KYBERNETIKA — VOLUME *62* (2026), NUMBER 1, PAGES 55–76

AN INFEASIBLE INTERIOR-POINT ALGORITHM FOR MONOTONE LINEAR COMPLEMENTARITY PROBLEMS BASED ON A FINITE HYPERBOLIC KERNEL FUNCTION

Safa Guerdouh, Wided Chikouche and Behrouz Kheirfam

This paper concerns an infeasible kernel-based interior-point algorithm (IPA) for monotone linear complementarity problems (LCPs). Our algorithm differs from other existing algorithms in the literature since its feasibility step is induced by a finite hyperbolic barrier term. The convergence analysis shows that the proposed algorithm is well-defined and its complexity bound coincides with the currently best-known iteration bound of infeasible interior-point methods for monotone LCPs. Moreover, the practical performance of our algorithm is validated by some extensive numerical tests. To the best of our knowledge, this is the first full-Newton step infeasible IPA based on a hyperbolic kernel function for solving monotone LCPs.

Keywords: linear complementarity problems, infeasible interior-point method, kernel function, polynomial complexity

Classification: 90C51, 90C33

1. INTRODUCTION

LCPs are considered as a classical class of problems in mathematical programming. Although LCP is not an optimization problem, it plays a key role in many scientific computing and engineering applications such as convex quadratic programming, bimatrix games, the free boundary problem, nonnegative constrained least squares problems, market equilibrium problems and many other applications. For more details on the LCP theory and applications, we refer the reader to [7, 9, 15, 24, 26].

Since the introduction of LCPs in the mid 1960's, various methods have been proposed to solve them. Interior-point methods (IPMs) are among the most popular and widely used approaches due to their theoretical foundations and practical efficiency. IPMs were first introduced by Karmarkar [16] for linear optimization (LO). However, the first IPM for LCP was proposed by Kojima et al. [24]. After that, Kojima et al. [23] suggested a unified framework of IPMs for a class of LCPs.

IPMs are classified according to their starting points into feasible IPMs and infeasible IPMs (IIPMs). Feasible IPMs start with a strictly feasible point neighboring the central path and generate a sequence of iterates that maintain the same conditions. Whereas,

DOI: 10.14736/kyb-2026-1-0055

IIPMs start with an arbitrary positive point that is not necessarily feasible, and feasibility is reached as optimality is approached. The first infeasible-start algorithm was proposed by Lustig [27]. Later, Roos [33] introduced the first full-Newton step infeasible IPA for LO based on the classical logarithmic kernel function (KF). In 2016, Kheirfam and Haghighi [20] presented the first IIPM based on a trigonometric KF proposed previously in [22] to solve LO problems.

Motivated by these works, and given that trigonometric and hyperbolic functions share similar geometric properties, we recently introduced an infeasible IPA in [11] to solve LO, based on the following hyperbolic sine KF

$$\psi_{\mathrm{G}}(t) = \frac{t^2-1}{4} - \int_1^t \frac{\sinh(1)}{2\sinh(y)}\,\mathrm{d}y,\ \forall t > 0.$$

As a continuation in this context, we propose in this paper the hyperbolic cosine KF defined below

$$\psi(t) = \frac{t^2-1}{2} - \int_1^t \frac{\cosh(1)}{\cosh(y)}\,\mathrm{d}y,\ \forall t > 0. \tag{1}$$

An interesting feature of this KF is that it has a finite value at the boundary, which differentiates it from the hyperbolic sine KF presented above and from all other hyperbolic KFs introduced in the literature to study feasible IPMs [4, 5, 6, 10, 12, 13, 14, 37, 38, 39].

Our contribution is an infeasible-start IPA for LCP. The algorithm uses the KF (1) with a finite hyperbolic barrier term to determine the search directions in the feasibility step while the classical search directions are used in the centering steps. To our knowledge, the only KF with a finite barrier term used in the literature for IIPMs [18, 25] is

$$\psi_{\text{finite}}(t) = \frac{1}{2}(t-1)^2,\ \forall t > 0.$$

We present some notations used throughout the paper. $\mathbb{R}^n_+$ and $\mathbb{R}^n_{++}$ denote the nonnegative and the positive orthants respectively. $\|\cdot\|$ and $\|\cdot\|_\infty$ denote the Euclidean and the infinity norms in $\mathbb{R}^n$. For given vectors $x, s \in \mathbb{R}^n$, $X = \mathrm{diag}(x)$ denotes the $n \times n$ diagonal matrix whose diagonal entries are the components of x, and the vector xs indicates the component-wise product of x and s. Finally, if $f(x), g(x) \geq 0$ are two real valued functions of a real nonnegative variable, the notation $f(x) = \mathcal{O}(g(x))$ means that $f(x) \leq Cg(x)$ for some positive constant C and $f(x) = \Theta(g(x))$ means that $C_1 g(x) \leq f(x) \leq C_2 g(x)$ for two positive constants C_1 and C_2.

The remainder of this paper is organized as follows : In the next section we recall the basic of kernel-based IIPMs. After that, we analyze the complexity of the proposed algorithm in Section 3. In Section 4, we present our preliminary numerical results. We finally end the paper by providing some concluding remarks in Section 5.

2. PRELIMINARIES

In this section, we briefly describe the basics of IIPMs using KFs for LCP. After that, we outline the statement of the proposed algorithm. We start by recalling the standard montone LCP problem

$$(P)\begin{cases} s = Mx + q, \\ x^T s = 0, \\ x \geq 0,\ s \geq 0, \end{cases}$$

with $x, s, q \in \mathbb{R}^n$ and $M \in \mathbb{R}^{n\times n}$, such that M is a positive semidefinite matrix, i.e.,

$$u^T M u \geq 0,\ \forall u \in \mathbb{R}^n.$$

Unlike feasible IPMs, in IIPMs : for $\epsilon > 0$, the pair (x, s) is called an $\epsilon-$ solution of (P) if the norm of the residual vector $r = s - Mx - q$ does not exceed ϵ, and also $x^T s \leq \epsilon$.

In accordance with IIPMs, we choose $x^0 > 0$ and $s^0 > 0$ such that $x^0 s^0 = \mu^0 e$ for some positive number μ^0 where e denotes the vector of all ones. In this work, we assume that

$$x^0 = \xi_p e, \quad s^0 = \xi_d e, \tag{2}$$

where μ^0 the initial duality gap and (ξ_p, ξ_d) satisfy the following inequalities

$$\|x^*\| < \xi_p,\ \|s^*\| < \xi_d,$$

with (x^*, s^*) a solution of (P). Since M is positive semidefinite, the existence of such solution can be insured if the problem (P) is feasible (Theorem 3.1.2 in [7]). In what follows we assume that (P) is feasible.

2.1. Perturbed problem and central path

For any ν with $0 < \nu \leq 1$, we consider the perturbed problem (P_ν)

$$(P_\nu)\begin{cases} s - Mx - q = \nu r^0, \\ xs = 0,\ x \geq 0,\ s \geq 0, \end{cases}$$

where $r^0 = s^0 - Mx^0 - q$. Note that if $\nu = 1$ then (P_ν) satisfies the interior-point condition (IPC), i.e., (P_ν) has a feasible solution $(x, s) > 0$ which implies that $(x, s) = (x^0, s^0)$ yields a strictly feasible solution of problem (P_ν).

