}$ be the maximal feasible step size and*

$$\bar{\alpha} = \lambda_{\min}(v)\sigma^{-1}. \tag{81}$$

Then

$$\alpha_{\max} \geq \bar{\alpha} \geq \sigma^{-1}\left(1 + \frac{q\sigma}{\nu_1}\right)^{-\frac{1}{q}}.$$

In view of Lemma 8.9, it is clear we can use any $\alpha \in (0, \bar{\alpha})$ as a step size. Note that, after such a step, we get a new primal-dual pair $(x + \alpha\Delta x, s + \alpha\Delta s)$ or the scaled pair $(v + \alpha d_x, v + \alpha d_s)$ and then we need to use the **NT** scaling scheme to transform the primal and dual vectors to the same vector, which we denote by v^+. On the other hand, according to (72), the proximity after this step is defined as $\Psi(v^+)$. Let us denote the difference between the proximity before and after one step as a function of the step size, that is

$$g(\alpha) = \Psi(v^+) - \Psi(v). \tag{82}$$

The main task now is to study the decreasing behavior of $g(\alpha)$ for $\alpha \in [0, \bar{\alpha})$.

¿From part (ii) of Proposition 8.5 one gets

$$g(\alpha) = \Psi(v^+) - \Psi(v) \leq \frac{1}{2}\left(\Psi\left(v + \alpha d_x\right) + \Psi(v + \alpha d_s)\right) - \Psi(v) =: g_1(\alpha).$$

Hence it suffices for us to estimate the decrease of the value of the functions $g_1(\alpha)$ after one step. In [48] Peng et al. compute

$$\begin{aligned} g_1'(\alpha) &= \frac{1}{2}\mathrm{Tr}(\psi'(v + \alpha d_x) \circ d_x + \psi'(v + \alpha d_s) \circ d_s), \\ g_1''(\alpha) &= \frac{1}{2}\frac{d^2}{d\alpha^2}\mathrm{Tr}(\psi(v + \alpha d_x) + \psi(v + \alpha d_s)), \end{aligned}$$

and report the following result that plays a crucial role in establishing the polynomial complexity of the algorithm.

Lemma 8.10 [48] *For any $\alpha \in (0, \bar{\alpha})$ holds*

$$g_1''(\alpha) \leq \frac{\nu_2\sigma^2}{2}\left((\lambda_{\max}(v) + \alpha\sigma)^{p-1} + (\lambda_{\min}(v) - \alpha\sigma)^{-q-1}\right).$$

The result stated in Lemma 8.10 shows a close analogy with the SDO case (see Lemma 7.9, page 32). The remaining statements for the SOCO case are very similar to the LO and SDO cases, and hence we just state the results and recall their SDO analogues.

Corollary 8.11 *For $\alpha < \bar{\alpha}$,*

$$g(\alpha) \leq g_1(\alpha) \leq -\frac{\sigma^2\alpha}{2} + \frac{1}{2}\nu_2\sigma^2 \int_0^\alpha \int_0^\xi \left((\lambda_{\max} v + \zeta\sigma)^{p-1} + (\lambda_{\min} v - \zeta\sigma)^{-q-1} \right) d\zeta d\xi. \tag{83}$$

Here we recall the SDO analogue of Corollary 8.11, i.e., Corollary 7.10 on page 32. We define now the function $g_2(\alpha)$ as the right hand side of the inequality in (83). It is straightforward to verify that $g_2(\alpha)$ is strictly convex and twice differentiable for all $\alpha \in [0, \bar{\alpha})$. Let α^* be the unique global minimizer of $g_2(\alpha)$ in the interval $[0, \bar{\alpha})$, i.e.,

$$\alpha^* = \arg \min_{\alpha \in [0,\bar{\alpha})} g_2(\alpha), \tag{84}$$

or equivalently α^* is the unique solution of the equation

$$\frac{\nu_2}{p} \left(\lambda_{\max} v + \alpha^* \sigma \right)^p - \lambda_{\max} v^p) + \frac{\nu_2}{q} \left(\lambda_{\min} v - \alpha^* \sigma \right)^{-q} - \lambda_{\min} v^{-q}) = \sigma.$$

The following lemma gives the estimation of the value of α^*.

