KYBERNETIKA — VOLUME *59* (2023), NUMBER 6, PAGES 827–860

COMPLEXITY OF PRIMAL-DUAL INTERIOR-POINT ALGORITHM FOR LINEAR PROGRAMMING BASED ON A NEW CLASS OF KERNEL FUNCTIONS

Safa Guerdouh, Wided Chikouche, Imene Touil and Adnan Yassine

In this paper, we first present a polynomial-time primal-dual interior-point method (IPM) for solving linear programming (LP) problems, based on a new kernel function (KF) with a hyperbolic-logarithmic barrier term. To improve the iteration bound, we propose a parameterized version of this function. We show that the complexity result meets the currently best iteration bound for large-update methods by choosing a special value of the parameter. Numerical experiments reveal that the new KFs have better results comparing with the existing KFs including $\log t$ in their barrier term.

To the best of our knowledge, this is the first IPM based on a parameterized hyperbolic-logarithmic KF. Moreover, it contains the first hyperbolic-logarithmic KF (Touil and Chikouche in Filomat 34:3957-3969, 2020) as a special case up to a multiplicative constant, and improves significantly both its theoretical and practical results.

Keywords: linear programming, primal-dual interior-point methods, kernel function, complexity analysis, large and small-update methods

Classification: 90C05, 90C51

1. INTRODUCTION

LP also referred as "linear optimization" is a specific class of mathematical optimization problems. It represents optimization problems that involve linear relationships between the decision variables and the linear objective function, as well as linear constraints. LP has several applications in real life such as production planning, transportation and distribution, resource allocation, portfolio optimization, and many more. Due to the theoretical and practical importance of LPs, various methods were proposed to solve them. IPMs rank among the most efficient methods both theoretically and practically. IPMs were first introduced by Narendra Karmarkar in 1984 [32], the main property of these methods is that they follow the central path in the interior of the feasible region of the problem. The most important results on IPMs for solving LP problems are summarized in the monographs written by Roos et al. [48], Wright [56], Ye [57] and Peng et al. [41].

DOI: 10.14736/kyb-2023-6-0827

Primal-dual IPMs based on KFs represent a fascinating subclass of IPMs. This approach combines two fundamental concepts: primal-dual IPMs and KFs. The choice of the KF plays an important role not only for the IPM analysis but also for the performance of the corresponding interior-point algorithm. Specifically, the KF is used to define an equivalent form of the central path of the IPM, to define the proximity measure, and to obtain search directions. In fact, the central path followed by IPMs is obtained by solving a parametric system that is characterized by a barrier function defined in terms of a KF. In addition, the gradient of the KF serves to define a measure of the distance between the iterates and the central path. Both the KF and the proximity measure effect the iteration bounds since some of their properties play an important role in the complexity analysis. That is why the complexity rate depends on the proposed KF.

The first primal-dual IPM based on the classical logarithmic barrier function was introduced by Roos et al. [48]. After that, Peng et al. introduced primal-dual IPMs based on the so-called self-regular barrier functions, in [39] for LP and semidefinite programming (SDP) and in [40] for second-order conic programming (SOCP). They significantly improved the theoretical complexity obtained for the classical logarithmic KF and obtained the currently best iteration bound for large-update IPMs namely, $\mathcal{O}(\sqrt{n}\log n\log\frac{n}{\epsilon})$, with n denotes the number of variables in the problem and ϵ denotes the desired accuracy in terms of the objective value. This was one of the main motivations for considering other KFs as a substitute for the classical logarithmic KF. In this context, Bai et al. [7] introduced in 2002 an IPM based on an exponential barrier term which has a finite value at the boundary of the feasible region. The growth term of this finite KF was later parametrized by El Ghami et al. [23] and the approach was extended to solve different types of optimization problems including SDP [26], convex quadratic programming (CQP) [15], $P_*(\kappa)$ horizontal linear complementarity problem [8] and Cartesian $P_*(\kappa)-$linear complementarity problem over symmetric cones (SCLCP) [54].

In 2004, Bai et al. [6] proposed a new class of eligible KFs which are not necessarily self-regular. This class includes the classical logarithmic KF and self-regular KFs as special cases. They presented a unified analysis of primal-dual IPMs based on eligible KFs for LP. The results obtained in [6] were successfully extended to SDP, CQP, convex quadratic semidefinite programming (CQSDP), SOCP, convex quadratic programming over symmetric cone (SCCQP), $P_*(\kappa)-$linear complementarity problem (LCP), the Cartesian $P_*(\kappa)-$SCLCP and symmetric programming in [9, 13, 16, 20, 37, 52, 53, 55] respectively.

In [2, 3, 4], published in 2005 and 2007, Amini and his co-authors proposed parametrizations of the two exponential KFs proposed for the first time in [6]. They improved the iteration complexity for large-update methods from $\mathcal{O}(\sqrt{n}\log^2 n\log\frac{n}{\epsilon})$ to $\mathcal{O}(\sqrt{n}\log n\log\frac{n}{\epsilon})$. The IPMs based on these parametric exponential KFs were extended to solve SDP [17, 42], $P_*(\kappa)-$LCP [5, 43] and SOCP [18].

Upon scrutiny of the works cited above that focus on IPMs based on KFs, it can be concluded that the theoretical complexity bounds remain unaffected by the type of problem when switching from one problem type to another using, of course, the same KF.

We cannot talk about KFs without mentioning trigonometric KFs. This type of functions has been extensively explored in the literature [14, 30, 31, 34, 36, 38, 44], starting

with the work of El Ghami et al. [25] where the authors studied an IPM based on the first trigonometric KF introduced in [6]. They established that the complexity bounds for large- and small-update methods are $\mathcal{O}(n^{\frac{3}{4}}\log\frac{n}{\epsilon})$ and $\mathcal{O}(\sqrt{n}\log\frac{n}{\epsilon})$ respectively. This function was later generalized by Bouafia et al. [12] and El Ghami el al. [45] in 2016 for LP and SDP respectively. They obtained $\mathcal{O}(\sqrt{n}\log n\log\frac{n}{\epsilon})$ iterations complexity for large-update methods which is a significant improvement from the results obtained by El Ghami et al. in [25].

Recently, Touil and Chikouche [51] introduced the first IPM based on a hyperbolic-logarithmic KF for SDP. They proved that the corresponding interior-point algorithm meets $\mathcal{O}(n^{\frac{2}{3}}\log\frac{n}{\epsilon})$ iterations as the worst case complexity bound for the large-update method. In another paper [50], they presented an IPM based on a pure hyperbolic barrier term. By parametrizing the latter, Guerdouh et al. in [29] improved the complexity bound for large-update IPMs from $\mathcal{O}(n^{\frac{3}{4}}\log\frac{n}{\epsilon})$ to the currently best one. Remaining within the LP case, they proposed an IPM based on a different KF with an exponential-hyperbolic barrier term. The obtained complexity bound for large-update methods meets $\mathcal{O}(\sqrt{n}\log^2 n\log\frac{n}{\epsilon})$ iterations [28].

Motivated by these works, we first introduce a primal-dual IPM for solving LP problems based on a new hyperbolic-logarithmic KF. Then, we parametrized the new function. The obtained class of KFs contains the hyperbolic-logarithmic KF proposed in [51] as a special case up to a multiplicative constant and improves its theoretical complexity bound to the currently best iteration bound for large-update methods namely, $\mathcal{O}(\sqrt{n}\log n\log\frac{n}{\epsilon})$.

We structure our paper as follows. In Section 2, we briefly recall the basics of IPMs for LP. In Section 3, we present the full complexity analysis for the basic KF followed by some numerical tests on eight different examples to illustrate the effectiveness of the proposed KF comparing to other available KFs. In section 4, we introduce a parametrization of the basic KF and derive its complexity bound with some additional numerical experiments. We finally end the paper by providing some concluding remarks.

Let us finish this introduction with some notations used in the whole paper :
$\|x\| = \sqrt{x^T x}$ denotes the Euclidean norm of a vector $x \in \mathbb{R}^n$. $\mathbb{R}^n_+$ and $\mathbb{R}^n_{++}$ denote the nonnegative and the positive orthants respectively, e denotes the all-one vector. 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 indicate 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.

2. PRELIMINARIES

In this section, we provide a brief description of the idea behind the IPMs based on KFs. We consider the standard form of LP problems

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

where $A \in \mathbb{R}^{m\times n}$, $c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$ are given, $\text{rank}(A) = m \leq n$ and $x \in \mathbb{R}^n$ is the vector of variables. The dual problem of (P) is given by

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

where $y \in \mathbb{R}^m$ and $s \in \mathbb{R}^n$ are the vectors of variables.

Finding an optimal solution of (P) and (D) is equivalent to solving the non-linear system of equations

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

The first and second pair of equations represent primal and dual feasibility. Replacing the last equation in (1), the so-called complementarity condition for (P) and (D), by the parameterized equation $xs = \mu e$, with parameter $\mu > 0$, implies that $x, s > 0$ and leads to the following system

$$\begin{cases} Ax = b, x > 0, \\ A^T y + s = c, s > 0, \\ xs = \mu e. \end{cases} \tag{2}$$

At this stage, we assume that the problems (P) and (D) satisfy the interior-point condition (IPC), i. e., there exists (x^0, y^0, s^0) such that

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

This condition means that the interiors of the feasible regions of problems (P) and (D) are both non-empty. However, if the problem doesn't satisfy the IPC, it can be modified so that it does and even in such a way that $x^0 = s^0 = e$, the details can be found in [48]. Since the IPC holds, system (2) has a unique solution (x_μ, y_μ, s_μ) for each $\mu > 0$, (see [48, Theorem II.4]). The set of unique solutions $\{(x_\mu, y_\mu, s_\mu) : \mu > 0\}$ forms the so-called central path of problems (P) and (D). If $\mu \to 0$, then the limit of the central path exists and yields optimal solutions for (P) and (D).