Lemma 2.1. (Mansouri et al. [29, Lemma 4.1]) If the original problem (P) is feasible, then the perturbed problem (P_ν) satisfy the IPC.

Since (P) is feasible, Lemma 2.1 implies that for every $\mu > 0$, the following system has a unique solution

$$\begin{cases} s - Mx - q = \nu r^0, x \geq 0,\ s \geq 0, \\ xs = \mu e. \end{cases}$$

This means that the central path exists and the set of unique solutions $\{(x(\mu,\nu), s(\mu,\nu)) : \mu > 0,\ 0 < \nu \leq 1\}$ forms the central path and $(x(\mu,\nu), s(\mu,\nu))$ are the $\mu-$centers of (P_ν) with $\mu = \nu\mu^0$. Furthermore, we denote $(x(\mu,\nu), s(\mu,\nu)) = (x_\mu, s_\mu)$ for simplicity purposes.

Fixing $\mu > 0$, let us defined the τ-neighbourhood $\mathcal{N}(\tau, \mu)$ for (P_ν) as follows

$$\mathcal{N}(\tau, \mu) = \{(x, s) \in \mathbb{R}^n \times \mathbb{R}^n : s - Mx - q = \nu r^0 \text{ and } \delta(x, s; \mu) \leq \tau\},$$

with τ the radius of the neighbourhood and δ a so-called proximity measure. δ is used to measure the distance between an iterate (x, s) and the $\mu-$center (x_μ, s_μ) of the perturbed problem (P_ν).

In this paper, we define the proximity measure δ as follows

$$\delta(x, s; \mu) := \delta(v) := \frac{1}{\sqrt{2}} \left\| v^{-1} - v \right\|, \text{ where } v := \sqrt{\frac{xs}{\mu}}. \tag{3}$$

Remark 2.2.

- According to [34], we call the step where the present iterate is in a certain neighbourhood of the current μ-center an outer iteration, and the procedure to get a pair (x, s) in the neighborhood of this μ-center an inner iteration. In the algorithm, we use the proximity $\delta(x, s; \mu)$ to control the iterates.
- The choice of the parameter θ is an important ingredient of IPMs. Generally, when θ is a constant independent of n, for example $\theta = \frac{1}{2}$, the algorithm is called a large-update (or long-step) method. On the other hand, if θ depends on the problem dimension n, for example $\theta = \frac{1}{2\sqrt{n}}$, then the algorithm is referred to as a small-update (or short-step) method.

2.2. An iteration of the algorithm

We outline the structure of one iteration of the algorithm. Each main iteration consists of a feasibility step, a μ-update, and centering steps.

- **Feasibility Step:** We first generate iterates (x^f, s^f) that are strictly feasible for the perturbed problem (P_{ν^+}), where $\nu^+ = (1 - \theta)\nu$ with $\theta \in (0, 1)$. This step ensures feasibility while reducing the residual norm. However, the new iterate may not be sufficiently close to $\tau-$neighborhood of the central path. Hence, the centering steps are needed to return to the neighborhood.
- **Centering Steps:** To restore centrality, we perform a limited number of centering iterations while keeping ν^+ fixed. These steps adjust the iterates toward the τ-neighborhood of the central path while reducing μ to $\mu^+ = (1 - \theta)\mu$. In fact, after the feasibility step, we perform a few centering steps in order to get the iterate (x^+, s^+) which is feasible for (P_{ν^+}) such that $\delta(x^+, s^+; \mu^+) \leq \tau$. Starting with the iterate (x^f, s^f), we solve system (4) to get the search directions $(\Delta x, \Delta s)$

$$\begin{cases} M\Delta x - \Delta s = \theta\nu r^0, \\ s\Delta x + x\Delta s = \mu e - xs. \end{cases} \tag{4}$$

  Since M is positive semidefinite, for any $x > 0$ and $s > 0$, this system uniquely defines the search directions $(\Delta x, \Delta s)$.

This process continues until the duality gap $x^T s$ and the norm of the residual vector r are both below the accuracy parameter ϵ.

2.3. Kernel functions in the feasibility step

To obtain iterates that are feasible for (P_{ν^+}), we need to solve the following system of equations

$$\begin{cases} M\Delta^f x - \Delta^f s = \theta\nu r^0, \\ s\Delta^f x + x\Delta^f s = \mu e - xs. \end{cases} \tag{5}$$

Then the new feasible iterates are

$$x^f = x + \Delta^f x, \quad s^f = s + \Delta^f s. \tag{6}$$

Let the scaled search directions d_x^f and d_s^f be defined as follows

$$d_x^f = \frac{v\Delta^f x}{x}, \quad d_s^f = \frac{v\Delta^f s}{s}. \tag{7}$$

System (5) is then rewritten in the following form

$$\begin{cases} \overline{M} d_x^f - d_s^f = \theta\nu v s^{-1} r^0, \\ d_x^f + d_s^f = v^{-1} - v, \end{cases} \tag{8}$$

where $\overline{M} = MS^{-1}X, \;\; S = \text{diag}(s), \;\; X = \text{diag}(x)$.

Observe that the right-hand side in the last equation of (8) is equal to minus gradient of the classical logarithmic scaled barrier (proximity) function

$$\Psi_c(v) = \sum_{i=1}^{n} \psi_c(v_i), \tag{9}$$

where

$$\psi_c(t) = \frac{t^2 - 1}{2} - \log t.$$

ψ_c is called the logarithmic KF of the logarithmic barrier function Ψ_c. The basic of kernel-based IPMs is to replace ψ_c by any strictly convex function ψ defined from $\mathbb{R}_{++}$ to $\mathbb{R}_+$ which is minimal at $t = 1$ with $\psi(1) = 0$. The corresponding proximity function Ψ is then obtained by replacing ψ_c with ψ in (9). System (8) is then converted to

$$\begin{cases} \overline{M} d_x^f - d_s^f = \theta\nu v s^{-1} r^0, \\ d_x^f + d_s^f = -\nabla\Psi(v). \end{cases} \tag{10}$$

In the sequel, we will use the norm-based proximity measure σ defined by

$$\sigma(x, s; \mu) = \sigma(v) = \left\| d_x^f + d_s^f \right\| = \left\| \nabla\Psi(v) \right\|.$$

Function σ is a suitable proximity that measures the closeness to the central path of the monotone LCP, because

$$\sigma(v) = 0 \Leftrightarrow \nabla\Psi(v) = 0 \Leftrightarrow v = e.$$

Note that a barrier function does not need to be kernel-based, although that is, most common, it's enough that it be a smooth, strictly convex function defined for all $v > 0$, which is minimal at $v = e$, with $\Psi(v) = 0$.

The effectiveness of kernel-based IPMs, both in theory and in practice, strongly depends on the selection of a suitable KF. The KF plays a fundamental role in shaping the central path of an IPM, defining the proximity measure, and determining the search directions. Particularly, the gradient of the barrier function derived from the KF is used to solve system (10), thereby determining the search directions. Moreover, the proximity measure σ is defined in terms of the KF. In addition, the KF is employed to establish a τ-neighborhood around the μ-center, which is used as part of the stopping criterion in the inner iterations of the algorithm. Therefore, both the KF and the associated proximity measure directly influence the iteration bounds, as their mathematical properties are essential for the complexity analysis. Hence, the overall complexity rate of a kernel-based IPM is highly dependent on the chosen KF.