Lemma 8.12 [48] *Let the constant α^* be defined by (84). Suppose that $\Psi(v) \geq \nu_1^{-1}$ and $v_{\max} > 1$ and let*

$$\nu_5 = \min \left\{ \frac{\nu_1}{2\nu_2(p+\mu_1) + \nu_1(p-1)}, \frac{\nu_1^2}{(1+\nu_1)\left(2\nu_2(\nu_1+q) + \nu_1 q\right)} \right\}. \tag{85}$$

Then

$$\alpha^* \geq \nu_5 \sigma^{-\frac{q+1}{q}}.$$

The following result estimates the decreasing value of the proximity.

Theorem 8.13 [48] *Let the function $g(\alpha)$ be defined by (82) with $\Psi(v) \geq \nu_1^{-1}$. Then the step size given by $\alpha = \alpha^*$ (see (84)) or $\alpha = \nu_5 \sigma^{\frac{q-1}{q}}$ is strictly feasible. Moreover,*

$$g(\alpha) \leq \frac{1}{2} g'(0)\alpha \leq -\frac{\nu_5 \nu_1^{\frac{q-1}{2q}}}{4} \Psi(v)^{\frac{q-1}{2q}}.$$

Note that Theorem 8.13 is similar to its SDO analogue Theorem 7.11, page 33. To get the total complexity result for the algorithm, we still need to describe the growth behavior of the proximity $\Psi(v)$. In [48] Peng et al. show that after the update of μ, the proximity is still bounded above by the number $\psi_0(\theta, \tau, 2N)$ defined by (64), where N is the number of cones. The following result is proved in [48] (compare with its analogue Lemma 7.12).

Lemma 8.14 [48] *Let $\Psi(x, s, \mu) \leq \tau$ and $\tau \geq \nu_1^{-1}$. Then after an update of the barrier parameter, no more than*

$$\left\lceil \frac{8q\nu_1^{-\frac{q-1}{2q}}}{\nu_5 (q+1)} \left(\psi_0(\theta, \tau, 2N) \right)^{\frac{q+1}{2q}} \right\rceil$$

iterations are needed to recenter. In the special case where $\psi(t) = \Upsilon_{p,q}(t)$ with $\nu_1 = \nu_2 = 1$, at most

$$\left\lceil \frac{8q \max\{3p+1, 6q+4\}}{q+1} \left(\psi_0(\theta, \tau, 2N) \right)^{\frac{q+1}{2q}} \right\rceil$$

inner iterations are needed to recenter.

Finally, the total complexity of the algorithm can be estimated as follows.

Theorem 8.15 [48] *If $\tau \geq \nu_1^{-1}$, the total number of iterations required by the primal-dual Newton algorithm is not more than*

$$\left\lceil \frac{8q\nu_1^{-\frac{q-1}{2q}}}{\nu_5\,(q+1)}\left(\psi_0(\theta,\tau,2N)\right)^{\frac{q+1}{2q}} \right\rceil \left\lceil \frac{1}{\theta}\log\frac{N}{\varepsilon} \right\rceil.$$

In the special case where $\psi(t) = \Upsilon_{p,q}(t)$ with $\nu_1 = \nu_2 = 1$, the total number of iterations required by the primal-dual Newton algorithm is less than or equal to

$$\left\lceil \frac{8q\max\{3p+1,6q+4\}}{q+1}\left(\psi_0(\theta,\tau,2N)\right)^{\frac{q+1}{2q}} \right\rceil \left\lceil \frac{1}{\theta}\log\frac{N}{\varepsilon} \right\rceil.$$

