



## A NEW CLASS OF POLYNOMIAL INTERIOR-POINT ALGORITHMS FOR $P_*(\kappa)$ -LINEAR COMPLEMENTARY PROBLEMS

Y.Q. BAI<sup>\*†</sup>, G. LESAJA<sup>‡</sup> AND C. ROOS<sup>§</sup>

*Dedicated to Professor Liansheng Zhang on the occasion of his 70th birthday.*

**Abstract:** We present a polynomial interior-point algorithm for  $P_*(\kappa)$  Linear Complementarity Problems (LCP) based on a class of parametric kernel functions, with parameters  $p \in [0, 1]$  and  $q \geq 1$ . The same class of kernel function was considered earlier for Linear Optimization (LO) by Bai *et al.* in [7]. This class is fairly general and includes the classical logarithmic function, the prototype self-regular function, and non-self-regular kernel functions as special cases. The iteration bounds obtained in this paper are  $O\left[(1 + 2\kappa)q(p + 1)n^{(p+q)/q(p+1)} \log \frac{n}{\epsilon}\right]$  for large-update methods and  $O\left[(1 + 2\kappa)q^2\sqrt{n} \log \frac{n}{\epsilon}\right]$  for small-update methods. These bounds match the best known existing iteration bounds. As far as we know this is the first result on interior-point methods for  $P_*(\kappa)$ -LCPs based on this class of kernel functions.

**Key words:** linear complementarity problem,  $P_*(\kappa)$  matrix, interior-point method, polynomial complexity

**Mathematics Subject Classification:** 90C22, 90C31

### 1 Introduction

In this paper, we consider a class of linear complementarity problems (LCP) formulated in the standard form: Given a matrix  $M \in \mathbf{R}^{n \times n}$  and a vector  $q \in \mathbf{R}^n$ , find  $(x, s) \in \mathbf{R}^{2n}$  such that

$$s = Mx + q, \quad xs = 0, \quad x, s \geq 0, \quad (1.1)$$

where  $xs$  denotes the componentwise product of the vectors of  $x$  and  $s$ .

Note that (1.1) is a feasibility problem, and not an optimization problem. However, it is well-known that it can easily be written as an optimization problem:

$$\min \{x^T s : Mx - s = -q, \quad x, s \geq 0\}.$$

<sup>\*</sup>The first author, corresponding author, kindly acknowledges the support of the grant of National Science Foundation of China 10771133 and the Program of Shanghai Pujiang 06PJ14039.

<sup>† §</sup> The first and third author kindly acknowledge support the Dutch Royal Academy of Sciences (KNAW) 06CDP022.

<sup>‡</sup>Research of the second author was supported in part by the Georgia Southern University 2006-2007 Faculty Research Grant.

On the other hand, the Karush-Kuhn-Tucker (KKT) optimality conditions for Linear Optimization (LO) and Convex Quadratic Optimization (CQO) can be written in the form of an LCP, showing the relevance of the LCP for these important classes of optimization problems. Moreover, LCP also has a close connection to variational inequalities: some classes of variational inequalities can be formulated as an LCP and vice versa. In addition, many important practical problems in economics theory (equilibrium problems), game theory, transportation planning (assignment problems), optimal control, engineering, etc. can be directly formulated as LCP. For a comprehensive treatment of LCP theory and practice we refer to the monographs of Cottle *et al.* [9] and Kojima *et al.* [12], and for a recent comprehensive treatment of variational inequalities and complementarity problems to the monograph of Facchinei and Pang [10].

Due to the theoretical and practical importance of LCP, efficient methods for solving LCPs are of a significant interest. The existing tradition of generalizing results for LO to LCP dates back to the early days of the development of simplex-type algorithms (pivoting algorithms) and it continues to this day. The Interior-Point Methods (IPMs) that has been a great success for LO are no exception. Various IPMs for LO have been successfully generalized to LCP. Besides the aforementioned monograph of Kojima *et al.* [12], and without any attempt to be complete we mention a few other relevant references: [2, 13, 14, 21, 19, 22, 23].

The majority of IPMs for LO are based on the use of the logarithmic barrier function in calculation of the search direction. Recently, Peng *et al.* designed primal-dual IPMs for LO based on a different class of barrier functions, the class of so-called self-regular barrier (or proximity) functions [16]. They derived the currently best known complexity bounds for large-update methods, namely  $O(\sqrt{n} \log n \log \frac{n}{\epsilon})$ . Subsequently, Y.Q. Bai *et al.* [4] presented primal-dual IPMs for LO based on yet another class of barrier functions that are not self-regular and obtained the same bounds as Peng *et al.*

The goal of this paper is to extend the results obtained for LO by Bai *et al.* in [7] to the class of  $P_*(\kappa)$ -LCPs whose definition is given in the next section. The outline of the algorithm is presented in Section 3 while the definition and properties of the of the kernel and barrier functions used in the design of the algorithm are discussed in the Section 4. The analysis and complexity results of the algorithm are presented in the Sections 5 and 6 respectively.

Some notation used throughout the paper is as follows. We use the standard notation,  $\mathbf{R}^n$ ,  $\mathbf{R}_+^n$  and  $\mathbf{R}_{++}^n$  to denote the set of (real) vectors with  $n$  components, the set of non-negative vectors and the set of positive vectors, respectively. The 2-norm of the vector  $x$  is denoted with  $\|x\|$ . For any  $x \in \mathbf{R}^n$ ,  $x_{min}$  ( or  $x_{max}$ ) denotes the smallest ( or largest) value of the components of the vector  $x$ . The bold symbol  $\mathbf{e}$  always denotes the all-one vector with  $n$  components. Finally, if  $g(x) \geq 0$  is a real valued function of a real nonnegative variable, the notation  $g(x) = O(x)$  means that  $g(x) \leq \bar{c}x$  for some positive constant  $\bar{c}$  and  $g(x) = \Theta(x)$  that  $c_1x \leq g(x) \leq c_2x$  for two positive constants  $c_1$  and  $c_2$ .

## **2** The $P_*(\kappa)$ -LCP

As indicated in the Introduction, in this paper we consider the LCP in the standard form (1.1). It is known that for general matrices  $M$  the problem is NP-complete [8]. Therefore, it is natural to look for classes of matrices  $M$  for which the corresponding LCPs can be solved in polynomial time. Different classes of matrices have been considered. We list the most important of these classes.

- Skew-symmetric matrices (SS):

$$x^T Mx = 0, \quad \forall x \in R^n.$$

- Positive semidefinite matrices (PSD):

$$x^T Mx \geq 0, \quad \forall x \in R^n.$$

- $P$ -matrices: Matrices with all principal minors positive or equivalently

$$\forall x(\neq 0) \in \mathbf{R} : \exists i \in I : x_i(Mx)_i > 0.$$

- $P_0$ -matrices: Matrices with all principal minors nonnegative or equivalently

$$\forall x(\neq 0) \in \mathbf{R} : \exists i \in I : x_i \neq 0 \text{ and } x_i(Mx)_i \geq 0.$$

- Column sufficient matrices (CSU)

$$\forall x \in R^n : (x_i(Mx)_i \leq 0, \forall i \in I \Rightarrow x_i(Mx)_i = 0, \forall i \in I.$$

- Row sufficient matrices (RSU):  $M^T$  is column sufficient.

- Sufficient matrices (SU):  $M$  is both column sufficient and row sufficient.

- $P_*(\kappa)$ :  $\kappa \geq 0$  and

$$(1 + 4\kappa) \sum_{i \in \mathcal{I}^+(x)} x_i(Mx)_i + \sum_{i \in \mathcal{I}^-(x)} x_i(Mx)_i \geq 0, \quad \forall x \in R^n,$$

where

$$\mathcal{I}^+(x) = \{i : x_i(Mx)_i > 0\}, \quad \mathcal{I}^-(x) = \{i : x_i(Mx)_i < 0\},$$

or equivalently

$$x^T Mx \geq -4\kappa \sum_{i \in \mathcal{I}^+(x)} x_i(Mx)_i, \quad \forall x \in R^n,$$

and

$$P_* = \bigcup_{\kappa \geq 0} P_*(\kappa).$$

The class of  $P_*$  matrices was introduced by Kojima *et al.* in their fundamental monograph on IPM's for LCP [12], while the other classes, and many additional classes not mentioned here, were listed in the classical monograph of Cottle *et al.* [9]. See also [3]. The relationship between some of the above classes is as follows:

$$SS \subset PSD \subset P_* = SU \subset CS \subset P_0, \quad P \subset P_*, \quad P \cap SS = \emptyset. \quad (2.1)$$

Some of these relations are obvious, like  $PSD = P_*(0) \subset P_*$  or  $P \subset P_*$ , while others require proof [9, 12, 18].

Most common and most studied is the class of *monotone*-LCPs, where matrix  $M$  is a positive-semidefinite matrix. This is largely due to the fact that the Karush-Kuhn-Tucker conditions (KKT) of Quadratic Optimization problem (QO) with the quadratic objective

function defined by the positive-semidefinite matrix can be formulated as monotone-LCP. In addition, most practical problems that can be directly formulated as LCP are usually monotone-LCP. We also recall that in the special case, that of a Linear Optimization (LO) problem, the matrix  $M$  becomes a skew-symmetric matrix. In this paper we consider the class of  $P_*(\kappa)$ -LCPs, which (as is clear from (2.1)) contains the class of monotone-LCP as a special case.