KFs were first introduced for feasible IPMs. The pioneering feasible primal-dual IPM based on the classical logarithmic barrier function was introduced by Roos et al. [34]. Later, Peng et al. [31] proposed primal-dual IPMs that uses the so-called self-regular (SR) barrier functions, significantly improving the theoretical complexity achieved with the classical logarithmic KF from $\mathcal{O}(n \log \frac{n}{\epsilon})$ to $\mathcal{O}(\sqrt{n} \log n \log \frac{n}{\epsilon})$, where n represents the number of variables, and ϵ is the desired accuracy in terms of the objective value. This success motivated the exploration of alternative KFs in lieu of the classical logarithmic KF. For instance, Bai et al. [2] introduced in 2002 an exponential barrier term with finite values at the boundary of the feasible region.

In 2004, Bai et al. [3] proposed a broader class of eligible KFs that contains both the classical logarithmic and the SR KFs. They presented a unified analysis of feasible primal-dual IPMs based on eligible KFs for LO. They also introduced the first KF with a trigonometric barrier term. Since then, many KFs were introduced and analysed. These functions can be classified chronologically according to the type of their barrier term into: simple algebraic, exponential, trigonometric and hyperbolic.

KFs were also introduced to define IIPMs. Roos [33] presented a full-Newton step infeasible IPA for LO, which was based on the classical logarithmic KF. The method was extended to semidefinite optimization (SDO) and monotone LCP by Mansouri et al. [28, 29]. Following a completely different approach from Roos's work, Salahi et al. [35, 36] proposed an IIPM for LO built upon the SR KF defined earlier in [31].

The first IIPM for LO employing a trigonometric KF was introduced by Kheirfam and Haghighi in 2016 [20], drawing on the function originally proposed in [22] for feasible IPMs. The following year, they proposed a new IIPM for solving P_*-matrix LCP [19] based on the trigonometric KF introduced in [17] for feasible IPMs.

Pirhaj et al. [32] extended the approach of [20] to solve monotone LCPs. Then, Moslemi and Kheirfam [30] proposed an IIPM for solving SDO problems based on a new trigonometric KF. After that, Kheirfam and Haghighi [21] proposed an IIPM for solving LO problems based on an other new trigonometric KF. Later, Guerdouh et al. [11] presented the first full-Newton step IIPM based on a hyperbolic KF to solve LO problems.

In this work, we consider the barrier function based on the new KF ψ defined previously in (1). We also give its first derivative

$$\psi'(t) = t - \frac{\cosh(1)}{\cosh(t)},\ \forall t > 0. \tag{11}$$

It is worth noting that ψ takes a finite value at the boundary of the feasible region. To be more precise

$$\lim_{t\to 0} \psi(t) = \psi(0) < \infty,$$

since the function $x \mapsto \frac{\cosh(1)}{\cosh(x)}$ is continuous on $[0, 1]$.

Based on our KF, the proximity measure is defined as follows

$$\sigma(x, s; \mu) = \sigma(v) = \left\| \frac{\cosh(e)}{\cosh(v)} - v \right\|. \tag{12}$$

We summarize the generic kernel-based IIPM for monotone LCP in Algorithm 1.

Algorithm 1 : kernel-based infeasible IPA for monotone LCP

Input
a threshold parameter $\tau > 0$;
an accuracy parameter $\epsilon > 0$;
a fixed barrier update parameter $\theta \in (0, 1)$;
a barrier function Ψ;
initialization parameters $\xi_p,\ \xi_d > 0$.
begin
$x := \xi_p e,\ s := \xi_d e,\ \mu := \mu^0$;
while $\max(x^T s, \|r\|) \geq \epsilon$ (outer iteration)
begin
feasibility step :
Solve system (10) and use (7) to obtain $(\Delta^f x, \Delta^f s)$;
Update $(x, s) := (x, s) + (\Delta^f x, \Delta^f s)$;
Update $v = \sqrt{\frac{xs}{\mu}}$;
Calculate $\delta(x, s; \mu) := \delta(v) := \frac{1}{\sqrt{2}} \left\| v^{-1} - v \right\|$;
$\mu-$ and $\nu-$updates:
$\mu :\ = (1 - \theta)\mu$;
$\nu :\ = (1 - \theta)\nu$;
centering steps:
while $\delta(x, s; \mu) := \delta(v) > \tau$ (inner iteration)
Solve system (4) to obtain $(\Delta x, \Delta s)$;
Update $(x, s) := (x, s) + (\Delta x, \Delta s)$;
Update $v = \sqrt{\frac{xs}{\mu}}$;
end while (inner iteration)
end while (outer iteration)
end

---

3. ANALYSIS OF THE ALGORITHM

In what follows, we will study Algorithm 1 based on the barrier function induced by the new KF (1). We show that the algorithm is well defined. We first state some useful technical lemmas. Then, we prove that the iterates obtained after the feasibility step are strictly feasible. After that, we derive an upper bound for the number of inner iterations required by the algorithm to obtain an optimal solution.

3.1. Technical lemmas

Let δ be the proximity previously defined in (3). For simplicity purposes, we put $\delta(x, s, \mu) = \delta$. Then, we recall the following well-known results.

Lemma 3.1. (Mansouri et al. [29, Lemma 5.6]) Let $\rho(\delta) := \frac{\sqrt{2}}{2}\delta + \sqrt{1 + \frac{1}{2}\delta^2}$. Then

$$\frac{1}{\rho(\delta)} \leq v_i \leq \rho(\delta),\ i = 1, \ldots, n.$$

Lemma 3.2. (Mansouri et al. [29, Lemma 3.5]) If $\delta < 1$, then x^+ and s^+ are positive and

$$\delta(x^+, s^+; \mu) \leq \frac{\delta^2}{\sqrt{2(1-\delta^2)}}.$$

A direct consequence of this lemma is the following corollary.

Corollary 3.3. (Mansouri et al. [29, Corollary 3.6]) If $\delta \leq \frac{1}{\sqrt{2}}$ then $\delta(x^+, s^+; \mu) \leq \delta^2$ i.e., quadratic convergence to the $\mu-$center is obtained.

Remark 3.4. Using Corollary 3.3, we conclude that $\delta(x^+, s^+; \mu^+) \leq \tau$ will hold after

$$\left\lceil \log_2 \left(\log_2 \frac{1}{\tau^2} \right) \right\rceil$$

centering steps.

The following lemma provides an important feature of the hyperbolic cosine function which will be very helpful in the complexity analysis of our algorithm.