For fixed $\mu > 0$, we apply Newton's method to the parameterized system (2). Thus, the search direction $(\Delta x, \Delta y, \Delta s)$ is determined by the following system of linear equations

$$\begin{cases} A\Delta x = 0, \\ A^T \Delta y + \Delta s = 0, \\ s\Delta x + x\Delta s = \mu e - xs. \end{cases} \tag{3}$$

By taking a step along the search direction, one constructs a new iterate point

$$x_+ := x + \alpha \Delta x, \;\; y_+ := y + \alpha \Delta y, \;\; s_+ := s + \alpha \Delta s,$$

for some $0 < \alpha \leq 1$ satisfying the positivity condition i. e., $(x_+, s_+) > 0$.
Now, we define the scaled vector v and the scaled search directions d_x and d_s as follows

$$v = \sqrt{\frac{xs}{\mu}}, \quad d_x = \frac{v\Delta x}{x}, \quad d_s = \frac{v\Delta s}{s}. \tag{4}$$

System (3) can be rewritten as follows

$$\begin{cases} \overline{A}d_x = 0, \\ \overline{A}^T\Delta y + d_s = 0, \\ d_x + d_s = v^{-1} - v, \end{cases} \tag{5}$$

where $\overline{A} = \frac{1}{\mu}AV^{-1}X$, $V = \mathrm{diag}(v)$, $X = \mathrm{diag}(x)$.
Observe that the right-hand side in the last equation of (5) is equal to minus gradient of the classical logarithmic scaled barrier (proximity) function

$$\Psi_c(v) = \sum_{i=1}^{n} \psi_c(v_i), \tag{6}$$

with

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

Thus the third equation in (5), often called the scaled centering equation, can be rewritten as follows

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

This indicates that Ψ_c essentially determines the search direction. In addition, it's easy to verify that

$$\nabla^2\Psi_c(v) = \mathrm{diag}(e + v^{-2}),$$

and that the matrix $\mathrm{diag}(e + v^{-2})$ is positive definite. This implies that the Hessian $\nabla^2\Psi_c(v)$ is positive definite which indicates that Ψ_c is strictly convex. Moreover,

$$\nabla\Psi_c(e) = 0.$$

It follows that $\Psi_c(v)$ attains its minimal value at $v = e$, with $\Psi_c(e) = 0$. This means that

$$\nabla\Psi_c(v) = 0 \Leftrightarrow \Psi_c(v) = 0 \Leftrightarrow v = e \Leftrightarrow x = x_\mu \text{ and } s = s_\mu,$$

where x_μ and s_μ are the first and third components of the solution of system (2) respectively. Therefore, (x_μ, s_μ) can be identified as the minimizers of the function Ψ_c. For this reason, Ψ_c serves mainly as a "proximity" measure of closeness for (x, s) with respect to the $\mu-$center. These observations concerning Ψ_c led to the creation of the concept of primal-dual IPMs based on KFs. The basic idea in these methods is to replace ψ_c by any strictly convex function $\psi :]0, +\infty[\to [0, +\infty[$ which is minimal at $t = 1$ with $\psi(1) = 0$. The corresponding proximity function Ψ is then obtained by replacing ψ_c by ψ in (6). This explains the reason for calling ψ the KF of the barrier function Ψ. Thus,

Ψ still serves as a proximity measure for closeness with respect to the $\mu-$center (x_μ, s_μ), and the inequality

$$\Psi(v) \leq \tau,$$

defines a $\tau-$neighbourhood of the $\mu-$center.

In the literature, many KFs were proposed with a barrier term constructed by $\log t$ combined with another type of functions. In Table 1, we can find all these KFs (to our knowledge) and the complexity results for the corresponding algorithms, starting with the classical logarithmic KF ψ_c in [48] and its generalized version $\psi_{1,p}$ proposed by El Ghami et al. in [22].

In this context, we consider in this paper a new univariate KF ψ defined as follows:

$$\psi(t) = \frac{t^2-1}{2} + \tanh^2(1)\big(\coth(t) - \log t\big) - \tanh(1), \ \forall t > 0, \tag{7}$$

and generalize the barrier term of ψ by applying a positive parameter p to obtain the following parametric KF

$$\psi_p(t) = \frac{t^2-1}{2} + \frac{\sinh^2(1)}{\sinh^2(1) + p\coth^{p-1}(1)}\Big(\coth^p(t) - \log t - \coth^p(1)\Big), \quad p \geq 2. \tag{8}$$

Remark 2.1. Taking $p = 1$ in (8) and using the hyperbolic identity

$$\cosh^2(t) - \sinh^2(t) = 1, \ \forall t \in \mathbb{R},$$

we obtain KF (7).

Coming back to system (5), we can convert it to

$$\begin{cases} \overline{A}d_x = 0, \\ \overline{A}^T\Delta y + d_s = 0, \\ d_x + d_s = -\nabla\Psi(v). \end{cases} \tag{9}$$

Since A has full row rank, system (9) has a unique solution. Furthermore, the vectors d_x and d_s are orthogonal and thus

$$d_x^T d_s = 0 \Leftrightarrow \nabla\Psi(v) = 0 \Leftrightarrow v = e \Leftrightarrow \Psi(v) = 0 \Leftrightarrow x = x_\mu \text{ and } s = s_\mu.$$

In the sequel, we will use the norm-based proximity measure $\sigma(v)$ defined by

$$\sigma(v) = \frac{1}{2}\left\|d_x + d_s\right\| = \frac{1}{2}\left\|\nabla\Psi(v)\right\|. \tag{10}$$

Type	Kernel functions	Reference & Type of problem	Complexity
Simple	$\psi_c(t) = \frac{t^2-1}{2} - \log t$ $\psi_{1,p}(t) = \frac{t^{p+1}-1}{p+1} - \log t, \quad 0 < p \leq 1$ $\psi_2(t) = 8t^2 - 11t + \frac{2}{\sqrt{t}} - 4\log t$ $\psi_{3,p}(t) = \frac{t^2-1-\log t}{2} + \frac{t^{1-p}-1}{2(p-1)}, \quad p \geq 2$	[48, LP] [22, LP] [25, LP] [11, LP]	$\mathcal{O}\left(n \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(n \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(n^{\frac{5}{6}} \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(pn^{\frac{p+1}{2p}} \log \frac{n}{\epsilon}\right)$
Exponential	$\psi_{4,p}(t) = \frac{t^2-1-\log t}{2} + \frac{e^{\frac{1}{t^p}-1}-1}{2p}, \quad p \geq 1$ $\psi_{5,p}(t) = t^2 - 1 - \log t + \frac{e^{p(\frac{1}{t}-1)}-1}{p}, \quad p \geq 1$	[19, LP] [10, LP]	$\mathcal{O}\left(p\sqrt{n}(\log n)^{\frac{p+1}{p}} \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(p^{-1}\sqrt{n}(\log n)^2 \log \frac{n}{\epsilon}\right)$
Trigonometric	$\psi_6(t) = \frac{t^2-1}{2} - \log t + \frac{1}{8}\tan^2(\pi \frac{1-t}{4t+2})$ $\psi_{6,\lambda}(t) = \frac{t^2-1}{2} - \log t + \lambda \tan^2(\pi \frac{1-t}{3t+2}), \quad 0 < \lambda \leq \frac{8}{25\pi}$ $\psi_{7,p}(t) = \frac{t^2-1}{2} - \log t - \int_1^t \frac{1}{2p(2+4x)^2} \tan^{2p}(\pi \frac{1-x}{4x+2})dx, \quad p \in \mathbb{N}$	[46, LP] [14, LP] [30, LCP]	$\mathcal{O}\left(n^{\frac{2}{3}} \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(n^{\frac{2}{3}} \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left((1+2k)n^{\frac{2p+1}{4p}} p^{\frac{2p+1}{2p}} \log \frac{n}{\epsilon}\right)$
Hyperbolic	$\psi_8(t) = \frac{1+2\coth(1)}{2\sinh^2(1)}(t^2-1) + \coth^2(t) - \coth^2(1) - \log t$ $\psi_{\text{new}}(t) = \frac{t^2-1}{2} + \tanh^2(1)\Big(\coth(t) - \log t\Big) - \tanh(1)$ $\psi_{\text{new},p}(t) = \frac{t^2-1}{2} + a\Big(\coth^p(t) - \log t - \coth^p(1)\Big),$ with $a = \frac{\sinh^2(1)}{\sinh^2(1) + p\coth^{p-1}(1)}$ and $p \geq 2$	[51, SDP] [new, LP] [new, LP]	$\mathcal{O}\left(n^{\frac{2}{3}} \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(n^{\frac{3}{4}} \log \frac{n}{\epsilon}\right)$ $\mathcal{O}\left(pn^{\frac{p+2}{2(p+1)}} \log \frac{n}{\epsilon}\right)$

Tab. 1. Existing kernel functions with $\log t$ in their barrier terms.

Now, we can describe the algorithm briefly. Given a strictly feasible point (x^0, y^0, s^0) which is in a τ-neighbourhood of the given μ-center, we reduce μ to $\mu_+ := (1-\theta)\mu$ for some fixed $0 < \theta < 1$ and then solve system (9) to obtain the search direction. The positivity condition of a new iterate is ensured by choosing an appropriate step size α. This procedure is repeated until we find a new iterate (x_+, y_+, s_+) that again belongs to the τ-neighbourhood of the current μ-center, that is, until $\Psi(v) \leq \tau$. Then, we update the parameter μ to μ_+ and we let $(x, y, s) = (x_+, y_+, s_+)$. This procedure is repeated until we find an iterate (x_+, y_+, s_+) such that $x_+^T s_+ < \epsilon$. In this case, an ϵ-approximate optimal solution of problems (P) and (D) is found. The generic IPM is outlined in Algorithm 1.

From the above, we can summarize the role of KFs in what follows

- The KF, more specifically the gradient of the barrier function is used to solve system (9) and hence to define new search directions.
- The proximity measure σ, which is used to measure the closeness of the triplet (x, y, s) to the $\mu-$ center, is defined in term of the KF.
- The KF serves to define a $\tau-$neighbourhood of the $\mu-$center and thus it intervenes in the stopping criterion of the inner iteration of the algorithm.

Algorithm 1 : Generic Interior-Point Algorithm for LP

Input
a threshold parameter $\tau \geq 1$;
an accuracy parameter $\epsilon > 0$;
a fixed barrier update parameter $\theta \in]0, 1[$; (x^0, y^0, s^0) satisfy the IPC and $\mu^0 = 1$ such that
$\Phi\left(x^0, s^0; \mu^0\right) := \Psi(v^0) \leq \tau$.
begin
$x := x^0; s := s^0; \mu := \mu^0$;
while $n\mu \geq \epsilon$
begin (outer iteration)
$\mu := (1-\theta)\mu$;
while $\Phi(x, s; \mu) := \Psi(v) > \tau$
begin (inner iteration)
Solve system (9) and use (4) to obtain $(\Delta x, \Delta y, \Delta s)$;
Choose a suitable step size α;
$x := x + \alpha\Delta x; y := y + \alpha\Delta y; s := s + \alpha\Delta s; v := \sqrt{\frac{xs}{\mu}}$;
end while (inner iteration)
end while (outer iteration)

3. THE NEW KERNEL FUNCTION

In this section, we give some properties of our new KF and derive the complexity analysis.

3.1. Some technical results

In the analysis of the algorithm based on ψ, we need its first three derivatives with respect to t which are given for all $t > 0$ by

$$\psi'(t) = t - \tanh^2(1)\left(\frac{1}{\sinh^2(t)} + \frac{1}{t}\right), \tag{11}$$

$$\psi''(t) = 1 + \tanh^2(1)\left(2\frac{\coth(t)}{\sinh^2(t)} + \frac{1}{t^2}\right), \tag{12}$$

and

$$\psi'''(t) = -\tanh^2(1)\left(\frac{2}{\sinh^4(t)} + 4\frac{\coth^2(t)}{\sinh^2(t)} + \frac{2}{t^3}\right) < 0. \tag{13}$$

Clearly, $\psi'(1) = \psi(1) = 0$, $\lim_{t\to 0^+} \psi(t) = \lim_{t\to+\infty} \psi(t) = +\infty$ and $\psi''(t) > 0$, $\forall t > 0$. This implies that ψ is a KF.

Furthermore, the two functions $\psi : [1, +\infty[\to [0, +\infty[$ and $\psi' :]0, 1] \to]-\infty, 0]$ are both monotonically increasing. So, we can define the following functions
$\varrho : [0, +\infty[\longrightarrow [1, +\infty[$ the inverse function of $\psi(t)$ for $t \geq 1$.
$\rho : [0, +\infty[\longrightarrow]0, 1]$ the inverse function of $-\frac{1}{2}\psi'(t)$ for $0 < t \leq 1$.

In what follows, we develop some technical lemmas. From (12), we see that

$$\psi''(t) \geq 1, \ \forall t > 0,$$

thus we have the following lemma.

