

UNIFIED ANALYSIS OF KERNEL-BASED INTERIOR-POINT METHODS FOR $P_*(\kappa)$ -LINEAR COMPLEMENTARITY PROBLEMS*

G. LESAJA[†] AND C. ROOS[‡]

Abstract. We present an interior-point method for the $P_*(\kappa)$ -linear complementarity problem (LCP) that is based on barrier functions which are defined by a large class of univariate functions called *eligible kernel functions*. This class is fairly general and includes the classical logarithmic function and the self-regular functions, as well as many non-self-regular functions as special cases. We provide a unified analysis of the method and give a general scheme on how to calculate the iteration bounds for the entire class. We also calculate the iteration bounds of both long-step and short-step versions of the method for several specific eligible kernel functions. For some of them we match the best known iteration bounds for the long-step method, while for the short-step method the iteration bounds are of the same order of magnitude. As far as we know, this is the first paper that provides a unified approach and comprehensive treatment of interior-point methods for $P_*(\kappa)$ -LCPs based on the entire class of eligible kernel functions.

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

AMS subject classifications. 90C22, 90C31

DOI. 10.1137/090766735

1. Introduction. In this paper we consider a class of linear complementarity problems (LCPs) 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

$$(1) \quad s = Mx + q, \quad xs = 0, \quad x, s \geq 0,$$

where xs denotes the componentwise (Hadamard) product of the vectors of x and s .

Although this is a feasibility and not an optimization problem, it is closely related to optimization problems. It is well known that the Karush–Kuhn–Tucker (KKT) optimality conditions for linear optimization (LO) and convex quadratic optimization can be written as LCPs. Moreover, LCPs also have 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 LCPs. For a comprehensive treatment of LCP theory and practice, we refer the reader to the monographs of Cottle, Pang, and Stone [11] and Kojima et al. [13], and for a recent comprehensive treatment of variational inequalities and complementarity problems to the monograph of Facchinei and Pang [12].

Due to the theoretical and practical importance of LCPs, efficient methods for solving LCPs are of significant interest. The existing tradition of generalizing results for LO to LCPs dates back to the early days of the development of simplex-type algorithms (pivoting algorithms) and continues to this day. The interior-point methods

*Received by the editors July 31, 2009; accepted for publication (in revised form) July 9, 2010; published electronically October 19, 2010.

<http://www.siam.org/journals/siopt/20-6/76673.html>

[†]Department of Mathematical Sciences, Georgia Southern University, Statesboro, GA 30460-8093 (goran@georgiasouthern.edu).

[‡]Department of Software Technology and Algorithms, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands (c.roos@tudelft.nl).

(IPMs) that have been a great success for LO are no exception. Various IPMs for LO have been successfully generalized to LCPs. Besides the aforementioned monograph of Kojima et al. [13], and without any attempt to be complete, we mention a few other relevant references: [2, 17, 18, 28, 29, 30].

There are basically two types of path-following IPMs: short-step (small-update) methods that take small steps near the central path and long-step (large-update) methods that take more aggressive steps and deviate farther from the central path. The theoretical iteration bound for large-update methods is $O(n \log \frac{n}{\epsilon})$, where n denotes the number of variables in the problem and ϵ denotes the desired accuracy in terms of the objective value. The iteration bound for small-update IPMs is substantially better: $O(\sqrt{n} \log \frac{n}{\epsilon})$. However, small-update IPMs are inefficient in practice, while large-update methods perform much better. This phenomenon is known as the *gap between theory and practice*, also called the *irony of IPMs* [24].

The above-mentioned iteration bounds were obtained by analyzing first order IPMs based on the logarithmic barrier function for monotone LCPs, which are the most common and most studied class of LCPs. Kojima et al. [13] were the first to derive the above-mentioned bound for short-step methods but in a larger framework than monotone LCPs. The authors introduced a new class of LCPs, the $P_*(\kappa)$ -LCPs, which include monotone LCPs as a special case when $\kappa = 0$, and derived an $O((1 + \kappa)\sqrt{n} \log \frac{n}{\epsilon})$ iteration bound. More about different classes of LCPs and their relationships can be found in the next section. Subsequently, many authors presented short-step methods that match this iteration bound. Worth mentioning is the method of Miao [16], which is superlinearly convergent. Potra and Sheng [23] further improved on the result of Miao by proposing a superlinearly convergent predictor-corrector method for sufficient LCPs that has a nice feature of not depending on κ (although its analysis and iteration bound does) and operates in a neighborhood that is slightly larger than the one used in Miao's method. The $O(\sqrt{n} \log \frac{n}{\epsilon})$ iteration bound remains the best known iteration bound for short-step methods.

In an attempt to improve theoretical complexity of long-step methods, two main approaches can be observed. The first consists of modifying and improving methods based on the logarithmic barrier function. Most of these methods are higher order methods with the exception of a few methods that are first order methods. We mention the first order method of Ai and Zhang [4] for monotone LCPs with $O(\sqrt{n} \log n \log \frac{n}{\epsilon})$ iteration complexity. Potra [21] proposed higher order method with the same complexity. Subsequently, Potra and Liu [14, 22] proposed higher order methods for sufficient LCPs that have even better complexity, namely, $O((1 + \kappa)\sqrt{n} \log \frac{n}{\epsilon})$, which is the same complexity as that of short-step methods. Furthermore, the methods are superlinearly convergent even in the absence of strict complementarity and, similarly to the method in [23], do not depend on κ . It is worth mentioning that although the number of iterations in higher order methods decreases, the computational cost per iteration increases. However, the increase is moderate. It is necessary to solve additional $O(\log n)$ linear systems with the same matrix as in the implementation of the first order method. This adds only $O(n^2 \log n)$ arithmetic operations to the $O(n^3)$ arithmetic operations needed to solve the linear system appearing at each iteration of the first order method.

The second approach consists of considering first order methods based on barrier functions that differ from the logarithmic barrier function. Peng, Roos, and Terlaky in [20] were the first to analyze primal-dual IPMs for LO based on a class of barrier functions that is defined by so-called self-regular kernel functions. They significantly improved the theoretical complexity of first order large-update IPMs, ob-

taining the currently best known iteration bound for these types of methods, namely, $O(\sqrt{n} \log n \log \frac{n}{\epsilon})$. They also managed to extend these results to $P_*(\kappa)$ -nonlinear complementarity problems (NCPs), of which the LCP is a special case, as well as to other important optimization problems such as semidefinite and second order cone optimization problems. This iteration bound has since been matched, as already mentioned, in [4] and numerous other papers, some of which will be mentioned in what follows.

Subsequently, Bai, El Ghami, and Roos [5] and Bai et al. [7] presented primal-dual IPMs for LO based on so-called *eligible kernel functions*, which are not necessarily self-regular. For some of them they matched the best iteration bounds for large-update IPMs. They managed to extend the results in [5] to semidefinite and second order cone optimization problems [27, 8]. Furthermore, the results for LO in [7], which were based on a specific eligible kernel function, were extended to $P_*(\kappa)$ -LCPs in Bai, Lesaja, and Roos [6], again matching the best known iteration bounds with the addition of a factor that depends affinely on κ .

The goal of this paper is to extend the results obtained for $P_*(\kappa)$ -LCPs by Bai, Lesaja, and Roos in [6] to the entire class of eligible kernel functions of which the kernel function considered in [6] is just a special case. We provide a unified and comprehensive analysis, similar to the analysis presented in [5] for LO problems, develop a standardized scheme which can be used to calculate iteration bounds for specific eligible kernel functions, and then use the scheme to find iterations bounds for ten specific eligible kernel functions.

The outline of this paper is as follows. In section 2 we present the $P_*(\kappa)$ -LCP and explain why we consider this problem. In section 3 we outline a generic IPM for solving the $P_*(\kappa)$ -LCP that is based on the use of the general barrier function. In section 4 we introduce a wide class of so-called eligible kernel functions and barrier functions based on them and discuss some of the properties. In section 5 we provide the analysis of the generic IPM outlined in section 3 based on the barrier functions defined by eligible kernel functions. In section 6 the complexity of the method is discussed. In section 7 a unified scheme for the calculation of iteration bounds is given and applied to a number of specific eligible kernel functions. Concluding remarks are presented in section 8.