We remark that the above classes enjoy the nice property that if matrix  $M$  belongs to one of these classes, then every principal submatrix of  $M$  also belongs to the class. In what follows, we state other properties that are relevant for the design of IPMs in this paper.

**Lemma 2.1 (Lemma 4.1. in [12]).** *The matrix*

$$\bar{M} = \begin{pmatrix} -M & I \\ S & X \end{pmatrix} \quad (2.2)$$

*is nonsingular for any positive diagonal matrices  $X, S$  if and only if  $M$  is  $P_0$ -matrix.*

Matrices of the form  $\bar{M}$  in (2.2) appear at each iteration of an interior-point method for LCP. Thus, the above lemma recognizes the class  $P_0$  as the largest class which guarantees the existence and uniqueness of the solution for a linear system with coefficient matrix  $\bar{M}$ . As was observed first in [12] this property makes  $P_0$ -LCPs amenable for being solved by an interior-point method.

However, it is well known that the sequence of the iterates of IPMs may not converge to the solution of LCP. To assure that each accumulation of the iteration sequence is a solution, the sequence must be bounded. The question arises: Which class of matrices implies boundedness of the sequence? In their fundamental work [12] Kojima *et al.* presented an IPM for LCP and proved its global convergence under the following condition which just formalizes the above discussion.

**Condition 2.2.**

- (1)  $M$  is a  $P_0$ -matrix;
- (2) the interior-point condition (IPC) is satisfied, i.e., there exists a strictly feasible point  $0 < (x^0, s^0) \in \mathcal{F} = \{(x, s) \geq 0 : s = Mx + q\}$ ;
- (3) the level set  $\mathcal{F}_t = \{(x, s) \in \mathcal{F} : x^T s \leq t\}$  is bounded for each  $t \geq 0$ .

Kojima *et al.* pointed out that, unfortunately, if  $M$  is  $P_0$ -matrix, then requirement (3) is generally not satisfied and they provided a counterexample. However, if  $M$  is a  $P_*$ -matrix, then requirement (3) is satisfied (Lemma 4.5 in [12]). Hence, in some sense  $P_*$  class is a maximal class which guarantees global convergence of IPMs. This fact is, at least from the theoretical point of view, an important reason why we consider the class of  $P_*(\kappa)$ -LCP in this paper.

### **3 The Generic Interior-point Algorithm for the $P_*(\kappa)$ -LCP**

As already indicated in the previous section, in this paper we consider the  $P_*(\kappa)$ -LCP in the standard form: Given a  $P_*(\kappa)$  matrix  $M \in R^{n \times n}$  and a vector  $q \in R^n$ , find  $(x, s) \in R^{2n}$  such that

$$s = Mx + q, \quad xs = 0, \quad x, s \geq 0. \quad (3.1)$$

We assume that the  $P_*(\kappa)$ -LCP satisfy the interior-point condition (IPC), that is, there exists a point  $x^0 > 0$  such that  $s^0 = Mx^0 + q > 0$ , which means that the interior of the

feasible region is not empty. The IPC can be assumed without loss of generality. In [12, Section 5.1] Kojima *et al.* presented a method of ‘reducing the LCP to an artificial LCP with an apparent interior feasible point’. Another approach to obtaining IPC is the approach of Andersen and Ye [1], who embed the original complementarity problem (CP) into an artificial homogeneous CP that satisfies the IPC. However, their approach is specifically designed for monotone CP and can not be extended to  $P_*(\kappa)$ -CP, see [16].

The basic idea of interior-point algorithms for LCP is to use a Newton-type method in a such a way that it guarantees global convergence and fast local convergence. However, direct application of the Newton-type method on the system in (3.1) will not work because the method will most likely ‘get stuck’ on the second equation which is commonly known as *complementarity equation*. The standard procedure to fix this problem is to replace the complementarity equation in (3.1) by the parameterized equation  $xs = \mu e$ , with  $\mu > 0$ . Thus, we consider the system

$$\begin{aligned} s &= Mx + q \\ xs &= \mu e, \\ x, s &> 0. \end{aligned}$$

Since we assume that IPC holds and  $M$  is a  $P_*(\kappa)$  matrix then according to the discussion in the previous section (Lemma 2.1 and the Condition 2.2) the parameterized system in (3.2) has a unique solution, for each  $\mu > 0$ . This solution is denoted as  $(x(\mu), s(\mu))$  and we call it the  $\mu$ -center of LCP. The set of  $\mu$ -centers (with  $\mu$  running through all positive real numbers) gives a homotopy path, which is called *the central path* of LCP. The relevance of the central path for Linear Optimization (LO) was first recognized by Megiddo [11] and then extended to LCP by Kojima *et al.* [12]. Under the above assumptions, if  $\mu \rightarrow 0$ , the limit of the central path exists and it is an optimal solution of LCP.

The limiting property of the central path mentioned above leads naturally to the main idea of the IPMs for solving LCP: Trace the central path while reducing  $\mu$  at each iteration. However, tracing the central path exactly would be too costly and inefficient. It has been shown that it is sufficient to trace the central path approximately.

As already indicated, due to the result of Kojima *et al.* [12], we may assume that IPC holds, that is, a strictly feasible point  $(x, s)$  exists. In addition, this point can be chosen in a such a way that it is ‘close’ to the  $\mu$ -center  $(x(\mu), s(\mu))$  for some positive  $\mu$ , where ‘closeness’ is measured using a barrier function, as discussed later in this section. We then decrease  $\mu$  to  $\mu_+ := (1 - \theta)\mu$ , for some  $\theta \in (0, 1)$  and, redefining  $\mu := \mu_+$ , a direct application of Newton’s method to (3.2) will lead to the following Newton system for the search direction  $(\Delta x, \Delta s)$ :

$$\begin{aligned} -M\Delta x + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs. \end{aligned} \tag{3.2}$$

Due to Lemma 2.1, this system has a unique solution for any  $(x, s) > 0$ . By taking a step along the search direction, one constructs a new pair  $(x_+, s_+)$  with

$$x_+ = x + \alpha\Delta x, \quad s_+ = s + \alpha\Delta s, \tag{3.3}$$

where  $\alpha \in (0, 1)$  denotes the step size, which has to be chosen appropriately. If necessary, we repeat the procedure until we find iterates that are in a certain neighborhood of  $(x(\mu), s(\mu))$ . Then  $\mu$  is again reduced by the factor  $1 - \theta$  and we apply Newton’s method targeting at the new  $\mu$ -centers, and so on. This process is repeated until  $\mu$  is small enough, for example, until

$n\mu \leq \epsilon$ , where  $\epsilon$  is a small positive number. At this stage we have found an  $\epsilon$ -approximate solution of LCP.

For the analysis of IPMs it is convenient to associate to any pair  $(x, s) \geq 0$  and  $\mu > 0$ , the vector

$$v := \sqrt{\frac{xs}{\mu}}. \quad (3.4)$$

Note that the pair  $(x, s)$  coincides with the  $\mu$ -center  $(x(\mu), s(\mu))$  if and only if  $v = \mathbf{e}$ . Next, the following *scaled search directions*  $d_x$  and  $d_s$  are introduced

$$d_x := \frac{v\Delta x}{x}, \quad d_s := \frac{v\Delta s}{s}, \quad (3.5)$$

where the operations are componentwise product and division. Using (3.4) and (3.5) the system (3.2) can be rewritten as

$$\begin{aligned} -\bar{M}d_x + d_s &= 0, \\ d_x + d_s &= v^{-1} - v, \end{aligned} \quad (3.6)$$

where  $\bar{M} := \mu VS^{-1}MXV^{-1}$ , with  $V := \text{diag}(v)$ ,  $X := \text{diag}(x)$ ,  $S := \text{diag}(s)$ .

Note that the pair  $(x, s)$  coincides with the  $\mu$ -center  $(x(\mu), s(\mu))$  if and only if  $v = \mathbf{e}$ .

A crucial observation is that the right hand side  $v^{-1} - v$  in the second equation of (3.6) equals minus the gradient of the function

$$\Psi_c(v) := \sum_{i=1}^n \left( \frac{v_i^2 - 1}{2} - \log v_i \right),$$

where  $v_i$  represents the  $i$ -th component of the vector  $v$ . In other words,

$$d_x + d_s = -\nabla \Psi_c(v).$$

This equation is called the *scaled centering equation*. Its importance arises from the fact that it essentially defines the search direction.

One may easily verify that  $\nabla^2 \Psi_c(v) = \text{diag}(\mathbf{e} + v^{-2})$ . Since this matrix is positive definite,  $\Psi_c(v)$  is strictly convex. Moreover, since  $\nabla \Psi_c(\mathbf{e}) = 0$ , it follows that  $\Psi_c(v)$  attains its minimal value at  $v\mathbf{e}$ , with  $\Psi_c(\mathbf{e}) = 0$ . Thus,  $\Psi_c(v)$  is nonnegative everywhere and vanishes if and only if  $v = \mathbf{e}$ , that is, if and only if  $x = x(\mu)$  and  $s = s(\mu)$ . Hence, we see that the  $\mu$ -center  $(x(\mu), s(\mu))$  can be characterized as the minimizer of the function  $\Psi_c(v)$ . Thus, the second important feature of the function  $\Psi_c(v)$  is that it essentially serves as a 'proximity' measure of closeness for  $(x, s)$  with respect to the  $\mu$ -center.

