ADV MATH SCI JOURNAL

Advances in Mathematics: Scientific Journal **11** (2022), no.5, 449–467
ISSN: 1857-8365 (printed); 1857-8438 (electronic)
https://doi.org/10.37418/amsj.11.5.3

A PRIMAL-DUAL *IPMS* FOR *SDO* PROBLEM BASED ON A NEW KERNEL FUNCTION WITH A LOGARITHMIC BARRIER TERM

Abderrahim Guemmaz, El Amir Djeffal[1], and Bachir Bounibane

ABSTRACT. In this paper, we consider primal-dual Interior Point Method $(IPMs)$ for semidefinite optimization problem (SDO) problems, based on a new kernel function with a logarithmic barrier term, which play an important role for generating a new design of primaldual (IPM) algorithms. New search directions and proximity functions are proposed, based on this kernel function.We proved that our algorithm has $\mathbf{O}\left(qsn^{\frac{sq+1}{2sq}}\log\left(\frac{n}{\epsilon}\right)\right)$ iteration bound for large-update methods and $\mathbf{O}\left(q^2s^2\sqrt{n}\log\left(\frac{n}{\epsilon}\right)\right)$ iteration bound for small-update methods. Finally, for its numerical tests, some strategies are used and indicate that the algorithm is efficient.

1. INTRODUCTION

The first paper dealing with SDO problems dates back to the early 1960s (Bellman and Fan, 1963). For the next many years, the whole topic of SDO stayed silent except for a few isolated results scattered in the literature. The situation changed dramatically around the beginning of the 1990s when SDO started to emerge as one of the fastest developing areas of mathematical programming.

[1]*corresponding author*

2020 *Mathematics Subject Classification.* 90C22, 90C51.
Key words and phrases. Complexity analysis; Iteration bound; Kernel function; Primal-dual interior point methods; Semidefinite optimization problems.
Submitted: 19.04.2022; *Accepted:* 06.05.2022; *Published:* 12.05.2022.

Since then it has become one of the most active research areas in mathematical programming (See [10]). The efficient new algorithms, interior-point methods $(IPMs)$ have led to increased interest both in the application and the research of SDO. In this paper we deal with so-called primal-dual $IPMs$. It is generally agreed that these $IPMs$ are most efficient from a computational point of view [11]. In 1984, Karmarkar [13] proposed a polynomial-time algorithm the so-called $IPMs$ for solving linear optimization (LO) problems. This method is extended to SDO, which an important contribution in this field was made by Nesterov and Todd [5, 14]. For a comprehensive study, the reader is referred to [3, 7, 12].

Many researchers have designed different types of primal-dual interior-point methods $(IPMs)$. Among them, $IPMs$ based on kernel functions have been designed. Several kernel functions have been introduced including the so-called self-regular kernel functions [2, 8] and the nonself-regular kernel functions [2, 15]. In principle, a kernel function gives rise to a search direction and hence to a primal-dual interior point method. In this paper, we consider the new kernel function with a logarithmic Barrier Term as follows

$$\psi_s(t) = \frac{(t^2-1)}{2} - \frac{\log(t)}{2} - \frac{1}{2s}\sum_{j=1}^{s}\frac{t^{1-jq}-1}{1-jq}, \quad q>1, s\in\mathbb{N}\backslash\{0\}. \tag{1.1}$$

We will formulate an interior-point methods for SDO by using a new proximity function and give its complexity analysis, and then we show that the iteration bounds are $\mathbf{O}\left(qsn^{\frac{sq+1}{2Sq}}\log\left(\frac{n}{\epsilon}\right)\right)$ and $\mathbf{O}\left(q^2s^2\sqrt{n}\log\left(\frac{n}{\epsilon}\right)\right)$ for large and small-update methods, respectively. The remainder of this paper is organized as follows. First in Sect.(1), define the central path and the classical NT search direction and the new search direction determined by Kernel Functions for SDO, then we present the generic primal-dual IPM algorithm. The new kernel function and its properties and study the matrix function $\Phi_S(V)$ are presented in Sect.(3). In Sect.(4), we analyse the algorithm and derive the complexity bound for SDO. Some numerical results are provided in Sect.(5). Finally, some concluding remarks follow in Sect.(6).

2. Statement of the problem

We consider the semidefinite optimization problem (SDO) in its primal format:

$$\text{(P)} \qquad \min \{C \bullet X : \ A_i \bullet X = b_i, \ 1 \leq i \leq m, \ X \succeq 0\},$$

along with its dual problem as

$$\text{(D)} \qquad \max \left\{ b^T y : \ \sum_{i=1}^{m} y_i A_i + S = C, \ S \succeq 0 \right\},$$

where C, $A_i \in \mathbf{S}^n$, $1 \leq i \leq m$, $b = (b_1, b_2, \ldots, b_m) \in \mathbb{R}^m$, $y \in \mathbf{S}^m$. Here $\mathbf{S}^n$ denotes the space of $n \times n$ real symmetric matrices. In addition $X \succeq 0$ indicates that X is a symmetric postive semidefinite and the operator $\bullet$ denotes the standard inner product in $\mathbf{S}^n$, i.e., $C \bullet X = T_r(CX) = \sum_{i=1}^{n} \sum_{i=1}^{n} C_{ij} X_{ij}$, where T_r denotes the trace. Throughout the paper, we assume that the A_i are linearly independent.

Now, we recall the notion of the central path with its properties and we derive the classical Nestrov-Todd search direction for SDO.