Some notation used throughout the paper 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 nonnegative 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-ones 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 $g(x) = \Theta(x)$ means 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). It is known that for general matrices M the problem is NP-complete [10]. 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. Below we list several important and frequently used classes:

- Skew-symmetric matrices (*SS*):

$$\forall x \in \mathbf{R}^n : x^T M x = 0.$$

- Positive semidefinite matrices (PSD):

$$\forall x \in \mathbf{R}^n : x^T Mx \geq 0.$$

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

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

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

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

- Column sufficient matrices (CSU):

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

- Row sufficient matrices (RSU): M^T is column sufficient.
- Sufficient matrices (SU): M is both column sufficient and row sufficient.
- $P_*(\kappa)$: For $\kappa \geq 0$ the following property holds:

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

where

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

or, equivalently,

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

Furthermore, we define

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

The class of P_* -matrices was introduced by Kojima et al. in their fundamental monograph on IPMs for LCPs [13], while the other classes, as well as a number of additional classes not mentioned here, were discussed in the classical monograph of Cottle, Pang, and Stone [11]. See also [3]. The relationships between some of the above classes are as follows:

$$(2) \quad SS \subset PSD \subset P_* = SU \subset CSU \subset P_0, \quad P \subset P_*, \quad P \cap SS = \emptyset.$$

Some of these relations are obvious, like $PSD = P_*(0) \subset P_*$ or $P \subset P_*$, while others require proof [11, 13, 26]. Especially interesting—and nontrivial—is the fact that P_* -matrices are just sufficient, which was proved by Väliaho in [26].

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 KKT conditions of the quadratic optimization problem with the quadratic objective function defined by the positive semidefinite matrix can be formulated as the monotone LCP. In addition, most practical problems that can be directly formulated as LCPs are usually monotone LCPs. We also recall that in the special case of an 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)) contains the class of monotone LCPs 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 [13]). *The matrix*

$$(3) \quad \bar{M} = \begin{pmatrix} -M & I \\ S & X \end{pmatrix}$$

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

Matrices of the form \bar{M} in (3) appear at each iteration of an IPM for LCPs. 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 [13], this property makes P_0 -LCPs amenable to being solved by an IPM.

However, it is well known that the sequence of the iterates of an IPM may not converge to the solution of an LCP. To assure that each limit point 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 [13], Kojima et al. presented an IPM for an LCP and proved its global convergence under the following condition, which just formalizes the above discussion.

CONDITION 2.2 (Condition 2.3 in [13]).

- (i) M is a P_0 -matrix;
- (ii) there exists a strictly feasible point $(x^0, s^0) > 0$ and $s^0 = Mx^0 + q$;
- (iii) the level set $\mathcal{F}_t = \{(x, s) \in \mathbf{R}_{++}^{2n} : s = Mx + q, \ x^T s \leq t\}$ is bounded for each $t \geq 0$.

Kojima et al. pointed out that, unfortunately, if M is a P_0 -matrix, then requirement (iii) is generally not satisfied, and they provided a counterexample. However, if M is a P_* -matrix, then requirement (iii) is satisfied (Lemma 4.5 in [13]). Hence, in some sense the P_* -class is maximal with respect to the property that 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 $P_*(\kappa)$ -LCPs in this paper.

3. The generic IPM 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 $M \in \mathbf{R}^{n \times n}$ and a vector $q \in \mathbf{R}^n$, find $(x, s) \in \mathbf{R}^{2n}$ such that

$$(4) \quad s = Mx + q, \quad xs = 0, \quad x \geq 0, \quad s \geq 0.$$

We assume that the $P_*(\kappa)$ -LCP satisfies 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 [13, section 5.1] Kojima et al. presented a method of “reducing the LCP to an artificial LCP with an apparent interior feasible point.”

The basic idea of IPMs for LCPs 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 (4) will not work because the method will most likely “get stuck” on the second equation, which is commonly known as the *complementarity equation*. The standard procedure for fixing this problem is to

replace the complementarity equation in (4) by the parameterized equation $xs = \mu \mathbf{e}$, with $\mu > 0$. Thus, we consider the system

$$(5) \quad \begin{aligned} s &= Mx + q, \\ xs &= \mu \mathbf{e}. \end{aligned}$$

Note that $x > 0$, $s > 0$ is enforced by the second equation above.

Since we assume that IPC holds and M is a $P_*(\kappa)$ -matrix, it is well known that the parameterized system (5) has a unique solution for each $\mu > 0$ (Lemma 4.3 of [13]). This solution is denoted as $(x(\mu), s(\mu))$ and is called the μ -center of the LCP. The set of μ -centers (with μ running through all positive real numbers) gives a homotopy path, which is called *the central path* of the LCP. The relevance of the central path for LO was first recognized by Megiddo [15] and then extended to LCPs by Kojima et al. [13]. Under the above assumptions, if $\mu \rightarrow 0$, the limit of the central path exists and is an optimal solution of the LCP (Theorem 4.4 of [13]).

The limiting property of the central path mentioned above leads naturally to the main idea of the IPMs for solving LCPs: Trace the central path while reducing μ 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. The “engine” that does so is a damped Newton method as discussed briefly in the paragraph below.

Suppose that the appropriate point $(x, s) > 0$ “close” to the central path is known. We will describe shortly how to measure the closeness to the central path. Using the point (x, s) , a direct application of Newton’s method to (5) will lead to the following Newton system for the search direction $(\Delta x, \Delta s)$:

$$(6) \quad \begin{aligned} -M\Delta x + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu \mathbf{e} - xs. \end{aligned}$$

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

$$(7) \quad x_+ = x + \alpha \Delta x, \quad s_+ = s + \alpha \Delta s,$$

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 close, i.e., in a certain neighborhood of $(x(\mu), s(\mu))$. Then μ is again reduced by a certain factor, and we again apply Newton’s method targeting the new μ -centers and so on. This process is repeated until μ is a small number close to zero. At this stage we have found an approximate solution of the LCP. In what follows we discuss the details of the method.

For the formulation and analysis of the generic IPM for the LCP that we want to present, the introduction of the following *variance vector* is critical:

$$(8) \quad v := \sqrt{\frac{xs}{\mu}},$$

where $(x, s) \geq 0$ and $\mu > 0$. Note that the pair (x, s) coincides with the μ -center $(x(\mu), s(\mu))$ if and only if $v = \mathbf{e}$ where \mathbf{e} is a vector of all ones. Next, the following *scaled search directions* d_x and d_s are introduced:

$$(9) \quad d_x := \frac{v\Delta x}{x}, \quad d_s := \frac{v\Delta s}{s},$$

where the operations are componentwise multiplication and division. Using (8) and (9), the system (6) can be rewritten as

$$(10) \quad \begin{aligned} -\bar{M}d_x + d_s &= 0, \\ d_x + d_s &= v^{-1} - v, \end{aligned}$$

where $\bar{M} := DMD$, with $D := X^{\frac{1}{2}}S^{-\frac{1}{2}}$, $X := \text{diag}(x)$, $S := \text{diag}(s)$.

A crucial observation is that the right-hand side $v^{-1} - v$ in the second equation of (10) equals the negative gradient of the function

$$(11) \quad \Psi_c(v) := \sum_{i=1}^n \left(\frac{v_i^2 - 1}{2} - \log v_i \right),$$

where v_i represents the i th component of the variance vector v . This is the well-known (*scaled*) *logarithmic barrier function*.

The following two observations about the logarithmic barrier function are important. First, the second equation in (10), often called the *scaled centering equation*, can be written as

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

which implies that Ψ_c essentially defines the search direction.