Lemma 3.5. One has

$$\left| t - \frac{\cosh(1)}{\cosh(t)} \right| \leq 2 \left| t - \frac{1}{t} \right|,\ \forall t > 0. \tag{13}$$

Proof. We clearly see that (13) is verified for $t = 1$. Let g be the function defined for $t > 0$ as follows

$$g(t) = \left| t - \frac{\cosh(1)}{\cosh(t)} \right| - 2 \left| t - \frac{1}{t} \right|$$

$$= \begin{cases} \dfrac{2}{t} - t - \dfrac{\cosh(1)}{\cosh(t)}, \ \forall t \geq 1, \\ t + \dfrac{\cosh(1)}{\cosh(t)} - \dfrac{2}{t}, \ \forall t \leq 1, \end{cases}$$

and its derivative is

$$g'(t) = \begin{cases} \dfrac{\cosh(1)\sinh(t)}{\cosh^2(t)} - \left(\dfrac{2}{t^2} + 1\right), \ \forall t > 1, \\ \left(\dfrac{2}{t^2} + 1\right) - \dfrac{\cosh(1)\sinh(t)}{\cosh^2(t)}, \ \forall t < 1. \end{cases}$$

An important observation is that $\lim_{t \to 0^+} g(t) = \lim_{t \to +\infty} g(t) = -\infty$ and $g(1) = 0$. So, to prove inequality (13), we need to verify that

$$g'(t) > 0, \ \forall 0 < t < 1, \ \text{ and } \ g'(t) < 0, \ \forall t > 1.$$

We start with the case $t > 1$. Since the hyperbolic cosine function is monotonically increasing, we get

$$g'(t) < \tanh(t) - 1 - \frac{2}{t^2}.$$

Moreover, using the fact that

$$\tanh(t) < 1, \ \forall t > 0,$$

we obtain the desired inequality. Now, we move to the second case, i.e., $t < 1$. We rewrite g' as follows

$$\begin{aligned} g'(t) &= \frac{t^2\cosh^2(t) + \left(2\cosh^2(t) - \cosh(1)t^2\sinh(t)\right)}{t^2\cosh^2(t)}, \\ &> \frac{2(\cosh^2(t) - t^2)}{t^2\cosh^2(t)}, \end{aligned}$$

where the last inequality is obtained using the increase of the hyperbolic sine function and the fact that $\sinh(1)\cosh(1) < 2$. Moreover, the Taylor expansion of the hyperbolic cosine function implies that

$$\cosh(t) > t, \ \forall t > 0.$$

It follows that $g'(t) > 0, \ \forall t < 1$, which completes the proof. □

3.2. Analysis of the feasibility step

In this subsection, we prove that after the feasibility step the new iterate (x^f, s^f) is strictly feasible for (P_{ν^+}) and is quadratically convergent to the μ^+−centers, i.e.

$$\delta(x^f, s^f; \mu^+) \leq \frac{1}{\sqrt{2}}.$$

The next lemma reveals a sufficient condition for the strict feasibility of the iterate (x^f, s^f).

Lemma 3.6. The new iterates (x^f, s^f) are strictly feasible if

$$\frac{v\cosh(e)}{\cosh(v)} + d_x^f d_s^f > 0. \tag{14}$$

Proof. We first introduce a step length $\alpha \in [0,1]$. Then, we define

$$x^\alpha = x + \alpha\Delta^f x, \quad s^\alpha = s + \alpha\Delta^f s.$$

Suppose that (14) is satisfied. Using (11), the last equation of (10) and notation (7), we get

$$\begin{aligned} x^\alpha s^\alpha &= \mu(v + \alpha d_x^f)(v + \alpha d_s^f) \\ &= \mu\left(v^2 + \alpha v(d_x^f + d_s^f) + \alpha^2 d_x^f d_s^f\right) \\ &= \mu\left(v^2 + \alpha\left(\frac{v\cosh(e)}{\cosh(v)} - v^2\right) + \alpha^2 d_x^f d_s^f\right) \\ &> \mu\left((1-\alpha)v^2 + \alpha(1-\alpha)\frac{v\cosh(e)}{\cosh(v)}\right). \end{aligned}$$

Obviously $(1-\alpha)v^2 + \alpha(1-\alpha)\dfrac{v\cosh(e)}{\cosh(v)} \geq 0$. Thus, for every $\alpha \in [0,1]$, $x^\alpha s^\alpha > 0$ which implies that none of the components of x^α and s^α vanishes. Taking $\alpha = 1$, we obtain $x^f > 0$ and $s^f > 0$, which completes the proof. □

The following lemma gives an upper bound for the proximity function after the feasibility step.

Lemma 3.7. If $\dfrac{v\cosh(e)}{\cosh(v)} + d_x^f d_s^f > 0$, then

$$\delta(v^f) \leq \frac{\theta\sqrt{n} + 3\sqrt{2}\delta\rho(\delta) + \frac{1}{2}(\|d_x^f\|^2 + \|d_s^f\|^2)}{\sqrt{2(1-\theta)\left(\frac{1}{\rho(\delta)^2} - 2\sqrt{2}\delta\rho(\delta) - \frac{1}{2}(\|d_x^f\|^2 + \|d_s^f\|^2)\right)}}. \tag{15}$$

Proof. From definition (3), we have

$$\delta(x^f, s^f; \mu+) := \delta(v^f) = \frac{1}{\sqrt{2}}\left\|(v^f)^{-1} - v^f\right\|, \text{ where } v^f = \sqrt{\frac{x^f s^f}{\mu^+}}.$$

This implies that

$$\delta(v^f) \leq \frac{1}{\sqrt{2}v_{\min}^f}\left\|e - (v^f)^2\right\|. \tag{16}$$

In addition, using (11) and the second equation of (10), we get

$$x^f s^f = \frac{xs}{v^2}(v + d_x^f)(v + d_s^f)$$

$$
\begin{aligned}
&= \mu(v^2 + v(d_x^f + d_s^f) + d_x^f d_s^f) \\
&= \mu\left(v^2 + v\left(\frac{\cosh(e)}{\cosh(v)} - v\right) + d_x^f d_s^f\right). \qquad (17)
\end{aligned}
$$

It follows that

$$
\begin{aligned}
\left\|e - (v^f)^2\right\| \leq & \frac{1}{1-\theta}\left\|(1-\theta)e - v^2 - \left(\frac{v\cosh(e)}{\cosh(v)} - v^2\right) - d_x^f d_s^f\right\| \\
= & \frac{1}{1-\theta}\left\|(e - v^2) - \theta e + v\left(\frac{\cosh(e)}{\cosh(v)} - v\right) - d_x^f d_s^f\right\| \\
\leq & \frac{1}{1-\theta}\left(\left\|v(v^{-1} - v)\right\| + \theta\sqrt{n} + \left\|v\left(\frac{\cosh(e)}{\cosh(v)} - v\right)\right\| + \left\|d_x^f d_s^f\right\|\right). \qquad (18)
\end{aligned}
$$

Moreover, using definition (3), Lemma 3.1 and Lemma 3.5 we obtain the following inequalities

$$
\begin{aligned}
\left\|v(\frac{1}{v} - v)\right\|^2 &= \sum_{i=1}^n \left|v_i\left(\frac{1}{v_i} - v_i\right)\right|^2 \\
&\leq \rho(\delta)^2 \sum_{i=1}^n \left|\frac{1}{v_i} - v_i\right|^2 \\
&= 2\rho(\delta)^2\delta^2, \qquad (19)
\end{aligned}
$$

and

$$
\begin{aligned}
\left\|v\left(v - \frac{\cosh(e)}{\cosh(v)}\right)\right\|^2 &= \sum_{i=1}^n \left|v_i\left(v_i - \frac{\cosh(1)}{\cosh(v_i)}\right)\right|^2 \\
&\leq \rho(\delta)^2 \sum_{i=1}^n \left|\frac{1}{v_i} - v_i\right|^2 \\
&= 8\rho(\delta)^2\delta^2. \qquad (20)
\end{aligned}
$$

Using (19), (20) and the fact that

$$
\left\|d_x^f d_s^f\right\| \leq \frac{1}{2}\left(\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2\right),
$$

we arrive at

$$
\left\|e - (v^f)^2\right\| \leq \frac{1}{1-\theta}\left(3\sqrt{2}\rho(\delta)\delta + \theta\sqrt{n} + \frac{1}{2}\left(\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2\right)\right). \qquad (21)
$$