Lemma 3.1. (Bai at al. [7, Lemma 2.1]) Let ψ be defined as in (7). Then

$$\frac{1}{2}(t-1)^2 \leq \psi(t) \leq \frac{1}{2}(\psi'(t))^2, \quad \forall t > 0.$$

Corollary 3.2. (Bai at al. [7, Corollary 2.2]) Let $\sigma(v)$ be defined by (10). Then, for any $v > 0$, we have

$$\sigma(v) \geq \sqrt{\frac{\Psi(v)}{2}}.$$

Remark 3.3. Throughout the paper, we assume that $\tau \geq 1$. Using Corollary 3.2 and the assumption that $\Psi(v) \geq \tau$, we have

$$\sigma(v) \geq \sqrt{\frac{1}{2}}.$$

Furthermore, we can write $\psi(t)$ as

$$\psi(t) = \frac{t^2-1}{2} + \psi_b(t),$$

where $\dfrac{t^2-1}{2}$ is the growth term and $\psi_b(t)$ is the barrier term of our KF. Moreover, $\psi_b(1) = 0$ and ψ_b is monotonically decreasing. Hence, we can take advantage of the following lemma.

Lemma 3.4. (Bai at al. [6, Lemma 6.2]) One has

$$\sqrt{1+2s} \leq \varrho\,(s) \leq 1 + \sqrt{2s},\ \forall s \in [0, +\infty[.$$

Now, we give an implicit lower bound for $\rho\,(s)$.

Lemma 3.5. For all $(z,t) \in [0,+\infty[\times]0,1]$ such that $z = -\frac{1}{2}\psi'(t)$, one has

$$\coth(t) \leq \coth(1)\,(2z+2)^{\frac{1}{2}}.$$

P r o o f. Let $z \geq 0$ and $t \in]0,1]$ such that $z = -\frac{1}{2}\psi'(t)$, then $\rho\,(z) = t$.
Since $\dfrac{1}{\sinh^2(t)} = \coth^2(t) - 1$, using (11), we have

$$\begin{aligned} 2z &= -\psi'(t) \\ &= -t + \tanh^2(1)\left(\coth^2(t) - 1 + \frac{1}{t}\right), \end{aligned}$$

which implies that

$$\begin{aligned} \coth^2(t) &= \coth^2(1)(2z+t) + 1 - \frac{1}{t} \\ &\leq \coth^2(1)(2z+2). \end{aligned}$$

This proves the lemma. □

Lemma 3.6. (Touil and Chikouche [50, Lemma 1]) One has

$$2t\coth(t) - 1 > 0,\ \forall t > 0. \tag{14}$$

The following lemma reveals some key properties of the new KF.

Lemma 3.7. Let ψ be as defined in (7). Then,

(i) $t\psi''(t) - \psi'(t) > 0,\ \forall t > 0.$

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

(iii) ψ'' is monotonically decreasing on $]0,+\infty[$.

P r o o f . For the first and second item, using (11) and (12), we have

$$t\psi''(t) - \psi'(t) = \tanh^2(1)\left(\frac{2t\coth(t)+1}{\sinh^2(t)} + \frac{2}{t}\right) > 0,$$

and

$$t\psi''(t) + \psi'(t) = 2t + \tanh^2(1)\left(\frac{2t\coth(t)-1}{\sinh^2(t)}\right) > 0,$$

by taking into acount (14) of Lemma 3.6.
For the third item, using (13), we have $\psi'''(t) < 0$, $\forall t > 0$, then $\psi''(t)$ decreases monotonically. This completes the proof. □

Lemma 3.8. (Peng et al. [41, Lemma 2.1.2]) A twice differentiable function $\psi : \mathbb{R}_{++} \to \mathbb{R}$ verifies property (ii) of Lemma 3.7 if and only if it verifies one of the following equivalent properties

- $\psi(\sqrt{t_1 t_2}) \leq \frac{1}{2}\left(\psi(t_1) + \psi(t_2)\right),\ \forall t_1, t_2 > 0.$
- the function $\xi \mapsto \psi(e^{\xi})$ is convex.

We say that ψ is exponentially convex or shortly e-convex.

3.2. Growth behaviour

Before updating μ in the generic interior-point algorithm, we have $\Psi(v) \leq \tau$. After updating μ in an outer iteration, the vector v is divided by the factor $(1-\theta)$, which generally leads to an increase of the value of $\Psi(v)$. Thus, during the inner iterations, the value of $\Psi(v)$ decreases until it passes the threshold τ. We proceed by studying the effect of updating the barrier parameter μ on the value of $\Psi(v)$.

Theorem 3.9. (Bai at al. [6, Theorem 3.2]) For any $v > 0$ and $\beta > 1$, we have

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

Applying this theorem with $\beta = \dfrac{1}{1-\theta}, 0 < \theta < 1$, and taking into account the increasing of ψ on $[1, +\infty[$, we get

Corollary 3.10. By the assumption $\Psi(v) \leq \tau$ just before the $\mu-$update to $(1-\theta)\mu$, we have

$$\Psi(v_+) \leq n\psi\left(\frac{\varrho\left(\dfrac{\tau}{n}\right)}{\sqrt{1-\theta}}\right), \tag{15}$$

where $v_+ = \dfrac{v}{\sqrt{1-\theta}}$.

Lemma 3.11. (Bai at al. [6, Lemma 6.3]) Let θ be such that $0 < \theta < 1$. If $\Psi(v) \leq \tau$, then

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

Corollary 3.12. Let θ be such that $0 < \theta < 1$. If $\Psi(v) \leq \tau$, then

$$\Psi(v_+) \leq \frac{3\coth(1)+1}{(1-\theta)} \left(\theta\sqrt{n} + \sqrt{2\tau}\right)^2 := \Psi_0.$$

Ψ_0 is an upper bound for $\Psi(v_+)$ during the process of the algorithm.

3.3. An estimation of the step size

The purpose of this subsection is to compute a default step size α such that the iterates (x_+, y_+, s_+) defined in Algorithm 1 are feasible and the proximity function decreases sufficiently. Note that during an inner iteration the parameter μ is fixed. Hence, using (4), we have

$$x_+ = \frac{x}{v}(v + \alpha d_x), \quad s_+ = \frac{s}{v}(v + \alpha d_s).$$

It follows that the new v-vector is given by

$$v_+ = \sqrt{\frac{x_+ s_+}{\mu}} = \sqrt{(v + \alpha d_x)(v + \alpha d_s)}.$$

The e-convexity property of ψ implies that

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

Now, we consider the decrease in Ψ as a function of α noted f defined by

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

In the remainder of this section, we put for simplicity $\sigma := \sigma(v)$.
From Lemmas 4.3, 4.4, 4.5 and Theorem 4.6 in [6], we have the following theorem.

Theorem 3.13. (Bai at al. [6]) The largest possible value of the step size α^* is

$$\alpha^* = \frac{\rho(\sigma) - \rho(2\sigma)}{2\sigma}.$$

Furthermore

$$\alpha^* \geq \frac{1}{\psi''(\rho(2\sigma))},$$

and we have for all $\alpha \in [0, \alpha^*]$:

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

Corollary 3.14. Let us set $\bar{\alpha} = \dfrac{1}{\psi''\left(\rho\left(2\sigma\right)\right)}$, as the default step size. Then

$$f(\bar{\alpha}) \leq -\frac{\sigma^2}{\psi''\left(\rho\left(2\sigma\right)\right)}. \tag{16}$$

We can obtain the upper bound for the decreasing value of the proximity in the inner iteration by the following theorem.

Theorem 3.15. If $\bar{\alpha}$ is the default step size and $\sigma \geq 1$, then we have

$$f\left(\bar{\alpha}\right) \leq -\frac{\Psi(v)^{\frac{1}{4}}}{273}. \tag{17}$$

P r o o f. Using (12) and (14) we have

$$\begin{aligned}\psi''(t) \leq & 1 + \tanh^2(1)\left(2\coth(t)(\coth^2(t) - 1) + 4\coth^2(t)\right)\\ & \leq 1 + \tanh^2(1)\left(2\coth^3(t) + 4\coth^2(t)\right)\\ & \leq 1 + 6\tanh^2(1)\coth^3(t).\end{aligned}$$

Hence, putting $t = \rho\left(2\sigma\right)$ and using Lemma 3.5 we get

$$\begin{aligned}\psi''\left(\rho\left(2\sigma\right)\right) & \leq \left(1 + 6\tanh^2(1)\right)\coth^3(\rho\left(2\sigma\right))\\ & \leq \coth^3(1)\left(1 + 6\tanh^2(1)\right)\left(4\sigma + 2\right)^{\frac{3}{2}}\\ & \leq \coth^3(1)\left(1 + 6\tanh^2(1)\right)\left(4\sigma + 2(2\sigma)\right)^{\frac{3}{2}}.\end{aligned}$$

Moreover,

$$f(\bar{\alpha}) \leq -\frac{\sigma^2}{\psi''\left(\rho\left(2\sigma\right)\right)}.$$

This implies that

$$\begin{aligned}f\left(\bar{\alpha}\right) & \leq -\frac{\sigma^2}{\coth^3(1)\left(1 + 6\tanh^2(1)\right)\left(8\sigma\right)^{\frac{3}{2}}}\\ & = -\frac{\sqrt{\sigma}}{8^{\frac{3}{2}}\coth^3(1)\left(1 + 6\tanh^2(1)\right)}.\end{aligned}$$

Using Corollary 3.2, we get

$$\begin{aligned}f\left(\bar{\alpha}\right) & \leq -\frac{\Psi(v)^{\frac{1}{4}}}{2^{\frac{1}{4}}8^{\frac{3}{2}}\coth^3(1)\left(1 + 6\tanh^2(1)\right)}\\ & \leq -\frac{\Psi(v)^{\frac{1}{4}}}{273}.\end{aligned}$$

Hence the theorem is proved. □

3.4. Iteration complexity

We need to compute how many inner iterations are required to return to the situation where $\Psi(v) \leq \tau$ after $\mu-$update. Let us define the value of $\Psi(v)$ after $\mu-$update as Ψ_0, and the subsequent values in the same outer iteration as $\Psi_k, k = 1, \ldots, K$, where K stands for the total number of inner iterations in the outer iteration. By the definition of $f(\alpha)$ and according to (17) , for $k = 1, \ldots, K-1$, we obtain

$$\Psi_{k+1} \leq \Psi_k - \frac{\Psi_k^{\frac{1}{4}}}{273}.$$

Lemma 3.16. (Peng et al. [39, Proposition 2.2]) Suppose $t_0, t_1, \ldots, t_k$ be a sequence of positive numbers such that

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

where $\beta > 0$ and $0 < \gamma \leq 1$. Then

$$K \leq \left\lceil \frac{t_0^{\gamma}}{\beta\gamma} \right\rceil.$$

As a consequence, by taking $t_k = \Psi_k$, $\beta = \dfrac{1}{273}$ and $\gamma = \dfrac{3}{4}$, we get the following lemma.

Lemma 3.17. Let K be the total number of inner iterations in the outer iteration. Then, we have

$$K \leq 364\, \Psi_0^{\frac{3}{4}}.$$

Now we derive the complexity bounds for large and small-update methods.

Theorem 3.18. The total number of iterations to obtain an approximate solution with $n\mu \leq \epsilon$ is bounded by

$$\left(364\, \Psi_0^{\frac{3}{4}} \frac{\log \frac{n}{\epsilon}}{\theta} \right). \tag{18}$$

P r o o f . It is well known that the number of outer iterations for the situation $n\mu \leq \epsilon$ is bounded above by $\frac{1}{\theta}(\log \frac{n}{\epsilon})$ (see [48, Lemma II.17, page 116]).
Knowing that an upper bound for the total number of iterations is obtained by multiplying the number of inner and outer iterations, we obtain the result thanks to the above lemma. □

For large-update IPMs with $\tau = \mathcal{O}(n)$ and $\theta = \Theta(1)$, we have $\mathcal{O}\left(n^{\frac{3}{4}} \log \frac{n}{\epsilon}\right)$ iteration complexity for LP problems based on our new KF.