Neglecting the influence of the constants in the expression in Theorem 8.15, one can safely conclude that for any fixed $\theta \in (0,1)$ with suitable $p, q \geq 1$, the algorithm with large-update for SOCO has an $\mathcal{O}(N^{\frac{q+1}{2q}}\log\frac{N}{\varepsilon})$ iterations bound, while the algorithm with small-update ($\theta = \mathcal{O}(\frac{1}{\sqrt{N}})$) remains with the complexity of $\mathcal{O}(\sqrt{N}\log\frac{N}{\varepsilon})$ iterations bound. Furthermore, from Theorem 8.15 one can readily see that if p is a constant and $q = \log N$, then the new large-update algorithm has a complexity $\mathcal{O}(\sqrt{N}\log N\log\frac{N}{\varepsilon})$ iterations bound.

Finally, as we mentioned earlier a SOCO problem can also be solved by casting it as a SDO problem in $\mathcal{S}^{n\times n}$. In such a situation, the iteration complexity of the algorithm for solving the reformulated SDO problem has a bound of $\mathcal{O}(n^{\frac{q+1}{2q}}\log\frac{n}{\varepsilon})$. If $K^j \in I\!R^2$ for all $j \in \mathcal{J}$, then the iteration complexity of the algorithm for SOCO is the same as that of its counterpart for SDO, but when $2N < n$, the algorithm that works directly on the original SOCO problem has a better iteration bound, see [48]. However, the improvement is significant if $N \ll n$.

9 Summary and Future Work

9.1 Summary

The monograph [48] of Peng et al. presents the methodology of Self-Regular functions that provide a new framework for the theory of primal-dual IPMs. It contains their earlier research [47, 49, 50, 51, 52] that was a breakthrough in reducing the gap between the theory and the practical performance of IPMs with respect to small and large-update methods. In this paper we review the most recent developments on SR-IPMs for LO, and give an overview of SR-IPMs for SDO and SOCO problems as well. First, we describe an algorithmic schema of standard IPMs (see Algorithm 1, page 3), and than we explain the motivation for deriving the new search directions (see Section 1). The new IPMs use SR-functions as kernel functions in formulating proximity measures. We provide some basic properties of SR-functions and SR proximities in Section 2. The interested reader can find more about these functions in [48]. In the remaining sections of this paper, we discuss several large-update path-following SR-IPMs for LO, SDO and SOCO, including their polynomial iteration complexity.

We present an adaptive single step large-update SR-IPM for LO including its complete complexity analysis in Section 3 (see also [57]). This method is developed for the $\Gamma_{1q},\ q > 1$ family

of kernel functions. The adaptive large-update algorithm chooses the target value adaptively at each step, and hence it is more flexible in updating the target value than classical IPMs (see Algorithm 2, page 12). This algorithm retains the so far best $\mathcal{O}\left(\sqrt{n}\log n\log\frac{n}{\epsilon}\right)$ iteration complexity of large-update IPMs.

In Section 4, we present a SR-proximity based predictor-corrector IPM for LO that enjoys polynomial complexities and asymptotic superlinear convergence (see also [53, 58]). For the predictor step this algorithm chooses either a SR step or an affine scaling step. The corrector step is recentering with the respect to the SR-neighborhood. A remarkable feature is that the SR-neighborhood includes the infinity neighborhood that is usually used in predictor-corrector IPMs. Due to the specific step size strategy, this algorithm has the so-far best iteration complexity. Although feasible predictor-corrector IPMs admit better convergence results, infeasible IPMs (IIPMs) are widely used in academic and commercial softwares. In Section 5, we present a SR-IIPM that is based on the $\Gamma_{13}(t)$ function as a kernel function (see also [56]). The new IIPM always takes large-updates and consists of only one Newton step for each updated target value (see Algorithm 4, page 23). The iteration complexity for the SR-IIPM is $\mathcal{O}\left(n^2\log\frac{n}{\epsilon}\right)$. The infeasible neighborhood which is used for designing the algorithm, is also defined in this section.