The above observations regarding the function  $\Psi_c(v)$  lead to an obvious generalization: we can replace  $\Psi_c(v)$  by any strictly convex function  $\Psi(v)$ ,  $v \in \mathbf{R}_{++}^n$ , such that  $\Psi(v)$  is minimal at  $v = \mathbf{e}$  and  $\Psi(\mathbf{e}) = 0$ . Thus, the new scaled centering equation becomes

$$d_x + d_s = -\nabla \Psi(v). \quad (3.7)$$

The function  $\Psi(v)$  is called a *(scaled) barrier function*. Of course, different barrier functions lead to different Newton directions, as they are calculated from the Newton system

$$\begin{aligned} -\bar{M}d_x + d_s &= 0, \\ d_x + d_s &= -\nabla \Psi(v). \end{aligned} \quad (3.8)$$

Since

$$\Psi(v) = 0 \quad \Leftrightarrow \quad \nabla \Psi(v) = 0 \quad \Leftrightarrow \quad v = \mathbf{e}$$

the function  $\Psi(v)$  still may serve as a proximity measure for closeness with respect to the  $\mu$ -center  $(x(\mu), s(\mu))$ . In the sequel we also use norm based proximity measure, namely

$$\delta(v) := \frac{1}{2} \|\nabla \Psi(v)\|. \quad (3.9)$$

The function  $\delta(v)$  is often called *proximity function* and it is easy to see that

$$\delta(v) = 0 \quad \Leftrightarrow \quad v = \mathbf{e}. \quad (3.10)$$

To simplify matters we will restrict ourselves to the case where the barrier function  $\Psi(v)$  is separable with identical coordinate functions  $\psi(v_i)$ . Thus,

$$\Psi(v) = \sum_{i=1}^n \psi(v_i), \quad (3.11)$$

where  $\psi : (0, +\infty) \rightarrow [0, +\infty)$  is twice differentiable and attains its minimum at  $t = 1$ , with  $\psi(1) = 0$ . Following the terminology introduced in [15, 4, 5], we call the univariate function  $\psi(t)$  the *kernel function* of the barrier function  $\Psi(v)$ . Obviously, in the case

$$\psi_c(t) = \frac{t^2 - 1}{2} - \log t, \quad (3.12)$$

and  $\psi(t) = \psi_c(t)$ , we have  $\Psi(v) = \Psi_c(v)$ . One may easily verify that  $\Psi_c(v)$  is the 'classical' logarithmic barrier function. We call  $\psi_c(t)$  the *classical kernel function*.

It is clear from the above discussion that the closeness of  $(x, s)$  to  $(x(\mu), s(\mu))$  can be measured by the value of  $\Psi(v)$ . Introducing a parameter  $\tau > 0$  as a threshold value, the inequality  $\Psi(v) \leq \tau$  defines a  $\tau$ -neighborhood of the  $\mu$ -center.

The generic form of this algorithm is shown in Figure 1. In this algorithm we write  $\Psi(x, s, \mu)$  instead of  $\Psi(v)$ , with  $v$  as defined in (3.4). If  $\Psi(v) \geq \tau$  we start a new *inner iteration* by computing the scaled search directions  $d_x$  and  $d_s$  at the current iterate and the current value of  $\mu$  from (3.8). Then, we compute the search directions  $\Delta x$  and  $\Delta s$  from  $d_x$  and  $d_s$  by using (3.5). Next, the new iterates are calculated using (3.3) with the appropriately calculated step-size  $\alpha$ . If necessary, the procedure is repeated until the iterate belongs to the  $\tau$ -neighborhood of the current  $\mu$ -center  $(x(\mu), s(\mu))$ , that is until  $\Psi(v) \leq \tau$ . Then, the *outer iteration* starts by reducing  $\mu$  by the factor  $1 - \theta$  with  $0 < \theta < 1$ . It is most likely that after this step the barrier function value will exceed the threshold value, that is,  $\Psi(v) > \tau$ , and hence, the algorithm enters the inner iteration loop again. This process is repeated until  $\mu$  is small enough, say until  $n\mu < \epsilon$ ; at this stage we have found an  $\epsilon$ -solution of LCP.

Note that the algorithm can be started, since, as we discussed at the beginning of this section, we may assume that a strictly feasible point  $x^0$  is given, and this point can be chosen such that  $\Psi(v^0) \leq \tau$  which means that it is in the  $\tau$ -neighborhood of the  $\mu$ -center.

The parameters  $\tau$ ,  $\theta$  and the step size  $\alpha$  in the algorithm should be tuned in such a way that the number of iterations required by the algorithm is as small as possible. In the literature two types of methods are distinguished: small-update methods and large-update methods, according to the value of the *barrier-update parameter*  $\theta$ . Large-update methods are characterized by the fact that  $\theta$  is a fixed constant ( $\theta \in (0, 1)$ ), independent of the

---

**Generic Interior-Point Algorithm for LCP**

---

**Input:**

A threshold parameter  $\tau \geq 1$ ;  
 an accuracy parameter  $\varepsilon > 0$ ;  
 a fixed barrier update parameter  $\theta$ ,  $0 < \theta < 1$ ;  
 $(x^0, s^0)$  and  $\mu^0 = (x^0)^T s^0 / n$  such that  $\Psi(x^0, s^0, \mu^0) \leq \tau$ .

**begin**

$x := x^0$ ;  $s := s^0$ ;  $\mu := \mu^0$ ;

**while**  $n\mu \geq \varepsilon$  **do**

**begin**

$\mu := (1 - \theta)\mu$ ;

**while**  $\Psi(x, s, \mu) > \tau$  **do**

**begin**

calculate search direction  $(\Delta x, \Delta s)$  using (3.8) and (3.5);

determine a step size  $\alpha$ ;

update  $x := s + \alpha\Delta x$ ;  $s := s + \alpha\Delta s$ ;

**end****end****end**

Figure 1: Generic Interior-Point Algorithm for LCP

dimension  $n$  of the problem, whereas small-update methods use a value of  $\theta$  that depends of the dimension of the problem, with  $\theta = O(\frac{1}{\sqrt{n}})$ .

The resulting iteration bound depends on a careful selection of these parameter values and it also heavily depends on the choice of the kernel function. The question which kernel function minimizes the iteration bound is still open. The goal in this paper is more limited. We will show that the Generic Algorithm described in Figure 1 for the class of the kernel functions that will be specified in the next section has the most favorable iteration bounds, that match the bounds obtained for Linear Optimization (LO) up to the scaling with a factor that depends affinely on  $\kappa$ .

#### **4 The Kernel and Barrier Function and their Properties**

As indicated above the iteration bound of the Generic Algorithm depends heavily on the choice of the kernel function. Almost all complexity results of IPMs for LCP, monotone as well as  $P_*(\kappa)$ , are based on the classical logarithmic kernel (barrier) function. The iteration bounds for large-update and small-update methods based on the classical logarithmic kernel function are  $O(n \log \frac{n}{\varepsilon})$  and  $O(\sqrt{n} \log \frac{n}{\varepsilon})$ , respectively for the monotone LCP. For the  $P_*(\kappa)$ -LCP the results are the same, up to a factor that depends affinely on  $\kappa$ . This factor equals 1 in the monotone case (when  $\kappa = 0$ ).

In the case of LO the first contribution in the design and analysis of the primal-dual IPMs based on the use of kernel functions different than the classical logarithmic kernel function was presented by Peng, Roos and Terlaky in [15]. The results were extended and

generalized by the same authors in the monograph [16]. The kernel functions introduced and analyzed in [15, 16] are so-called self-regular functions. Based upon these functions an  $O(\sqrt{n} \log n \log \frac{n}{\epsilon})$  iteration bound was obtained for large-update methods, which is currently the best iteration bound for these methods. It is a considerable improvement over the bound for large-update methods based on the classical logarithmic kernel function, which is  $O(n \log \frac{n}{\epsilon})$ . In [16] the authors extended these results for LO to  $P_*(\kappa)$ -NCP (Nonlinear Complementarity Problems), of which  $P_*(\kappa)$ -LCP is a special case, as well as to other classes of problems such as semidefinite and conic programming problems. They obtained essentially the same iteration bound as for LO, up to a factor containing  $\kappa$ .

Recently Bai *et al.* proposed kernel functions that are not necessarily self-regular [4, 5]. For some of these functions they managed to match the best known iteration bounds for LO, as just mentioned. Also, they could extend these results to semidefinite and second-order conic optimization ([6, 20]). This paper is a first attempt to generalize some of these results to  $P_*(\kappa)$ -LCP case.

The class of parametric kernel functions considered in this paper was introduced by Bai *et al.* in [7], where they managed to achieve the best known iteration bounds for LO. Our goal is to generalize these favorable complexity results to the  $P_*(\kappa)$ -LCP case. The function is given by

$$\psi_{p,q}(t) = \begin{cases} \frac{t^{p+1} - 1}{p+1} + \frac{t^{1-q} - 1}{q-1}, & t > 0, \quad p \in [0, 1], \quad q > 1 \\ \frac{t^{p+1} - 1}{p+1} - \log t, & t > 0, \quad p \in [0, 1], \quad q = 1 \end{cases} \quad (4.1)$$

where  $p$  and  $q$  are *growth* and *barrier* parameters respectively. Note that  $\lim_{q \downarrow 0} \psi_{p,q}(t) = \psi_{p,1}(t)$ . In the sequel we restrict ourselves in the analysis to the case where  $q > 1$ . But by continuity the results also apply to the case where  $q = 1$ .