Throughout the paper, we assume that (P) and (D) satisfy the interior point condition (IPC), i.e., there exists a strictly feasible pair $(X^0, \ y^0, \ S^0)$ such that

$$A_i \bullet X^0 = b_i, i = 1, 2, \ldots, m, \quad \sum_{i=1}^{m} y_i^0 A_i + S^0 = C, \quad X_0 \succ, \quad S^0 \succ 0.$$

If the (IPC) holds, then the Karush-Kuhn-Tucker (KKT) optimality conditions for both problems (P) and (D) can be expressed as follows

$$\text{(2.1)} \qquad \begin{cases} A_i \bullet X = b_i, \quad 1 \leq i \leq m, \ \ X \succeq 0, \\ \sum\limits_{i=1}^{m} y_i A_i + S = C, \quad S \succeq 0, \\ XS = 0. \end{cases}$$

The basic idea of primal-dual $IPMs$ is to replace the last equation in system (2.1), which is so called complementarity condition by the parameterized equation $XS =$

μI with $X \succ 0$, $S \succ 0$, and $\mu > 0$. This leads us to the following system:

$$\text{(2.2)} \qquad \begin{cases} A_i \bullet X = b_i, 1 \leq i \leq m, \quad X \succeq 0, \\ \sum\limits_{i=1}^{m} y_i A_i + S = C; \quad S \succeq 0, \\ XS = \mu I. \end{cases}$$

The IPC implies that system (2.2) has a unique solution $(X(\mu), y(\mu), S(\mu))$, for each $\mu > 0$. We call $X(\mu)$ the μ-center of (P) and $(y(\mu), S(\mu))$ is known as the μ-center of (D). As μ goes to zero [15, 16], $(X(\mu), y(\mu), S(\mu))$ converges to the optimal solution of the problem (P) and (D). We call the set $\{(X(\mu), y(\mu), S(\mu))|\mu > 0\}$ the central path of the problems (P) and (D). Now to obtain the search direction for SDO, we apply Newton's method to the system (2.2) for a given strictly feasible primal-dual point (X, y, S), which yields the following linear system of equations:

$$\text{(2.3)} \qquad \begin{cases} A_i \bullet \Delta X = b_i, \ 1 \leq i \leq m, \ X \succeq 0, \\ \sum\limits_{i=1}^{m} \Delta_{y_i} A_i + \Delta S = 0, \ S \succeq 0, \\ X\Delta S + \Delta X S = \mu I - XS. \end{cases}$$

The system (2.3) can be rewritten as

$$\text{(2.4)} \qquad \begin{cases} A_i \bullet \Delta X = b_i, \quad 1 \leq i \leq m; \quad X \succeq 0, \\ \sum\limits_{i=1}^{m} \Delta_{y_i} A_i + \Delta S = 0, \quad S \succeq 0, \\ X\Delta S S^{-1} + \Delta X = \mu S^{-1} - X. \end{cases}$$

System (2.4) has a unique solution [16], in which ΔX is not necessarily symmetric. because $\Delta X \ SS^{-1}$ may be not symmetric. Many researchers have proposed methods for symmetrizing the third equation in the Newton system (2.3) such that the resulting new system has a unique symmetric solution.

2.1. **A new search directions.** In order to provide the scaled Newton system has a unique symmetric solution, Zhang [17] introduced the following symmetrization operator

$$H_P(M) = \frac{1}{2}\left(PMP^{-1} + \left(PMP^{-1}\right)^T\right), \ \forall M \in \mathbb{R}^{n \times n}.$$

One can easily verify that

$$H_P(M) = \mu I,$$

for, any symetric matrix M, and where the scaling matrix P determines the symmetrisation strategy. For any given nonsingular matrix P, the system (2.2) is equivalent to

$$\left\{\begin{array}{l} A_i \bullet X = b_i,\ 1 \leq i \leq m,\ X \succeq 0, \\ \sum\limits_{i=1}^{m} y_i A_i + S = C,\ S \succeq 0, \\ H_P\left(XS\right) = \mu I. \end{array}\right. \tag{2.5}$$

Applying Newton's method to the system (2.5), we obtain the Newton system as follows

$$\left\{\begin{array}{l} A_i \bullet \Delta X = 0,\ 1 \leq i \leq m, \\ \sum\limits_{i=1}^{m} \Delta y_i A_i + \Delta S = 0, \\ H_P\left(X\Delta S + \Delta X S\right) = \mu I - H_P\left(XS\right). \end{array}\right. \tag{2.6}$$

The search direction obtained through the system (2.6) is called the Monteiro-Zang (MZ) unified direction. Different choices of the matrix P result in different search directions (see. eg., [3, 6, 12]). In this paper, we use the NT symmetrization scheme [6], from which the NT search direction is derived. Let us define the matrix.

$$P := X^{\frac{1}{2}}(X^{\frac{1}{2}}SX^{\frac{1}{2}})^{-\frac{1}{2}}X^{\frac{1}{2}} = S^{-\frac{1}{2}}\left(S^{\frac{1}{2}}XS^{\frac{1}{2}})\right)^{\frac{1}{2}} S^{-\frac{1}{2}},$$

and also define $D = P^{\frac{1}{2}}$, where $P^{\frac{1}{2}}$ denotes the symmetric square root of P. The matrix D can be used to rescale X and S to the same matrix V, defined by

$$V = \frac{1}{\sqrt{\mu}} D^{-1} X D^{-1} = \frac{1}{\sqrt{\mu}} DSD = \frac{1}{\sqrt{\mu}} (D^{-1}XSD)^{\frac{1}{2}}. \tag{2.7}$$

Note that both matrices D and V are symmetric and positive definite. We have

$$V^2 = \frac{1}{\sqrt{\mu}} D^{-1} XSD.$$

On the other hand, we define:

$$\begin{array}{l} \bar{A}_i = \frac{1}{\sqrt{\mu}} D A_i D,\ i = 1, \ldots, m, \\ D_X = \frac{1}{\sqrt{\mu}} D^{-1} \Delta X D^{-1}, \\ D_S = \frac{1}{\sqrt{\mu}} D \Delta S D. \end{array} \tag{2.8}$$

From (2.7), after some elementary reductions, the NT search direction $(D_X, \Delta y, D_S)$ satisfies the following system

$$\left\{\begin{array}{l} \bar{A}_i \bullet D_X = 0, \quad 1 \le i \le m, \\ \sum\limits_{i=1}^{m} \Delta y_i \bar{A}_i + D_S = 0, \\ D_X + D_S = V^{-1} - V. \end{array}\right. \tag{2.9}$$

Again, this system has a unique solution as $(D_X, \Delta y, D_S)$, which is called the NT search direction.