In a similar way we have

$$
\begin{aligned}
(v_{\min}^f)^2 &= \min_i \frac{1}{1-\theta}\left({v_i}^2 + v_i\left(\frac{\cosh(1)}{\cosh(v_i)} - v_i\right) + d_{x_i}^f d_{s_i}^f\right) \\
&\geq \frac{1}{1-\theta}\left(v_{\min}^2 - \left\|v\left(v - \frac{\cosh(e)}{\cosh(v)}\right)\right\| - \left\|d_x^f d_s^f\right\|\right)
\end{aligned}
$$

$$\geq \frac{1}{1-\theta}\left(\frac{1}{\rho(\delta)^2} - 2\sqrt{2}\rho(\delta)\delta - \frac{1}{2}\left(\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2\right)\right). \tag{22}$$

Substituting (21) and (22) into (16) yields the desired inequality. □

Corollary 3.8. Let $n \geq 2$, $\delta \leq \tau = \frac{1}{16}$ and $\theta \leq \frac{1}{22\sqrt{n}}$. If

$$\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2 \leq \frac{3}{5},$$

then after the feasibility step iteration (x^f, s^f) is strictly feasible and lies in the quadratic convergence neighborhood with respect to the μ^+-center of (P_{ν^+}).

Proof. From (22)

$$\begin{aligned}\left({v_i}^2 + v_i\left(\frac{\cosh(1)}{\cosh(v_i)} - v_i\right) + {d_{x_i}}^f {d_{s_i}}^f\right) &\geq \left(\frac{1}{\rho(\delta)^2} - 2\sqrt{2}\rho(\delta)\delta - \frac{1}{2}\left(\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2\right)\right)\\ &\geq \left(\frac{1}{\rho(\tau)^2} - 2\sqrt{2}\rho(\tau)\tau - \frac{3}{10}\right)\\ &\simeq 0.1307 > 0,\end{aligned}$$

where the second inequality is due to the fact that $\rho(\delta)$ is monotonically increasing with respect to δ. Hence, $\dfrac{v\cosh(e)}{\cosh(v)} + d_x^f d_s^f > 0$, which implies using Lemma 3.6 that (x^f, s^f) is strictly feasible. In addition, from (15) we get

$$\begin{aligned}\delta(v^f) \leq &\frac{\frac{1}{22} + \frac{3\sqrt{2}}{16}\left(\frac{\sqrt{2}}{32} + \sqrt{1 + \frac{1}{2}\frac{1}{16^2}}\right) + \frac{3}{10}}{\sqrt{2\left(1 - \frac{1}{22\sqrt{2}}\right)\left(\frac{1}{\left(\frac{\sqrt{2}}{32} + \sqrt{1 + \frac{1}{2}\frac{1}{16^2}}\right)^2} - \frac{2\sqrt{2}}{16}\left(\frac{\sqrt{2}}{32} + \sqrt{1 + \frac{1}{2}\frac{1}{16^2}}\right) - \frac{3}{10}\right)}}\\ &\leq 0.682,\end{aligned}$$

which implies that $\delta(v^f) \leq \dfrac{1}{2}$. Therefore, (x^f, s^f) lies in the quadratic convergence neighborhood with respect to the μ^+-center of (P_{ν^+}). □

3.3. The scaled search directions and the value of the parameter θ

In this subsection, we will proceed as in Sections 5.2 and 5.3 in [29]. We apply Lemma 5.5 in [29] on system (10) with $u = d_x^f,\ z = d_s^f,\ a = \theta\nu v s^{-1} r_q^0$ and $b = \dfrac{\cosh(e)}{\cosh(v)} - v$, and we get

$$\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2 \leq \left\|\frac{\cosh(e)}{\cosh(v)} - v\right\|^2 + 2\left(\left\|\theta\nu v s^{-1} r_q^0\right\|^2 + \left\|\frac{\cosh(e)}{\cosh(v)} - v\right\|\left\|\theta\nu v s^{-1} r_q^0\right\|\right).$$

Following the same procedure used to obtain (20), we obtain

$$\left\|d_x^f\right\|^2 + \left\|d_s^f\right\|^2 \leq 8\delta^2 + 2\left(\left\|\theta\nu v s^{-1} r_q^0\right\| + 2\sqrt{2}\delta\right)\left\|\theta\nu v s^{-1} r_q^0\right\|.$$

From [29], we have

$$\left\|\theta\nu v s^{-1} r_q^0\right\| \leq 3n\theta\rho(\delta)\left(2+\rho(\delta)^2\right).$$

These last two inequalities imply that

$$\|d_x^f\|^2 + \|d_s^f\|^2 \leq 8\delta^2 + 6n\theta\rho(\delta)\left(2+\rho(\delta)^2\right)\left(3n\theta\rho(\delta)\left(2+\rho(\delta)^2\right) + 2\sqrt{2}\delta\right). \qquad (23)$$

At this stage, we specify the value of the parameter θ as follows

$$\theta = \frac{1}{22n}.$$

Using this special value of θ, we can easily verify that

$$8\delta^2 + 6n\theta\rho(\delta)\left(2+\rho(\delta)^2\right)\left(3n\theta\rho(\delta)\left(2+\rho(\delta)^2\right) + 2\sqrt{2}\delta\right) \leq \frac{3}{5}. \qquad (24)$$

By Corollary 3.8 we conclude that for $n \geq 2$, $\delta \leq \tau = \frac{1}{16}$ and $\theta = \frac{1}{22n}$, the iterate obtained after the feasibility step is strictly feasible and lies in the quadratic convergence neighborhood with respect to the μ^+-center of (P_{ν^+}).

3.4. Iteration bound

We arrive at the final result of this section which summarizes the complexity bound. As we found in the previous sections, starting from an iterate (x, s) satisfying $\delta(x, s; \mu) \leq \tau$ with τ and θ previously defined, the new iterate (x_+, s_+) is strictly feasible and $\delta(x_+, s_+; \mu^+) \leq \frac{1}{\sqrt{2}}$. Therefore, we may state the main result of the paper.

Theorem 3.9. If (P) has an optimal solution (x^*, s^*) such that $\|x^*\|_\infty \leq \xi_p$ and $\|s^*\|_\infty \leq \xi_d$, then after at most

$$88n \log \frac{\max\{(x^0)^T s^0, \|r^0\|\}}{\epsilon}$$

iterations the algorithm finds an $\epsilon-$solution for the monotone LCP (P).