The above class of kernel functions (4.1) contains several well-known kernel functions:

- For  $p = 1$  and  $q = 1$ ,  $\psi(t) = \psi_c(t)$  which is the classical logarithmic kernel function, see (3.12).
- For  $p = 1$  and  $q > 1$ ,  $\psi(t)$  is the prototype self-regular kernel function, see [16].
- For  $p = 0$  and  $q = 2$ ,  $\psi(t)$  is the simple kernel function analyzed in [4].

Let us recall from [7] that for  $0 \leq p < 1$  and  $q > 1$ ,  $\psi_{p,q}(t)$  is not a self-regular function. To simplify the notation below we denote  $\psi_{p,q}(t)$  simply as  $\psi(t)$ . According to (3.11), the corresponding scaled barrier function  $\Psi(v)$  is given by

$$\Psi(v) = \sum_{i=1}^n \psi(v_i) = \sum_{i=1}^n \left( \frac{v_i^{p+1} - 1}{p+1} + \frac{v_i^{1-q} - 1}{q-1} \right), \quad v \in \mathbf{R}_{++}^n, \quad 0 \leq p \leq 1 \quad \text{and} \quad q > 1. \quad (4.2)$$

We conclude this section by listing some useful properties of  $\psi(t)$  and  $\Psi(v)$  that are used in the complexity analysis of the algorithm in Figure 1 for  $P_*(\kappa)$ -LCPs. For the proofs of these results we refer to [7, Section 2], or [5].

First, we list the derivatives of the  $\psi(t)$  since they will play a crucial role in the analysis of the algorithm.

$$\psi'(t) = t^p - t^{-q}, \quad \psi''(t) = pt^{p-1} + qt^{-q-1}, \quad \psi'''(t) = p(p-1)t^{p-2} - q(q+1)t^{-q-2}. \quad (4.3)$$

It is quite straightforward to verify the following

$$\psi(1) = \psi'(1) = 0, \quad \lim_{t \rightarrow 0} \psi(t) = \lim_{t \rightarrow \infty} \psi(t) = +\infty.$$

Moreover, from (4.3) we conclude that  $\psi(t)$  is strictly convex and  $\psi''(t)$  is monotonically decreasing on the interval  $t \in (0, +\infty)$ .

**Lemma 4.1.** *If  $t_1 > 0$  and  $t_2 > 0$ , then*

$$\psi(\sqrt{t_1 t_2}) \leq \frac{1}{2}(\psi(t_1) + \psi(t_2)).$$

**Lemma 4.2.** *Let  $\rho(s) : [0, \infty) \rightarrow (0, 1]$  be the inverse function of  $-\frac{1}{2}\psi'(t)$  for  $t \leq 1$ . The following inequality holds*

$$\rho(s) \geq \frac{1}{(1 + 2s)^{\frac{1}{q}}}.$$

**Lemma 4.3.** *Let  $\varrho : [0, \infty) \rightarrow [1, \infty)$  be the inverse function of  $\psi(t)$  for  $t \geq 1$ . The following inequalities hold*

$$(1 + (p + 1)s)^{\frac{1}{p+1}} \leq \varrho(s) \leq 1 + s + \sqrt{s^2 + 2s}.$$

*If  $q \geq 2 - p$ , then*

$$\varrho(s) \leq 1 + \sqrt{s + s^2 + s\sqrt{s^2 + 2s}}.$$

The following result gives a lower bound for  $\delta(v)$  in terms of  $\Psi(v)$ .

**Theorem 4.4.** *The following inequality holds*

$$\delta(v) \geq \frac{1}{2}\psi'(\varrho(\Psi(v))).$$

**Corollary 4.5.** *If  $\Psi(v) \geq \tau \geq 1$ , then*

$$\delta(v) \geq \frac{1}{6}[\Psi(v)]^{\frac{p}{1+p}}.$$

## 5 The Analysis of the Algorithm

### 5.1 Growth Behavior of the Barrier Function During an Outer Iteration

In this subsection we discuss the growth behavior of the barrier function. Note that at the start of each outer iteration of the algorithm, just before the update of  $\mu$  with the factor  $1 - \theta$ , we have  $\Psi(v) \leq \tau$ . Due to the update of  $\mu$  the vector  $v$  is divided by the factor  $\sqrt{1 - \theta}$ , with  $0 < \theta < 1$ , which in general leads to an increase in the value of  $\Psi(v)$ . Then, during the subsequent inner iterations,  $\Psi(v)$  decreases until it passes the threshold value  $\tau$  again. Hence, during the course of the algorithm the largest values of  $\Psi(v)$  occur just after the updates of  $\mu$ . That is why in this section we derive an estimate for the effect of a  $\mu$ -update on the value of  $\Psi(v)$ . The results are stated without proofs; these can be found in [7, Section 3.1].

**Theorem 5.1.** *Let  $\varrho : [0, \infty) \rightarrow [1, \infty)$  be the inverse function of the kernel function  $\psi(t)$  for  $t \geq 1$ .*

*Then for any positive vector  $v$  and any  $\beta \geq 1$  the following inequality holds:*

$$\Psi(\beta v) \leq n\psi\left(\beta\varrho\left(\frac{\Psi(v)}{n}\right)\right).$$

**Corollary 5.2.** *Let  $0 \leq \theta \leq 1$  and  $v_+ = \frac{v}{\sqrt{1-\theta}}$ . If  $\Psi(v) \leq \tau$ , then*

$$\Psi(v_+) \leq n\psi \left( \frac{\varrho\left(\frac{\tau}{n}\right)}{\sqrt{1-\theta}} \right).$$

From this corollary and Lemma 4.3 the following lemma can be obtained.

**Lemma 5.3.** *We have the following upper bounds on the value of  $\Psi(v_+)$  after a  $\mu$ -update.*

$$\Psi(v_+) \leq L_1 := n\psi \left( \frac{1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}}}{\sqrt{1-\theta}} \right), \quad q > 1; \quad (5.1)$$

and

$$\Psi(v_+) \leq L_2 := n\psi \left( \frac{1 + \sqrt{\frac{\tau}{n} + \frac{\tau^2}{n^2} + \frac{\tau}{n} \sqrt{\frac{\tau^2}{n^2} + \frac{2\tau}{n}}}}{\sqrt{1-\theta}} \right), \quad q \geq 2 - p. \quad (5.2)$$

## 5.2 Determining the Default Step Size

In this subsection, we determine a default step size which not only keeps the iterations feasible but also gives rise to a sufficiently large decrease of the barrier function defined in (4.2) in each inner iteration. In each inner iteration we first compute the search directions  $\Delta x$  and  $\Delta s$  from the system (3.8). After a step-size  $\alpha$  is determined the new iterate  $(x_+, s_+)$ , is calculated by (3.3).

Recall that during an inner iteration the parameter  $\mu$  is fixed. Hence, after the step in the direction  $(\Delta x, \Delta s)$  with the step-size  $\alpha$  the new  $v$ -vector is given by

$$v_+ = \sqrt{\frac{x_+ s_+}{\mu}}.$$

Since

$$\begin{aligned} x_+ &= x \left( e + \alpha \frac{\Delta x}{x} \right) = x \left( e + \alpha \frac{d_x}{v} \right) = \frac{x}{v} (v + \alpha d_x), \\ s_+ &= s \left( e + \alpha \frac{\Delta s}{s} \right) = s \left( e + \alpha \frac{d_s}{v} \right) = \frac{s}{v} (v + \alpha d_s), \end{aligned}$$

we obtain, using  $xs = \mu v^2$ ,

$$v_+ = \sqrt{(v + \alpha d_x)(v + \alpha d_s)}.$$

Next, we consider the decrease in  $\Psi$  as a function of  $\alpha$ . We define two functions

$$f(\alpha) = \Psi(v_+) - \Psi(v),$$

and

$$f_1(\alpha) := \frac{1}{2}(\Psi(v + \alpha d_x) + \Psi(v + \alpha d_s)) - \Psi(v).$$

Lemma 4.1 implies that

$$\Psi(v_+) = \Psi(\sqrt{(v + \alpha d_x)(v + \alpha d_s)}) \leq \frac{1}{2}(\Psi(v + \alpha d_x) + \Psi(v + \alpha d_s)).$$

This inequality shows that  $f_1(\alpha)$  is an upper bound of  $f(\alpha)$ . Obviously,

$$f(0) = f_1(0) = 0.$$

Taking the derivative with respect to  $\alpha$ , we get

$$f_1'(\alpha) = \frac{1}{2} \sum_{i=1}^n (\psi'(v_i + \alpha d_{xi}) d_{xi} + \psi'(v_i + \alpha d_{si}) d_{si}).$$

To simplify the notation we used (and will use below) the following conventions:  $d_{xi} := (d_x)_i$  and  $d_{si} := (d_s)_i$ . From the above equation and using (3.7) we obtain

$$f_1'(0) = \frac{1}{2} \nabla \Psi(v)^T (d_x + d_s) = -\frac{1}{2} \nabla \Psi(v)^T \nabla \Psi(v) = -2\delta(v)^2.$$

Differentiating once again, we get

$$f_1''(\alpha) = \frac{1}{2} \sum_{i=1}^n (\psi''(v_i + \alpha d_{xi}) d_{xi}^2 + \psi''(v_i + \alpha d_{si}) d_{si}^2) > 0, \quad \text{unless } d_x d_s = 0. \quad (5.3)$$

During an inner iteration  $x$  and  $s$  are not both at the  $\mu$ -center since  $\Psi(v) \geq \tau > 0$ , so we may conclude that  $f_1(\alpha)$  is strictly convex in  $\alpha$ . It is worth pointing out that in general  $f(\alpha)$  is not convex.