The first two equations in system (2.9) imply that D_X and D_S are orthogonal:

$$Tr(D_X D_S) = Tr(D_S D_X) = 0.$$

Hence, using the third equation in (2.9) we obtain

$$(\|D_X + D_S\|)^2 = \|D_X\|^2 + \|D_S\|^2 = \left\|V^{-1} - V\right\|^2 = \|\psi'(V)\|^2.$$

This implies that D_X, D_S are both zero if and only if $V^{-1} - V = 0$. In this case, X and S satisfy $XS = \mu I$, which indicates that X and S are the $\mu-$centers.

In fact, the right-hand side of the third equation in (2.9) is the negative gradient of the matrix barrier function $\Phi_c(V)$ with the classical kernel function $\psi_c(t) = \dfrac{t^2 - 1}{2} - \log(t)$, while $\psi_c(t)$ satisfies:

$$\psi_c'(1) = \psi_c(1) = 0, \psi_c''(t) > 0, t > 0 \text{ and } \lim_{t \longrightarrow 0^+} \psi_c(t) = \lim_{t \longrightarrow +\infty} \psi_c(t) = +\infty.$$

Then,

$$\begin{aligned} \Phi_c(V) &= Tr(\psi_c(V)) = \sum_{i=1}^{n} (\psi_c(\lambda_i(V))) \\ &= \sum_{i=1}^{n} \left(\tfrac{\lambda_i(V)^2 - 1}{2} - \log(\lambda_i(V)) \right). \end{aligned}$$

That is to say

$$\nabla \Phi_c(V) = V^{-1} - V.$$

Moreover, we call $\psi_c(t)$ the kernel function of the logarithmic barrier function $\Phi_c(V)$. Thus, the system (2.9) becomes

$$\left\{\begin{array}{l} \bar{A}_i \bullet D_X = 0, \quad 1 \le i \le m, \\ \sum\limits_{i=1}^{m} \Delta y_i \bar{A}_i + D_S = 0, \\ D_X + D_S = -\nabla \Phi_c(V). \end{array}\right. \tag{2.10}$$

In this paper, we replace $\Phi_c(V)$ by a new barrier function $\Phi_s(V)$ and $\psi_c(t)$ by a new kernel function $\psi_s(t)$, where ψ_s is defined in (1.1).

We replace the right-hand side of the third equation in (2.10) by $-\nabla\Phi_s(V)$. Thus this system can be rewritten as

$$\begin{cases} \bar{A}_i \bullet D_X = 0, \quad 1 \leq i \leq m, \\ \sum\limits_{i=1}^{m} \Delta y_i \bar{A}_i + D_S = 0, \\ D_X + D_S = -\nabla\Phi_s(V). \end{cases} \tag{2.11}$$

In the algorithm, we use the barrier function $\Phi_s(V)$ as a measure function and also we introduce the norm-based proximity measure $\delta(V)$ to the central path as follows:

$$\delta(V) := \frac{1}{2}\left\|\psi_s'(V)\right\| = \frac{1}{2}\sqrt{\sum_{i=1}^{m}\left[\psi_s'(\lambda_i(V))\right]^2} = \frac{1}{2}\left\|D_X + D_S\right\|, \tag{2.12}$$

and we have

$$D_X = D_S = 0_{n\times n} \quad \Leftrightarrow \quad \delta(V) = 0_{n\times n}$$
$$\Leftrightarrow \quad V = I \Leftrightarrow \quad \Phi_s(V) = 0 \quad \Leftrightarrow \quad X = X(\mu), \ S = S(\mu).$$

By taking a step along the search direction, with the step size α defined by some line search rules, we construct a new triple (X, y, S) according to

$$X_+ = X + \alpha\Delta X, \quad y_+ = y + \alpha\Delta y, S_+ = S + \alpha\Delta S.$$

Hence, the value of $\delta(V)$ can be considered as a measure for the distance between the matrix pair (X, y, S) and the central path.

2.2. **A generic primal–dual IPM for SDO.** We can now describe the algorithm briefly. according to the definition of the matrix V, it is determined by the current iterates (X, S) and the center parameter μ. Thus $\Phi_s(X, S, \mu)$ in the algorithm is another expression of the matrix function, we denote $\Phi_s(V) = \Phi_s(X, S, \mu)$. Hence we can use $\Phi_s(V)$ as a proximity function to measure the distance between the current iteration and the corresponding $\mu-$center. The primal-dual interior-point algorithm for SDO works as follows: Assume that $\tau \geq 1$ and there is a strictly feasible point (X, y, S) which is in a $\tau-$neighborhood of the given $\mu-$center. We update μ to $\mu_+ := (1-\theta)\mu$, for some fixed $\theta \in (0,1)$, and then solve the system (2.11) so that $(\Delta X, \Delta y, \Delta S)$ is computed via (2.8) to obtain the NT search direction. The positivity condition of a new iteration is ensured with the right choice of the step size α. This procedure is repeated until we find a new iteration (X_+, y_+, S_+) which is in a $\tau-$neighborhood of the μ_+-center and then we let $\mu := \mu_+$ and $(X, y, S) := (X_+, y_+, S_+)$. We repeat the process until $n\mu < \varepsilon$.The generic form of the algorithm is shown in Fig.1