Second, 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 μ -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 μ -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 barrier function $\Psi(v)$, $v \in \mathbf{R}_{++}^n$, such that $\Psi(v)$ is minimal at $v = \mathbf{e}$ and $\Psi(\mathbf{e}) = 0$; i.e.,

$$(12) \quad \Psi(v) = 0 \quad \Leftrightarrow \quad \nabla \Psi(v) = 0 \quad \Leftrightarrow \quad v = \mathbf{e}.$$

Thus, the function $\Psi(v)$ still serves as a proximity measure for closeness with respect to the μ -center $(x(\mu), s(\mu))$. Introducing a parameter $\tau > 0$ as a threshold value, the inequality $\Psi(v) \leq \tau$ defines a τ -neighborhood of the μ -center. In what follows we also often use the norm-based proximity measure, namely,

$$(13) \quad \delta(v) := \frac{1}{2} \|\nabla \Psi(v)\|.$$

It is easy to see that

$$(14) \quad \delta(v) = 0 \quad \Leftrightarrow \quad v = \mathbf{e}.$$

The new barrier function determines the calculation of the search directions as the new scaled centering equation becomes

$$(15) \quad d_x + d_s = -\nabla \Psi(v),$$

and the Newton steps are calculated from the Newton system

$$(16) \quad \begin{aligned} -\bar{M}d_x + d_s &= 0, \\ d_x + d_s &= -\nabla\Psi(v). \end{aligned}$$

The original Δx and Δs are then calculated using (8) and (9). However, they can also be obtained directly by solving the following system:

$$(17) \quad \begin{aligned} -M\Delta x + \Delta s &= 0, \\ s\Delta x + x\Delta s &= -\mu v \nabla\Psi(v). \end{aligned}$$

Now we can outline the generic IPM that uses the general barrier function as defined above. Suppose that the current iterate (x, s) is known and is in the τ -neighborhood of the corresponding μ -center, i.e., $\Psi(v) \leq \tau$. Next, the value of μ is reduced by the factor $1 - \theta$ with $0 < \theta < 1$, which changes the value of v according to (8) and defines a new μ -center $(x(\mu), s(\mu))$. This will likely cause the increase of the value of the barrier function above the threshold value of τ , i.e., $\Psi(v) \geq \tau$. Now we start the *inner iteration* by calculating new iterates (7) where Δx and Δs are calculated from (16) and (9) or directly from (17) and the step size α is chosen appropriately with the goal of reducing the value of barrier function $\Psi(v)$. If necessary, the procedure is repeated until we find the iterate that again belongs to the τ -neighborhood of the current μ -center, that is, until $\Psi(v) \leq \tau$. It is important to mention that during the inner iteration the value of μ is kept constant. At this point we start a new *outer iteration* by reducing the value of μ again. This process is repeated until μ is small enough, say until $n\mu < \epsilon$ for a certain (small) value of the accuracy parameter $\epsilon > 0$.

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, s^0) is given and, moreover, that this point can be chosen such that $\Psi(v^0) \leq \tau$, which means that it is in the τ -neighborhood of the μ -center.

The generic IPM outlined above is summarized in Figure 1. In what follows we will refer to this method simply as the *Algorithm*.

As already mentioned, there are two types of methods: small-update methods that take small steps close to the central path, and large-update methods that take more aggressive steps farther away from the central path. This is reflected by the choice of values for parameters τ and θ . Large-update methods are characterized by the fact that θ is a fixed constant ($\theta \in (0, 1)$), independent of the dimension n of the problem, i.e., $\theta = O(1)$ while $\tau = O(n)$, whereas small-update methods use a value of θ that depends on the dimension of the problem, with $\theta = O(\frac{1}{\sqrt{n}})$ while τ is a constant, i.e., $\tau = O(1)$.

The resulting iteration bound not only depends on a careful selection of the parameters in the Algorithm but also depends heavily on the choice of the barrier function. We will analyze the Algorithm for a wide class of barrier functions based on the so-called *eligible kernel functions* that will be specified in the next section.

4. The eligible kernel functions. As indicated above, the iteration bound of the Algorithm depends heavily on the choice of the barrier function. Until recently almost all complexity results of IPMs for LCPs, monotone as well as $P_*(\kappa)$, were based on the classical logarithmic barrier function $\Psi_c(v)$ given in (11). By introducing the univariate function $\psi_c : (0, \infty) \rightarrow [0, \infty)$ defined as

$$(18) \quad \psi_c(t) = \frac{t^2 - 1}{2} - \log t,$$

Algorithm. Generic IPM for LCP.

Input:

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

begin

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

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

begin

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

$v = \sqrt{\frac{xs}{\mu}}$;

while $\Psi(v) > \tau$ **do**

begin

calculate search direction $(\Delta x, \Delta s)$ using (17);

determine the step size α according to (34);

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

end

end

end

FIG. 1. *Generic IPM for LCP.*

the function Ψ_c can be written as

$$\Psi_c(v) = \sum_{i=1}^n \psi_c(v_i),$$

i.e., as a separable function with identical univariate functions for each component v_i . The univariate function ψ_c is called the *logarithmic kernel function*.

In this paper we consider more general barrier functions of the form

$$(19) \quad \Psi(v) := \sum_{i=1}^n \psi(v_i),$$

where the univariate function $\psi : (0, \infty) \rightarrow [0, \infty)$ is called the *kernel function* of $\Psi(v)$. We require that the general kernel function ψ has properties similar to those of the logarithmic kernel function, namely,

$$(20a) \quad \psi'(1) = \psi(1) = 0,$$

$$(20b) \quad \psi''(t) > 0,$$

$$(20c) \quad \lim_{t \downarrow 0} \psi(t) = \lim_{t \rightarrow \infty} \psi(t) = \infty.$$

Clearly, (20a) and (20b) indicate that $\psi(t)$ is a nonnegative strictly convex function such that $\psi(t)$ achieves its minimum at $t = 1$; i.e., $\psi(1) = 0$. This implies that

since $\psi(t)$ is twice differentiable, it is completely determined by its second derivative:

$$(21) \quad \psi(t) = \int_1^t \int_1^\xi \psi''(\zeta) d\zeta d\xi.$$

Moreover, (20c) expresses that $\psi(t)$ is coercive and has the barrier property. Furthermore, the conditions imply that $\psi(t)$ is decreasing for $0 < t < 1$ and increasing for $t > 1$. Thus, the growth behavior of $\psi(t)$ is described for $t > 1$, while barrier behavior $\psi(t)$ is described for $0 < t < 1$. In addition, the consequence of (20b) is that $\psi'(t)$ is increasing for $t > 0$. In what follows we assume that a kernel function satisfies (20a)–(20c).

The presentation below is closely related to that in [5]. Many proofs and details are therefore omitted since they can be found there. However, there are some important differences that are explained in what follows. In order to prove favorable complexity results, we impose additional conditions on the kernel functions:

$$(22a) \quad t\psi''(t) + \psi'(t) > 0, \quad t < 1,$$

$$(22b) \quad \psi'''(t) < 0, \quad t > 0,$$

$$(22c) \quad 2\psi''(t)^2 - \psi'(t)\psi'''(t) > 0, \quad t < 1,$$

$$(22d) \quad \psi''(t)\psi'(\beta t) - \beta\psi'(t)\psi''(\beta t) > 0, \quad t > 1, \quad \beta > 1.$$

The kernel functions that satisfy conditions (20a)–(20c) and (22a)–(22d) are called *eligible kernel functions*.

Furthermore, in [5] another condition is also discussed, namely,

$$(23) \quad t\psi''(t) - \psi'(t) > 0, \quad t > 1.$$

This condition is listed because conditions (23) and (22b) imply condition (22d) (Lemma 4.4 of [5]). The reason for the introduction of condition (23) is that it is easier to check condition (23) than (22d), which is more technically involved. In addition, many eligible functions do satisfy condition (23). Thus, the kernel function is also eligible if it satisfies conditions (22a)–(22d) and (23). It is also shown in [5] that conditions (22a), (22b), (22c), and (23) are logically independent.

Note that condition (22a) is obviously satisfied if $t \geq 1$, because then $\psi'(t) \geq 0$. Similarly, condition (23) is satisfied if $t \leq 1$, since then $\psi'(t) \leq 0$. Another obvious but important consequence of condition (22b) is that $\psi''(t)$ is decreasing for $t > 0$. This fact is used often in what follows.