Recently, Bai et al. [8, 9, 10] define new kernel functions that are not SR but attain similar iteration bounds as SR-IPMs do. They derive these functions by replacing the first condition SR.1 in the definition of SR functions, by a set of simpler conditions (see conditions (41)–(43)), and therefore simplify the analysis of IPMs. In Section 6, we list the non-SR kernel functions, and give their iteration complexity. All of the presented functions are closely related to SR functions. We show that the non-SR function $\frac{t^2-1}{2}+e^{\frac{1}{2}-1}-1$, is the limit point of a sequence of SR functions. Hence, it follows that the cone of SR functions is not a closed cone, and the exponential barrier functions are on the boundary of the SR-cone.

In Section 7 and Section 8, we explain the basics of SR-IPMs for SDO and SOCO, respectively. We show that the concept of SR-proximity based primal-dual IPMs for SDO is a natural extension of SR-IPMs for LO (see also [49]). Since the Newton system of the algorithm for SDO in general does not obtain a symmetric solution, some symmetrization schema should be applied. The algorithm given here, chooses the NT symmetrization schema (see Algorithm 5, page 30). The resulting large-update algorithm has an $\mathcal{O}(\sqrt{n}\log n\log\frac{n}{\epsilon})$ iteration bound which is up to date the best known one for SDO when large-update IPMs are applied. We show also that the derived SR-IPMs for SOCO admit similar complexity results as SR-IPMs for SDO do (see also [48, 52]). The described algorithm for SOCO uses the NT scaling as well. Here we also provide the major results for SR-proximity functions for SOCO.

To sum up, all presented variants of IPMs, except IIPMs, improve the worst case iteration bound of large-update IPMs. Due to the remarkable results on improved theoretical complexity of IPMs, the theory of SR-IPMs is one of the hottest research area of IPMs.

9.2 Future Work

Theoretical and implementation aspects of SR-IMPs are still not fully explored. Here we list some of the challenging problems/questions that are worth to study in future research. Some of them are already under consideration.

1. Is it possible to design pure primal (or dual) IPMs for LO based on some barrier functions similar to the SR functions? If the result is positive, how to do it for SOCO and SDO

problems?

2. Can we design SR-IPMs for SDO and SOCO based on the various scaling techniques other than NT scaling?

3. Can we analyze SR-proximity based adaptive-large-update IPMs for SDO and SOCO?

4. Can we analyze SR-proximity based PC-IPMs for SDO and SOCO?

5. The analysis of SR-proximity based infeasible IPMs (IIPMs) was very complex for the LO case. The questions are:
   • How to generalize Algorithm 4 to the $\Gamma_{1q}(\cdot)$ family?
   • Can such generalization be combined with an adaptive update of the algorithm?
   • How to derive efficiently the generalization of IIPMs for SDO and SOCO?

6. How to generalize the analysis of IPMs based on non-SR functions to the whole family of non-SR functions that satisfy conditions (41)–(43)?

7. How to extend the SR-IPMs to Nonlinear Optimization?

8. Can we close the cone of SR functions?

9. Can we give a unified analysis for all kernel functions in the closure of the cone of SR functions?

10. Identifying other interesting sub–families and functions on the border of the SR cone.

Some preliminary numerical experiments with the new algorithms for LO and SOCO are given in [53, 74]. They show that the number of iterations of the SR-based IPMs is usually less than, or equal to that of the standard large-update IPMs. The preliminary results are promising, but still extensive numerical testing is needed to explore the efficiency of the approach. The following problems regarding the implementation should be addressed:

1. What choice of q and line search provide the best practical performance of SR based IPMs algorithms?

2. Implementation issues regarding SR-IPMs for LO still should be further explored.

3. Implementing SR-IPMs for SOCO and SDO should be fully considered.

Acknowledgments: This research was supported by the MITACS project "New IPMs and Software for Convex Conic–Linear Optimization and Their Application to Solve VLSI Circuit Layout Problems ", by an NSERC discovery grant, and the CRC program. The first author would also like to thank the Iranian Ministry of Science, Research and Technology for supporting his research.