For small-update IPMs with $\tau = \mathcal{O}(1)$ and $\theta = \Theta\left(\frac{1}{\sqrt{n}}\right)$, we get the currently best known iteration bound, namely $\mathcal{O}\left(\sqrt{n} \log \frac{n}{\epsilon}\right)$ iterations.

3.5. Numerical tests

In this subsection, we carry out numerical experiments of the interior-point algorithm based on different KFs. Our experiments are implemented in MATLAB R2012b and performed on Supermicro dual-2.80 GHz Intel Core i5 server with 4.00 Go RAM.

We have taken $\epsilon = 10^{-8}$ and $\tau = n$. We chose a practical step size α as in [33] i. e., $\alpha = \min(\alpha_x, \alpha_s)$, with

$$\alpha_x = \min_{i=1..n} \begin{cases} -\dfrac{x_i}{\Delta x_i} & \text{if } \Delta x_i < 0, \\ 1 & \text{elsewhere,} \end{cases}$$

and

$$\alpha_s = \min_{i=1..n} \begin{cases} -\dfrac{s_i}{\Delta s_i} & \text{if } \Delta s_i < 0, \\ 1 & \text{elsewhere.} \end{cases}$$

This choice of α guarantees the strict positivity of the new point. Moreover, we increase the step size by a fixed factor $0 < \beta < 1$ (in our case we chose $\beta = 0.9$).

Our purpose is to show that the generic interior-point algorithm based on our hyperbolic-logarithmic KF can be very efficient in solving LP problems. As in [24], we conducted comparative numerical tests between the KFs provided in Table 1 on the following eight test problems with different sizes, ranging from very small to big size problems.

3.5.1. Examples with fixed size

All examples with fixed size are taken from [1].

Example 3.19. $m = 2, n = 4$.

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & -3 \end{pmatrix}, \quad b = (1, 0.5)^T, \quad c = (1, 2, 3, 4)^T,$$

where the initial feasible solutions are defined as follows

$$x^0 = (0.5, 0.27, 0.14, 0.09)^T, \quad y^0 = (0, 0)^T, \quad s^0 = (1, 2, 3, 4)^T.$$

The optimal solution is
$x^* = (0.87500, 0, 0, 0.12500)^T$,
$y^* = (1.75000, -0.75000)^T$
$s^* = (0.00000, 1.00000, 1.25000, 0.00000)^T$.

Example 3.20. $m = 3$, $n = 5$.

$$A = \begin{pmatrix} 2 & 1 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad b = (8, 7, 3)^T, \quad c = (-4, -5, 0, 0, 0)^T,$$

where the initial feasible solutions are defined as follows
$x^0 = (2.85, 1.9, 0.4, 0.35, 1.1)^T, y^0 = (-1.2, -1.8, -0.5)^T, s^0 = (0.2, 0.3, 1.2, 1.8, 0.5)^T$.
The optimal solution is

$x^* = (3.0000, 2.0000, 0.0000, 0.0000, 1.0000)^T$,
$y^* = (-1.0000, -2.0000, -0.0000)^T$,
$s^* = (0.0000, 0.0000, 1.0000, 2.0000, 0.0000)^T$.

Example 3.21. $m = 3, n = 6$.

$$A = \begin{pmatrix} 2 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \quad b = (0, 0, 1)^T, \quad c = (3, -1, 1, 0, 0, 0)^T,$$

where the initial feasible solutions are defined as follows
$x^0 = (0.06757, 0.13258, 0.13302, 0.26774, 0.13302, 0.2664)^T$, $y^0 = (-2, -2, -3)^T$,
$s^0 = (10, 4, 6, 1, 5, 1)^T$.
The optimal solution is
$x^* = (0.0000, 0.5000, 0.0000, 0.5000, 0.0000, 0.0004)^T$,
$y^* = (-0.5000, -0.5000, -0.5000)^T$,
$s^* = (4.5000, 0.0000, 2.0000, 0.0000, 1.0000, 0.0000)^T$.

Example 3.22. $m = 5, n = 9$.

$$A = \begin{pmatrix} 0 & 1 & 2 & -1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 3 & 4 & -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & -2 & 1 & 2 & 0 & 0 & 1 & 0 \\ 1 & 2 & 0 & -1 & -2 & 0 & 0 & 0 & 1 \\ 1 & 3 & 4 & 2 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 2 \\ 1 \end{pmatrix}, \quad c = (1, 0, -2, 1, 1, 0, 0, 00)^T,$$

where the initial feasible solutions are defined as follows
$x^0 = (0.1819, 0.0699, 0.063, 0.1105, 0.2012, 0.6732, 1.1885, 2.835, 2.1912)^T$,
$y^0 = (-1.3843, -0.8751, -0.4241, -0.4463, -3.0424)^T$,
$s^0 = (4.9398, 13.1544, 14.7156, 9.1788, 4.5072, 1.3843, 0.8751, 0.4241, 0.4463)^T$.
The optimal solution is
$x^* = (0.0000, 0.0000, 0.2664, 0.0000, 0.0000, 0.4269, 1.1406, 3.5729, 2.0000)^T$,
$y^* = (-0.0000, -0.0000, -0.0000, -0.0000, -0.4999)^T$,
$s^* = (1.5000, 1.4999, 0.0000, 1.9999, 1.4999, 0.0000, 0.0000, 0.0000, 0.0000)^T$.

3.5.2. Examples with variable size

Example 3.23. (Bouafia et al. [12]) The matrix A is defined as

$$A(i, j) = \begin{cases} 1 & \text{if } i = j \text{ or } j = i + m, \\ 0 & \text{otherwise,} \end{cases}$$

$c(i) = -1$, $c(i + m) = 0$ and $b(i) = 2$, for $i = 1, \ldots, m$.
We start by an initial point (x^0, y^0, s^0) such that
$x^0 = e$, $y^0(i) = -2$ and $s^0(i) = 1$, $s^0(i + m) = 2$, for $i = 1, \ldots, m$.
The optimal solution is
$x^*(i) = 2$, $x^*(i + m) = 0$, $y^*(i) = -1$, $s^*(i) = 0$, $s^*(i + m) = 1$, for $i = 1, \ldots, m$.

Example 3.24. (Anane [1, Example 5]) The matrix A is defined as

$$A(i,j) = \begin{cases} 1 & \text{if } i = j \text{ or } j = i + m, \\ 0 & \text{otherwise,} \end{cases}$$

$c = -e$ and $b(i) = 2$, $i = 1, \ldots, m$.
We start by an initial point (x^0, y^0, s^0) such that
$x^0 = s^0 = e$ and $y^0(i) = -2$, for $i = 1, \ldots, m$.
The optimal solution is
$x^* = e$, $s^* = 0$, and $y^*(i) = -1$, for $i = 1, \ldots, m$.

Example 3.25. (Touil et al. [49, Example 4]) The matrix A is defined as

$$A(i,j) = \begin{cases} 1 & \text{if } i = j \text{ or } j = i + m, \\ 0 & \text{otherwise,} \end{cases}$$

$c = -e$ and $b(i) = 2$, $i = 1, \ldots, m$.
We start by an initial point (x^0, y^0, s^0) such that
$x^0(i) = 1.5$, $x^0(i+m) = 0.5$, $y^0(i) = -2$, for $i = 1, \ldots, m$, and $s^0 = e$.
The optimal solution is
$x^*(i) = 1.4793$, $x^*(i+m) = 0.5207$, $y^*(i) = -1$, for $i = 1, \ldots, m$, and $s^* = 0$.

Example 3.26. The matrix A is defined as

$$A(i,j) = \begin{cases} 1 & \text{if } i = j \text{ or } j = i + m, \\ 0 & \text{otherwise,} \end{cases}$$

$c(i) = \frac{1}{4}$, $c(i+m) = \frac{3}{4}$ and $b(i) = 2$, $i = 1, \ldots, m$.
We start by an initial point (x^0, y^0, s^0) such that
$x^0(i) = \frac{5}{4}$, $x^0(i+m) = \frac{3}{4}$, $y^0(i) = \frac{-7}{12}$, and $s^0(i) = \frac{5}{6}$, $s^0(i+m) = \frac{4}{3}$, for $i = 1, \ldots, m$.
The optimal solution is
$x^*(i) = 2$, $x^*(i+m) = 0$, $y^*(i) = \frac{1}{4}$, and $s^*(i) = 0$, $s^*(i+m) = \frac{1}{2}$, for $i = 1, \ldots, m$.

Problems with fixed size were tested for multiple values of θ, $\theta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. The table corresponding to fixed size problems (Table 2) shows that, in most cases, larger θ gives better iteration numbers. Thus, for variable size problems, we only choose $\theta \in \{0.7, 0.9, 0.99\}$. This reduces the number of experiments since we perform Algorithm 1 with the considered KFs on four test problems for seven different sizes $n = 2m$ where $m \in \{5, 25, 50, 100, 200, 400, 1000\}$. This left us with 104 experiments for each KF. The summary of results is given in tables below.

Ex	θ	ψ_c	$\psi_{1,\frac{1}{2}}$	ψ_2	$\psi_{3,2}$	$\psi_{3,\frac{\log n}{2}}$	$\psi_{4,1}$	$\psi_{4,2}$	$\psi_{5,2}$	$\psi_{5,\log n}$	ψ_6	$\psi_{6,0.05}$	$\psi_{7,2}$	$\psi_{7,[\log n]}$	ψ_8	ψ_{new}
Example 1	0.1	**188**	**188**	–	**188**	**188**	**188**	**188**	190	**188**	**188**	**188**	**188**	**188**	200	**188**
	0.3	**56**	**56**	104	**56**	**56**	**56**	**56**	60	**56**	**56**	**56**	**56**	**56**	70	**56**
	0.5	**29**	38	47	**29**	**29**	**29**	32	31	**29**	**29**	**29**	**29**	**29**	44	**29**
	0.7	**17**	41	36	**17**	**17**	**17**	**17**	**17**	**17**	**17**	**17**	**17**	**17**	23	**17**
	0.9	**11**	57	18	**11**	33	**11**	13	13	12	**11**	**11**	**11**	**11**	14	**11**
Example 2	0.1	**191**	**191**	243	**191**	**191**	**191**	**191**	215	**191**	**191**	**191**	**191**	**191**	215	**191**
	0.3	**57**	**57**	63	**57**	**57**	**57**	**57**	75	**57**	**57**	**57**	**57**	**57**	75	**57**
	0.5	**29**	36	31	**29**	**29**	**29**	**29**	35	33	**29**	**29**	**29**	**29**	39	**29**
	0.7	**17**	36	19	**17**	**17**	**17**	**17**	25	21	**17**	**17**	**17**	**17**	21	**17**
	0.9	11	45	12	10	11	10	10	**9**	**9**	11	11	11	11	**9**	10
Example 3	0.1	**192**	**192**	–	**192**	**192**	197	197	206	195	**192**	**192**	**192**	**192**	204	**192**
	0.3	**57**	**57**	61	**57**	**57**	**57**	62	69	62	**57**	**57**	**57**	**57**	66	**57**
	0.5	**30**	41	32	**30**	**30**	**30**	**30**	37	**30**	**30**	**30**	**30**	**30**	33	**30**
	0.7	**17**	44	**17**	18	**17**	20	20	23	21	**17**	**17**	**17**	**17**	20	18
	0.9	33	73	34	28	33	**19**	21	22	22	39	26	42	–	25	21
Example 4	0.1	**196**	**196**	240	**196**	**196**	**196**	**196**	220	214	**196**	**196**	**196**	**196**	212	**196**
	0.3	**58**	**58**	112	**58**	**58**	**58**	**58**	66	74	**58**	**58**	**58**	**58**	78	**58**
	0.5	31	38	34	**30**	31	33	**30**	36	36	31	31	31	31	42	33
	0.7	28	45	22	**21**	28	**21**	24	28	28	22	22	22	22	24	**21**
	0.9	24	67	41	21	24	24	24	22	23	23	23	23	23	**20**	21

Tab. 2. Number of inner iterations for fixed size examples.

