IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 15, Issue 2 Ser. II (Mar – Apr 2019), PP 12-18
www.iosrjournals.org

Implementation of Interior-Point Methods for $P_*(\kappa)$-Linear Complementarity Problem Based On A Kernel Function

Wei Chen, Luhua Yin, Jinwei Su, Hao Liu
School of Mathematics, Physics and Statistics, Shanghai University of Engineering Science, Shanghai, 201620, P.R. China
Corresponding Author: Wei Chen, Luhua Yin

Abstract: *In this paper, we present the implementation of interior-point methods for $P_*(\kappa)$ -linear complementarity problem based on a kernel function. Some preliminary numerical results are provided to demonstrate the computational performance of the proposed kernel-based interior-point methods.*
Keywords: *Linear complementarity problems, interior-point methods, kernel function*
MR (2000) Subject Classification: *90C33, 90C51*

Date of Submission: 27-03-2019 Date of acceptance: 12-04-2019

I. Introduction

Given a matrix $M \in \mathbb{R}^{n\times n}$ and a vector $q \in \mathbb{R}^n$, the standard linear complementarity problem (LCP) is to find a pair of vectors $(x, s) \in \mathbb{R}^{2n}$ such that

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

where xs denotes the coordinatewise product of the vectors x and s. It is well known that the Karush-Kuhn-Tucker optimality conditions for linear optimization and convex quadratic optimization can be written as LCPs. In addition, many important practical problems in such areas as economics theory (equilibrium problems), game theory, transportation planning (assignment problems), optimal control, engineering etc. can be formulated as LCPs. For a comprehensive treatment of LCPs theory and applications, we refer the reader to the monographs of Cottle et al. [1] and Kojima et al. [5].

We call LCP the $P_*(\kappa)$-LCP if the matrix M is a $P_*(\kappa)$-matrix, i.e.,

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

where $I_+(x) = \{i : x_i(Mx)_i > 0\}$ and $I_-(x) = \{i : x_i(Mx)_i < 0\}$. The union of all the $P_*(\kappa)$-matrices is defined by

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

i.e., M is a P_*-matrix if $M \in P_*(\kappa)$ for some $\kappa \geq 0$. There are many solution approaches for $P_*(\kappa)$-LCP. Among them, kernel-function based interior-point methods (IPMs) gain much more attention. For some other related kernel-function based IPMs we refer to the monograph [18] and the references [6,7,14-17,19,20].

The aim of the paper is to propose the implementation of IPMs for $P_*(\kappa)$-LCP based on the following kernel function, namely,

$$\varphi(t) = (t-1)^2 / 2, \tag{1}$$

which has a finite value at the boundary of feasible region and first considered by Liu and Sun [11]. The search direction decided by the kernel function (1) coincides with the equivalent reformulation of the central path by Zhang and Xu [24], and is only up to a constant comparing with the equivalent algebraic transformation $\varphi(t) = \sqrt{t}$ considered by Darvay [2]. The numerical example serves to demonstrate the influence of the update parameters θ and the dimension of the problem on the number of the iterations.

The outline of the paper is as follows. In Section 2, we briefly recall the kernel-based IPMs for $P_*(\kappa)$-LCP. In Section 3, we report some preliminary numerical results to demonstrate the computational performance of the proposed methods. Finally, some conclusions and remarks are made in Section 4.

II. The Generic Kernel-Based IPMs

In this section, we briefly recall the outline of the kernel-based IPMs for $P_*(\kappa)$-LCP, which includes the central path, the new search directions and the generic kernel-based IPMs for $P_*(\kappa)$-LCP. The detailed can be found in [8,13,14].