Input:
A threshold parameter $\tau \geq 1$;
an accuracy parameter $\varepsilon > 0$;
a fixed barrier update parameter $\theta, 0 < \theta < 1$;.
a strictly feasible pair (X^0, S^0) and $\mu^0 = \frac{Tr(X^0S^0)}{n}$ such that $\Phi_s(X^0, S^0; \mu^0) \leq \tau$.
begin
$X := X^0; y := y^0; S := S^0; \mu := \mu^0;$
while $n\mu \geq \epsilon$ do
begin
$\mu := (1-\theta)\mu;$
while $\Phi_s (X, S, \mu) > \tau$ do
begin
Solve system (2.11) and use (2.8) to obtain $(\Delta X, \Delta y, \Delta S)$;
Determine a suitable step size α;
Update $(X, y, S) := (X, y, S) + \alpha (\Delta X, \Delta y, \Delta S)$.
end while
end while
end

Fig.1 Generic Primal-Dual Algorithm for SDO.

3. The properties of the New Kernel Function

We will now address a new kernel function with its properties are provided. Let's define the new univariate function

$$\psi_s(t) = \frac{(t^2-1)}{2} - \frac{\log(t)}{2} - \frac{1}{2s}\sum_{j=1}^{s} \frac{t^{1-jq}-1}{1-jq}, \quad q > 1,\, s \in \mathbb{N}\backslash\{0\}.$$

It is easy to observe that as $t \to 0$ or $t \to \infty$, then $\psi(t) \to \infty$. So, $\psi_s(t)$ is without a doubt a kernel function.

We'll need the first three derivatives of $\psi_s(t)$,we provide them as below

$$\psi_s^{'}(t) = t - \frac{1}{2t} - \frac{1}{2s}\sum_{j=1}^{s} t^{-jq},$$

$$\psi_s^{''}(t) = 1 + \frac{1}{2t^2} + \frac{1}{2s}\sum_{j=1}^{s} qjt^{-jq-1},$$

$$\psi_s^{'''}(t) = -\frac{1}{t^3} - \frac{1}{2s}\sum_{j=1}^{s} jq(jq+1)t^{-jq-2}.$$

If $s = 1$, we obtain the kernel function (12) given by Bouaafia et al. in [18]. The following lemma establishes the efficiency of the new kernel function (1.1).

Lemma 3.1. *Let $\psi_S(t)$ be as defined in (1.1) and $t > 0$. Then,*

$$\psi_S^{''}(t) > 1,$$

$$\psi_S^{'''}(t) < 0,$$

(3.1) $$t\psi_S^{''}(t) - \psi_S^{'}(t) > 0,$$

(3.2) $$t\psi_S^{''}(t) + \psi_S^{'}(t) > 0.$$

The last property (3.2) in lemma 3.1 is equivalent to convexity of composed functions $t \to \psi_s(e^t)$ and this holds if only if $\psi_s(\sqrt{t_1 t_2}) \leq \frac{1}{2}\left(\psi_s(t_1) + \psi_s(t_2)\right)$, for any $t_1, t_2 \geq 0$. This property is well-known in the literature, and numerous researchers have demonstrated it (see [7, 21]). We have the following theorem

Theorem 3.1. *[Proposition 5.2.6 in [1]]Let $V_1, V_2 \in \mathbf{S}^n_{++}$, and Φ_s is the real valued matrix function induced by the matrix function ψ_s. Then,*

$$\Phi_s\left(\left[V_1^{\frac{1}{2}} V_2 V_1^{\frac{1}{2}}\right]\right) \leq \frac{1}{2}\left(\Phi_s\left(V_1\right) + \Phi_s\left(V_2\right)\right).$$

We provide some technical findings of the new kernel function in preparation for later.

Lemma 3.2. *For $\psi_s(t)$, we've got*

$$\frac{1}{2}(t-1)^2 \leq \psi_s(t) \leq \frac{1}{2}\left[\psi_s'(t)\right]^2, \quad t > 0. \tag{3.3}$$

$$\psi_s(t) \leq \left[\frac{6+q(s+1)}{8}\right](t-1)^2, \quad t > 1. \tag{3.4}$$

Let $\sigma : [0, \infty[\to [1, +\infty[$ be the inverse function of $\psi_s(t)$ for $t \geq 1$ and $\rho : [0, \infty[\to]0, 1]$ be the inverse function of $-\frac{1}{2}\psi_s'(t)$ for all $t \in]0, 1]$. Then we have the following lemma.

Lemma 3.3. *For $\psi_s(t)$, we have*

$$1 + \sqrt{\frac{8s}{6+q(s+1)}} \leq \sigma(s) \leq 1 + \sqrt{2s}, \quad s \geq 0. \tag{3.5}$$

$$\rho(z) > \left[\frac{1}{4z+2}\right]^{\frac{1}{sq}}, \quad z > 0. \tag{3.6}$$

Proof. For (3.5), let $s = \psi_S(t), t \geq 1$, i.e., $\sigma(s) = t, t \geq 1$. By (3.3), we have $\psi_S(t) \geq \frac{1}{2}(t-1)^2$. Then $s \geq \frac{1}{2}(t-1)^2$, $t \geq 1$. This means that $t = \sigma(s) \leq 1 + \sqrt{2s}$. By (3.4), we get

$$s = \psi_S(t) \leq \left[\frac{6+q(S+1)}{8}\right](t-1)^2, \quad t \geq 1.$$

So, $t = \sigma(s) \geq 1 + \sqrt{\frac{8s}{6+q(S+1)}}$.