P r o o f. Recall that in each main iteration, both the duality gap and the norm of the residual r are reduced by $1-\theta$. Thus letting K be the number of $\mu-$updates before one $\epsilon-$solution is obtained, one has

$$e(x, s) \leq (1-\theta)^K\, e(x_0, s_0) \leq \epsilon,$$

with $e(x, s) = \max(x^T s, \|r\|)$ and $r = s - Mx - q$. Thus,

$$K \leq \frac{1}{\log(1-\theta)} \log \frac{e(x_0, s_0)}{\epsilon} = \frac{1}{\log(1-\theta)} \log \frac{\max((x^0)^T s^0, \|r^0\|)}{\epsilon}.$$

Using the fact that $\theta > \log(1-\theta)$ for any $\theta \in (0,1)$, we arrive at

$$K \leq \frac{1}{\theta} \log \frac{\max((x^0)^T s^0, \|r^0\|)}{\epsilon}.$$

Setting $\theta = \frac{1}{22n}$, we conclude that the total number of $\mu-$updates (outer iterations) before one $\epsilon-$solution is obtained is bounded by

$$22n \log \frac{\max((x^0)^T s^0, \|r^0\|)}{\epsilon}.$$

Moreover, according to Remark 3.4, the number of centering steps required to obtain iterates that satisfy $\delta(x^+, s^+; \mu^+) \leq \tau$ is at most 3. This implies that the number of inner iterations in each main iteration is at most 4. Knowing that an upper bound for the total number of iterations is obtained by multiplying the number of inner and outer iterations, we deduce that the total number of iterations needed by the algorithm to obtain an $\epsilon-$solution of the monotone LCP (P) is bounded by

$$4 \times 22n \log \frac{\max((x^0)^T s^0, \|r^0\|)}{\epsilon} = 88n \log \frac{\max((x^0)^T s^0, \|r^0\|)}{\epsilon}.$$

□

4. NUMERICAL TESTS

In this section, we showcase the practical efficiency of the proposed algorithm outlined in Algorithm 1 by performing some preliminary numerical tests. Our experiments were directly implemented in MATLAB R2012b and performed on Supermicro dual-2.80 GHz Intel Core i5 server with 4.00 Go RAM.

4.1. Comparison with the infeasible interior-point algorithm outlined in [40]

First, we compare the practical performance of our algorithm (Alg. 2) with the infeasible IPA (Alg. 1) proposed by Zhang et al. [40] which is based on the locally KF ψ_l defined as follows

$$\psi_l(t) = (1-t)^2, \ \forall t > 0.$$

For each algorithm, we choose the suitable theoretical values of the parameters τ and θ that guarantee the convergence. Both considered algorithms share the same threshold parameter $\tau = \dfrac{1}{16}$ and accuracy parameter $\epsilon = 10^{-4}$ while having different $\theta : \theta = \dfrac{1}{22n}$ for Alg. 2 and $\theta = \dfrac{1}{33n}$ for Alg. 1. Both algorithms were implemented in MATLAB and were tested on 6 LO problems from the Netlib repository. We first need to transform them into LCP form as follows.

Let's consider the linear program in primal-dual standard form

$$(LP_P) \quad \min c^T x \text{ s.t. } Ax = b, \quad x \geq 0,$$

$$(LP_D) \quad \max b^T y \text{ s.t. } A^T y + s = c, \quad s \geq 0,$$

with $A \in \mathbb{R}^{m \times n}$, $x, s, c \in \mathbb{R}^n$ and $b, y \in \mathbb{R}^m$. It's well-known that solving (LP_P) and (LP_D) is equivalent to solving

$$\begin{cases} Ax = b, x \geq 0, \\ A^T y + s = c, s \geq 0, \\ xs = 0. \end{cases} \tag{25}$$

Defining the matrix M and the vector q as follows

$$M = \begin{pmatrix} 0_{n,n} & A^T \\ -A & 0_{m,m} \end{pmatrix} \text{ and } q = \begin{pmatrix} c \\ b \end{pmatrix},$$

system (25) can be reformulated as a LCP of form (P).

The results are summarized in the table below. For each example, we use **bold** font to highlight the best (i.e., the smallest) iteration number.

Problem	Alg. 1	Alg. 2
afiro	59067	**39374**
blend	133426	**88947**
sc50a	86698	**57795**
sc50b	86326	**57548**
sc105	197944	**131959**
sc205	221576	**147713**

Tab. 1. Total number of iterations for some Netlib problems.

Based on the results presented in Table 1, we can clearly see that our algorithm outperformed Alg. 1 with a difference in iteration numbers that can amount up to 73863. In fact, our algorithm achieved the best iteration number in all the realized experiments.

4.2. The effect of the parameter θ

We first investigate the effect of the barrier update parameter θ on the computational performance of the algorithm. The latter was tested for several values of the parameter θ to solve the monotone LCP (P) with M and q defined as follows

$$M(i,j) = \begin{cases} 1 & \text{if } i = j, \\ 2 & \text{if } i < j, \\ 0 & \text{otherwise,} \end{cases}$$

and $q = -e$. We start by an initial point $(x^0, s^0) = (\xi_p e, \xi_d e)$ with $\xi_p = \frac{1}{2}$ and $\xi_d = 1$. For this example, we set $\epsilon = 10^{-4}$, $\tau = \frac{1}{16}$, and

$$\theta \in \left\{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, \frac{1}{22}, \frac{1}{16}, \frac{1}{22\sqrt{n}}, \frac{1}{22n}, \frac{1}{16\sqrt{n}}, \frac{1}{\sqrt{10n}}, \frac{1}{20+n}\right\}.$$

Our algorithm solved the aforementioned example for sizes

$$n \in \{5, 10, 25, 50, 100, 200, 500, 1000\}.$$

The results are summarized in Table 2.

θ \ n	5	10	25	50	100	200	300	500	1000
0.01	1037	1173	1330	1440	1547	1652	1713	1789	1893
0.05	204	230	261	283	303	324	336	351	371
0.1	99	112	127	138	148	158	164	171	181
0.2	47	53	60	65	70	75	78	81	86
0.3	30	34	38	41	44	47	49	51	54
0.4	21	24	27	29	31	33	34	36	38
0.5	16	18	20	21	23	24	25	26	28
0.6	12	13	15	16	17	19	19	20	21
0.7	**9**	**10**	**12**	**13**	**13**	**14**	**15**	**15**	**16**
$\frac{1}{22}$	225	254	288	312	355	357	370	387	409
$\frac{1}{16}$	126	183	208	225	241	258	267	279	295
$\frac{1}{\sqrt{10n}}$	69	112	205	317	484	734	934	1263	1893
$\frac{1}{22n}$	1142	2587	7344	15908	34177	73003	113556	–	–
$\frac{1}{22\sqrt{n}}$	508	815	1464	2244	3411	5155	6548	8835	–
$\frac{1}{16\sqrt{n}}$	368	591	1063	1630	2479	3747	4760	6423	–
$\frac{1}{20+n}$	256	348	595	1006	1857	3643	5498	9340	–

Tab. 2. Number of iterations for different values of θ.

From Table 2, we can see that the number of iteration is effected by θ. In fact, $\theta = 0.7$ outperformed all the other values with a difference in iteration numbers that can amount up to 113551, while for $\theta = 0.6$, the obtained iteration numbers are always close to the best ones. Furthermore, the algorithms with $\theta \in \left\{ \frac{1}{22\sqrt{n}}, \frac{1}{22n}, \frac{1}{16\sqrt{n}}, \frac{1}{20+n} \right\}$ failed to converge for the number of variables considered in this subsection.

The same conclusion can't be drawn for other LCP problems. However, although the convergence of the algorithm for constant values of θ is not theoretically guaranteed, these constant values can perform better than the theoretical values of θ which depend on the size of the problem.