In what follows we present several lemmas that are needed to obtain a suitable default value for the step-size  $\alpha$ . These lemmas are variants of lemmas that occur in [5] or [7] for the LO case. In the current LCP case there is a difficulty that does not appear in the LO case, namely that  $d_x$  and  $d_s$  are not (necessarily) orthogonal. In essence, this difficulty is dealt with in the first lemma below (Lemma 5.4). After this the other lemmas easily follow by adapting the proofs of the corresponding lemmas for the LO case to the current case.

**Lemma 5.4.** *The following inequality holds:*

$$\|(d_x; d_s)\| \leq 2\delta\sqrt{1+2\kappa}.$$

*Proof.* Since  $M$  is a  $P_*(\kappa)$ -matrix and, according to (3.2),  $\Delta s = M\Delta x$ , it follows that

$$\Delta x^T \Delta s = \Delta x^T M \Delta x \geq -4\kappa \sum_{i \in \mathcal{I}^+(\Delta x)} \Delta x_i (M \Delta x)_i = -4\kappa \sum_{i \in \mathcal{I}^+(\Delta x)} \Delta x_i \Delta s_i.$$

Because of (3.4) and (3.5) we have  $\Delta x \Delta s = \mu d_x d_s$ . Hence it follows that

$$d_x^T d_s \geq -4\kappa \sum_{i \in \mathcal{I}^+(d_x)} d_{xi} d_{si} \quad (5.4)$$

Using the arithmetic-geometric mean inequality  $ab \leq \frac{1}{4}(a+b)^2$  we obtain

$$\sum_{i \in \mathcal{I}^+(d_x)} d_{xi} d_{si} \leq \frac{1}{4} \sum_{i \in \mathcal{I}^+(d_x)} (d_{xi} + d_{si})^2 \leq \frac{1}{4} \sum_{i=1}^n (d_{xi} + d_{si})^2 = \frac{1}{4} \|d_x + d_s\|^2 = \frac{1}{4} \|\nabla \Psi(v)\|^2 = \delta^2,$$

where we used the definition (3.9) of  $\delta$  and (3.8). Substitution of this inequality into (5.4) yields

$$d_x^T d_s \geq -4\kappa \delta^2.$$

As a consequence we may now write:

$$\begin{aligned}\|(d_x; d_s)\|^2 &= \sum_{i=1}^n (d_{x_i}^2 + d_{s_i}^2) = \sum_{i=1}^n ((d_{x_i} + d_{s_i})^2 - 2d_{x_i}d_{s_i}) \\ &= \|d_x + d_s\|^2 - 2d_x^T d_s = 4\delta^2 - 2d_x^T d_s \\ &\leq 4\delta^2 + 8\kappa\delta^2 = 4(1 + 2\kappa)\delta^2.\end{aligned}$$

This proves the lemma.  $\square$

Immediate consequences of the above lemma are the following inequalities

$$\|d_x\| \leq 2\delta\sqrt{1 + 2\kappa}, \quad \|d_s\| \leq 2\delta\sqrt{1 + 2\kappa}. \quad (5.5)$$

Let us point out that similar inequalities can be found in other papers on  $P_*(\kappa)$ -LCP, e.g., [2, 16]. Equipped with these inequalities we can deal with the lemmas that follow in the rest of this section.

From now on we assume, without loss of generality, that the coordinates of  $v$  are ordered such that

$$v_1 \leq v_2 \leq \dots \leq v_n.$$

As a consequence we have  $v_{\min} = v_1$ .

**Lemma 5.5.** *The following inequality holds*

$$f_1''(\alpha) \leq 2(1 + 2\kappa) \delta^2 \psi''(v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}).$$

*Proof.* Using (5.5) we obtain the following inequalities

$$v_i + \alpha d_{x_i} \geq v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}, \quad v_i + \alpha d_{s_i} \geq v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}, \quad 1 \leq i \leq n.$$

Substituting these inequalities into the expression (5.3) for  $f_1''(\alpha)$ , while using the fact that  $\psi''(t)$  is monotonically decreasing (due to (4.3)), we get

$$f_1''(\alpha) \leq \frac{1}{2} \psi''(v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}) \sum_{i=1}^n (d_{x_i}^2 + d_{s_i}^2) = \frac{1}{2} \psi''(v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}) \|(d_x; d_s)\|^2.$$

Now using Lemma 5.4 we obtain the desired inequality.  $\square$

**Lemma 5.6.** *If the step size  $\alpha$  satisfies*

$$-\psi'(v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}) + \psi'(v_1) \leq \frac{2\delta}{\sqrt{1 + 2\kappa}}, \quad (5.6)$$

then  $f_1'(\alpha) \leq 0$ .

*Proof.* Using Lemma 5.5 we have the following derivation

$$\begin{aligned}f_1'(\alpha) &= f_1'(0) + \int_0^\alpha f_1''(\zeta) d\zeta \\ &\leq -2\delta^2 + 2\delta^2(1 + 2\kappa) \int_0^\alpha \psi''(v_1 - 2\zeta\delta\sqrt{1 + 2\kappa}) d\zeta \\ &= -2\delta^2 - \delta\sqrt{1 + 2\kappa} \int_0^\alpha \psi''(v_1 - 2\zeta\delta\sqrt{1 + 2\kappa}) d(v_1 - 2\zeta\delta\sqrt{1 + 2\kappa}) \\ &= -2\delta^2 + \delta\sqrt{1 + 2\kappa} (-\psi'(v_1 - 2\alpha\delta\sqrt{1 + 2\kappa}) + \psi'(v_1)) \\ &\leq -2\delta^2 + \delta\sqrt{1 + 2\kappa} \frac{2\delta}{\sqrt{1 + 2\kappa}} = 0.\end{aligned}$$

which proves the lemma. The first inequality is due to Lemma 5.5, while the second inequality follows from the hypothesis (5.6) of the lemma.  $\square$

**Lemma 5.7.** *The largest possible value of the step size  $\alpha$  satisfying (5.6) is given by*

$$\bar{\alpha} : \frac{1}{2\delta\sqrt{1+2\kappa}} \left[ \rho(\delta) - \rho \left( \left( 1 + \frac{1}{\sqrt{1+2\kappa}} \right) \delta \right) \right]. \quad (5.7)$$

*Proof.* We want to compute the step size  $\alpha$  such that (5.6) holds, with  $\alpha$  as large as possible. The derivative with respect to  $v_1$  of the left-hand side in the inequality (5.6) is  $-\psi''(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) + \psi''(v_1)$ . Since  $\psi''(t)$  is decreasing, this derivative is negative. Therefore, the left-hand side is a decreasing function of  $\alpha$ . Hence, fixing  $\delta$ , the smaller  $v_1$  is, the smaller the maximal value of  $\alpha$  will be. We have

$$\delta = \frac{1}{2} \|\nabla\Psi(v)\| \geq \frac{1}{2} |\psi'(v_1)| \geq -\frac{1}{2}\psi'(v_1).$$

Equality holds if and only if  $v_1$  is the only coordinate in  $v$  that differs from 1, and  $v_1 \leq 1$  (in which case  $\psi'(v_1) \leq 0$ ). Hence, the worst situation for the step size occurs when  $v_1$  satisfies

$$-\frac{1}{2}\psi'(v_1) = \delta. \quad (5.8)$$

The derivative with respect to  $\alpha$  of the left-hand side in (5.6) equals

$$2\delta\sqrt{1+2\kappa} \psi''(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) \geq 0,$$

and, hence, the left side of the inequality (5.6) is increasing in  $\alpha$ . Thus, the largest possible value of  $\alpha$  satisfying (5.6), satisfies

$$-\psi'(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) + \psi'(v_1) = \frac{2\delta}{\sqrt{1+2\kappa}}.$$

Due to (5.8) the above equation can be written as

$$-\frac{1}{2}\psi'(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) \left( 1 + \frac{1}{\sqrt{1+2\kappa}} \right) \delta. \quad (5.9)$$

By the definition of the inverse function  $\rho$  (cf. Lemma 4.2), the equations (5.8) and (5.9) can be written as

$$v_1 = \rho(\delta), \quad v_1 - 2\alpha\delta\sqrt{1+2\kappa} = \rho \left[ \left( 1 + \frac{1}{\sqrt{1+2\kappa}} \right) \delta \right].$$

Thus, it follows that

$$\alpha = \frac{1}{2\delta\sqrt{1+2\kappa}} \left( \rho(\delta) - \rho \left( \left( 1 + \frac{1}{\sqrt{1+2\kappa}} \right) \delta \right) \right),$$

and the lemma is proved.  $\square$

The term  $1 + \frac{1}{\sqrt{1+2\kappa}}$  appears frequently in the sequel, so in order to simplify the notation we introduce the following abbreviation:

$$K := 1 + \frac{1}{\sqrt{1+2\kappa}}. \quad (5.10)$$

**Lemma 5.8.** *With  $\rho$ ,  $\bar{\alpha}$  and  $K$  as defined above, we have*

$$\bar{\alpha} \geq \frac{1}{(1+2\kappa)\psi''(\rho(K\delta))}. \quad (5.11)$$

*Proof.* By the definition of  $\rho$ , we have  $-\psi'(\rho(\delta)) = 2\delta$ . Taking the derivative with respect to  $\delta$ , we find

$$-\psi''(\rho(\delta))\rho'(\delta) = 2,$$

which leads to

$$\rho'(\delta) = -\frac{2}{\psi''(\rho(\delta))} < 0.$$

Hence,  $\rho$  is monotonically decreasing. Using the fundamental theorem of calculus, the expression (5.7) for the step size  $\bar{\alpha}$  can be transformed as follows:

$$\bar{\alpha} = \frac{1}{2\delta\sqrt{1+2\kappa}}(\rho(\delta) - \rho(K\delta)) = \frac{1}{2\delta\sqrt{1+2\kappa}} \int_{K\delta}^{\delta} \rho'(\sigma) d\sigma = \frac{1}{\delta\sqrt{1+2\kappa}} \int_{\delta}^{K\delta} \frac{d\sigma}{\psi''(\rho(\sigma))}. \quad (5.12)$$