θ	m	ψ_c	$\psi_{1,\frac{1}{2}}$	ψ_2	$\psi_{3,2}$	$\psi_{3,\frac{\log n}{2}}$	$\psi_{4,1}$	$\psi_{4,2}$	$\psi_{5,2}$	$\psi_{5,\log n}$	ψ_6	$\psi_{6,0.05}$	$\psi_{7,2}$	$\psi_{7,[\log n]}$	ψ_8	ψ_{new}
$\theta = 0.7$	5	**18**	44	–	**18**	**18**	**18**	21	26	32	**18**	**18**	**18**	**18**	35	**18**
	25	**19**	46	–	**19**	**19**	**19**	22	27	36	**19**	**19**	**19**	**19**	36	**19**
	50	**20**	49	–	**20**	**20**	**20**	23	28	39	**20**	**20**	**20**	**20**	37	**20**
	100	**20**	49	–	**20**	**20**	**20**	23	28	46	**20**	**20**	**20**	**20**	37	**20**
	200	**21**	52	–	**21**	**21**	**21**	24	31	62	**21**	**21**	**21**	**21**	40	**21**
	400	**21**	52	–	**21**	**21**	**21**	24	31	49	**21**	**21**	**21**	**21**	40	**21**
	1000	**22**	54	–	**22**	**22**	**22**	25	34	77	**22**	**22**	**22**	**22**	41	**22**
$\theta = 0.9$	5	21	58	–	14	25	13	23	31	26	16	16	15	15	**12**	14
	25	23	63	–	15	15	14	24	32	–	17	17	17	20	**13**	15
	50	23	63	–	15	15	14	25	32	–	17	17	17	22	**13**	15
	100	24	69	–	16	16	16	25	33	–	19	18	18	17	**14**	17
	200	24	69	–	16	15	16	25	33	–	19	18	18	17	**14**	17
	400	24	69	–	16	16	16	25	33	–	19	18	18	18	**14**	17
	1000	25	69	–	18	17	17	26	34	–	20	20	19	20	**15**	18
$\theta = 0.99$	5	21	89	20	29	21	39	40	**15**	32	28	30	37	37	21	28
	25	21	89	20	29	30	39	40	**15**	41	28	30	37	45	21	28
	50	24	104	22	31	29	41	42	**17**	42	31	33	40	57	23	30
	100	24	104	22	31	27	41	42	**17**	42	31	33	40	57	23	30
	200	24	104	22	31	30	41	42	**17**	46	31	33	40	57	23	30
	400	24	104	22	31	24	41	42	**17**	–	31	33	40	55	23	30
	1000	24	104	22	31	31	41	42	**17**	–	31	33	40	54	23	30

Tab. 3. Number of inner iterations for Example 5 with different sizes $n = 2m$.

θ	m	ψ_c	$\psi_{1,\frac{1}{2}}$	ψ_2	$\psi_{3,2}$	$\psi_{3,\frac{\log n}{2}}$	$\psi_{4,1}$	$\psi_{4,2}$	$\psi_{5,2}$	$\psi_{5,\log n}$	ψ_6	$\psi_{6,0.05}$	$\psi_{7,2}$	$\psi_{7,[\log n]}$	ψ_8	ψ_{new}
$\theta = 0.7$	5	**18**	39	54	**18**	**18**	**18**	**18**	34	30	**18**	**18**	**18**	**18**	28	**18**
	25	**19**	41	–	**19**	**19**	**19**	**19**	37	28	**19**	**19**	**19**	**19**	29	**19**
	50	**20**	44	–	**20**	**20**	**20**	**20**	40	45	**20**	**20**	**20**	**20**	32	**20**
	100	**20**	44	–	**20**	**20**	**20**	**20**	40	32	**20**	**20**	**20**	**20**	32	**20**
	200	**21**	46	–	**21**	**21**	**21**	**21**	41	–	**21**	**21**	**21**	**21**	33	**21**
	400	**21**	46	–	**21**	**21**	**21**	**21**	41	–	**21**	**21**	**21**	**21**	33	**21**
	1000	**22**	48	–	**22**	**22**	**22**	**22**	42	–	**22**	**22**	**22**	**22**	34	**22**
$\theta = 0.9$	5	11	40	**9**	10	11	10	10	**9**	**9**	11	11	11	11	**9**	10
	25	12	45	**10**	11	11	11	11	**10**	**10**	12	12	12	12	**10**	11
	50	12	45	**10**	11	11	11	11	**10**	**10**	12	12	12	12	**10**	11
	100	13	51	**11**	12	12	12	12	**11**	**11**	13	13	13	13	**11**	12
	200	13	51	**11**	12	12	12	12	**11**	**11**	13	13	13	13	**11**	12
	400	13	51	**11**	12	12	12	12	**11**	**11**	13	13	13	13	**11**	12
	1000	15	55	**12**	14	13	13	13	**12**	**12**	14	14	15	15	**12**	14
$\theta = 0.99$	5	11	60	**10**	**10**	11	**10**	**10**	**10**	**10**	11	11	11	11	**10**	**10**
	25	11	60	**10**	**10**	**10**	**10**	**10**	**10**	**10**	11	11	11	11	**10**	**10**
	50	13	73	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	100	13	73	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	200	13	73	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	400	13	73	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	1000	13	73	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13

Tab. 4. Number of inner iterations for Example 6 with different sizes $n = 2m$.

θ	m	ψ_c	$\psi_{1,\frac{1}{2}}$	ψ_2	$\psi_{3,2}$	$\psi_{3,\frac{\log n}{2}}$	$\psi_{4,1}$	$\psi_{4,2}$	$\psi_{5,2}$	$\psi_{5,\log n}$	ψ_6	$\psi_{6,0.05}$	$\psi_{7,2}$	$\psi_{7,[\log n]}$	ψ_8	ψ_{new}
$\theta = 0.7$	5	**18**	42	–	**18**	**18**	**18**	**18**	–	–	**18**	**18**	**18**	**18**	–	**18**
	25	**19**	44	–	**19**	**19**	**19**	**19**	–	–	**19**	**19**	**19**	**19**	–	**19**
	50	**20**	47	–	**20**	**20**	**20**	**20**	–	–	**20**	**20**	**20**	**20**	–	**20**
	100	**20**	47	–	**20**	**20**	**20**	**20**	–	–	**20**	**20**	**20**	**20**	–	**20**
	200	**21**	49	–	**21**	**21**	**21**	**21**	–	–	**21**	**21**	**21**	**21**	–	**21**
	400	**21**	49	–	**21**	**21**	**21**	**21**	–	–	**21**	**21**	**21**	**21**	–	**21**
	1000	**22**	52	–	**22**	**22**	**22**	**22**	–	–	**22**	**22**	**22**	**22**	–	**22**
$\theta = 0.9$	5	11	44	**9**	10	11	10	10	**9**	**9**	11	11	11	11	**9**	10
	25	12	49	**10**	11	11	11	11	**10**	**10**	12	12	12	12	**10**	11
	50	12	49	**10**	11	11	11	11	**10**	**10**	12	12	12	12	**10**	11
	100	13	54	**11**	12	12	12	12	**11**	**11**	13	13	13	13	**11**	12
	200	13	54	**11**	12	12	12	12	**11**	**11**	13	13	13	13	**11**	12
	400	13	54	**11**	12	12	12	12	**11**	**11**	13	13	13	13	**11**	12
	1000	15	60	**12**	14	13	13	13	**12**	14	15	15	15	15	**12**	14
$\theta = 0.99$	5	11	64	**10**	**10**	11	**10**	**10**	**10**	**10**	11	11	11	11	**10**	**10**
	25	11	64	**10**	**10**	**10**	**10**	**10**	**10**	**10**	11	11	11	11	**10**	**10**
	50	13	80	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	100	13	80	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	200	13	80	**12**	13	13	13	13	**12**	**12**	13	13	13	13	**12**	13
	400	13	80	**12**	13	13	13	13	**12**	14	13	13	13	13	**12**	13
	1000	13	80	**12**	13	13	13	13	**12**	14	13	13	13	13	**12**	13

Tab. 5. Number of inner iterations for Example 7 with different sizes $n = 2m$.

θ	m	ψ_c	$\psi_{1,\frac{1}{2}}$	ψ_2	$\psi_{3,2}$	$\psi_{3,\frac{\log n}{2}}$	$\psi_{4,1}$	$\psi_{4,2}$	$\psi_{5,2}$	$\psi_{5,\log n}$	ψ_6	$\psi_{6,0.05}$	$\psi_{7,2}$	$\psi_{7,[\log n]}$	ψ_8	ψ_{new}
$\theta = 0.7$	5	**18**	42	50	**18**	**18**	**18**	**18**	27	29	**18**	**18**	**18**	**18**	28	**18**
	25	**19**	45	53	**19**	**19**	**19**	**19**	28	31	**19**	**19**	**19**	**19**	31	**19**
	50	**20**	47	56	**20**	**20**	**20**	**20**	29	52	**20**	**20**	**20**	**20**	32	**20**
	100	**20**	47	56	**20**	**20**	**20**	**20**	29	59	**20**	**20**	**20**	**20**	32	**20**
	200	**21**	50	59	**21**	**21**	**21**	**21**	30	–	**21**	**21**	**21**	**21**	33	**21**
	400	**21**	50	59	**21**	**21**	**21**	**21**	30	–	**21**	**21**	**21**	**21**	33	**21**
	1000	**22**	52	62	**22**	**22**	**22**	**22**	31	–	**22**	**22**	**22**	**22**	36	**22**
$\theta = 0.9$	5	**12**	56	32	**12**	**12**	13	16	16	16	**12**	**12**	**12**	**12**	**12**	**12**
	25	14	61	33	**13**	**13**	14	17	17	49	14	14	14	14	**13**	**13**
	50	14	61	33	**13**	**13**	14	17	17	25	14	14	14	15	**13**	**13**
	100	15	67	34	**14**	**14**	15	19	18	50	15	15	15	16	**14**	**14**
	200	15	67	34	**14**	**14**	15	19	18	92	15	15	15	16	**14**	**14**
	400	15	67	34	**14**	**14**	15	19	18	32	15	15	15	16	**14**	**14**
	1000	16	72	35	**15**	16	16	20	19	29	16	16	16	18	**15**	**15**
$\theta = 0.99$	5	21	82	23	26	22	28	36	28	33	24	24	27	27	**19**	23
	25	21	82	23	26	26	28	36	28	31	24	24	27	32	**19**	23
	50	23	96	25	28	26	30	38	30	36	26	26	29	40	**21**	26
	100	23	96	25	28	22	30	38	30	100	26	26	29	40	**21**	26
	200	23	96	25	28	23	30	38	30	38	26	26	29	40	**21**	26
	400	23	96	25	28	24	30	38	30	45	26	26	29	33	**21**	26
	1000	23	96	25	28	28	30	38	30	43	26	26	29	34	**21**	26

Tab. 6. Number of inner iterations for Example 8 with different sizes $n = 2m$.