References

[1] I. Adler and F. Alizadeh. Primal-dual interior point algorithms for convex quadratically constrained and semidefinite optimization problems. Technical Report RRR–111–95, Rutcor, Rutgers Center for Operations Research, P.O. Box 5062, New Brunswick, NJ, 1995.

[2] W. Achtziger, M. Bendsoe, A. Ben–Tal, and J. Zowe. Equivalent displacement based formulations for maximum strength truss topology design. *Impact of Computing in Science and Engineering*, 4(4):315–345, 1992.

[3] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1995.

[4] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton. Complementarity and nondegeneracy in semidefinite programming. *Mathematical Programming*, 77:111–128, 1997.

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

[6] E.D. Andersen, C. Roos, and T. Terlaky. On implementing a primal-dual interior-point method for conic quadratic optimization. *Mathematical Programming Ser. B*, 95:249–277, 2003.

[7] M. Anjos and A. Vanelli. An attractor-repeller approach to floorplanning. *Mathematical Methods of Operations Research*, 56(1):3–27, 2002.

[8] Y.Q. Bai and C. Roos. A primal-dual interior-point method based on a new kernel function with linear growth rate. To appear in: *Proceedings Perth Industrial Optimization Meeting.*

[9] Y.Q. Bai, C. Roos, and M. El Ghami. A comparative study of new barrier functions for primal-dual interior–point algorithms in linear optimization. Working paper, 2003.

[10] Y.Q. Bai, C. Roos, and M. El Ghami. A new efficient large-update primal-dual interior-point method based on a finite barrier. *SIAM Journal on Optimization,* 13(3):766–782, 2003.

[11] R. Bellman. *Introduction to Matrix Analysis.* Classics in Applied Mathematics, SIAM, Philadelphia, 12, 1995.

[12] M. Bendose, A. Ben–Tal, and J. Zowe. Optimization methods for truss geometry and topology design. *Structural Optimization*, 7:141–159, 1994.

[13] A. Ben-Tal and A. Nemirovski. *Convex Optimization in Engineering: Modeling, Analysis, Algorithms.* MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2001.

[14] S. Boyd, L. El Gahaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM Studies in Applied Mathematics, Philadelphia, PA 19101, USA, 15:1994.

[15] F.T. Cheng and D.E. Orin. Efficient algorithm for optimal force distribution – the compact–dual LP method. *IEEE Transactions on Robotics and Automation*, 6(2):178–187, 1990.

[16] E.W. Cheney. *Introduction to Approximation Theory.* Chelsea Publishing Company, New York, 1982.

[17] J. Faraut and A. Korányi. *Analysis on Symmetric Cones.* Oxford University Press, NY, USA, 1994.

[18] B. Fares, D. Noll, and P. Apkarian. Robust control via sequential semidefinite programming. *SIAM Journal on Control and Optimization*, 40:1791–1820, 2002.

[19] L. Faybusovich. Euclidean Jordan algebras and interior-point algorithms. *Positivity*, 1:331–357, 1997.

[20] L. Faybusovich. A Jordan-algebraic approach to potential-reduction algorithms. Technical Report, Department of Mathematics, University of Notre Dame, IN, 1998.

[21] M. Fukushima, Z.Q. Luo, and P. Tseng. Smoothing functions for second-order-cone complementarity problems. *SIAM Journal on Optimization*, 12(2):438–460, 2001.

[22] M.X. Goemans. Semidefinite programming in combinatorial optimization. *Mathematical Programming*, 79:143–161, 1997.

[23] O. Güler. Limiting behavior of the weighted central paths in linear programming. *Mathematical Programming*, 65:215–228, 1994.

[24] O. Güler and Y. Ye. Convergence behavior of interior point algorithms. *Mathematical Programming*, 60:215–228, 1993.

[25] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on Optimization*, 6:342–361, 1996.

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

[27] B. Jansen, C. Roos, T. Terlaky, and Y. Ye. Improved complexity using higher order correctors for primal-dual Dikin affine scaling. *Mathematical Programming, Series B*, 76:117–130, 1997.