Throughout the paper, we assume that $P_*(\kappa)$-LCP satisfy the interior-point condition (IPC), i.e., there exists a pair $(x^0, s^0) > 0$ such that $s^0 = Mx^0 + q$, which implies the existence of a solution for $P_*(\kappa)$-LCP. In fact, the IPC can be assumed without loss of generality. For this and some other properties mentioned below, we refer to Kojima et al. [5]

Finding an approximate solution of $P_*(\kappa)$-LCPs is equivalent to solving the following system

$$\begin{pmatrix} -Mx+s \\ xs \end{pmatrix} = \begin{pmatrix} q \\ 0 \end{pmatrix}, \ x, s \geq 0. \tag{2}$$

The standard approach is to replace the second equation in (2), the so-called complementarity condition for $P_*(\kappa)$-LCPs, by the parameterized equation $xs = \mu e$, with $\mu > 0$. This leads to the following system

$$\begin{pmatrix} -Mx+s \\ xs \end{pmatrix} = \begin{pmatrix} q \\ \mu e \end{pmatrix}, \ x, s \geq 0. \tag{3}$$

From Lemma 4.3 in [5], the parameterized system (3) has a unique solution for each $\mu > 0$ due to the fact that M is a $P_*(\kappa)$-matrix and the IPC holds. We denote this solution as $(x(\mu), s(\mu))$ and call it the μ-center of $P_*(\kappa)$-LCP. The set of μ-centers gives a homotopy path, which is called the central path of $P_*(\kappa)$-LCP. If $\mu \to 0$, then the limit of the central path exists and since the limit points satisfy the complementarity condition, i.e., $xs = 0$, the limit yields a solution for $P_*(\kappa)$-LCP (cf. Theorem 4.4 in [5]).

IPMs follow the central path approximately and find an approximate solution of $P_*(\kappa)$-LCP as μ goes to zero. A natural way to define a search direction is to follow Newton's approach and linearize the second equation in (3). This yields to the following system

$$\begin{pmatrix} -M\Delta x + \Delta s \\ s\Delta x + x\Delta s \end{pmatrix} = \begin{pmatrix} 0 \\ \mu e - xs \end{pmatrix}. \tag{4}$$

It follows from Lemma 4.1 in [5] that the modified Newton-system (4) has a unique solution.

Let

$$v := \sqrt{\frac{xs}{\mu}}, \tag{5}$$

and

$$d_x := \frac{v\Delta x}{x}, \ d_s := \frac{v\Delta s}{s}. \tag{6}$$

Then, we have

$$\begin{pmatrix} -\bar{M}d_x + d_s \\ d_x + d_s \end{pmatrix} = \begin{pmatrix} 0 \\ v^{-1} - v \end{pmatrix}, \tag{7}$$

where $\bar{M} := DMD$ with $D := X^{\frac{1}{2}}S^{-\frac{1}{2}}$, $X := \mathrm{diag}(x)$ and $S := \mathrm{diag}(s)$. It is obvious that the right-hand side $v^{-1} - v$ in the second equation of the system (7) equals minus the derivative of the classic barrier function $\Phi_c(v)$, i.e.,

$$\Phi_c(v) := \sum_{i=1}^{n} \varphi_c(v_i), \ v \in \mathbb{R}^n_{++} \tag{8}$$

where

$$\varphi_c(t) := \frac{t^2 - 1}{2} - \log t$$

is the kernel function of the classic barrier function $\Phi_c(v)$. Thus, the system (7) can be rewritten as the following system.

$$\begin{pmatrix} -\bar{M}d_x + d_s \\ d_x + d_s \end{pmatrix} = \begin{pmatrix} 0 \\ -\nabla\Phi_c(v) \end{pmatrix}. \tag{9}$$

For obtain the new search directions, we define the barrier function $\Phi(v): \mathbb{R}^n_{++} \to \mathbb{R}_+$ based on the parametric kernel function given by (1) as follows

$$\Phi(x, s; \mu) := \Phi(v) := \sum_{i=1}^{n} \varphi(v_i). \tag{10}$$

Due to the properties of the kernel function $\varphi(t)$ given by (1), we can conclude that $\Phi(v)$ is a strictly convex function and attains minimal value at $v = e$ and $\Phi(e) = 0$, i.e.,

$$\nabla\Phi(v) = 0 \Leftrightarrow \Phi(v) = 0 \Leftrightarrow v = e. \tag{11}$$