To obtain a lower bound for  $\bar{\alpha}$  we want to replace the argument of the last integral by its minimal value. Thus, we would like to know when  $\psi''(\rho(\sigma))$  is maximal for  $\sigma \in (\delta, K\delta)$ . Since  $\psi''$  is monotonically decreasing,  $\psi''(\rho(\sigma))$  is maximal for  $\sigma \in (\delta, K\delta)$  when  $\rho(\sigma)$  is minimal. Since  $\rho$  is monotonically decreasing, this occurs when  $\sigma = K\delta$ . Therefore, using (5.12) and (5.10) we obtain

$$\bar{\alpha} = \frac{1}{\delta\sqrt{1+2\kappa}} \int_{\delta}^{K\delta} \frac{d\sigma}{\psi''(\rho(\sigma))} \geq \frac{1}{\delta\sqrt{1+2\kappa}} \frac{1}{\psi''(\rho(K\delta))} \int_{\delta}^{K\delta} d\sigma = \frac{1}{1+2\kappa} \frac{1}{\psi''(\rho(K\delta))},$$

and the lemma is proved.  $\square$

**Theorem 5.9.** *If  $\bar{\alpha}$  is defined by (5.7), then the following inequality holds*

$$\bar{\alpha} \geq \frac{1}{(1+2\kappa)(p+q)(1+2K\delta)^{\frac{1+q}{q}}}.$$

*Proof.* By Lemma 4.2 we have

$$\rho(K\delta) \geq (1+2K\delta)^{-\frac{1}{q}}.$$

Since  $\psi''(t)$  is monotonically decreasing in  $t \in (0, \infty)$ , we get

$$\psi''(\rho(K\delta)) \leq \psi''\left((1+2K\delta)^{-\frac{1}{q}}\right).$$

Hence, using (5.11) and (4.3) we obtain

$$\begin{aligned} \bar{\alpha} &\geq \frac{1}{(1+2\kappa)\psi''(\rho(K\delta))} \geq \frac{1}{1+2\kappa} \frac{1}{p(1+2K\delta)^{\frac{1-p}{q}} + q(1+2K\delta)^{\frac{1+q}{q}}} \\ &\geq \frac{1}{(1+2\kappa)(p+q)(1+2K\delta)^{\frac{q+1}{q}}}, \end{aligned}$$

which is the desired inequality.  $\square$

In the analysis of the Generic Algorithm described in the Figure 1 we use

$$\tilde{\alpha} := \frac{1}{(1+2\kappa)(p+q)(1+2K\delta)^{\frac{q+1}{q}}}. \quad (5.13)$$

as the default step size. Note that  $\tilde{\alpha} \leq \bar{\alpha}$ .

### 5.3 Decrease of the Barrier Function During an Inner Iteration

In this section we show that the default step size (5.13) yields sufficient decrease of the barrier function value during each inner iteration.

**Lemma 5.10.** *If the step size  $\alpha$  is such that  $\alpha \leq \bar{\alpha}$ , where  $\bar{\alpha}$  is defined by (5.7), then*

$$f(\alpha) \leq -\alpha\delta^2.$$

*Proof.* Let the univariate function  $h$  be such that

$$h(0) = f_1(0) = 0, \quad h'(0) = f_1'(0) = -2\delta^2, \quad h''(\alpha) = 2(1+2\kappa)\delta^2\psi''(v_1 - 2\alpha\delta\sqrt{1+2\kappa}).$$

According to Lemma 5.5 we have  $f_1''(\alpha) \leq h''(\alpha)$  which implies  $f_1'(\alpha) \leq h'(\alpha)$  and  $f_1(\alpha) \leq h(\alpha)$ . Taking  $\alpha \leq \bar{\alpha}$ , with  $\bar{\alpha}$  as defined by (5.7), and using the fundamental theorem of calculus we get

$$\begin{aligned} h'(\alpha) &= \int_0^\alpha h''(\xi) d\xi + h'(0) \\ &= -2\delta^2 + 2(1+2\kappa)\delta^2 \int_0^\alpha \psi''(v_1 - 2\xi\delta\sqrt{1+2\kappa}) d\xi \\ &= -2\delta^2 - \delta\sqrt{1+2\kappa} (\psi'(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) - \psi'(v_1)) \\ &\leq -2\delta^2 - \delta\sqrt{1+2\kappa} \frac{2\delta}{\sqrt{1+2\kappa}} = 0. \end{aligned}$$

The last inequality is due to the definition of  $\bar{\alpha}$ , which guarantees that if  $\alpha \leq \bar{\alpha}$  then inequality (5.6) in Lemma 5.6 holds. Since  $\psi'''(t) < 0$  for  $t > 0$ ,  $\psi''(t)$  is decreasing in  $t$ , and therefore  $h''(\alpha)$  is increasing in  $\alpha$ . Using Lemma A.2, we get

$$f_1(\alpha) \leq h(\alpha) \leq \frac{1}{2}\alpha h'(0) = -\alpha\delta^2.$$

As we mentioned before,  $f_1(\alpha)$  is an upper bound of  $f(\alpha)$ , hence, the lemma is proved.  $\square$

**Theorem 5.11.** *If  $\tilde{\alpha}$  is the default step size, as defined by (5.13), then*

$$f(\tilde{\alpha}) \leq -\frac{\Psi(v)^{\frac{p(q-1)}{q(p+1)}}}{100(1+2\kappa)(p+q)}.$$

*Proof.* Since  $\tilde{\alpha} \leq \bar{\alpha}$  we apply Lemma 5.10 with  $\alpha = \tilde{\alpha}$ . Also using the definition (5.13) of  $\tilde{\alpha}$  we obtain

$$f(\tilde{\alpha}) \leq -\tilde{\alpha}\delta^2 = -\frac{\delta^2}{(1+2\kappa)(p+q)(1+2K\delta)^{\frac{q+1}{q}}}.$$

Since  $q > 1$ , it can easily be verified that the last expression above is monotonically decreasing in  $\delta$ . Hence, using this fact and Corollary 4.5 we get

$$f(\tilde{\alpha}) \leq -\frac{\Psi(v)^{\frac{2p}{p+1}}}{36(1+2\kappa)(p+q) \left[1 + \frac{K}{3}\Psi(v)^{\frac{p}{p+1}}\right]^{\frac{q+1}{q}}}. \quad (5.14)$$

The rest of the proof consist of simplifying the last expression. Using the fact that  $1 < \tau \leq \Psi(v)$  we may write

$$\left(1 + \frac{K}{3}\Psi(v)^{\frac{p}{p+1}}\right)^{\frac{q+1}{q}} \leq \left(\Psi(v)^{\frac{p}{p+1}} + \frac{K}{3}\Psi(v)^{\frac{p}{p+1}}\right)^{\frac{q+1}{q}} = \left(1 + \frac{K}{3}\right)^{\frac{q+1}{q}} \Psi(v)^{\frac{p}{p+1} \frac{q+1}{q}}.$$

Since  $K \leq 2$  and  $q > 1$ , it follows that

$$\left(1 + \frac{K}{3} \Psi(v)^{\frac{p}{p+1}}\right)^{\frac{q+1}{q}} \leq \left(1 + \frac{2}{3}\right)^{\frac{q+1}{q}} \Psi(v)^{\frac{p}{p+1} \frac{q+1}{q}} \leq \left(\frac{5}{3}\right)^2 \Psi(v)^{\frac{p}{p+1} \frac{q+1}{q}} = \frac{25}{9} \Psi(v)^{\frac{p}{p+1} \frac{q+1}{q}}.$$

Substitution into (5.14) leads to

$$f(\tilde{\alpha}) \leq -\frac{\Psi(v)^{\frac{p(q-1)}{q(p+1)}}}{100(1+2\kappa)(p+q)}$$

which is the desired inequity. Thus, the theorem has been proved.  $\square$

## 6 Complexity of the Algorithm

In this section the name *Algorithm* will refer to the Generic Algorithm of Figure 1 for the specific case when the kernel function is defined by (4.1) and the step size is given by (5.13). In the previous sections we have found all the ingredients that we need for deriving an upper bound for the number of iterations required by the Algorithm. In the first subsection below we derive an iteration bound for large-update methods and in the second subsection an iteration bound for small-update methods.