The following lemma lists some equivalent representations of condition (22a).

LEMMA 4.1 (Lemma 2.1.2 in [20]). *Let $\psi(t)$ be a twice differentiable function for $t > 0$. Then the following three properties are equivalent:*

(i) $\psi(\sqrt{t_1 t_2}) \leq \frac{1}{2}(\psi(t_1) + \psi(t_2))$ for $t_1, t_2 > 0$.

(ii) $\psi'(t) + t\psi''(t) \geq 0$, $t > 0$.

(iii) $\psi(e^\xi)$ is convex.

Following [20], the property described in Lemma 4.1 is called *exponential convexity*, or *e-convexity*. This property has proved to be essential in the analysis of primal-dual IPMs based on kernel functions (cf. [19, 20]).

The following quadratic bounds of the eligible kernel function $\psi(t)$ have proved to be very useful.

LEMMA 4.2 (Lemma 2.6 of [5]). *Given an eligible kernel function $\psi(t)$, the*

following inequalities hold:

$$\begin{aligned}\frac{1}{2}\psi''(t)(t-1)^2 &< \psi(t) < \frac{1}{2}\psi''(1)(t-1)^2 & \text{if } t > 1, \\ \frac{1}{2}\psi''(1)(t-1)^2 &< \psi(t) < \frac{1}{2}\psi''(t)(t-1)^2 & \text{if } t < 1.\end{aligned}$$

Furthermore, two inverse functions related to the kernel function have proved to be essential in the analysis of the Algorithm and are defined below. Note that the inverse is taken with respect to the composition of functions.

DEFINITION 4.3. *Given the kernel function ψ , we define the following functions:*

- (i) $\varrho: [0, \infty) \rightarrow [1, \infty)$ is the inverse function of $\psi(t)$ for $t \geq 1$.
- (ii) $\rho: [0, \infty) \rightarrow (0, 1]$ is the inverse function of $-\frac{1}{2}\psi'(t)$ for $t \leq 1$.

In what follows we use the fact that ϱ is an increasing function, since $\psi(t)$ is increasing for $t \geq 1$. Similarly, ρ is a decreasing function because $\psi'(t)$ is increasing, whence $-\frac{1}{2}\psi'(t)$ is decreasing.

In Table 1 ten eligible kernel functions are listed. It is also worth noting that the kernel functions ψ_4 , ψ_7 , and ψ_8 are special cases of the function ψ_{10} for a particular choice of parameters p and q . Moreover, functions the ψ_1 and ψ_9 can also be considered as special cases of the function ψ_{10} because $\frac{t^{1-q}-1}{q-1} \rightarrow \log t$ when $q \rightarrow 1$.

TABLE 1
Ten eligible kernel functions.

i	Kernel functions $\psi_i(t)$
1	$\frac{t^2-1}{2} - \log t$
2	$\frac{t^2-1}{2} + \frac{t^{1-q}-1}{q(q-1)} - \frac{q-1}{q}(t-1), \quad q > 1$
3	$\frac{t^2-1}{2} + \frac{(e-1)^2}{e} \frac{1}{e^t-1} - \frac{e-1}{e}$
4	$\frac{1}{2} \left(t - \frac{1}{t}\right)^2$
5	$\frac{t^2-1}{2} + e^{\frac{1}{t}-1} - 1$
6	$\frac{t^2-1}{2} - \int_1^t e^{\frac{1}{\xi}-1} d\xi$
7	$\frac{t^2-1}{2} + \frac{t^{1-q}-1}{q-1}, \quad q > 1$
8	$t-1 + \frac{t^{1-q}-1}{q-1}, \quad q > 1$
9	$\frac{t^{1+p}-1}{1+p} - \log t, \quad p \in [0, 1]$
10	$\frac{t^{p+1}-1}{p+1} + \frac{t^{1-q}-1}{q-1}, \quad p \in [0, 1], \quad q > 1$

We conclude this section by mentioning that the results for $P_*(\kappa)$ -LCPs in [6] were obtained by using the kernel function ψ_{10} . These results will be confirmed in section 7, but using the larger and more general framework developed in this paper. In addition, the iteration bounds for all of the functions in Table 1 will also be established.

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 eligible barrier function used

in the Algorithm. Note that at the start of each outer iteration, just before the update of μ with the factor $1 - \theta$, we have $\Psi(v) \leq \tau$. Due to the update of μ , 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 τ again. Hence, during the course of the algorithm the largest values of $\Psi(v)$ occur just after the updates of μ . Thus, it is important to estimate the effect of a μ -update on the value of $\Psi(v)$. The result is stated in the theorem below.

THEOREM 5.1 (Theorem 3.2 of [5]). *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*

$$(24) \quad \Psi(v_+) \leq n\psi\left(\frac{\varrho(\frac{\tau}{n})}{\sqrt{1 - \theta}}\right) \leq \frac{n}{2}\psi''(1)\left(\frac{\varrho(\frac{\tau}{n})}{\sqrt{1 - \theta}} - 1\right)^2.$$

Proof. The first inequality follows from Theorem 5.1 by taking $\beta = \frac{1}{\sqrt{1 - \theta}}$, while the second inequality follows from Lemma 4.2. \square

5.2. Determining the default step size. In this subsection, we determine a default step size which not only keeps the iterates feasible but also leads to a sufficiently large decrease of the value of the barrier function in each inner iteration of the Algorithm.

In each inner iteration we first compute the search directions Δx and Δs from the system (16). After a step size α is determined, the new iterate (x_+, s_+) is calculated from (7). Recall that during an inner iteration the parameter μ is fixed. Hence, after the step in the direction $(\Delta x, \Delta s)$ with the step size α , the new variance vector v is given by

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

Given the fact that

$$x_+ = x \left(e + \alpha \frac{\Delta x}{x} \right) = x \left(e + \alpha \frac{d_x}{v} \right) = \frac{x}{v} (v + \alpha d_x),$$

$$s_+ = s \left(e + \alpha \frac{\Delta s}{s} \right) = s \left(e + \alpha \frac{d_s}{v} \right) = \frac{s}{v} (v + \alpha d_s),$$

and using the fact that $xs = \mu v^2$, we obtain

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

Our main goal is to consider the decrease in Ψ as a function of α . Hence, we define a function

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

Lemma 4.1 implies that

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

Next, we define a function

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

The advantage of working with this function instead of the original function $f(\alpha)$ will soon become clear. First, it is easy to see that $f_1(\alpha)$ is an upper bound of $f(\alpha)$ and that $f(0) = f_1(0) = 0$. Next, in order to analyze the function $f_1(\alpha)$, it is important to know its derivatives with respect to α . We have

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

where $d_{xi} := (d_x)_i$ and $d_{si} := (d_s)_i$. From the above equation and by using (15), 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 with respect to α , we get the second derivative

$$(25) \quad f_1''(\alpha) = \frac{1}{2} \sum_{i=1}^n (\psi''(v_i + \alpha d_{xi}) d_{xi}^2 + \psi''(v_i + \alpha d_{si}) d_{si}^2) > 0.$$

During an inner iteration x and s are not both at the μ -center since $\Psi(v) \geq \tau > 0$; hence, we conclude that $f_1(\alpha)$ is strictly convex in α . It is worth pointing out that in general $f(\alpha)$ is not convex. That is an important advantage of using the function $f_1(\alpha)$ instead of using the original decrease function $f(\alpha)$.

In order to find the default step size, it is necessary to estimate the behavior of the function $f_1(\alpha)$, which naturally leads to the analysis and estimation of its derivatives. From the expressions of these derivatives we see that this leads to estimating the lengths of the search directions d_x and d_s . In the 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 5.3 (Lemma 5.4 of [6]). *The following inequalities hold:*

$$(26) \quad \|d_x\| \leq 2\delta\sqrt{1+2\kappa}, \quad \|d_s\| \leq 2\delta\sqrt{1+2\kappa}.$$

Equipped with these inequalities, we can deal with the lemmas that follow in the rest of this section. The proofs of these lemmas are similar to those in [6]; however, the class of kernel functions under consideration in this paper is much larger; thus the proofs in [6] had to be carefully reexamined, and at places different arguments had to be used.