By replacing the right-hand side of the second equation in (9) by $-\nabla\Phi(v)$, we have

$$\begin{pmatrix} -\bar{M}d_x + d_s \\ d_x + d_s \end{pmatrix} = \begin{pmatrix} 0 \\ -\nabla\Phi(v) \end{pmatrix}. \tag{12}$$

The new search directions d_x and d_s are obtained by solving (12) so that Δx and Δs are computed through (6). We can conclude that Δx and Δs both vanish if and only if $v = e$, i.e., if and only if $x = x(\mu), s = s(\mu)$. Otherwise, we will use $(\Delta x, \Delta s)$ as the new search direction. Then, we have

$$x_+ := x + \alpha\Delta x, \quad s_+ := s + \alpha\Delta s, \tag{13}$$

where α is the default step size defined by some line search rules. Furthermore, we can easily verify that

$$d_x = d_s = 0 \Leftrightarrow \nabla\Phi(v) = 0 \Leftrightarrow \Phi(v) = 0 \Leftrightarrow v = e. \tag{14}$$

This implies that the value of $\Phi(v)$ can be considered as a measure for the distance between the given pair (x, s) and the corresponding μ-center $(x(\mu), s(\mu))$.

Now we can outline the generic kernel-based IPMs that uses the barrier function defined by (10). Suppose that $(x(\mu), s(\mu))$ is known for some positive μ and is in the τ-neighborhood of the corresponding μ-center, i.e., $\Phi(v) \le \tau$. For example, due to the above assumption we may assume this for $\mu = 1$, with $x(1) = s(1) = e$. Then, we decrease μ to $\mu := (1-\theta)\mu$ with $\theta \in (0,1)$, which changes the value of v according to (5) and defines a new μ-center $(x(\mu), s(\mu))$. This may cause the increase of the value of the barrier function above the threshold value of τ, i.e., $\Phi(v) > \tau$. Now we start the inner iteration by solving the scaled Newton system (12) and via (6) to get the new search direction $(\Delta x, \Delta s)$. The new iterate (x_+, s_+) is calculated by (13). If necessary, we repeat the procedure until we find iterates that are in the neighborhood of $(x(\mu), s(\mu))$. 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. Then we apply Newton's method targeting at the new μ-centers, and so on. This process is repeated until μ is small enough, say until $n\mu < \varepsilon$, at this stage we have found an ε-solution of $P_*(\kappa)$-LCPs. The generic form of kernel-based IPMs is shown in Fig. 1.

Generic Kernel-Based IPMs for $P_*(\kappa)$-LCP

Inputs:
a threshold parameter $\tau \ge 1$;
an accuracy parameter $\varepsilon > 0$;
a fixed barrier update parameter θ, $0 < \theta < 1$;
a strictly feasible point (x^0, s^0) and $\mu^0 = (x^0)^T s^0 / n$ such that $\Phi(x^0, s^0; \mu^0) \le \tau$.
begin
$x := x^0$; $s := s^0$; $\mu := \mu^0$;
while $n\mu \ge \varepsilon$ **do**

begin
$\mu := (1-\theta)\mu$;
while $\Phi(x,s;\mu) > \tau$ **do**
begin
compute the search directions $(\Delta x, \Delta s)$;
choose a suitable step size α;
update $(x,s) := (x,s) + \alpha(\Delta x, \Delta s)$.
end
end
end

Figure 1: Algorithm

III. Numercal Results

In this section, we give an implementation of the kernel-based IPMs for $P_*(\kappa)$-LCP, which demonstrates to reveal the influence of the update parameters θ and the dimension of the problem on the number of the iterations.