3.5.3. Comments

Recall that the numerical results were obtained by performing Algorithm 1 with the KFs defined in Table 1 on eight test problems.
When there are parameters $p \geq 1$ involved in the definition of a KF, we used two values of these parameters: the parameter that gives the best theoretical iteration bound and a common value $p = 2$, except for $\psi_{1,p}$ and $\psi_{6,\lambda}$ where we chose $p = \frac{1}{2}$ and $\lambda = 0.05$ since the parameters p and λ take their values in the intervals $]0, 1]$ and $]0, \frac{8}{25\pi}]$ respectively. This left us with 15 different KFs.

For each example, we used **bold** font to highlight the best, i. e., the smallest, iteration number.

From Tables 2 – 6, we may draw a few conclusions:

- For $\psi_{1,p}$ (the classical logarithmic barrier function ψ_c occurs if $p = 1$), although the theoretical iteration bound of the algorithm is independent of the parameter p, numerical tests show that p influences the iteration count. In both types of problems (fixed or variable size), $p = 1$ gives better results which is in accordance with the analysis carried out by [22].

- The function $\psi_{1,\frac{1}{2}}$ never gives the smallest iteration number in examples with variable size, even for examples with fixed size it gives the smallest iteration number only for the values 0.1 and 0.3 of θ.

- For KFs $\psi_2, \psi_{5,2}, \psi_{5,\log n}, \psi_{7,[\log n]}$ and ψ_8, the dashes in the corresponding columns of Tables 2 – 6 indicate that the algorithms require more than 10^4 iterations to obtain an optimal solution. Despite this, $\psi_2, \psi_{5,2}, \psi_{5,\log n}$ and ψ_8, (and exclusively ψ_8) are the only ones to give the smallest iteration number for $\theta = 0.9$ in Examples 6 and 7 (in Example 5), while for $\theta = 0.99$ in Example 5 (resp. in Example 8), $\psi_{5,2}$ (resp. ψ_8) is the only function to achieve the best iteration number.

- The iteration numbers of the algorithm based on our KF ψ_{new} depend on the values of the parameter θ. In fact, the value 0.9 of θ gives better iteration numbers in general.

- In all examples with variable size for $\theta = 0.7$, the KF ψ_{new} has the smallest iterations number with the KFs $\psi_c, \psi_{3,p}, \psi_{4,p}, \psi_6, \psi_{6,0.05}, \psi_{7,2}, \psi_{7,[\log n]}$ except $\psi_{4,2}$ in Example 5.

- For our KF, the obtained iteration numbers coincided with, or at worst was close to the best ones with a slackness of at most 4 iterations, except the case $\theta = 0.99$ in Example 5, where the slackness attain thirty.

To confirm the superiority of our algorithm in terms of the total number of iterations, we compute, for each KF the percentage of cases where the KF gives the best iteration number. As an illustration, we plot a histogram that we use as a statistical tool to compare the performance of the algorithms.

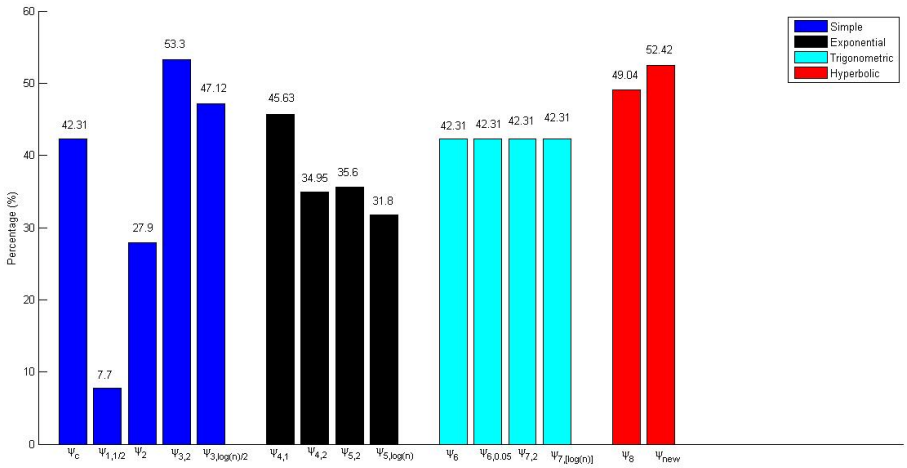

Fig. 1. Performance comparison between KFs in Table 1.

Supported by the performance bar graph, We may conclude some remarks:

- For $\psi_{4,p}$, the parameter which gives the best theoretical complexity bound has better percentage than $p = 2$. But the concordance between the theoretical and the numerical results is not always satisfied, as we can see in the case of $\psi_{3,p}$, $\psi_{5,p}$ and $\psi_{7,p}$.
- We can easily see that the algorithms based on hyperbolic KFs attain the most wins, on average, among the considered algorithms based on other types.
- Although the theoretical complexity obtained for ψ_8 is better than the one of ψ_{new}, but numerical tests reveal that ψ_{new} has better percentage. In fact, a thorough analysis shows that in the examples with fixed size, it's ψ_{new} that gives the smallest iteration number with a slackness which can amount up to 24 iterations. As for examples with variable size for $\theta = 0.7$, ψ_{new} performs better than ψ_8 in Examples 5, 6 and 8 while in Examples 7, ψ_8 doesn't have the ability to complete the run successfully. In contrast, for the values 0.9 and 0.99 of θ, ψ_8 meets or exceeds ψ_{new} with a slackness of no more than 7 iterations.

The numerical effectiveness of ψ_{new} comparing with all KFs with logarithmic barrier term motivates us to propose a generalization of this function in order to achieve a better theoretical complexity.

4. PARAMETRIZED VERSION

In the previous section, we performed a well-detailed complexity analysis for ψ_{new}. Thus, we discuss more briefly the analysis for the generalized KF $\psi_{\text{new},p}$.

$$\psi_{\text{new},p}(t) = \frac{t^2 - 1}{2} + \psi_{b,p}(t), \tag{19}$$

with $\psi_{b,p}(t) = a\Big(\coth^p(t) - \log t - \coth^p(1)\Big)$, $a = \dfrac{\sinh^2(1)}{\sinh^2(1) + p\coth^{p-1}(1)}$ and $p \geq 2$.

Remark 4.1. This function can also be considered as a generalization, up to the multiplicative constant $1/a$, of the KF ψ_8 introduced in [51] (see Table 1).

Using simple calculations, we can easily prove that $\psi_{\text{new},p}$ is indeed a KF and we have:

$\psi'_{\text{new},p}(t) = t - a\Big(\dfrac{p\coth^{p-1}(t)}{\sinh^2(t)} + \dfrac{1}{t}\Big)$
$\psi''_{\text{new},p}(t) = 1 + a\Big(\dfrac{p(p-1)\coth^{p-2}(t)}{\sinh^4(t)} + \dfrac{2p\coth^p(t)}{\sinh^2(t)} + \dfrac{1}{t^2}\Big) > 1$
$\psi'''_{\text{new},p}(t) = -a\Big(\dfrac{p(p-1)(p-2)\coth^{p-3}(t)}{\sinh^6(t)} + \dfrac{(6p^2-4p)\coth^{p-1}(t)}{\sinh^4(t)} + \dfrac{4p\coth^{p+1}(t)}{\sinh^2(t)} + \dfrac{2}{t^3}\Big) < 0$
$t\psi''_{\text{new},p} - \psi'_{\text{new},p} = a\Big(\dfrac{p(p-1)t\coth^{p-2}(t)}{\sinh^4(t)} + \dfrac{2pt\coth^p(t)}{\sinh^2(t)} + \dfrac{2}{t} + \dfrac{p\coth^{p-1}(t)}{\sinh^2(t)}\Big)$
$t\psi''_{\text{new},p} + \psi'_{\text{new},p} = 2t + a\Big(\dfrac{p(p-1)t\coth^{p-2}(t)}{\sinh^4(t)} + \dfrac{p\coth^{p-1}(t)}{\sinh^2(t)}\big(2t\coth(t) - 1\big)\Big)$

Tab. 7. The first three derivatives of $\psi_{\text{new},p}$ with $t\psi''_{\text{new},p} \pm \psi'_{\text{new},p}$.

Thus, $\psi_{\text{new},p}$ satisfies all the results of Lemma 3.1, Corollary 3.2, Lemma 3.4 and Lemma 3.7. Whereas, the equivalent of Lemma 3.5 for $\psi_{\text{new},p}$ is

Lemma 4.2. For all $(z,t) \in [0,+\infty) \times (0,1]$ such that $z = -\frac{1}{2}\psi'_{\text{new},p}(t)$, we have

$$\coth(t) \leq \Big(\big(\sinh^2(1) + \coth^{p-1}(1)\big)\coth^2(1)\Big)^{\frac{1}{p+1}}(2z+1)^{\frac{1}{p+1}}.$$

4.1. Complexity analysis

The complexity analysis for the parameterized KF $\psi_{\text{new},p}$ proceeds in the same way as in the previous section for ψ_{new}. To avoid repetition, we do not present all the details of the computations. We only present the outcome of each step of our computational scheme in Table 8.

$f(\bar{\alpha}) \leq -\dfrac{\sqrt{2}\Psi(v)^{\frac{p}{2(p+1)}}}{72\left(1 + 2\sinh^2(1)\coth^5(1)(p+5)\right)}$	$\Psi(v_+) \leq \dfrac{\coth(1)(p+2)}{(1-\theta)}\left(\theta\sqrt{n} + \sqrt{2\tau}\right)^2 := \Psi_0$
$\beta = \dfrac{\sqrt{2}}{72\left(1 + 2\sinh^2(1)\coth^5(1)(p+5)\right)}$ and $\gamma = \dfrac{p+2}{2(p+1)}$	$K \leq \left\lceil \dfrac{72\sqrt{2}(5+4p)(p+1)}{(p+2)} \right\rceil \Psi_0^{\frac{p+2}{2(p+1)}}$

Tab. 8. Outcomes of the computational scheme.

For small-update method with $\tau = \mathcal{O}(1)$ and $\theta = \Theta\left(\frac{1}{\sqrt{n}}\right)$, the complexity of the primal-dual interior-point algorithm for LP problems based on the new parametric KF

is $\mathcal{O}(p^2\sqrt{n}\log\frac{n}{\epsilon})$ iterations complexity.

As for large-update method i. e., $\tau = \mathcal{O}(n)$ and $\theta = \Theta(1)$, the substitution of these values into

$$K \leq \left[\frac{72\sqrt{2}(5+4p)(p+1)}{(p+2)}\right]\Psi_0^{\frac{p+2}{2(p+1)}}$$

does not give the best possible bound. A better bound is obtained using the following lemma

Lemma 4.3. (El Ghami et al. [21]) Let $0 \leq \theta < 1$, $v_+ = \frac{v}{\sqrt{1-\theta}}$. If $\Psi(v) \leq \tau$, then we have

$$\Psi(v_+) \leq \frac{\theta n + 2\tau + 2\sqrt{2\tau n}}{2(1-\theta)} := \Psi_0.$$

Ψ_0 is an upper bound for $\Psi(v_+)$ during the process of the algorithm.