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

$$v_1 \leq v_2 \leq \cdots \leq v_n.$$

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

LEMMA 5.4. *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 (26) we obtain the following inequalities:

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

Substituting these inequalities into the expression (25) for $f_1''(\alpha)$, while using the fact that $\psi''(t)$ is monotonically decreasing (due to (22b)), we get

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

Now using Lemma 5.3 we obtain the desired inequality. \square

LEMMA 5.5. *If the step size α satisfies the condition*

$$(27) \quad -\psi'(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) + \psi'(v_1) \leq \frac{2\delta}{\sqrt{1+2\kappa}},$$

then

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

Proof. Using Lemma 5.4 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 5.4, while the second inequality follows from the hypothesis (27) of the lemma. \square

LEMMA 5.6. *The largest possible value of the step size α satisfying (27) is given by*

$$(28) \quad \bar{\alpha} := \frac{1}{2\delta\sqrt{1+2\kappa}} \left[\rho(\delta) - \rho\left(\left(1 + \frac{1}{\sqrt{1+2\kappa}}\right)\delta\right) \right].$$

Proof. We want to compute the step size α such that (27) holds, with α as large as possible. The derivative with respect to v_1 of the left-hand side in inequality (27) is $-\psi''(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) + \psi''(v_1)$. Since $\psi''(t)$ is decreasing (due to (22b)), this derivative is negative. Therefore, the left-hand side is a decreasing function of α . Hence, fixing δ , the smaller v_1 is, the smaller the maximal value of α 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

$$(29) \quad -\frac{1}{2}\psi'(v_1) = \delta.$$

The derivative with respect to α of the left-hand side in (27) equals

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

and, hence, the left-hand side of the inequality (27) is increasing in α . Thus, the largest possible value of α satisfying (27) satisfies

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

Due to (29), the above equation can be written as

$$(30) \quad -\frac{1}{2}\psi'(v_1 - 2\alpha\delta\sqrt{1+2\kappa}) \left(1 + \frac{1}{\sqrt{1+2\kappa}}\right) \delta.$$

Using the definition of the inverse function ρ , (29) and (30) 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 what follows, so in order to simplify the notation we introduce the following abbreviation:

$$(31) \quad \bar{\kappa} := 1 + \frac{1}{\sqrt{1+2\kappa}}.$$

LEMMA 5.7. *With ρ , $\bar{\alpha}$, and $\bar{\kappa}$ as defined above, we have*

$$(32) \quad \bar{\alpha} \geq \frac{1}{(1+2\kappa)\psi''(\rho(\bar{\kappa}\delta))} \geq \frac{1}{(1+2\kappa)\psi''(\rho(2\delta))}.$$

Proof. Using the fundamental theorem of calculus and the fact that ρ is monotonically decreasing, expression (28) for the step size $\bar{\alpha}$ can be transformed as follows:
(33)

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

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 $\sigma \in (\delta, \bar{\kappa}\delta)$. Since ψ'' is monotonically decreasing, $\psi''(\rho(\sigma))$ is maximal for $\sigma \in (\delta, \bar{\kappa}\delta)$.

when $\rho(\sigma)$ is minimal. Since ρ is monotonically decreasing, this occurs when $\sigma = \bar{\kappa}\delta$. Therefore, using (33) and (31) we obtain

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

From the definition of $\bar{\kappa}$ given by (31) it is easily seen that $1 < \bar{\kappa} \leq 2$. Using again the fact that ψ'' and ρ are monotonically decreasing functions, it immediately follows that

$$\frac{1}{1+2\kappa} \frac{1}{\psi''(\rho(\bar{\kappa}\delta))} \geq \frac{1}{1+2\kappa} \frac{1}{\psi''(\rho(2\delta))},$$

and the lemma is proved. \square

Given the above lemma we use

$$(34) \quad \tilde{\alpha} = \frac{1}{(1+2\kappa)\psi''(\rho(2\delta))}$$

as the default step size in the analysis of the Algorithm.

We conclude this section by noting the following remarkable fact.

Remark 5.8. After a careful analysis, it turns out that the only difference between the above default step size and the one developed in [5] for the LO case (cf. formula (29)) is the factor $\frac{1}{1+2\kappa}$. This will have an important consequence for the derivation of the iteration bounds.

5.3. Decrease of the barrier function during an inner iteration. In this section we show that the default step size (34) yields sufficient decrease of the barrier function value during each inner iteration.

We first state a technical lemma that will be used below.

LEMMA 5.9 (Lemma 12 of [19]). *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*

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

Now the following lemma can be proved.

LEMMA 5.10. *If the step size α is such that $\alpha \leq \bar{\alpha}$, where $\bar{\alpha}$ is defined by (28), 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 5.4, 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 (28), 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 (27) in Lemma 5.5 holds. Hence, $h(\alpha)$ is a decreasing function on $(0, \bar{\alpha}]$, and therefore $h(\alpha)$ attains its minimum at $\alpha = \bar{\alpha}$. Furthermore, since $\psi''(t)$ is decreasing in t , it follows that $h''(\alpha)$ is increasing for $\alpha \in (0, \bar{\alpha}]$. Thus, the assumptions of Lemma 5.9 are satisfied, and we may conclude that

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

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

Using the default step size (34), we get the following immediate consequence of the above lemma.

THEOREM 5.11. *If $\tilde{\alpha}$ is the default step size defined by (34), then*

$$(35) \quad f(\tilde{\alpha}) \leq -\frac{\delta^2}{(1+2\kappa)\psi''(\rho(2\delta))}.$$

Our next goal is to express the bound in the above theorem in terms of Ψ instead of δ . This is achieved by recalling the following important theorem.

THEOREM 5.12 (Theorem 4.9 of [5]). *One has*

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

Next, we need the following simple but important lemma.

LEMMA 5.13 (Lemma 4.7 of [5]). *The expression*

$$-\frac{\delta^2}{\psi''(\rho(2\delta))}$$

in (35) is monotonically decreasing in δ .

The immediate consequence of Lemma 5.13 and Theorem 5.12 is the following theorem.

THEOREM 5.14. *One has*

$$(36) \quad f(\tilde{\alpha}) \leq -\frac{(\psi'(\varrho(\Psi(v))))^2}{4(1+2\kappa)\psi''(\rho(\psi'(\varrho(\Psi(v)))))}.$$

In the next section we use the above theorem to estimate the number of inner iterations per outer iteration. This is the key step in finding an upper bound for the total number of iterations of the Algorithm.

6. Complexity of the Algorithm. In this section we find an upper bound on the total number of iterations of the Algorithm based on the eligible kernel function and with the step size given by (34).

As indicated in the previous section, the value of the $\Psi(v)$ increases after the μ -update at the start of each outer iteration of the Algorithm. In the subsequent inner iterations the value of μ is kept constant and the value of $\Psi(v)$ is reduced using the damped Newton method with the default step size (34). Thus, an upper bound for the total number of iterations is obtained by taking the product of an upper bound on the total number of outer iterations and an upper bound on the total number of inner iterations per one outer iteration.

First, we calculate an upper bound on the number of inner iterations required per outer iteration; that is, we find an upper bound on the number of iterations needed to return to the situation where $\Psi(v) \leq \tau$ after a μ -update. We denote the value of $\Psi(v)$ after the μ -update as Ψ_0 , and the subsequent values in the same outer iteration are denoted as Ψ_k , $k = 1, 2, \dots, K$, where K denotes the total number of inner iterations in the outer iteration.