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

[29] E. de Klerk. *Aspects of Semidefinite Programming: Interior Point Algorithms and Selected Applications*. Applied Optimization Series, Kluwer Academic Publishers, 65, 2002.

[30] E. de Klerk, C. Roos, and T. Terlaky. Infeasible–start semidefinite programming algorithms via self–dual embeddings. In: *Topics in semidefinite and interior–point methods*, P. Pardalos and H. Wolkowicz, editors, (Toronto, ON, 1996), 215–236, Fields Inst. Commun., 18, American Mathematical Society, Providence, RI, 1998.

[31] M. Kojima, S. Shindoh, and S. Hara. Interior–point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM Journal on Optimization*, 7:86–125, 1997.

[32] M. Kojima, S. Mizuno, and A. Yoshise. A primal-dual interior point algorithm for linear programming. In: *Progress in Mathematical Programming: Interior Point and Related Methods*, N. Megiddo, editor, Springer Verlag, New York, 29–47, 1989.

[33] J.B. Lasserre. An Explicit exact SDP relaxation for nonlinear 0-1 programs. *SIAM Journal on Optimization*, 12:756–769, 2002.

[34] M. Laurent. Lower bound for the number of iterations in semidefinite relaxations for the cut polytope. *Mathematics of Operations Research*, 28(4):871–883, 2003.

[35] H. Lebret and S. Boyd. Antenna array pattern synthesis via convex optimization. *IEEE Trans. on Signal Processing*, 45(3):526–532, 1997.

[36] L. Li, Z.Q. Luo, T.N. Davidson, K.M. Wong, and E. Bosse. Robust filtering via semidefinite programming with applications to target tracking. *SIAM Journal on Optimization*, 12(3):740–755, 2002.

[37] M.S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and Its Applications*, 284:193–228, 1998.

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

[39] S. Mizuno, M.J. Todd, and Y. Ye. On adaptive step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research*, 18:964–981, 1993.

[40] R.D.C. Monteiro, I. Adler, and M.G.C. Resende. A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extensions. *Mathematics of Operations Research*, 15:191–214, 1990.

[41] R.D.C. Monteiro. Primal-dual path following algorithm for semidefinite programming. *SIAM Journal on Optimization*, 7:663–678, 1997.

[42] A.S. Nemirovski and K. Scheinberg. Extension of Karmarkar's algorithm onto convex quadratically constrained quadratic programming. *Mathematical Programming*, 72:273–289, 1996.

[43] Y.E. Nesterov and A.S. Nemirovski. *Interior Point Polynomial Algorithms in Convex Programming.* SIAM Studies in Applied Mathematics, Volume 13, SIAM, Philadelphia, USA, 1994.

[44] Y.E. Nesterov and M.J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Mathematics of Operations Research*, 22(1):1–42, 1997.

[45] Y.E. Nesterov and M.J. Todd. Primal-dual interior point methods for self-scaled cones. *SIAM Journal on Optimization*, 8:324–364, 1998.

[46] J. Peng, C. Roos, and T. Terlaky. A new efficient large-update interior-point method for linear optimization. *Journal of Computational Technologies*, 6(4):61–80, 2001.

[47] J. Peng, C. Roos, and T. Terlaky. A new class of polynomial primal-dual methods for linear and semidefinite optimization. *European Journal of Operations Research*, 143(2):234–256, 2002.

[48] J. Peng, C. Roos, and T. Terlaky. *Self-Regularity: A New Paradigm for Primal-Dual Interior-Point Methods.* Princeton University Press, Princeton, NY, 2002.

[49] J. Peng, C. Roos, and T. Terlaky. Self-regular functions and new search directions for linear and semidefinite optimization. *Mathematical Programming*, 93:129–171, 2002.

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

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

[52] J. Peng, C. Roos, and T. Terlaky. New primal-dual algorithms for second-order conic optimization based on self-regular proximities. *SIAM Journal on Optimization*, 13(1):179–203, 2002.