### 6.1 Iteration Bound for the Large-Update Method

First we count how many inner iterations are required by the Algorithm to return to the situation where  $\Psi(v) \leq \tau$  after a  $\mu$ -update. We denote the value of  $\Psi(v)$  after the  $\mu$ -update as  $\Psi_0$ , and the subsequent values in the same outer iteration are denoted as  $\Psi_k$ ,  $k = 1, 2, \dots, \bar{K}$ , where  $\bar{K}$  denotes the total number of inner iterations in the outer iteration. By using (5.1) in Lemma 5.3, we have

$$\Psi_0 \leq n\psi \left( \frac{1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}}}{\sqrt{1-\theta}} \right).$$

Since  $\psi(t) \leq \frac{t^{p+1}-1}{p+1}$  when  $t \geq 1$

$$\Psi_0 \leq \frac{n}{p+1} \left( \left[ \frac{1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}}}{\sqrt{1-\theta}} \right]^{p+1} - 1 \right).$$

For  $a > 0$  and  $p \in [0, 1]$ , by using Lemma A.1, one has

$$a^{p+1} - 1 = a \cdot (1 + (a-1))^p - 1 \leq a(1 + p(a-1)) - 1 = (a-1)(1 + ap)$$

which leads to

$$\begin{aligned} \Psi_0 &\leq \frac{n}{p+1} \left( \frac{1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}}}{\sqrt{1-\theta}} - 1 \right) \left( 1 + \frac{1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}}}{\sqrt{1-\theta}} p \right) \\ &= \frac{n}{(p+1)(1-\theta)} \left( 1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}} - \sqrt{1-\theta} \right) \left( \sqrt{1-\theta} + \left( 1 + \frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}} \right) p \right). \end{aligned}$$

Assuming that

$$\frac{\tau}{n} + \sqrt{\left(\frac{\tau}{n}\right)^2 + \frac{2\tau}{n}} \leq 2$$

and using the inequality  $1 - \sqrt{1 - \theta} \leq \theta$ , we obtain

$$\Psi_0 \leq \frac{4(n\theta + \tau + \sqrt{\tau^2 + 2\tau n})}{(p+1)(1-\theta)}.$$

Due to the Theorem 5.11 we have

$$\Psi_{k+1} \leq \Psi_k - \beta(\Psi_k)^{1-\gamma}, \quad k = 0, 1, \dots, \bar{K} - 1,$$

with

$$\beta = \frac{1}{100(1+2\kappa)(p+q)} \quad \text{and} \quad \gamma = \frac{p+q}{q(p+1)}.$$

Hence, by Lemma A.3, we obtain the following upper bound for the number  $\bar{K}$  of inner iterations.

$$\bar{K} \leq 100(1+2\kappa)q(p+1)(\Psi_0)^{\frac{p+q}{q(p+1)}} \quad (6.1)$$

$$\leq 100(1+2\kappa)q(p+1) \left( \frac{4(n\theta + \tau + \sqrt{\tau^2 + 2\tau n})}{(p+1)(1-\theta)} \right)^{\frac{p+q}{q(p+1)}}. \quad (6.2)$$

Now we can derive an upper bound for the total number of iterations needed by the large-update version of the Algorithm.

**Theorem 6.1.** *Given that  $\theta = \Theta(1)$ , and  $\tau = O(n)$ , which are characteristics of large-update methods, the Algorithm will obtain an  $\epsilon$ -approximate solution of  $P_*(\kappa)$ -LCP in at most*

$$O\left((1+2\kappa)q(p+1)n^{\frac{p+q}{q(p+1)}} \log \frac{n}{\epsilon}\right) \quad (6.3)$$

iterations.

*Proof.* It is well known that the number of outer iterations is bounded above by [17, Lemma II.17, page 116]

$$\frac{1}{\theta} \log \frac{n}{\epsilon}. \quad (6.4)$$

By multiplying this number with the upper bound for the number of inner iterations per outer iteration we get the upper bound for the total number of iterations, namely

$$100(1+2\kappa) \frac{q(p+1)}{\theta} \left( \frac{4(n\theta + \tau + \sqrt{\tau^2 + 2\tau n})}{(p+1)(1-\theta)} \right)^{\frac{p+q}{q(p+1)}} \log \frac{n}{\epsilon}.$$

One may easily verify that this agrees with the iteration bound stated in the theorem.  $\square$

One also easily verifies that if  $p > 0$  then the expression  $q(1+p)n^{\frac{p+q}{q(p+1)}}$  is convex in  $q$ . Moreover, it is minimal for  $q = \frac{p \log n}{p+1}$  and then it equals  $p e^{-1} n^{\frac{1}{1+p}} \log n$ , where  $e$  is the base of the natural logarithm function. Hence, if  $\log n > \frac{p+1}{p}$  and  $q = \frac{p \log n}{p+1}$  then the iteration bound (6.3) in Theorem 6.1 becomes

$$O\left((1+2\kappa)p^{p+1} \sqrt[p]{n} (\log n) \log \frac{n}{\epsilon}\right), \quad q = \frac{p \log n}{p+1} > 1. \quad (6.5)$$

Note that the last inequality implies

$$p > \frac{1}{\log n - 1}.$$

In the limiting case, when the above inequality holds with equality (which corresponds to  $q = 1$ ), the bound (6.5) becomes

$$O\left((1 + 2\kappa)\frac{n \log n}{\log n - 1} \log \frac{n}{\epsilon}\right) \equiv O\left((1 + 2\kappa)n \log \frac{n}{\epsilon}\right). \quad (6.6)$$

It is interesting to apply the above bounds to some special cases of the parameters  $p$  and  $q$ .

- $p = 1, q > 1$  : In this case  $\psi(t)$  is the prototype self-regular kernel function. From (6.5) we obtain the iteration bound  $O\left((1 + 2\kappa)\sqrt{n} \log n \log \frac{n}{\epsilon}\right)$ , which is currently the best known bound for large-update methods. This result matches the result obtained by Peng *et al.* [14].
- $p = 1, q = 1$  : In this case  $\psi(t)$  is the classical logarithmic kernel function. The bound in Theorem 6.1 simplifies to  $O\left((1 + 2\kappa)n \log \frac{n}{\epsilon}\right)$  which is the usual bound for large-update methods based on the logarithmic kernel function (see, e.g., [2, 12]).
- $p = 0, q = 2$  : In this case  $\psi(t) = t - \frac{1}{t} - 2$  which is, from an algebraic point of view, the simplest kernel function. Theorem 6.1 yields the same bound as for the logarithmic barrier function.

It is worth pointing out again that the class of kernel functions  $\psi(t)$  defined in (4.1) is not self-regular for  $0 \leq p < 1$ . Thus, the complexity results for  $0 \leq p < 1$  are new, since up to now all complexity results for  $P_*(\kappa)$ -LCP were developed for either the classical logarithmic kernel function or for the self-regular kernel functions.

## 6.2 Iteration Bound for the Small-Update Method

When applying the above analysis to small-update methods the resulting iteration bound is not as good as it can be for these types of methods. A better result is obtained by using (5.2) in Lemma 5.3:

$$\Psi_0 \leq n\psi\left(\frac{1 + \sqrt{\frac{\tau}{n} + \frac{\tau^2}{n^2} + \frac{\tau}{n}\sqrt{\frac{\tau^2}{n^2} + \frac{2\tau}{n}}}}{\sqrt{1 - \theta}}\right), \quad q \geq 2 - p.$$

Using the fact that  $\psi(t) \leq \frac{1}{2}\psi''(1)(t - 1)^2$  and  $\psi''(1) = p + q$ , we get (cf. [7, Lemma 2.2])

$$\Psi_0 \leq \frac{(p + q)n}{2} \left( \frac{1 + \sqrt{\frac{\tau}{n} + \frac{\tau^2}{n^2} + \frac{\tau}{n}\sqrt{\frac{\tau^2}{n^2} + \frac{2\tau}{n}}}}{\sqrt{1 - \theta}} - 1 \right)^2.$$

Applying again the inequality  $1 - \sqrt{1 - \theta} \leq \theta$ , it follows that

$$\Psi_0 \leq \frac{p + q}{2(1 - \theta)} \left( \theta\sqrt{n} + \sqrt{\tau + \frac{\tau^2}{n} + \tau\sqrt{\frac{\tau^2}{n^2} + \frac{2\tau}{n}}} \right)^2.$$

Therefore, by following the same line of arguments as for large-update methods, we get the following upper bound for the number  $\bar{K}$  of inner iterations during one outer iteration:

$$\bar{K} \leq 100(1 + 2\kappa)q(p + 1) \left( \frac{p + q}{2(1 - \theta)} \left[ \theta\sqrt{n} + \sqrt{\tau + \frac{\tau^2}{n} + \tau\sqrt{\frac{\tau^2}{n^2} + \frac{2\tau}{n}}} \right]^2 \right)^{\frac{p+q}{q(p+1)}}.$$

Using the upper bound (6.4) for the number of outer iterations we obtain the following bound for the total number of iterations:

$$100(1 + 2\kappa)\frac{q(p + 1)}{\theta} \left( \frac{p + q}{2(1 - \theta)} \left[ \theta\sqrt{n} + \sqrt{\tau + \frac{\tau^2}{n} + \tau\sqrt{\frac{\tau^2}{n^2} + \frac{2\tau}{n}}} \right]^2 \right)^{\frac{p+q}{q(p+1)}} \log \frac{n}{\epsilon}.$$

For small-update methods we have  $\theta = \Theta(\frac{1}{\sqrt{n}})$  and  $\tau = O(1)$ . Then, the term between the square brackets in the above expression is  $O(1)$ . The exponent  $\frac{p+q}{q(p+1)}$  is also  $O(1)$  since it is at most 1 (which happens if  $q = 1$ ). Thus, we get the bound

$$O\left((1 + 2\kappa)q(p + 1)(p + q)\sqrt{n}\log\frac{n}{\epsilon}\right).$$

Since  $p \in [0, 1]$ , we have  $q(p + 1)(p + q) = O(q^2)$  (in fact  $q(p + 1)(p + q) = q^2$  if  $p = 0$ ). Hence, we have proved the following result.