All our numerical experiments are carried out on a PC with Intel (R) Core (TM) i7-6700 Duo CPU @ 2.60 GHz and 20.0 GB of physical memory. The PC runs MATLAB version: 9.2.0.538062 (R2017a) on Windows 10 Home 64-bit operating system. In all test problems, we use $\varepsilon = 10^{-6}$ and $\tau = \sqrt{n}$ as our default accuracy parameter.

In general, though exists $(x^0, s^0) > 0$ for $P_*(\kappa)$-LCP satisfies the IPC, we don't know the value of (x^0, s^0). To overcome this difficulty, we replace the system (12) by the following system

$$\begin{pmatrix} -M\Delta x + \Delta s \\ s\Delta x + x\Delta s \end{pmatrix} = \begin{pmatrix} q + Mx - s \\ \mu e - xs \end{pmatrix} \tag{15}$$

to compute the search directions. Noting that $\mu e - xs = -\mu v \nabla \Phi_c(v)$, as describe in Section 2, we will replace $\nabla \Phi_c(v)$ by $\nabla \Phi(v)$ in the second equation of system (15). Then we have

$$\begin{pmatrix} -M\Delta x + \Delta s \\ s\Delta x + x\Delta s \end{pmatrix} = \begin{pmatrix} q + Mx - s \\ \mu v \nabla \Phi(v) \end{pmatrix}. \tag{16}$$

It should be noted that the step size selected during each inner iteration is small enough for analyzing the algorithm, while in practice the step size during each inner iteration should be large enough for the efficiency of the algorithm. Then the step sizes α_x and α_s during each inner iteration in this experiment are chosen according to the following strategy (see, e.g., [12]).

First, compute the maximum allowable step sizes by the following strategy [12]

$$\alpha_x^{max} = \frac{1}{max_{i=1,2,\ldots,n}\left\{1, -\frac{\Delta x_i}{x_i}\right\}}, \ \alpha_s^{max} = \frac{1}{max_{i=1,2,\ldots,n}\left\{1, -\frac{\Delta s_i}{s_i}\right\}}.$$

Then the maximum allowable step sizes are slightly reduced by a fixed factor $0 < p_0 < 1$ to prevent hitting the boundary, i.e., $\alpha_x = p_0 \alpha_x^{max}$ and $\alpha_s = p_0 \alpha_s^{max}$. In this paper, we choose $p_0 = 0.90$. The new iteration point is defined by

$$x_+ = x + \alpha_x \Delta x, \ \ s_+ = s + \alpha_s \Delta s.$$

In this paper, we consider the following five examples.

Problem 1 (Random $P_*(0)$-LCP)

By taking $M = AA^T$, where $A = rand(n,n)$ is an n-by-n matrix containing pseudorandom values drawn from the standard uniform distribution on the open interval $(0,1)$, $x = s = e$ and $q = s - Mx$. We can easily verify that $x^0 = s^0 = e$ are strictly feasible initial point and satisfies the condition $\Phi(v) = 0 \le \tau = \sqrt{n}$.

Problem 2 ($P_*(1/4)$ -LCP, Lee's example in [6])

$$M = \begin{pmatrix} 0 & 1 \\ -2 & 0 \end{pmatrix}, q = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

with the strictly feasible initial points $x^0 = (0.4, 0.45)^T$ and $s^0 = (2.45, 2.2)^T$.

Problem 3 ($P_*(0)$ -LCP, Fathi's example in [3])

$$M = \begin{pmatrix} 1 & 2 & 2 & \cdots & 2 \\ 2 & 5 & 6 & \cdots & 6 \\ 2 & 6 & 9 & \cdots & 10 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 2 & 6 & 10 & \cdots & 4n-3 \end{pmatrix}, q = \begin{pmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \end{pmatrix}.$$

Problem 4 ($P_*(0)$ -LCP, Waston's third example in [22])

$$M = \begin{pmatrix} 6 & -4 & 2 & \cdots & 0 \\ -4 & 6 & -4 & \cdots & 0 \\ 2 & -4 & 6 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & 6 \end{pmatrix}, q = \begin{pmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \end{pmatrix}.$$