The decrease on each inner iteration is given by (36) in Theorem 5.14. The key to estimating the number of inner iterations is based on the following important assumption: There exist positive constants β and $\gamma \in (0, 1]$ such that

$$(37) \quad \frac{(\psi'(\varrho(\Psi(v))))^2}{4(1+2\kappa)\psi''(\rho(\psi'(\varrho(\Psi(v)))))} \geq \beta\Psi(v)^{1-\gamma}.$$

This assumption holds for $\gamma = 1$ and

$$\beta = \frac{(\psi'(\varrho(\tau)))^2}{4(1+2\kappa)\psi''(\rho(\psi'(\varrho(\tau))))} \leq \frac{(\psi'(\varrho(\Psi(v))))^2}{4(1+2\kappa)\psi''(\rho(\psi'(\varrho(\Psi(v)))))}$$

because $\Psi(v) \geq \tau > 0$. However, this is the trivial case; our goal is to find γ as small as possible for the reason that is explained in Lemma 6.2 below. Thus, the appropriate values of β and γ have to be found separately for each eligible kernel function.

We first state the following technical lemma.

LEMMA 6.1 (Proposition 2.2 of [20]). *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 $K \leq \lfloor \frac{t_0^\gamma}{\beta\gamma} \rfloor$.

The upper bound on the number of inner iterations per outer iteration is then given by the following lemma.

LEMMA 6.2. *If K denotes the number of inner iterations per one outer iteration of the Algorithm, then we have*

$$(38) \quad K \leq \left\lfloor \frac{\Psi_0^\gamma}{\beta\gamma} \right\rfloor.$$

Proof. The definition of K implies $\Psi_{K-1} > \tau$ and $\Psi_K \leq \tau$. Furthermore, the definition of $f(\alpha)$ together with (36) and (37) leads to the following inequality:

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

Now, by applying Lemma 6.1 with $t_k = \Psi_k$, the desired inequality follows. \square

Next, we calculate an upper bound on the number of outer iterations which coincides with the number of barrier parameter θ updates until we obtain $n\mu < \epsilon$. It is well known [25, Lemma II.17, page 116] that the number of outer iterations is bounded above by

$$(39) \quad \frac{1}{\theta} \log \frac{n}{\epsilon}.$$

An upper bound on the total number for iterations now easily follows and is given in the theorem below.

THEOREM 6.3. *The total number of iterations N in the Algorithm is bounded above by*

$$(40) \quad N \leq \frac{\Psi_0^\gamma}{\theta\beta^\gamma} \log \frac{n}{\varepsilon}.$$

A few remarks are in order regarding the calculation of the iteration bounds.

1. The iteration bounds for the large-update version of the Algorithm can now be calculated by setting $\tau = O(n)$ and $\theta = \Theta(1)$ and for the small-update version by setting $\tau = O(1)$ and $\theta = \Theta(\frac{1}{\sqrt{n}})$.
2. Since Ψ_0 in (40) is unknown, we need an upper bound. According to the Corollary 5.2, the value Ψ_0 is bounded above by

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

3. Inequality (37) implies that the constant β can be written in the form $\beta = \frac{\bar{\beta}}{1+2\kappa}$, where $\bar{\beta}$ does not depend on κ . Thus, the iteration bound in (40) can be written as

$$(1+2\kappa)\frac{\Psi_0^\gamma}{\theta\bar{\beta}^\gamma} \log \frac{n}{\varepsilon},$$

where the fraction does not depend on κ .

4. The previous remark implies that the calculation of iteration bounds for the $P_*(\kappa)$ -LCP case are almost the same as for the LO case except that the iteration bounds in the $P_*(\kappa)$ -LCP case are multiplied with the factor $1+2\kappa$. This remarkable fact, which by no means was obvious or expected, is primarily caused by the fact that the default step size is almost the same as for the LO case modulo factor $\frac{1}{1+2\kappa}$, as noted in Remark 5.8.

The iteration bounds in Theorem 6.3 depend on the parameters β and γ and the upper bound on Ψ_0 . Since these are different for each specific case of eligible kernel function, the iteration bounds will also vary, as explained in more detail in the next section.

7. Iteration bounds for the specific eligible kernel functions.

7.1. The scheme. The procedure for calculating the iteration bounds for the specific eligible kernel function that was discussed in the previous two sections can be summarized in the Scheme stated in Figure 2, which is a modified version of the similar scheme in [5].

Our goal is to use the Scheme in Figure 2 to compute iteration bounds for large- and small-update versions of the Algorithm based on the ten kernel functions listed in Table 1.

In the next section we use the Scheme to calculate iteration bounds of the Algorithm for the kernel function ψ_{10} in Table 1. This kernel function was considered in the analysis of the IPMs for $P_*(\kappa)$ -LCPs in [6]. The derivation in the next section will confirm the results obtained in that paper, but from a different perspective and in a larger and more general framework.

7.2. Iteration bounds for $\psi_{10}(t)$. As indicated, we consider the kernel function

$$\psi_{10}(t) = \frac{t^{p+1} - 1}{p+1} + \frac{t^{1-q} - 1}{q-1}, \quad p \in [0, 1], \quad q > 1,$$

Step 0: Specify a kernel function $\psi(t)$; an update parameter θ , $0 < \theta < 1$; a threshold parameter τ ; and an accuracy parameter ϵ .

Step 1: Solve the equation $-\frac{1}{2}\psi'(t) = s$ to get $\rho(s)$, the inverse function of $-\frac{1}{2}\psi'(t)$, $t \in (0, 1]$. If the equation is hard to solve, derive a lower bound for $\rho(s)$.

Step 2: Calculate the decrease of $\Psi(v)$ during an inner iteration in terms of δ for the default step size $\tilde{\alpha}$ from

$$f(\tilde{\alpha}) \leq -\frac{\delta^2}{(1+2\kappa)\psi''(\rho(2\delta))}.$$

Step 3: Solve the equation $\psi(t) = s$ to get $\varrho(s)$, the inverse function of $\psi(t)$, $t \geq 1$. If the equation is hard to solve, derive lower and upper bounds for $\varrho(s)$.

Step 4: Derive a lower bound for δ in terms of $\Psi(v)$ by using

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

Step 5: Substitute the results of Steps 3 and 4 into Step 2 and find a valid inequality of the form

$$f(\tilde{\alpha}) \leq -\frac{\bar{\beta}}{1+2\kappa}\Psi(v)^{1-\gamma}$$

for some positive constants $\bar{\beta}$ and γ , with $\gamma \in (0, 1]$ as small as possible.

Step 6: Calculate an upper bound of Ψ_0 from one of the inequalities

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

Step 7: Calculate an upper bound for the total number of iterations by substituting the results of Steps 5 and 6 into the expression

$$(1+2\kappa)\frac{\Psi_0^\gamma}{\theta\bar{\beta}^\gamma}\log\frac{n}{\epsilon}.$$

Step 8: Set $\tau = O(n)$ and $\theta = \Theta(1)$ to calculate a complexity bound for large-update methods, and set $\tau = O(1)$ and $\theta = \Theta(\frac{1}{\sqrt{n}})$ to calculate a complexity bound for small-update methods.

FIG. 2. Scheme for calculating iteration bounds of the Algorithm.

and provide detailed calculations of each step of the Scheme.