4.3. Comparison with feasible interior-point algorithm outlined in [1]

In this subsection, numerical tests were carried out to compare our algorithm (Algorithm 2) with the primal-dual IPA introduced by Amini and Peyghami (Algorithm 1) [1]. The latter is based on the following exponential KF

$$\psi_A(t) = \frac{t^2-1}{2} - \int_1^t \frac{e^{y^{-p}}}{e}\,\mathrm{d}y, \ \forall t > 0,$$

with $p = 1$. Since Algorithm 1 is a feasible IPM, it requires a strictly feasible starting point. That's the reason we used the self-dual embedding model [34] to enable the start of the algorithm. Both algorithms were implemented to solve a set of 6 problems from the Netlib repository. Each algorithm was tested for $\epsilon = 10^{-8}$, $\tau = 2n$ and $\theta \in \{0.1, 0.2, 0.25, 0.3\}$. The number of iterations and the CPU time (i.e., the time required to obtain an $\epsilon-$approximate solution) for every problem are presented in tables below.

Problem		Algorithm 1	Algorithm 2
afiro	iter	**246**	306
	CPU	21.5334	0.6426
blend	iter	**235**	292
	CPU	34.1202	**4.3859**
sc50a	iter	**241**	283
	CPU	23.1407	**2.3707**
sc50b	iter	**244**	282
	CPU	22.5959	**2.1385**
sc105	iter	**245**	300
	CPU	49.8902	**7.2078**
sc205	iter	**281**	332
	CPU	169.9907	**29.1676**

Tab. 3. Total number of iterations for $\theta = 0.1$.

Problem		Algorithm 1	Algorithm 2
afiro	iter	276	**145**
	CPU	26.2152	**0.7731**
blend	iter	**129**	139
	CPU	29.4119	**2.3022**
sc50a	iter	256	**134**
	CPU	27.6686	**1.1886**
sc50b	iter	256	**133**
	CPU	27.6358	**1.1362**
sc105	iter	267	**142**
	CPU	57.1004	**3.4786**
sc205	iter	274	**157**
	CPU	284.2934	**13.2209**

Tab. 4. Total number of iterations for $\theta = 0.2$.

Problem		Algorithm 1	Algorithm 2
afiro	iter	292	**122**
	CPU	26.4725	0.6402
blend	iter	185	**107**
	CPU	37.5712	**1.5542**
sc50a	iter	272	**104**
	CPU	27.4236	**1.0647**
sc50b	iter	277	**104**
	CPU	28.047	**1.0695**
sc105	iter	281	**110**
	CPU	57.5442	**3.0222**
sc205	iter	287	**122**
	CPU	169.428	**10.1915**

Tab. 5. Total number of iterations for $\theta = 0.25$.

Problem		Algorithm 1	Algorithm 2
afiro	iter	305	**91**
	CPU	27.26	**0.5055**
blend	iter	235	**87**
	CPU	39.4938	**1.3347**
sc50a	iter	291	**84**
	CPU	28.7867	**0.9437**
sc50b	iter	297	**84**
	CPU	28.742	**0.9012**
sc105	iter	299	**89**
	CPU	59.3844	**2.3971**
sc205	iter	305	**98**
	CPU	126.4622	**8.5558**

Tab. 6. Total number of iterations for $\theta = 0.3$.

Based on the results presented in Tables 3-6, we can clearly see that

- our algorithm outperformed Algorithm 1 in most cases, achieving the lowest iteration count in over 70% of the conducted experiments.
- the CPU time required by Algorithm 1 is considerably higher than that of our proposed method. In certain cases, Algorithm 1 required up to 53 times more computation time. Although Algorithm 1 might involve fewer iterations (Tables 3 and 4), it is slower because it employs the self-dual embedding model [34] to ensure a strictly feasible starting point, which effectively increased the problem size compared to the original formulation. Another contributing factor is the presence of the expression $e^{\frac{1}{t}}$ in the definition of the KF of Algorithm 1. For values of $t \lesssim 0.0014$, this expression exceeds the numerical range that can be represented in MATLAB, resulting in overflow ([8]). Since such small values of t arise in the vector v during execution for some test problems, Algorithm 1 may require substantially more time to reach the desired solution.

5. CONCLUSIONS AND REMARKS

In this paper, a full-Newton step infeasible IPM for solving monotone LCP problems based on a new hyperbolic KF is proposed. The new KF has a finite value at the boundary of the feasible region. With appropriate choices of the threshold parameter τ and the barrier update parameter θ, we proved that our algorithm is well-defined and has the same complexity as the currently best known infeasible IPAs for monotone LCPs. Finally, the numerical results indicate that the algorithm performs well on a set of LCPs generated from some Netlib problems. However, more numerical testing is necessary to draw more definite conclusions about the behaviour of the algorithm.

ACKNOWLEDGEMENTS

This work has been supported by the general direction of scientific research and technological development (DGRSDT) under project PRFU No. C00L03UN180120220008. Algeria. The authors would like to thank the anonymous reviewers for the valuable comments and suggestions which improved the quality of the paper.

(Received September 29, 2024)

REFERENCES

[1] K. Amini and M. R. Peyghami: An interior-point method for linear programming based on a class of kernel functions. Bull. Austral. Math. Soc. *71* (2005), 139–153. DOI:10.1017/S0004972700038090

[2] Y. Q. Bai, M. El Ghami, and C. Roos: A new efficient large-update primal-dual interior-point method based on a finite barrier. SIAM J. Optim. *13* (2002), 766–782. DOI:10.1137/s1052623401398132

[3] 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), 101–128. DOI:10.1137/S1052623403423114

[4] Y. Bouhenache, W. Chikouche and S. Guerdouh: An interior-point algorithm for LCP based on a parameterized hyperbolic kernel function. Comput. Math. Math. Phys. *65* (2025), 1181–1194. DOI:10.1134/S096554252570054X

[5] Y. Bouhenache, W. Chikouche and I. Touil: A large-update primal-dual interior-point algorithm for convex quadratic optimization based on a new bi-parameterized bi-hyperbolic kernel function. Lobach. J. Math. *45*(3) (2024), 992–1007. DOI:10.1134/S1995080224600560

[6] Y. Bouhenache, W. Chikouche, I. Touil and S. F. Hafshejani: Complexity analysis of primal-dual interior-point methods for convex quadratic programming based on a new twice parameterized kernel function. J. Math. Model. *12*(2) (2024), 247–265. DOI:10.22124/jmm.2024.25394.2257

[7] R. W. Cottle, J.-S. Pang and R. E. Stone: The Linear Complementarity Problem. Academic Press, Boston, MA 1992.

[8] M. El Ghami, I. D. Ivanov and T. Steihaug: Primal-dual interior-point methods solver based on kernel functions for linear optimization. In: Proc. International Multiconference on Computer Science and Information Technology, Mragowo 2009, pp. 743–749. DOI:10.1109/IMCSIT.2009.5352756

[9] M. C. Ferris and J. S. Pang: Engineering and economic applications of complementarity problems. SIAM Rev. *39*(4) (1997), 669–713. DOI:10.1137/S0036144595285130

[10] S. Guerdouh and W. Chikouche: A primal-dual large-update interior-point algorithm for symmetric cone optimization based on a new class of kernel functions. Palest. J. Math. *13*(3) (2024), 320–332.