Problem 5 ($P_*(0)$ -LCP, Murty's example in [11])

$$M = \begin{pmatrix} 1 & 2 & 2 & \cdots & 2 \\ 0 & 1 & 2 & \cdots & 2 \\ 0 & 0 & 1 & \cdots & 2 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}, q = \begin{pmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \end{pmatrix}.$$

The above five test problems are solved by applying the proposed kernel-based IPMs, some preliminary numerical results are provided to demonstrate the dependency of the iteration number (It.) with respect to the update parameters θ and the dimension (DIM) of the problem. For Problems 3-5, we use $x^0 = s^0 = e$ as their starting points, which may not satisfy the condition $\Phi(v) \le \tau = \sqrt{n}$. The numerical results are reported in Tables 1-2.

θ				
0.1	0.3	0.5	0.7	0.9
25	22	24	18	14
DIM=10				

θ				
0.1	0.3	0.5	0.7	0.9
39	32	25	24	18
DIM=50				

θ				
0.1	0.3	0.5	0.7	0.9
56	53	37	31	26
DIM=200				

θ				
0.1	0.3	0.5	0.7	0.9
71	56	45	34	27
DIM=600				

Table 1 Numbers of iterations of Problem 1 for several choices of θ and DIM

It is clear from Tables 1-2 that the iteration number of the algorithm depends on the update parameter θ and the dimension of the problem. It is quite surprising that the larger θ gives the lowest iteration count in all cases. The iteration number of the algorithm increases as the dimension of the problem is increased.

θ				
0.1	0.3	0.5	0.7	0.9
16	15	14	12	11
Problem 2				

θ				
0.1	0.3	0.5	0.7	0.9
42	41	27	25	21
Problem 3 with DIM=120				

θ				
0.1	0.3	0.5	0.7	0.9
48	52	30	31	25
Problem 4 with DIM=120				

θ				
0.1	0.3	0.5	0.7	0.9
42	45	26	27	21
Problem 5 with DIM=120				

Table 2 Numbers of iterations of Problems 2-5 for several choices of θ and DIM

IV. Conclusion

In this paper, we have proposed an implementation of the kernel-based IPMs for $P_*(\kappa)$-LCPs based on the kernel function $\varphi(t)$ given by (1). Some preliminary numerical results are provided to reveal the influence of the update parameters θ and the dimension of the problem on the number of iterations.

Some interesting topics remain for further research. Such as the extensions to general nonlinear complementarity problems or symmetric complementarity problems deserve to be investigated.

Acknowledgements

This work is supported by University Students' Innovative Training Program of Shanghai (No. CS1821002).