For (3.6), let $z = -\frac{1}{2}\psi_S'(t)$, $t \in]0, 1] \Leftrightarrow 2z = -\psi_S'(t), \quad t \in]0, 1]$. According to the definition of $\psi_S'(t)$, we have

$$2z = -t + \frac{1}{2t} + \frac{1}{2S}\sum_{j=1}^{S} t^{-jq} > -1 + \frac{1}{2S}\sum_{j=1}^{S} t^{-Sq} = -1 + \frac{1}{2}t^{-Sq},$$

which implies $t = \rho(z) > \left[\frac{1}{4z+2}\right]^{\frac{1}{Sq}}$. The proof is finished. □

Lemma 3.4. *[Lemma 5 in [12]] Let $\sigma : [0, \infty[\to [1, +\infty[$ is the inverse function of $\psi_s(t)$, $t \geq 1$. We have*

$$\Phi_s(\beta V) \leq n\psi_s\left(\beta\sigma\left(\frac{\Phi_s(V)}{n}\right)\right), \quad v \in \mathbb{R}^*, \beta \geq 1.$$

Lemma 3.5. *Let $0 \leq \theta < 1$, $V_+ = \frac{V}{\sqrt{1-\theta}}$. If $\Phi_s(V) \leq \tau$, then we have*

$$\Phi_s(V_+) \leq \frac{\theta n + 2\tau + 2\sqrt{2\tau n}}{2(1-\theta)}.$$

Proof. Because of $\frac{1}{\sqrt{1-\theta}} \geq 1$ and $\sigma\left(\frac{\Phi_S(v)}{n}\right) \geq 1$, then $\frac{\sigma\left(\frac{\Phi_S(v)}{n}\right)}{\sqrt{1-\theta}} \geq 1$. And for $t \geq 1$, we obtain $\psi_S(t) \leq \frac{t^2-1}{2}$.
We use Lemma 3.4 with $\beta = \frac{1}{\sqrt{1-\theta}}$, (3.5), and $\Phi_S(v) \leq \tau$, we obtain

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

This completes the proof. □

Denoting

$$(\Phi_s)_0 = \frac{\theta n + 2\tau + 2\sqrt{2\tau n}}{2(1-\theta)} = L\left(n, \theta, \tau\right).$$

So, during the algorithm's execution, $(\Phi_s)_0$ is an upper bound for $\Phi_s(V_+)$.

4. Complexity Analysis

In the next subsection, we compute a default step size α and the resulting decrease in the barrier function.

4.1. An estimation of the step size.

We devoted this section to calculating a default step size α and the consequent decrease in the barrier function. After a damped step, we obtain

$$X_+ := X + \alpha\Delta X, y_+ := y + \alpha\Delta y, S_+ := S + \alpha\Delta S$$

From (2.8)we have

$$X_{+} := X + \alpha \Delta X = X + \alpha \sqrt{\mu} D D_X D = \sqrt{\mu} D \left(V + \alpha D_X\right) D,$$

and

$$S_{+} = S + \alpha \Delta S = S + \alpha \sqrt{\mu} D D_S D = \sqrt{\mu} D^{-1} \left(V + \alpha D_S\right) D^{-1}.$$

On the other hand, we have

$$V_{+}^{2} = \frac{D^{-1} X_{+} S_{+} D}{\mu} = \left(V + \alpha D_X\right)\left(V + \alpha D_S\right),$$

and it is clear that the matrix V_{+}^{2} is similar to the matrix

$$\left(V + \alpha D_X\right)^{\frac{1}{2}} \left(V + \alpha D_S\right) \left(V + \alpha D_X\right)^{\frac{1}{2}},$$

since is convex and from Theorem (3.1) we have

$$\begin{aligned}\Phi_s\left(V_{+}\right) &= \Phi_s\left(\left(V + \alpha D_X\right)^{\frac{1}{2}} \left(V + \alpha D_S\right) \left(V + \alpha D_X\right)^{\frac{1}{2}}\right)\\ &\leq \frac{1}{2}\left(\Phi_s\left(V + \alpha D_X\right) + \Phi_s\left(V + \alpha D_S\right)\right).\end{aligned}$$

Defining for $\alpha > 0$,

$$f\left(\alpha\right) = \Phi_s\left(V_{+}\right) - \Phi_s\left(V\right).$$

We thus have $f\left(\alpha\right) \leq f_1\left(\alpha\right)$, where

$$f_1\left(\alpha\right) = \frac{1}{2}\left(\Phi_s\left(V + \alpha D_X\right) + \Phi_s\left(V + \alpha D_S\right)\right) - \Phi_s\left(V\right)$$

Then $f\left(\alpha\right)$ is the difference of the proximity between a new iterate and a current iterate for a fixed $\mu > 0$. It is easily seen that, $f_1\left(\alpha\right) = f\left(\alpha\right) = 0$.

Now, to estimate the decrease of the proximity during one step, we need the two successive derivatives of $f_1\left(\alpha\right)$ with respect to α.

By using the rule of differentiability [4, 19], we get

$$f_1'(\alpha) = \frac{1}{2} Tr\left(\psi_s'\left(V + \alpha D_X\right) D_X + \psi_s'\left(V + \alpha D_S\right) D_S\right),$$

$$f_1''\left(\alpha\right) = \frac{1}{2} Tr\left(\Psi''\left(V + \alpha D_X\right) D_X^2 + \Psi''\left(V + \alpha D_S\right) D_S^2\right).$$

By using (2.7) and (2.12), we obtain

$$f_1'\left(0\right) = \frac{1}{2} Tr\left(\psi_s'\left(V\right)\left(D_X + D_S\right)\right) = \frac{1}{2} Tr\left(-\psi_s'\left(V\right)^2\right) = -2\delta^2\left(V\right).$$

Noted by $\delta\left(V\right):=\delta$, and $\Phi_s=\Phi_s(V)$.