Step 1. To obtain the inverse function $t = \rho(s)$ of $-\frac{1}{2}\psi'(t) = \frac{1}{2}(t^p - t^{-q})$ for $t \in (0, 1]$ we need to solve t from the equation

$$-t^p + t^{-q} = 2s, \quad t \in (0, 1].$$

Using that $t \leq 1$, we conclude that $t^{-q} = 2s + t^p \leq 2s + 1$, which implies

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

Step 2. Since $\psi''(t) = qt^{-q-1}$, it follows that

$$\begin{aligned} f(\tilde{\alpha}) &\leq -\frac{1}{1+2\kappa} \frac{\delta^2}{\psi''(\rho(2\delta))} \\ &= -\frac{1}{1+2\kappa} \frac{\delta^2}{p(\rho(2\delta))^{p-1} + \frac{1}{(\rho(2\delta))^{q+1}}} \\ &\leq -\frac{1}{1+2\kappa} \frac{1}{p(4\delta+1)^{\frac{1-p}{q}} + q(4\delta+1)^{\frac{q+1}{q}}}. \end{aligned}$$

Since $(4\delta + 1)^{1-p} \leq (4\delta + 1)^{q+1}$ for $p \in [0, 1]$ and $q \geq 1$, it follows that

$$(41) \quad f(\tilde{\alpha}) \leq -\frac{1}{1+2\kappa} \frac{1}{(p+q)(4\delta+1)^{\frac{q+1}{q}}}.$$

Step 3. The inverse function of $s = \psi(t)$ for $t \in [1, \infty)$ is $t = \varrho(s)$. Thus, we have

$$\frac{t^{1+p} - 1}{1+p} = \frac{1 - t^{1-q}}{q-1} + s,$$

which leads to the following lower and upper bounds for $\varrho(s) = t$:

$$(42) \quad (1 + (1+p)s)^{\frac{1}{1+p}} \leq \varrho(s) = t \leq \left((1+p)s + \frac{p+q}{q-1} \right)^{\frac{1}{1+p}}.$$

Step 4. Now, using that $\delta(v) \geq \frac{1}{2}\psi'(\varrho(\Psi(v)))$ and ψ' is monotonically increasing for $t \geq 1$, we may replace $\varrho(\Psi(v))$ by a smaller value. Thus, omitting the argument v and assuming $\Psi \geq \tau \geq 1$, we obtain

$$\begin{aligned} \delta(v) &\geq \frac{1}{2} \left((1 + (1+p)\Psi)^{\frac{p}{1+p}} - \frac{1}{(1+(1+p)\Psi)^{\frac{q}{1+p}}} \right) \\ &\geq \frac{1}{2} \left((1 + (1+p)\Psi)^{\frac{p}{1+p}} - \frac{1}{(1+(1+p)\Psi)^{\frac{1}{1+p}}} \right), \end{aligned}$$

whence

$$(43) \quad \delta(v) \geq \frac{(1+p)\Psi(v)}{2(1+(1+p)\Psi)^{\frac{1}{1+p}}} \geq \frac{\Psi(v)}{2(1+2\Psi(v))^{\frac{1}{1+p}}} \geq \frac{1}{6}\Psi(v)^{\frac{p}{1+p}}.$$

Step 5. Substituting (43) into (41) and noting that the right-hand side expression in (41) is monotonically decreasing in δ , we obtain

$$f(\tilde{\alpha}) \leq -\frac{1}{1+2\kappa} \frac{\Psi^{\frac{2p}{1+p}}}{36(p+q) \left(\frac{2}{3}\Psi^{\frac{p}{1+p}} + 1 \right)^{\frac{q+1}{q}}} \leq -\frac{1}{1+2\kappa} \frac{1}{100(p+q)} \Psi^{\frac{p(q-1)}{q(1+p)}}.$$

Hence, we get $\bar{\beta} = \frac{1}{100(p+q)}$ and $\gamma = \frac{q+p}{q(1+p)}$. Thus, the number K of inner iterations per one outer iteration is bounded above by

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

Step 6. Now, we need to find an upper bound of Ψ_0 . The fact that $\psi(t) \leq \frac{t^{1+p}}{1+p}$ for $t \geq 1$ leads to

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

Step 7. Hence, an upper bound for the total number of iterations is

$$(1+2\kappa)\frac{\Psi_0^\gamma}{\theta\beta^\gamma}\log\frac{n}{\epsilon} \leq \frac{1}{1+2\kappa}\frac{100(1+p)q}{\theta(1-\theta)^{\frac{p+q}{2q}}}\left(\frac{(1+p)\tau + \frac{q+p}{q-1}n}{1+p}\right)^{\frac{p+q}{q(1+p)}}\log\frac{n}{\epsilon}.$$

Step 8. For large-update methods the right-hand side expression becomes

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

and for small-update methods

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

The bound for small-update methods is not sharp and can be improved as we show below. We need to go back to Step 3 and improve the upper bound for the inverse function $t = \varrho(s)$ of $s = \psi(t)$.

Step 3. We have

$$t \leq 1 + \sqrt{t\psi(t)}.$$

Substituting $t \leq ((1+p)\psi(t) + \frac{q+p}{q-1})^{\frac{1}{1+p}}$ from (7.2) into the above inequality, we obtain

$$t = \varrho(s) \leq 1 + \sqrt{s}\left((1+p)s + \frac{p+q}{q-1}\right)^{\frac{1}{2(1+p)}}.$$

As a consequence, Step 6 and subsequent steps will change.

Step 6. We obtain the following upper bound of Ψ_0 :

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

By using the fact that

$$(44) \quad 1 - \sqrt{1-\theta} = \frac{\theta}{1 + \sqrt{1-\theta}} \leq \theta,$$

we get

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

TABLE 2
Complexity results for ten eligible kernel functions.

i	Kernel functions ψ_i	Small-update methods	Large-update methods
1	$\frac{t^2-1}{2} - \log t$	$O((1+2\kappa)\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)n \log \frac{n}{\epsilon})$
2	$\frac{t^2-1}{2} + \frac{t^{1-q}-1}{q(q-1)} - \frac{q-1}{q}(t-1), \quad q > 1$	$O((1+2\kappa)q\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)qn^{\frac{q+1}{2q}} \log \frac{n}{\epsilon})$
3	$\frac{t^2-1}{2} + \frac{(e-1)^2}{e} \frac{1}{e^t-1} - \frac{e-1}{e}$	$O((1+2\kappa)\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)n^{\frac{3}{4}} \log \frac{n}{\epsilon})$
4	$\frac{1}{2} \left(t - \frac{1}{t}\right)^2$	$O((1+2\kappa)\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)n^{\frac{2}{3}} \log \frac{n}{\epsilon})$
5	$\frac{t^2-1}{2} + e^{\frac{1}{t}-1} - 1$	$O((1+2\kappa)\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)\sqrt{n} \log^2 n \log \frac{n}{\epsilon})$
6	$\frac{t^2-1}{2} - \int_1^t e^{\frac{1}{\xi}-1} d\xi$	$O((1+2\kappa)\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)\sqrt{n} \log^2 n \log \frac{n}{\epsilon})$
7	$\frac{t^2-1}{2} + \frac{t^{1-q}-1}{q-1}, \quad q > 1$	$O((1+2\kappa)q^2\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)qn^{\frac{q+1}{2q}} \log \frac{n}{\epsilon})$
8	$t-1 + \frac{t^{1-q}-1}{q-1}, \quad q > 1$	$O((1+2\kappa)q^2\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)qn \log \frac{n}{\epsilon})$
9	$\frac{t^{1+p}-1}{1+p} - \log t, \quad p \in [0, 1]$	$O((1+2\kappa)\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)n \log \frac{n}{\epsilon})$
10	$\frac{t^{p+1}-1}{p+1} + \frac{t^{1-q}-1}{q-1}, \quad p \in [0, 1], \quad q > 1$	$O((1+2\kappa)q^2\sqrt{n} \log \frac{n}{\epsilon})$	$O((1+2\kappa)qn^{\frac{p+q}{q(1+p)}} \log \frac{n}{\epsilon})$

Step 7. Thus, the total number of iterations is bound above by

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

Since $\frac{p+q}{q(1+p)} \leq 1$ for all $p \in [0, 1]$ and $q \geq 2$, the bound can be simplified to

$$(1+2\kappa) \frac{50q(1+p)(p+q)}{\theta(1-\theta)} \left(\theta\sqrt{n} + \tau \left((1+p) \frac{\tau}{n} + \frac{p+q}{q-1} \right)^{\frac{1}{2(1+p)}} \right)^2 \log \frac{n}{\epsilon}.$$

Step 8. Now, for small-update methods and $p \in [0, 1]$, the above expression becomes

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

7.3. Summary of results. As already mentioned in Remark 5.8 and remarks at the end of section 6, which are then used in the Scheme, it turns out that the calculation of iterations bounds for LO and the $P_*(\kappa)$ -LCP case are almost the same. The difference is that in the $P_*(\kappa)$ -LCP case the iteration bounds are multiplied with the factor $1+2\kappa$. Thus, the results and iteration bounds obtained in [5] can be used in this paper. This covers the first seven functions in Table 1. The iteration bounds for the function ψ_{10} were analyzed in the previous subsection. For the functions ψ_8 and ψ_9 the calculations of the iteration bounds using the Scheme are similar to those for the function ψ_{10} ; therefore, the details are omitted.