References

[1]. Cottle, R.W., Pang, J.-S., Stone R.E.: The Linear Complementarity Problem. Academic Press, Boston, MA (1992)
[2]. Darvay, Z.: New interior-point algorithms in linear programming. Adv. Model Optim., 5(1), 51-92 (2003)
[3]. Fathi, Y.: Computational complexity of LCPs associated with positive definite matrices. Math. Program., 17(1), 335-344 (1979)
[4]. El Ghami, M., Wang, G.Q.: Interior-point methods for $P_*(\kappa)$-linear complementarity problem based on generalized trigonometric barrier function. Int. J. Appl. Math., 30(1): 11-33, 2017.
[5]. Kojima, M., Megiddo, N., Noma, T., Yoshise, A.: A Unified Approach to Interior Point Algorithms for Linear Complementarity problems. Lecture Notes in Computer Science 538, Springer-Verlag, New York (1991)
[6]. Lee, Y.H., Cho, Y.Y., Cho, G.M.: Interior-point algorithms for $P_*(\kappa)$-LCP based on a new class of kernel functions. J. Global Opti., 58(1), 137-149 (2014)
[7]. Lesaja, G., Roos, C.: Unified analysis of kernel-based interior-point methods for $P_*(\kappa)$-linear complementarity problems. SIAM J. Optim., 20(6), 3014-3039 (2010)
[8]. Li, L., Tao, J.Y., El Ghami, M., Cai, X.Z., Wang, G.Q.: A new parametric kernel function with a trigonometric barrier term for $P_*(\kappa)$-linear complementarity problems.Pac. J. Optim., 13(2), 255-278 (2017)
[9]. Li, X., Zhang, M.W., Chen, Y.: An interior-point algorithm for $P_*(\kappa)$-LCP based on a new trigonometric kernel function with a double barrier term. J. Appl. Math. Comput., 53(1-2), 1-20 (2016)
[10]. Liu, Z.Y., Sun, W.Y.: A full-NT-step infeasible interior-point algorithm for SDP based on kernel functions. Appl. Math. Comput., 217(10), 4990-4999 (2011)
[11]. Murty, K.G.: Linear Complementarity, Linear and Nonlinear Programming. Helderman-Verlag, Berlin (1988)
[12]. Roos, C., Terlaky, T., Vial, J.-Ph.: Theory and Algorithms for Linear Optimization. An Interior-Point Approach. John Wiley & Sons, Chichester, UK (1997)
[13]. Sajad, F.H., Alireza, F.J., Reza, P.M., et al.: Complexity of interior point methods for a class of linear complementarity problems using a kernel function with trigonometric growth term. J. Optim. Theory Appl., 178:935–949 (2018).
[14]. Wang, G.Q., Bai, Y.Q.: Polynomial interior-point algorithms for $P_*(\kappa)$ horizontal linear complementarity problem. J. Comput. Appl. Math., 233(2), 248-263 (2009)
[15]. Wang, G.Q.: A new polynomial interior-point algorithm for the monotone linear complementarity problem over symmetric cones with full NT-steps. Asia-Pac. J. Oper. Res., 29(2), 1250015 (20pp) (2012)
[16]. Wang, G.Q., Bai, Y.Q.: Polynomial interior-point algorithms for $P_*(\kappa)$ horizontal linear complementarity problem. J. Comput. Appl. Math., 233(2), 248-263 (2009)
[17]. Wang, G.Q., Bai, Y.Q.: A class of polynomial interior-point algorithms for the Cartesian P-Matrix linear complementarity problem over symmetric cones. J. Optim. Theory Appl., 152(3), 739-772 (2012)

[18]. Wang, G.Q., Bai, Y.Q.: Interior-Point Methods for Symmetric Cone Complementarity Problems: Theoretical Analysis and Algorithm Implementation. Harbin Institute of Technology Press, Harbin (2014). In Chinese

[19]. Wang, G.Q., Lesaja, G.: Full Nesterov-Todd step feasible interior-point method for the Cartesian $P_*(\kappa)$-SCLCP. Optim. Methods Softw., 28(3), 600-618 (2013)

[20]. Wang G.Q., Zhu D.T.: A class of polynomial interior-point algorithms for the Cartesian $P_*(\kappa)$ second-order cone linear complementarity problem. Nonlinear Anal., 73(12), 3705-3722 (2010)

[21]. Wang, W.W., Bi, H.M. Liu, H.W.: A full-Newton step interior-point algorithm for linear optimization based on a finite barrier, Oper. Res. Lett., 44(6), 750-753 (2016)

[22]. Watson, L.T.: Solving the nonlinear complementarity problem by a homotopy method. SIAM J. Optim., 17(1), 36-46 (1979)

[23]. Ye, Y.: On homogeneous and self-dual algorithm for LCP. Math. Program., 76(1), 211-222 (1997)

[24]. Zhang, L.P., Xu, Y.H.: A full-Newton step interior-point algorithm based on modified newton direction, Oper. Res. Lett., 39(5), 317–322 (2011)

Wei Chen, Luhua Yin. "Implementation of Interior-Point Methods for $P_*(\kappa)$ -Linear Complementarity Problem Based On A Kernel Function." IOSR Journal of Mathematics (IOSR-JM) 15.2 (2019): 12-18.