Lemma 4.1. *Let $\delta(v)$ defined in (2.12). Then*

$$\delta(V)\geq\sqrt{\frac{\Phi_s(V)}{2}}. \tag{4.1}$$

Proof. Using (3.3) and (2.12), we have

$$\Phi_s(v)=\sum_{i=1}^{n}\psi_s(\lambda_i(V))\leq\sum_{i=1}^{n}\frac{1}{2}\left[\psi_s^{'}(\lambda_i(V))\right]^2=\frac{1}{2}\sum_{i=1}^{n}\left[\psi_s^{'}(\lambda_i(V))\right]^2=2\delta(V)^2,$$

and so $\delta(V)\geq\sqrt{\frac{1}{2}\Phi_S(V)}$. This completes the proof. □

Lemma 4.2. *[Lemma 3.4.4 in [20]] One has*

$$f_1''\left(\alpha\right)\leq 2\delta^2\psi_s''\left(\lambda_n\left(V\right)-2\alpha\delta\right).$$

Lemma 4.3. *[Lemma 3.4.5 in [20]] If the step size α satisfies*

$$\psi_s'\left(\lambda_n\left(V\right)-2\alpha\delta\right)+\psi_s'\left(\lambda_n\left(V\right)\right)\leq 2\delta,$$

one has $f_1'\left(\alpha\right)\leq 0$

Lemma 4.4. *[Lemma 3.4.6 in [20]] Let $\rho:[0,\infty)\longrightarrow(0,1]$ denote the inverse function of the restriction of $-\frac{1}{2}\psi_s'\left(t\right)$ on the interval $(0,1]$, then the largest possible value of the step size of α satisfying $f_1'\left(\alpha\right)\leq 0$ is given by $\overline{\alpha}=\dfrac{\rho(\delta)-\rho(2\delta)}{2\delta}$*

Lemma 4.5. *[Lemma 4.4 in [1]] Let ρ and $\overline{\alpha}$ be as defined in Lemma 4.4.Then*

$$\frac{1}{\psi_s''\left(\rho(2\delta)\right)}\leq\overline{\alpha}\leq\frac{1}{\psi_s''\left(\rho(\delta)\right)}.$$

We use the notation

$$\widetilde{\alpha}=\frac{1}{\psi_s''\left(\rho(2\delta)\right)}$$

We are able to demonstrate the following Lemma

Lemma 4.6. *Let ρ and $\overline{\alpha}$ be as determined in Lemma 4.5. If $\Phi_s(V)\geq\tau\geq 1$, so we have*

$$\overline{\alpha}\geq\frac{2s}{2s+s\left(4\delta+2\right)^{\frac{2}{sq}}+q\sum_{j=1}^{s}j\left(4\delta+2\right)^{\frac{jq+1}{sq}}}.$$

Proof. Using Lemma 4.5, (3.1) and (3.6), we have

$$\begin{aligned}
\overline{\alpha} &\geq \frac{1}{\psi_s''\left(\rho(2\delta)\right)} \\
&= \frac{1}{1+\frac{1}{2}\rho(2\delta)^{-2}+\frac{q}{2s}\sum_{j=1}^{s} j\left(\rho(2\delta)\right)^{-jq-1}} \\
&\geq \frac{1}{1+\frac{1}{2}\left(4\delta+2\right)^{\frac{2}{sq}}+\frac{q}{2s}\sum_{j=1}^{s} j\left(4\delta+2\right)^{\frac{jq+1}{sq}}} \\
&= \frac{2s}{2s+s\left(4\delta+2\right)^{\frac{2}{sq}}+q\sum_{j=1}^{s} j\left(4\delta+2\right)^{\frac{jq+1}{sq}}}.
\end{aligned}$$

This completes the proof. □

Denoting

$$\widetilde{\alpha} = \frac{2s}{2s+s\left(4\delta+2\right)^{\frac{2}{sq}}+q\sum_{j=1}^{s} j\left(4\delta+2\right)^{\frac{jq+1}{sq}}}, \tag{4.2}$$

we have that $\widetilde{\alpha}$ is the default step size and that $\widetilde{\alpha} \leq \overline{\alpha}$.

Lemma 4.7. *[Lemma 3.12 in [7]] Let h be a convex and twice differentiable function with $h(0)=0$, $h'(0)<0$, which attains its minimum at $t^*>0$. If h'' is increasing for $t\in[0,t^*]$, then*

$$h(t) \leq \frac{th(0)}{2}, \quad 0 \leq t \leq t^*.$$

The following result is of great importance.

Lemma 4.8. *[Lemma 4.5 in [1]] If the step size α satisfies $\alpha \leq \overline{\alpha}$, then*

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

Lemma 4.9. *Let $\Phi_s(V) \geq 1$ and let $\widetilde{\alpha}$ be the default step size as defined in (4.2). Then, we have*

$$f(\widetilde{\alpha}) \leq -\frac{2s}{8\sqrt{2}(s+8)\left(1+4qs\right)}\left[\Phi_s(V)\right]^{\frac{sq-1}{2sq}}.$$

Proof. Since $\Phi_s(v) \geq 1$, then from (4.1), we have

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

Using Lemma 4.8 (Lemma 4.5 in [1]) with $\alpha = \widetilde{\alpha}$ and (4.2). This completes the proof. □

4.2. **Iteration bound.** Following the updating of μ to $(1-\theta)\mu$, we obtain

$$\Phi_s(V_+) \leq (\Phi_s)_0 = \frac{\theta n + 2\tau + 2\sqrt{2\tau n}}{2(1-\theta)} = L(n, \theta, \tau).$$

After μ-update to $(1-\theta)\mu$, it is necessary tocount how many inner iterations are required to come back to the situation, where $\Phi_s(V_+) \leq \tau$. We declare the value of $\Phi_s(V)$ after the updating of μ as $(\Phi_s)_0$ and we denote by $(\Phi_s)_k$, $k = 1, 2, \ldots, K$ the values of subsequent in the same outer iteration, such that K represent the total number of inner iterations per the outer iteration.