[11] S. Guerdouh, W. Chikouche and B. Kheirfam: A full-Newton step infeasible interior-point algorithm based on a kernel function with a new barrier term. J. Appl. Math. Comput. *69* (2023), 2935–2953. DOI:10.1007/s12190-023-01858-8

[12] S. Guerdouh, W. Chikouche and I. Touil: An efficient primal-dual interior-point algorithm for linear optimization problems based on a novel parameterized kernel function with a hyperbolic barrier term. Preprint, HAL, 2021. DOI:halshs-03228790

[13] S. Guerdouh, W. Chikouche and I. Touil: A primal-dual interior-point algorithm based on a kernel function with a new barrier term. Stat. Optim. Inf. Comput. *11* (2023), 773–784. DOI:10.19139/soic-2310-5070-1381

[14] S. Guerdouh, W. Chikouche, I. Touil and A. Yassine: Complexity of primal-dual interior-point algorithm for linear programming based on a new class of kernel functions. Kybernetika *59*(6) (2023), 827–860. DOI:10.14736/kyb-2023-6-0827

[15] C. Kanzow: Some non-interior continuation methods for linear complementarity problems. SIAM J. Matrix Anal. Appl. *17*(4) (1996), 851–868. DOI:10.1137/S0895479894273134

[16] N. K. Karmarkar: A new polynomial-time algorithm for linear programming. In: Proc. 16th Annual ACM Symposium on Theory of Computing *4*, 1984, pp. 373–395. DOI:10.1145/800057.808695

[17] B. Kheirfam: Primal-dual interior-point algorithm for semidefinite optimization based on a new kernel function with trigonometric barrier term. Numer. Algorithms *61* (2012), 659–680. DOI:10.1007/s11075-012-9557-y

[18] B. Kheirfam: A full-Newton step infeasible interior-point algorithm for linear complementarity problems based on a kernel function. Algorithmic Oper. Res. *7* (2013), 103–110.

[19] B. Kheirfam and M. Haghighi: An infeasible interior-point method for the P_*-matrix linear complementarity problem based on a trigonometric kernel function with full-Newton step. Commun. Comb. Optim. *3*(1) (2018), 51–70. DOI:10.22049/CCO.2018.25801.1038

[20] B. Kheirfam and M. Haghighi: A full-Newton step infeasible interior-point method for linear optimization based on a trigonometric kernel function. Optimization *65*(4) (2016), 841–857. DOI:10.1080/02331934.2015.1080255

[21] B. Kheirfam and M. Haghighi: A full-Newton step infeasible interior-point method based on a trigonometric kernel function without centering steps. Numer. Algorithms *85* (2020), 59–75. DOI:10.1007/s11075-019-00802-x

[22] B. Kheirfam and M. Moslemi: A polynomial-time algorithm for linear optimization based on a new kernel function with trigonometric barrier term. Yugosl. J. Oper. Res. *25*(2) (2015), 233–250. DOI:10.2298/YJOR120904006K

[23] M. Kojima, N. Megiddo, T. Noma and A. Yoshise: A unified approach to interior-point algorithms for linear complementarity problems. In: Lecture Notes in Computer Science, vol. 538, Springer, Berlin 1991. DOI:10.1007/3-540-54509-3

[24] M. Kojima, S. Mizuno and A. Yoshise: A polynomial-time algorithm for a class of linear complementarity problems. Math. Program. *44* (1989), 1–26. DOI:10.1007/BF01587074

[25] Z. Liu and W. Sun: A full-NT-step infeasible interior-point algorithm for SDP based on kernel functions. Appl. Math. Comput. *217* (2011), 4990–4999. DOI:10.1016/j.amc.2010.11.049

[26] T. De Luca, F. Facchinei and C. Kanzow: A semismooth equation approach to the solution of nonlinear complementarity problems. Math. Program. *75* (1996), 407–439. DOI:10.1007/BF02592192

[27] I. J. Lustig: Feasibility issues in a primal-dual interior-point method for linear programming. Math. Program. *49*(1–3) (1990), 145–162. DOI:10.1007/BF01588785

[28] H. Mansouri and C. Roos: A new full-Newton step infeasible interior-point algorithm for semidefinite optimization. Numer. Algorithms *52*(2) (2009), 225–255. DOI:10.1007/s11075-009-9270-7

[29] H. Mansouri, M. Zangiabadi and M. Pirhaji: A full-Newton step $\mathcal{O}(n)$ infeasible interior-point algorithm for linear complementarity problems. Nonlinear Anal. Real World Appl. *12*(1) (2011), 545–561. DOI:10.1016/j.nonrwa.2010.06.039

[30] M. Moslemi and B. Kheirfam: Complexity analysis of infeasible interior-point method for semidefinite optimization based on a new trigonometric kernel function. Optim. Lett. *13* (2019), 127–145. DOI:10.1007/s11590-018-1257-7

[31] J. Peng, C. Roos, and T. Terlaky: A new class of polynomial primal-dual methods for linear and semidefinite optimization. European J. Oper. Res. *143* (2002), 234–256. DOI:10.1016/S0377-2217(02)00275-8

[32] M. Pirhaji, M. Zangiabadi and H. Mansouri: An infeasible interior-point algorithm for monotone linear complementarity problem based on a specific kernel function. J. Appl. Math. Comput. *54* (2017), 469–483. DOI:10.1007/s12190-016-1019-6

[33] C. Roos: A full-Newton step $\mathcal{O}(n)$ infeasible interior-point algorithm for linear optimization. SIAM J. Optim. *16*(4) (2006), 1110–1136. DOI:10.1137/050623917

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

[35] M. Salahi, M. R. Peyghami and T. Terlaky: New complexity analysis of IIPMs for linear optimization based on a specific self-regular function. Eur. J. Oper. Res. *186*(2) (2008), 466–485. DOI:10.1016/j.ejor.2007.02.008

[36] M. Salahi, T. Terlaky and G. Zhang: The complexity of self-regular proximity based infeasible IPMs. Comput. Optim. Appl. *33* (2006), 157–185. DOI:10.1007/s10589-005-3064-1

[37] I. Touil and W. Chikouche: Primal-dual interior-point methods for semidefinite programming based on a new type of kernel functions. Filomat *34*(12) (2020), 3957–3969. DOI:10.2298/FIL2012957T

[38] I. Touil and W. Chikouche: Novel kernel function with a hyperbolic barrier term to primal-dual interior-point algorithm for SDP problems. Acta Math. Appl. Sin. Engl. Ser. *38* (2022), 44–67. DOI:10.1007/s10255-022-1061-0

[39] I. Touil and W. Chikouche: Polynomial-time algorithm for linear programming based on a kernel function with hyperbolic-logarithmic barrier term. Palest. J. Math. *11*(II) (2022), 127–135.

[40] L. Zhang, Y. Q. Bai and Y. Xu: A full-Newton step infeasible interior-point algorithm for monotone LCP based on a locally kernel function. Numer. Algorithms *61* (2012), 57–81. DOI:10.1007/s11075-011-9530-1

Safa Guerdouh, LMPA Laboratory, Faculty of Exact Sciences and Computer Science, University of Jijel, 18000 Jijel. Algeria.
Current affiliation: Department of Mathematics, Faculty of Sciences, University of Tizi-Ouzou, BP 17, 15000, Tizi-Ouzou, Algeria.
e-mail: guerdouhsafa@gmail.com, safa.guerdouh@ummto.dz

Wided Chikouche, LMPA Laboratory, Faculty of Exact Sciences and Computer Science, University of Jijel, 18000 Jijel. Algeria.
e-mail: wided.chikouche@gmail.com

Behrouz Kheirfam, Department of mathematics, Azarbaijan Shahid Madani University, Tabriz. Iran.
e-mail: b.kheirfam@azaruniv.ac.ir