P r o o f. For simplicity, we denote

$$\psi(t) := \psi_{\text{new},p}(t).$$

We can easily verify that $\psi_{b,p}(1) = 0$, $\psi'_{b,p}(1) = -1$, and $\psi_{b,p}$ is monotonically decreasing on $\mathbb{R}_{++}$. Hence, we have

$$\psi(t) \leq \frac{t^2-1}{2}, \ \forall t \geq 1.$$

Putting $t = \dfrac{\varrho(\frac{\tau}{n})}{\sqrt{1-\theta}} \geq 1$ and using (15), we get

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

Moreover, using Lemma 3.4, we obtain

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

□

Using the upper bound defined in Lemma 4.3, we get $\mathcal{O}\left((p+1)n^{\frac{p+2}{2(p+1)}}\log\frac{n}{\epsilon}\right)$ iterations complexity for large-update methods. This expression is minimal at $p = \dfrac{\log n}{2} - 1$ and then is equal to $\mathcal{O}\left(\sqrt{n}\log n\log\frac{n}{\epsilon}\right)$.

4.2. Numerical tests

Now, we would like to investigate the influence of parameterizing the KF (7) on the computational behavior of the generic primal-dual algorithm for LP presented in Algorithm 1. For this purpose, we present some numerical simulations for implementing the algorithm based on the new parametric KF (8) on the aforementioned set of problems with the same values of the parameters θ, μ and τ and for different values of the parameter p. We also compare with the KF that gave the best performance in the previous section $\psi_{3,2}$ and with ψ_8 which is of the same hyperbolic-logarithmic type. The obtained results are listed in tables below.

Ex	θ	ψ_{new}	$\psi_{\text{new},2}$	$\psi_{\text{new},3}$	$\psi_{\text{new},4}$	$\psi_{\text{new},\frac{\log n}{2}-1}$	$\psi_{3,2}$	ψ_8
Example 1	0.1	**188**	**188**	**188**	**188**	**188**	**188**	200
	0.3	**56**	**56**	**56**	**56**	**56**	**56**	70
	0.5	**29**	**29**	**29**	**29**	**29**	**29**	44
	0.7	**17**	**17**	**17**	**17**	**17**	**17**	23
	0.9	11	11	**10**	12	24	11	14
Example 2	0.1	**191**	**191**	**191**	**191**	**191**	**191**	215
	0.3	**57**	**57**	**57**	**57**	**57**	**57**	75
	0.5	**29**	**29**	**29**	**29**	**29**	**29**	39
	0.7	**17**	**17**	**17**	**17**	**17**	**17**	21
	0.9	10	**9**	**9**	**9**	11	10	**9**
Example 3	0.1	**192**	**192**	**192**	**192**	**192**	**192**	204
	0.3	**57**	**57**	**57**	**57**	**57**	**57**	66
	0.5	**30**	**30**	**30**	**30**	**30**	**30**	33
	0.7	18	19	18	19	**17**	18	20
	0.9	21	**19**	**19**	22	33	28	25
Example 4	0.1	**196**	**196**	**196**	**196**	**196**	**196**	212
	0.3	**58**	**58**	**58**	**58**	**58**	**58**	78
	0.5	33	**30**	**30**	**30**	31	**30**	42
	0.7	21	**20**	22	24	28	21	24
	0.9	21	**18**	22	20	24	21	20

Tab. 9. Number of inner iterations for fixed size examples.

θ	m	Example 5							Example 6						
		ψ_{new}	$\psi_{\text{new},2}$	$\psi_{\text{new},3}$	$\psi_{\text{new},4}$	$\psi_{\text{new},\frac{\log n}{2}-1}$	$\psi_{3,2}$	ψ_8	ψ_{new}	$\psi_{\text{new},2}$	$\psi_{\text{new},3}$	$\psi_{\text{new},4}$	$\psi_{\text{new},\frac{\log n}{2}-1}$	$\psi_{3,2}$	ψ_8
$\theta = 0.7$	5	**18**	**18**	21	22	**18**	**18**	35	**18**	**18**	**18**	**18**	**18**	**18**	28
	25	**19**	**19**	22	23	**19**	**19**	36	**19**	**19**	**19**	**19**	**19**	**19**	29
	50	**20**	**20**	23	24	**20**	**20**	37	**20**	**20**	**20**	**20**	**20**	**20**	32
	100	**20**	**20**	23	24	**20**	**20**	37	**20**	**20**	**20**	**20**	**20**	**20**	32
	200	**21**	**21**	24	25	**21**	**21**	40	**21**	**21**	**21**	**21**	**21**	**21**	33
	400	**21**	**21**	24	25	**21**	**21**	40	**21**	**21**	**21**	**21**	**21**	**21**	33
	1000	**22**	**22**	25	26	**22**	**22**	41	**22**	**22**	**22**	**22**	**22**	**22**	34
$\theta = 0.9$	5	14	14	16	16	22	14	**12**	10	**9**	**9**	**9**	11	10	**9**
	25	15	15	17	17	15	15	**13**	11	11	**10**	**10**	11	11	**10**
	50	15	16	17	17	16	15	**13**	11	11	**10**	**10**	11	11	**10**
	100	17	16	18	18	15	16	**14**	12	12	**11**	**11**	12	12	**11**
	200	17	16	18	18	16	16	**14**	12	12	**11**	**11**	12	12	**11**
	400	17	16	18	18	15	16	**14**	12	12	**11**	**11**	12	12	**11**
	1000	18	17	19	19	16	18	**15**	14	13	**12**	**12**	13	14	**12**
$\theta = 0.99$	5	28	24	29	30	**21**	29	**21**	**10**	**10**	**10**	**10**	11	**10**	**10**
	25	28	24	29	30	28	29	**21**	**10**	**10**	**10**	**10**	**10**	**10**	**10**
	50	30	27	31	32	38	31	**23**	13	**12**	**12**	**12**	**12**	13	**12**
	100	30	27	31	32	31	31	**23**	13	**12**	**12**	**12**	**12**	13	**12**
	200	30	27	31	32	27	31	**23**	13	**12**	**12**	**12**	**12**	13	**12**
	400	30	27	31	32	25	31	**23**	13	**12**	**12**	**12**	**12**	13	**12**
	1000	30	27	31	32	31	31	**23**	13	**12**	**12**	**12**	**12**	13	**12**

Tab. 10. Number of inner iterations for Example 5 and 6 with different sizes $n = 2m$.

θ	m	Example 7							Example 8						
		ψ_{new}	$\psi_{\text{new},2}$	$\psi_{\text{new},3}$	$\psi_{\text{new},4}$	$\psi_{\text{new},\frac{\log n}{2}-1}$	$\psi_{3,2}$	ψ_8	ψ_{new}	$\psi_{\text{new},2}$	$\psi_{\text{new},3}$	$\psi_{\text{new},4}$	$\psi_{\text{new},\frac{\log n}{2}-1}$	$\psi_{3,2}$	ψ_8
$\theta = 0.7$	5	**18**	**18**	**18**	**18**	**18**	**18**	–	**18**	**18**	**18**	**18**	**18**	**18**	28
	25	**19**	**19**	**19**	**19**	**19**	**19**	–	**19**	**19**	**19**	**19**	**19**	**19**	31
	50	**20**	**20**	**20**	**20**	**20**	**20**	–	**20**	**20**	**20**	**20**	**20**	**20**	32
	100	**20**	**20**	**20**	**20**	**20**	**20**	–	**20**	**20**	**20**	**20**	**20**	**20**	32
	200	**21**	**21**	**21**	**21**	**21**	**21**	–	**21**	**21**	**21**	**21**	**21**	**21**	33
	400	**21**	**21**	**21**	**21**	**21**	**21**	–	**21**	**21**	**21**	**21**	**21**	**21**	33
	1000	**22**	**22**	**22**	**22**	**22**	**22**	–	**22**	**22**	**22**	**22**	**22**	**22**	36
$\theta = 0.9$	5	10	**9**	**9**	**9**	11	10	**9**	**12**	**12**	**12**	**12**	**12**	**12**	**12**
	25	11	11	**10**	**10**	11	11	**10**	**13**	**13**	**13**	**13**	**13**	**13**	**13**
	50	11	11	**10**	**10**	11	11	**10**	**13**	**13**	**13**	**13**	**13**	**13**	**13**
	100	12	12	**11**	**11**	12	12	**11**	**14**	**14**	**14**	**14**	**14**	**14**	**14**
	200	12	12	**11**	**11**	12	12	**11**	**14**	**14**	**14**	**14**	**14**	**14**	**14**
	400	12	12	**11**	**11**	12	12	**11**	**14**	**14**	15	15	15	**14**	**14**
	1000	14	13	**12**	**12**	13	14	**12**	**15**	**15**	16	16	16	**15**	**15**
$\theta = 0.99$	5	**10**	**10**	**10**	**10**	11	**10**	**10**	23	21	24	22	21	26	**19**
	25	**10**	**10**	**10**	**10**	**10**	**10**	**10**	23	21	24	22	26	26	**19**
	50	13	**12**	**12**	**12**	**12**	13	**12**	26	23	26	24	27	28	**21**
	100	13	**12**	**12**	**12**	**12**	13	**12**	26	23	26	24	25	28	**21**
	200	13	**12**	**12**	**12**	**12**	13	**12**	26	23	26	24	23	28	**21**
	400	13	**12**	**12**	**12**	**12**	13	**12**	26	23	26	24	24	28	**21**
	1000	13	**12**	**12**	**12**	**12**	13	**12**	26	23	26	24	82	28	**21**

Tab. 11. Number of inner iterations for Example 7 and 8 with different sizes $n = 2m$.

ψ_i	ψ_{new}	$\psi_{\text{new},2}$	$\psi_{\text{new},3}$	$\psi_{\text{new},4}$	$\psi_{\text{new},\frac{\log n}{2}-1}$	$\psi_{3,2}$	ψ_8
%	50.48	66.99	68.93	66.99	58.25	51.45	55.33

Tab. 12. The percentage of cases where the KF gives the best iteration number.

4.2.1. Comments

By comparing the results in Tables 9 – 12, we notice that:

- The number of iterations clearly depends on the value of the parameter p as the gap between two different values of p can amount up to 59. It should also be noted that $\psi_{\text{new},3}$ significantly reduces the number of iterations although $\psi_{\text{new},\frac{\log n}{2}-1}$ has the best complexity bound theoretically.

- Comparing with ψ_{new}, the KFs $\psi_{\text{new},2}, \psi_{\text{new},3}, \psi_{\text{new},4}$ and $\psi_{\text{new},\frac{\log n}{2}-1}$ were able to produce even better iteration numbers especially in Examples 6 and 7 for $\theta = 0.9$ and $\theta = 0.99$, while maintaining similar performance for $\theta = 0.7$ in Examples 6, 7 and 8.

- $\psi_{\text{new},p}$, for all tested parameters, outperformed $\psi_{3,2}$ which had the best performance in the previous section. This confirms that the parametrization has effected the number of iterations of the algorithm positively.

- $\psi_{\text{new},2}$ far outperformed ψ_8 despite having the same complexity bounds and more than, the same expression up to a multiplicative constant (see Remark 4.1).