Lemma 4.10. *[Lemma 14 in [7]] Let $t_0, t_1, \ldots, t_k$ 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 $K \leq \left[\dfrac{t_0^\gamma}{\beta\gamma}\right]$.

Thus, it follows that

$$(\Phi_s)_{k+1} \leq (\Phi_s)_k - k(\Phi s)^{1-\gamma}, k = 0, 1, \ldots, K-1,$$

with

$$\kappa = \frac{2s}{8\sqrt{2}(s+8)(1+4qs)}, \gamma = 1 - \frac{sq-1}{2sq} = \frac{sq+1}{2sq}.$$

Lemma 4.11. *Let K be the total number of inner iterations in the outer iteration. Then we have*

$$K \leq \frac{8\sqrt{2}q(s+8)(1+4qs)}{1+sq}\left[(\Phi_s)_0\right]^{\frac{sq+1}{2sq}}.$$

Proof. By Lemma 1.3.2 in [7], we have $K \leq \frac{\left[(\Phi_s)_0\right]^\gamma}{\kappa\gamma} = \frac{8\sqrt{2}q(s+8)(1+4qs)}{sq+1}\left[(\Phi_s)_0\right]^{\frac{sq+1}{2sq}}$. □

Now, we estimate the total number of iterations of our algorithm.

We recall that the number of outer iterations is limited from above by $\dfrac{\log\left(\frac{n}{\epsilon}\right)}{\theta}$ (see Lemma II.17, page 116 in [9]). We can establish an upper bound on the total number of iterations by multiplying the number of outer iterations by the number of inner iterations, such as

$$\frac{8\sqrt{2}q(s+8)(1+4qs)}{sq+1}\left[(\Phi s)_0\right]^{\frac{sq+1}{2sq}}\frac{\log\left(\frac{n}{\epsilon}\right)}{\theta}. \tag{4.3}$$

In the methods of large-update with $\tau = \mathbf{O}(n)$ and $\theta = \Theta(1)$, we have

$$\mathbf{O}\left(qsn^{\frac{sq+1}{2sq}} \log\left(\frac{n}{\epsilon}\right)\right) \text{ iterations complexity.}$$

This is the best well-known complexity results for large-update methods.

n the methods of small-update, the replacement of $\tau = \mathbf{O}(1)$ and $\theta = \Theta\left(\frac{1}{\sqrt{n}}\right)$ in (4.3) does not provide the best possible bound. The best bound is obtained as follows.

By (3.4), with $\psi_s(t) \leq \left[\frac{6+q(s+1)}{8}\right](t-1)^2$, $t > 1$. We have

$$\begin{aligned}
\Phi_s(V_+) &\leq n\psi_s\left(\frac{1}{\sqrt{1-\theta}}\sigma\left(\frac{\Phi_s(V)}{n}\right)\right) \\
&\leq n\left[\frac{6+q(s+1)}{8}\right]\left(\frac{1}{\sqrt{1-\theta}}\sigma\left(\frac{\Phi_s(V)}{n}\right)-1\right)^2 \\
&= \frac{n\left(6+q(s+1)\right)}{8(1-\theta)}\left(\sigma\left(\frac{\Phi_s(V)}{n}\right)-\sqrt{1-\theta}\right)^2.
\end{aligned}$$

Using (3.5), we have

$$\begin{aligned}
&\frac{n\left(6+q(s+1)\right)}{8(1-\theta)}\left(\sigma\left(\frac{\Phi_s(V)}{n}\right)-\sqrt{1-\theta}\right)^2 \\
\leq\ &\frac{n\left(6+q(s+1)\right)}{8(1-\theta)}\left(\left(1+\sqrt{2\frac{\Phi_s(V)}{n}}\right)-\sqrt{1-\theta}\right)^2 \\
=\ &\frac{n\left(6+q(s+1)\right)}{8(1-\theta)}\left(\left(1-\sqrt{1-\theta}\right)+\sqrt{2\frac{\Phi_s(V)}{n}}\right)^2 \\
\leq\ &\frac{n\left(6+q(s+1)\right)}{8(1-\theta)}\left(\theta+\sqrt{2\frac{\tau}{n}}\right)^2 \\
=\ &\frac{\left(6+q(s+1)\right)}{8(1-\theta)}\left(\theta\sqrt{n}+\sqrt{2\tau}\right)^2 = \left(\Phi_s\right)_0,
\end{aligned}$$

where we utilized that as well $1-\sqrt{1-\theta} = \frac{\theta}{1+\theta} \leq \theta$ and $\Phi_s(v) \leq \tau$, utilizing this upper bound for $\left(\Phi_s\right)_0$, we obtain the following iteration bound

$$\frac{8\sqrt{2}q(s+8)\left(1+4qs\right)}{sq+1}\left[\left(\Phi_s\right)_0\right]^{\frac{sq+1}{2sq}}\frac{\log\left(\frac{n}{\epsilon}\right)}{\theta}.$$

Note now $(\Phi_s)_0 = \mathbf{O}(qs)$, and the iteration bound it be given as follows

$$\mathbf{O}\left(q^2 s^2 \sqrt{n} \log\left(\frac{n}{\epsilon}\right)\right) \text{ iterations complexity.}$$

5. Numerical tests

The algorithm has been tested on some benchmark problems issued from the library of test problems SDPLIB [22]. Here we used Nbr which means the iterations number produced by the algorithm. The implementation is manipulated in $C++$. Our tolerance is $\varepsilon = 10^{-8}$. For the update parameter, we have vary $0 < \theta < 1$.