**Theorem 6.2.** *Given that  $\theta = \Theta(\frac{1}{\sqrt{n}})$  and  $\tau = O(1)$ , which are characteristics of the small-update methods, the Algorithm will obtain an  $\epsilon$ -approximate solution in at most*

$$O\left((1 + 2\kappa)q^2\sqrt{n}\log\frac{n}{\epsilon}\right)$$

*iterations.*

## 7 Conclusion

In this paper we have analyzed large- and small-update versions of the IPM for  $P_*(\kappa)$ -LCP described in the Figure 1, while using the class of kernel functions (4.1) and with the default step size (5.13). The class of kernel functions (4.1) is important because it is fairly general and includes the classical logarithmic kernel function, the prototype self-regular kernel function, and non-self-regular kernel functions as special cases. This class was first introduced by Bai *et al.* in [7] for LO. As far as we are aware this is the first result on IPM for  $P_*(\kappa)$ -LCP based on this class of kernel functions.

The generalization from LO to  $P_*(\kappa)$ -LCP required several new arguments. In the LO case the orthogonality of the scaled search directions  $d_x$  and  $d_s$  makes it much easier to get upper bounds for  $\|d_x\|$  and  $\|d_s\|$  in terms of the proximity function  $\delta(v)$ . In the  $P_*(\kappa)$ -LCP case Lemma 5.4 serves this purpose. The new upper bounds, which depend on the parameter  $\kappa$ , required careful reexamination and modification of all the subsequent results. At the end, the iteration bound differ from the ones obtained in LO case only by a factor  $1 + 2\kappa$ .

The iteration bounds obtained in this paper are as good as they can be in the current state-of-the-art. They can be adapted to match the best known bounds for self-regular kernel functions (see [14, 16]). They also match well-known results for the logarithmic kernel

function (see for example [2, 12]). However, self-regular and logarithmic kernel functions are just special members of the class considered in this paper. They occur by taking  $p = 1, q > 1$  and  $p = 1, q = 1$  respectively. For  $0 \leq p < 1$ , our kernel functions are not self-regular. Thus, the algorithm and complexity results for  $0 \leq p < 1$  are new, because up to now all complexity results for  $P_*(\kappa)$ -LCP were developed for either the classical logarithmic or self-regular kernel functions.

Possible directions for further research include numerical studies to compare the new method with existing methods. Favorable theoretical complexity bounds in general do not necessarily mean that the numerical behavior of this method will be competitive with other methods. Additional, more theoretical, questions of interest include the development and complexity analysis of algorithms based on different kernel functions. Also, it would be interesting if our results could be generalized to nonlinear complementarity problems.

### **A** Three Technical Lemmas

We list three simple technical lemmas because they are used in the complexity analysis of the algorithms. The original proofs of these lemmas can be found in [4, 15, 16], respectively.

**Lemma A.1 (Lemma 20 in [4]).** *If  $\alpha \in [0, 1]$  and  $t \geq -1$ , then  $(1 + t)^\alpha \leq 1 + \alpha t$ .*

**Lemma A.2 (Lemma 12 in [15]).** *Let  $h(t)$  be a twice differentiable convex function with  $h(0) = 0$  and  $h'(0) < 0$ , and let  $h(t)$  attain its (global) minimum at  $t^* > 0$ . If  $h''(t)$  is monotonically increasing for  $t \in [0, t^*]$ , then one has*

$$h(t) \leq \frac{th'(0)}{2}, \quad 0 \leq t \leq t^*.$$

**Lemma A.3 (Proposition 2.2 in [16]).** *Let  $t_0, t_1, \dots, t_K$  be a sequence of positive numbers such that*

$$t_{k+1} \leq t_k - \beta t_k^{1-\gamma}, \quad k = 0, 1, \dots, K-1,$$

*where  $\beta > 0$  and  $0 < \gamma \leq 1$ . Then  $K \leq \left\lfloor \frac{t_0^\gamma}{\beta\gamma} \right\rfloor$ .*

### References

- [1] E.D. Andersen and Y. Ye, On a homogeneous algorithm for the monotone complementarity problem, *Math. Program.* 84 (1999) 375–399.
- [2] M. Anitescu, G. Lesaja, and F.A. Potra, An infesible interior-point predictor-corrector algorithm for the  $P_*$ -geometric LCP, *Appl. Math. Optim.* 36 (1997) 203–228.
- [3] M. Anitescu, G. Lesaja, and F.A. Potra, Equivalence between different formulations of the linear complementarity problem, *Optim. Methods Softw.* 7 (1997) 265–290.
- [4] Y.Q. Bai, C. Roos, A polynomial-time algorithm for linear optimization based on a new simple kernel function, *Optim. Methods Softw.* 18 (2003) 631–646.
- [5] Y.Q. Bai, M.E. Ghami and C. Roos, A comparative study of kernel functions for primal-dual interior-point algorithms in linear optimization, *SIAM J. Optim.* 15 (2004) 101–128.

- [6] Y.Q. Bai, G.Q. Wang, Primal-dual interior-point algorithms for second-order cone optimization based on a new parametric kernel function, *Acta Math. Sin. English Series*, DOI: 10.1107/s10114-007-0967-z, Sep. 10, 2007.
- [7] Y.Q. Bai, G. Lesaja, C. Roos, G.Q. Wang and M. El Ghami, A class of large-update and small-update primal-dual interior-point algorithms for linear optimization, Accepted by *J. Optim. Theory Appl.* September, 2007.
- [8] S.J. Chung, A note on the complexity of LCP. The LCP is strongly NP-complete. Technical Report 792. Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, Michigan, 1979.
- [9] R.W. Cottle, J.S. Pang and R.E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, MA, 1992.
- [10] F. Facchinei and J.S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problem*, Springer, New York, 2003.
- [11] N. Megiddo, Pathways to the optimal set in linear programming, in *Progress in Mathematical Programming: Interior Point and Related Methods*, N. Megiddo (ed.), Springer Verlag, New York, 1989, pp. 131–158. Identical version in *Proceedings of the 6th Mathematical Programming Symposium of Japan, Nagoya, Japan*, 1986, pp. 1–35.
- [12] M. Kojima, N. Megiddo, T. Noma and A. Yoshise, *A Unified Approach to Interior Point Algorithms for Linear Complementarity problems*, Lecture Notes in Computer Science 538, Springer-Verlag, New York, 1991.
- [13] R.D.C. Monteiro and S.J. Wright, A superlinear infeasible-interior-point affine scaling algorithm for LCP, *SIAM J. Optim.* 6 (1996) 1–20, .
- [14] J. Peng, C. Roos and T. Terlaky, New complexity analysis of primal-dual Newton methods for  $P_*(k)$  - nonlinear complementarity problems, in *High Performance Optimization*, H. Frenk, C. Roos, T. Terlaky and S. Zhang (eds.), Kluwer Academic Publication, Boston, 1999, pp. 245–265.
- [15] J. Peng, C. Roos and T. Terlaky, Self-regular functions and new search directions for linear and semidefinite optimization, *Math. Program. (Ser. A)* 93 (2002) 129–171.
- [16] J. Peng, C. Roos and T. Terlaky, *Self-Regularity: A New Paradigm for Primal-Dual Interior-Point Algorithms*, Princeton University Press, 2002.
- [17] C. Roos, T. Terlaky and J.P. Vial, *Theory and Algorithms for Linear Optimization. An Interior-Point Approach*, Springer Science, 2005.
- [18] H. Väliäho,  $P_*$ -matrices are just sufficient, *Linear Algebra Appl.* 239 (1999) 103–108.
- [19] J. Sun and Z.H. Huang, A smoothing Newton algorithm for the LCP with a sufficient matrix that terminates finitely at a maximally complementary solution, *Optim. Methods Softw.* 21 (2006) 597–615.
- [20] G.Q. Wang, Y.Q. Bai and C. Roos, Primal-dual interior-point algorithms for semidefinite optimization based on a simple kernel function, *J. Math. Model. Algorithms* 4 (2005) 409–433.

- [21] S.J. Wright and Y. Zhang, A superquadratic infeasible-interior-point method for linear complementarity problems, *Math. Program.* 73 (1996) 269–289.
  - [22] Y. Ye, On homogeneous and self-dual algorithms for LCP, *Math. Program.* 76 (1997) 211–221.
  - [23] P.A. Zhang and X.S. Li, An infeasible-start path-following method for monotone LCPs, *Math. Comput. Modelling* 38 (2003) 23–31.
- 

*Manuscript received 5 February 2007*

*revised 23 June 2007*

*accepted for publication 1 July 2007*

Y.Q. BAI  
Department of Mathematics, Shanghai University  
Shanghai, 200444, China  
E-mail address: [yqbai@shu.edu.cn](mailto:yqbai@shu.edu.cn)

G. LESAJA  
Mathematical Sciences, Georgia Southern University  
Statesboro, GA 30460-8093, USA  
E-mail address: [goran@GeorgiaSouthern.edu](mailto:goran@GeorgiaSouthern.edu)

C. ROOS  
Faculty of Information Technology and Systems, Delft University of Technology  
E-mail address: [c.roos@tudelft.nl](mailto:c.roos@tudelft.nl)