5. CONCLUSIONS AND REMARKS

In this paper, we investigate an interior-point algorithm for solving LP problems based on a new KF which is a combination of the classic KF and a hyperbolic barrier term. The complexity analysis for large-update primal-dual IPMs based on this KF yields an $\mathcal{O}\left(n^{\frac{3}{4}} \log \frac{n}{\epsilon}\right)$ iteration bound which improves the classical iteration bound with a factor $n^{\frac{1}{4}}$. For small-update methods, the proposed algorithm enjoys the favourable iteration bound. To achieve the best iteration complexity, we introduce a class of KFs by parametrizing the new hyperbolic-logarithmic KF. The simulation results illustrate that the proposed algorithm is efficient and robust.

As a further research we would like to extend our approach to more general optimization problems. Another question of interest is whether the new KF can be used to design an efficient infeasible interior-point algorithm as in [27, 35, 47].

ACKNOWLEDGEMENTS

This work has been supported by the general direction of scientific research and technological development (DGRSDT) under project PRFU No. C00L03UN180120220008. Algeria.

(Received June 5, 2023)

REFERENCES

[1] N. Anane: Méthodes de points intérieurs pour la programmation linéaire basées sur les fonctions noyaux. (Magister Thesis), Ferhat Abbas University, Setif 2012.

[2] K. Amini and K. A. Haseli: A new proximity function generating the best known iteration bounds for both large-update methods and small-update. ANZIAM J. *49* (2007), 259–270. DOI:10.1017/S1446181100012827

[3] K. Amini and M. R. Peyghami: An interior-point algorithm for linear optimization based on a new kernel function. Southeast Asian Bull. Math. *29* (2005), 651–667.

[4] 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

[5] 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(2)* (2009), 501–513. DOI:10.1016/j.amc.2008.11.002

[6] 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

[7] 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

[8] Y. Q. Bai and G. Q. Wang: Polynomial interior-point algorithms for $P_*(\kappa)$ horizontal linear complementarity problem. J. Comput. Appl. Math. *233* (2009), 248–263. DOI:10.1016/j.cam.2009.07.014

[9] Y. Q. Bai, G. Q. Wang and C. Roos: Primal-dual interior-point algorithms for second-order cone optimization based on kernel functions. Nonlinear Anal. *70* (2009), 10, 3584–3602. DOI:10.1016/j.na.2008.07.016

[10] A. Benhadid and K. Saoudi: A new parameterized logarithmic kernel function for linear optimization with a double barrier term yielding the best known iteration bound. Commun. Math. *28* (2020), 27–41. DOI:10.2478/cm-2020-0003

[11] M. Bouafia, D. Benterki, and A. Yassine: An efficient parameterized logarithmic kernel function for linear optimization. Optim. Lett. *12* (2018), 1079–1097. DOI:10.1007/s11590-017-1170-5

[12] M. Bouafia, D. Benterki, and A. Yassine: An efficient primal-dual interior-point method for linear programming problems based on a new kernel function with a trigonometric barrier term. J. Optim. Theory Appl. *170* (2016), 528–545. DOI:10.1007/s10957-016-0895-0

[13] X. Z. Cai, L. Li, M. El Ghami, T. Steihaug, and G. Q. Wang: A new parametric kernel function yielding the best known iteration bounds of interior-point methods for the Cartesian $P_*(\kappa)-$LCP over symmetric cones. Pac. J. Optim. *13* (2017), 4, 547–570.

[14] X. Z. Cai, G. Q. Wang, M. El Ghami, and Y. J. Yue: Complexity analysis of primal-dual interior-point methods for linear optimization based on a new parametric kernel function with a trigonometric barrier term. Abstr. Appl. Anal. (2014) Article ID 710158. DOI:10.1155/2014/710158

[15] X. Z. Cai, G. Q. Wang, and Z. H. Zhang: Complexity analysis and numerical implementation of primal-dual interior-point methods for convex quadratic optimization based on a finite barrier. Numer. Algorithms *62* (2013), 289–306. DOI:10.1007/s11075-012-9581-y

[16] X. Z. Cai, L. Wu, Y. J. Yue, M. M. L, and G. Q. Wang: Kernel-function based primal dual interior- point methods for convex quadratic optimization over symmetric cone. J. Inequal. Appl. (2014), 308, 22 pp.

[17] B. K. Choi and G. M. Lee: On complexity analysis of the primal-dual interior-point method for semidefinite optimization problem based on a new proximity function. Nonlinear Anal. *71* (2009), 2628–2640. DOI:10.1016/j.na.2009.05.078

[18] B. K Choi and G. M. Lee: On complexity analysis of the primal-dual interior-point method for second-order cone optimization problem. J. Korean Soc. Ind. Appl. Math. *14* (2010), 2, 93–111.

[19] L. Derbal and Z. Kebbiche: Theoretical and numerical result for linear optimization problem based on a new kernel function. J. Sib. Fed. Univ. Math. Phys. *12* (2019), 2, 160–172. DOI:10.17516/1997-1397-2019-12-2-160-172

[20] M. El Ghami, Y. Q. Bai, and C. Roos: Kernel-function based algorithms for semidefinite optimization. RAIRO Oper. Res. *43* (2009), 189–199. DOI:10.1051/ro/2009011

[21] M. El Ghami, Z. A. Guennoun, S. Bouali, and T. Steihaug: Interior-point methods for linear optimization based on a kernel function with a trigonometric barrier term. J. Comput. Appl. Math. *236* (2012), 3613–3623. DOI:10.1016/j.cam.2011.05.036

[22] M. El Ghami, I. D. Ivanov, C. Roos, and T. Steihaug: A polynomial-time algorithm for LO based on generalized logarithmic barrier functions. Int. J. Appl. Math. *21* (2008), 99-115.

[23] M.El. Ghami, I. D. Ivanov, J.B.M. Melissen, C. Roos and T. Steihaug: A polynomial-time algorithm for linear optimization based on a new class of kernel functions. J. Comput. Appl. Math. *224* (2009), 2, 500–513. DOI:10.1016/j.cam.2008.05.027

[24] 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

[25] M. El Ghami and C. Roos: Generic primal-dual interior-point methods based on a new kernel function. RAIRO Oper. Res. *42* (2008), 199–213.DOI:10.1051/ro:2008009

[26] M. El. Ghami, C. Roos, and T. Steihaug: A generic primal–dual interior-point method for semidefinite optimization based on a new class of kernel functions. Optim. Methods Softw. *25* (2010), 387–403. DOI:10.1080/10556780903239048

[27] 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

[28] 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

[29] 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. 2021. halshs-03228790

[30] S. F. Hafshejani, M. Fatemi, and M. R. Peyghami: An interior-point method for $P_*(\kappa)-$linear complementarity problem based on a trigonometric kernel function. J. Appl. Math. Comput. *48* (2015), 111–128. DOI:10.1007/s12190-014-0794-1

[31] S. F. Hafshejani and A. F. Jahromi: An interior-point method for $P_*(\kappa)-$horizontal linear complementarity problem based on a new proximity function. J. Appl. Math. Comput. *62* (2020), 281–300. DOI:10.1007/s12190-019-01284-9

[32] 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.

[33] A. Keraghel: Étude adaptative et comparative des principales variantes dans l'algorithme de karmarkar. Ph.D. Thesis, Grenoble Institute of Technology, 1989.

[34] 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

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

[36] 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* (2015), 2, 233–250. DOI:10.2298/YJOR120904006K

[37] G. Lesaja and C. Roos: Unified analysis of kernel-based interior-point methods for $P_*(\kappa)-$linear complementarity problems. SIAM J. Optim. *20* (2010), 6, 3014–3039. DOI:10.1137/090766735

[38] X. Li and M. W. Zhang: Interior-point algorithm for linear optimization based on a new trigonometric kernel function. Oper. Res. Lett. *43* (2015), 471–475. DOI:10.1016/j.orl.2015.06.013

[39] 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

[40] J. Peng, C. Roos, and T. Terlaky: Primal-dual interior-point methods for second order conic optimization based on self-regular proximities. SIAM J. Optim. *13* (2002), 1, 179–203.DOI:10.1137/S1052623401383236

[41] J. Peng, C. Roos, and T. Terlaky: Self-Regularity: A New Paradigm for Primal-Dual interior-point Algorithms. Princeton University Press, Princeton, New Jersey 2002.

[42] M. R. Peyghami: An interior-point approach for semidefinite optimization using new proximity functions. Asia-Pac. J. Oper. Res. *26* (2009), 3, 365–382. DOI:10.1142/S0217595909002250

[43] M. R. Peyghami and K. Amini: A kernel function based interior-point methods for solving $P_*(\kappa)-$linear complementarity problem. Acta Math. Appl. Sin. Engl. Ser. *26* (2010), 1761–1778. DOI:10.1007/s10114-010-7529-5

[44] M. R. Peyghami and S. F. Hafshejani: Complexity analysis of an interior-point algorithm for linear optimization based on a new proximity function. Numer. Algorithms. *67* (2014), 33–48. DOI:10.1007/s11075-013-9772-1

[45] M. R. Peyghami, S. F. Hafshejani, and S. Chen: A primal-dual interior-point method for semidefinite optimization based on a class of trigonometric barrier functions. Oper. Res. Lett. *44* (2016), 3, 319–323. DOI:10.1016/j.orl.2016.02.013

[46] M. R. Peyghami, S. F. Hafshejani, and L. Shirvani: Complexity of interior-point methods for linear optimization based on a new trigonometric kernel function. J. Comput. Appl. Math. *255* (2014), 74–85. DOI:10.1016/j.cam.2013.04.039

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

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

[49] I. Touil, D. Benterki, and A. Yassine: A feasible primal-dual interior-point method for linear semidefinite programming. J. Comput. Appl. Math. *312* (2017), 216–230. DOI:10.1016/j.cam.2016.05.008

[50] 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

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

[52] M. V. C. Vieira: Interior-point methods based on kernel functions for symmetric optimization. Optim. Methods Softw. *27* (2012), 3, 513–537. DOI:10.1007/s10852-005-3561-310.1007/s10852-005-3561-3

[53] G. Q. Wang, Y. Q. Bai, Y. Liu, and M. Zhang: A new primal-dual interior-point algorithm for convex quadratic optimization. J. Shanghai Univ. (English Edition), *12* (2008), 3, 189–196. DOI:10.1007/s11741-008-0301-3

[54] G. Q. Wang, M. M. Li, Y. J. Yue, and X. Z. Cai: New complexity analysis of interior-point methods for the Cartesian $P_*(\kappa)-$SCLCP. J. Inequal. Appl. (2013) Article number 285.

[55] G. Q. Wang and D. T. Zhu: A unified kernel function approach to primal-dual interior-point algorithms for convex quadratic SDO. Numer. Algorithms *57* (2011), 4, 537–558. DOI:10.1080/10556788.2010.544877

[56] S. J. Wright: Primal-dual interior-point methods. SIAM, 1997. DOI:10.1137/1.9781611971453

[57] Y. Y. Ye: Interior-point algorithms. Theory and Analysis. Wiley–Interscience Series in Discrete Mathematics and Optimization. John Wiley and Sons, New York 1997.

Safa Guerdouh, LMPA Laboratory, Faculty of Exact Sciences and Computer Science, University of Jijel, 18000 Jijel. Algeria.
e-mail: guerdouhsafa@gmail.com, safa.guerdouh@univ-jijel.dz

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

Imene Touil, LMPA Laboratory, Faculty of Exact Sciences and Computer Science, University of Jijel, 18000 Jijel. Algeria.
e-mail: i_touil@yahoo.fr

Adnan Yassine, LMAH – ULH, FR CNRS 3335, Normandie University, 25 rue Philippe Lebon, 76600 Le Havre. France.
e-mail: adnan.yassine@univ-lehavre.fr