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

[54] J. Renegar. *A Mathematical View of Interior-Point Methods in Convex Optimization.* SIAM, Philadelphia, USA, 2001.

[55] C. Roos, T. Terlaky, and J.-Ph. Vial. *Theory and Algorithms for Linear Optimization. An Interior Point Approach.* John Wiley & Sons, Chichester, United Kingdom, 1997.

[56] M. Salahi, T. Terlaky, and G. Zhang. The complexity of self–regular proximity based infeasible IPMs. Submitted to: *Computational Optimization and Applications*, 2003.

[57] M. Salahi and T. Terlaky. An adaptive self-regular proximity based large-update interior point methods for linear optimization. Submitted to: *Mathematical Programming*, 2003.

[58] M. Salahi and T. Terlaky. An adaptive large neighborhood self-regular predictor-corrector IPM for linear optimization. Working paper, 2004.

[59] G. Sonnevend. An "analytic center" for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming. In: *System Modeling and Optimization: Proceeding of the 12th IFIP Conference*, A. Prékopa, J. Szelezsán, and B. Strazicky, editors, Budapest, Hungary, September 1985, Volume 84, Lecture Notes in Control and Information Sciences, pp. 866–876. Springer Verlag, Berlin, 1986.

[60] J.F. Sturm. *Theory and Algorithms of Semidefinite Programming.* In: High Performance Optimization, H. Frenk, C. Roos, T. Terlaky, and S. Zhang, editors, Kluwer Academic Publishers, Boston, 1–194, 1999.

[61] J.F. Sturm and S. Zhang. Symmetric primal-dual path following algorithms for semidefinite programming. *Applied Numerical Mathematics*, 29:301–315, 1999.

[62] M.J. Todd. A study of search directions in primal-dual interior-point methods for semidefinite programming. *Optimization Methods and Software*, 11:1–46, 1999.

[63] M.J. Todd, K.C. Toh, and R.H. Tütüncü. On the Nesterov-Todd direction in semidefinite programming. *SIAM Journal on Optimization*, 8:769–796, 1998.

[64] T. Tsuchiya. A polynomial primal-dual path-following algorithm for second-order cone programming. Technical Report 649, The Institute of Statistical Mathematics, Tokyo, Japan, December, 1997.

[65] T. Tsuchiya. A convergent analysis of the scaling-invariant primal-dual path-following algorithms for second-order cone programming. *Optimization Methods and Software*, 11/12:141–182, 1999.

[66] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38:49–95, 1996.

[67] H. Wolkowicz, R. Saigal, and L. Vandenberghe. *Handbook of Semidefinite Programming (Theory, Algorithms and Applications)*. Kluwer Academic Publishers, Boston, 2000.

[68] S.J. Wright. *Primal-dual Interior-Point Methods*. SIAM, Philadelphia, USA, 1997.

[69] S.P. Wu, S. Boyd, and L. Vandenberghe. FIR filter design via spectral factorization and convex optimization. In: *Applied and Computational Control, Signals and Circuits*, B. Datta editor, Birkhauser, Chapter 5, 1:215–245, 1998.

[70] Y. Ye and K.M. Anstreicher. On quadratic and $O(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for LCP. *Mathematical Programming*, 62:537–551, 1993.

[71] Y. Ye. *Interior Point Algorithms, Theory and Analysis*. John Wiley & Sons, Chichester, United Kingdom, 1997.

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

[73] Q. Zhao, S.E. Karisch, F. Rendl, and H. Wolkowicz. Semidefinite programming relaxations for the Quadratic Assignment Problem. *Journal of Combinatorial Optimization*, 2:71–109, 1998.

[74] X. Zhu, J. Peng, T. Terlaky, and G. Zhang. On implementing Self-Regular proximity based feasible IPMs. Technical Report 2003/2, Advanced Optimization Lab. Department of Computing and Software, McMaster University, 2003. Submitted to: *INFORMS Journal on Computing*.