The summary of complexity results is given in Table 2.

For large-update methods, the resulting iteration bounds are summarized in the third column of Table 2. For ψ_2 and ψ_7 the bound is minimal if we choose $q = \frac{1}{2} \log n$,

and for ψ_{10} the bound is minimal if we choose $p = 1$ and $q = \frac{1}{2} \log n$. This gives the best bound known so far for large-update IPMs:

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

Note that the small-update methods all have the same complexity as the small-update method based on the logarithmic barrier function, namely,

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

8. Concluding remarks. In this paper we have analyzed large- and small-update versions of the IPM for the $P_*(\kappa)$ -LCP described in Figure 1, based on the large class of eligible kernel functions and with the default step size (34).

The class of eligible kernel functions is important because it is very general and includes the classical logarithmic kernel function, the self-regular kernel function, and a number of non-self-regular kernel functions as special cases. This class was first introduced by Bai, El Ghami, and Roos in [5] for LO.

In what follows we list the main contributions of the paper.

- We provide a unified analysis of IPMs for $P_*(\kappa)$ -LCPs for the entire class of eligible kernel functions, extending the similar analysis presented in [5] for LO problems. As far as we are aware, this is the first paper that offers such a comprehensive treatment of kernel-based IPMs for $P_*(\kappa)$ -LCPs.
- After a careful analysis we observed a remarkable fact: It turns out that the calculation of iteration bounds for the $P_*(\kappa)$ -LCP case is almost the same as for the LO case except that the iteration bounds in the $P_*(\kappa)$ -LCP case are multiplied with the factor $1 + 2\kappa$. This was by no means obvious or expected.
- We calculated the iteration bounds for ten eligible kernel functions listed in Table 1.
- The results obtained for specific kernel functions in several papers [6, 1, 9] can be derived as special cases of the unified approach presented in this paper. This was specifically shown for the function ψ_{10} in Table 1 that was used in [6].

The iteration bounds obtained in this paper are as good as they can be in the current state of the art. For large-update methods the best iteration bound is $O((1 + 2\kappa)\sqrt{n}(\log n) \log \frac{n}{\varepsilon})$, and for the small-update methods all iteration bounds are the same, namely, $O((1 + 2\kappa)\sqrt{n} \log \frac{n}{\varepsilon})$.

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 that of other methods. Another, more theoretical, direction for further research includes investigating the possibility of generalizing these results to conic LCPs and NCPs.

REFERENCES

- [1] K. AMINI AND M. R. PEYGHAMI, *Exploring complexity of large update interior-point methods for $P_*(\kappa)$ linear complementarity problem based on kernel function*, Appl. Math. Comput., 207 (2009), pp. 501–513.
- [2] M. ANITESCU, G. LESAJA, AND F. A. POTRA, *An infeasible interior-point predictor-corrector algorithm for the P_* -geometric LCP*, Appl. Math. Optim., 36 (1997), pp. 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), pp. 265–290.

- [4] W. AI AND S. ZHANG, *An $O(\sqrt{n})L$ iteration primal-dual path-following method, based on wide neighborhoods and large updates, for monotone LCP*, SIAM J. Optim., 16 (2005), pp. 400–417.
- [5] Y. Q. BAI, M. EL GHAMI, AND C. ROOS, *A comparative study of kernel functions for primal-dual interior-point algorithms in linear optimization*, SIAM J. Optim., 15 (2004), pp. 101–128.
- [6] Y. Q. BAI, G. LESAJA, AND C. ROOS, *A new class of polynomial interior-point algorithms for linear complementarity problems*, Pac. J. Optim., 4 (2008), pp. 19–41.
- [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*, J. Optim. Theory Appl., 138 (2008), pp. 341–359.
- [8] Y. Q. BAI, G. Q. WANG, AND C. ROOS, *A new primal-dual interior-point algorithm for second-order cone optimization*, Nonlinear Anal., 70 (2009), pp. 3584–3602.
- [9] G. M. CHO, M. K. KIM, AND Y. H. LEE, *Complexity of large-update interior point algorithm for $P_*(\kappa)$ linear complementarity problems*, Comput. Math. Appl., 53 (2007), pp. 948–960.
- [10] 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, University of Michigan, Ann Arbor, MI, 1979.
- [11] R. W. COTTLE, J.-S. PANG, AND R. E. STONE, *The Linear Complementarity Problem*, Academic Press, Boston, MA, 1992.
- [12] F. FACCHINEI AND J.-S. PANG, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer-Verlag, New York, 2003.
- [13] M. KOJIMA, N. MEGIDDO, T. NOMA, AND A. YOSHISE, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, Lecture Notes in Comput. Sci. 538, Springer-Verlag, Berlin, 1991.
- [14] X. LIU AND F. A. POTRA, *Corrector-predictor methods for sufficient linear complementarity problems in a wide neighborhood of the central path*, SIAM J. Optim., 17 (2006), pp. 871–890.
- [15] 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.
- [16] J. M. MIAO, *A quadratically convergent $O((\kappa+1)\sqrt{n}L)$ -iteration algorithm for the $P_*(\kappa)$ -matrix linear complementarity problem*, Math. Programming Ser. A, 69 (1995), pp. 355–368.
- [17] R. D. C. MONTEIRO AND S. J. WRIGHT, *A superlinear infeasible-interior-point affine scaling algorithm for LCP*, SIAM J. Optim., 6 (1996), pp. 1–18.
- [18] J. PENG, C. ROOS, AND T. TERLAKY, *New complexity analysis of primal-dual Newton methods for $P_*(\kappa)$ linear complementarity problems*, in High Performance Optimization, H. Frenk, C. Roos, T. Terlaky, and S. Zhang, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 245–265.
- [19] 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), pp. 129–171.
- [20] J. PENG, C. ROOS, AND T. TERLAKY, *Self-Regularity: A New Paradigm for Primal-Dual Interior-Point Algorithms*, Princeton University Press, Princeton, NJ, 2002.
- [21] F. A. POTRA, *A superlinearly convergent predictor-corrector method for degenerate LCP in a wide neighborhood of the central path with $O(\sqrt{n}L)$ -iteration complexity*, Math. Program. Ser. A, 100 (2004), pp. 317–337.
- [22] F. A. POTRA AND X. LIU, *Predictor-corrector methods for sufficient linear complementarity problems in a wide neighborhood of the central path*, Optim. Methods Softw., 20 (2005), pp. 145–168.
- [23] F. A. POTRA AND R. SHENG, *A large-step infeasible-interior-point method for the P_* -matrix LCP*, SIAM J. Optim., 7 (1997), pp. 318–335.
- [24] J. RENEGAR, *A Mathematical View of Interior-Point Methods in Convex Optimization*, MPS/SIAM Ser. Optim. 3, SIAM, Philadelphia, 2001.
- [25] C. ROOS, T. TERLAKY, AND J.-PH. VIAL, *Theory and Algorithms for Linear Optimization. An Interior-Point Approach*, 2nd ed., Springer Science, Heidelberg, Boston, 2005.
- [26] H. VÁLIAHO, *P_* -matrices are just sufficient*, Linear Algebra Appl., 239 (1996), pp. 103–108.
- [27] 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), pp. 409–433.

- [28] S. J. WRIGHT AND Y. ZHANG, *A superquadratic infeasible-interior-point method for linear complementarity problems*, Math. Programming Ser. A, 73 (1996), pp. 269–289.
- [29] Y. YE, *On homogeneous and self-dual algorithms for LCP*, Math. Programming Ser. B, 76 (1997), pp. 211–221.
- [30] P. A. ZHANG AND X. S. LI, *An infeasible-start path-following method for monotone LCPs*, Math. Comput. Modelling, 38 (2003), pp. 23–31.