<table>
<tr><th>Examples</th><th>Size (m,n)</th><th>Nbr. of iterations from [22]</th><th colspan="5">Results of our Algorithm</th></tr>
<tr><td rowspan="2">control1</td><td rowspan="2">$(21, 15)$</td><td rowspan="2">106</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>104</td><td>97</td><td>46</td><td>34</td></tr>
<tr><td rowspan="2">hinf1</td><td rowspan="2">$(13, 14)$</td><td rowspan="2">27</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>31</td><td>27</td><td>20</td><td>18</td></tr>
<tr><td rowspan="2">hinf2</td><td rowspan="2">$(13, 16)$</td><td rowspan="2">43</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>39</td><td>32</td><td>27</td><td>23</td></tr>
<tr><td rowspan="2">hinf3</td><td rowspan="2">$(13, 16)$</td><td rowspan="2">109</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>103</td><td>99</td><td>78</td><td>69</td></tr>
<tr><td rowspan="2">hinf4</td><td rowspan="2">$(13, 16)$</td><td rowspan="2">39</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>31</td><td>29</td><td>17</td><td>11</td></tr>
<tr><td rowspan="2">hinf5</td><td rowspan="2">$(13, 16)$</td><td rowspan="2">42</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>37</td><td>30</td><td>19</td><td>15</td></tr>
<tr><td rowspan="2">hinf7</td><td rowspan="2">$(13, 16)$</td><td rowspan="2">38</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>31</td><td>27</td><td>23</td><td>17</td></tr>
<tr><td rowspan="2">hinf9</td><td rowspan="2">$(13, 16)$</td><td rowspan="2">28</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>27</td><td>19</td><td>13</td><td>11</td></tr>
<tr><td rowspan="2">hinf10</td><td rowspan="2">$(21, 18)$</td><td rowspan="2">57</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>49</td><td>36</td><td>29</td><td>17</td></tr>
<tr><td rowspan="2">truss1</td><td rowspan="2">$(6, 13)$</td><td rowspan="2">17</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>11</td><td>6</td><td>5</td><td>3</td></tr>
<tr><td rowspan="2">truss4</td><td rowspan="2">$(12, 19)$</td><td rowspan="2">21</td><td>θ</td><td>0.15</td><td>0.35</td><td>0.75</td><td>0.95</td></tr>
<tr><td>**Nbr**</td><td>17</td><td>11</td><td>9</td><td>6</td></tr>
</table>

6. Conclusion

In this paper, we have improved the algorithmic complexity of (IPM) methods for (SDO) problems by a new kernel function. More specifically, we have proved large-update and small-update of primal-dual algorithm based on a new kernel function with a logarithmic barrier term defined by (1.1). Future research might focus on the extension to symmetric cone optimization. Finally, for the numerical tests, some strategies are used and indicate that our kernel function used in the algorithm is efficient

References

[1] Y.Q. Bai, M. El Ghami, C. Roos: *A comparative study of kernel functions for primal-dual interior point algorithms in linear optimization*, SIAM J. Optimiz **15**(1) (2004), 101-128.

[2] Y.Q. Bai, M. El Ghami, C. Roos: *A new efficient large-update primal-dual interior-point method based on a finite barrier*, SIAM J. Optim. **13**(3) (2003), 766-782.

[3] E. De Klerk: *Aspects of Semidefinite Programming*, Applied Optimization, vol. 65. Kluwer Academic, Dordrecht, 2002.

[4] R.A> Horn, C.R. Johnson: *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.

[5] Y.E. Nesterov, M.J. Todd: *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res. bf22(1) (1997),1-42

[6] B. Kheirfam, M. Moslem: *Apolynomial-time algorithme for linear optimization based on a new kernel function with trigonometric barrier term*, Yujor **25**(2) (2015), 233-250.

[7] J. Peng, C. Roos, T. Terlaky: *Self-regular functions and new search directions for linear and semidefinite optimization*, Math. Programming, bf93(1) (2002), 129–171.

[8] J. Peng, C. Roos, T. Terlaky: *Self-Regularity. A New Paradigm for Primal-Dual Interior-Point Algorithms*, Princeton University Press, Princeton, 2002.

[9] C. Roos. T. Terlaky. J.P. Vial: *Theory and Algorithms for Linear Optimization*, An Interior Point Approach. Wiley, Chichester, 1997.

[10] M.R. Peyghami: *An interior point approach for semidefinite optimization using new proximity functions. Asia-Pacific*, Journal of Operational Research **26**(03) (2002), 365-382.

[11] E.D. Andersen, J. Gondzio, Cs. Mészáros, X. Xu: *Implementation of interior point methods for large scale linear programming*, in T. Terlaky (ed.), Interior Point Methods of Mathematical Programming, Kluwer Academic Publishers, The Netherlands. (1996) 189–252.

[12] E.A. Djeffal, L. Djeffal: *A path following interior-point algorithm for semidefinite optimization problem based on new kernel function*, Journal of Mathematical Modeling, **4**(1) (2016), 35–58.

[13] N.K. KARMARKAR: *A new polynomial-time algorithm for linear programming*, Combinatorica **4** (1984), 375–395.

[14] M.J. TODD: *A study of search directions in primal-dual interior point methods for semidefinite programming*, Optim. Methods Softw., **11** (1999), 1-46.

[15] G.Q. WANG, Y.Q. BAI, C. ROOS: *Primal-dual interior-point algorithms for semidefinite optimization based on a simple kernel function*, J. Math. Model. Algorithms bf4(4) (2005), 409-433.

[16] Y. ZHANG: *On extending primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim. **8**(2) (1998), 365-386.

[17] M.W. ZHANG: *A large-update interior-piont algorithm for convex quadratic semi-definite optimization based on a new kernel function[J]*, Acta Math, Sinica (Engl Ser). **28**(11) (2012), 2313-2328.

[18] M. BOUAAFIA, D. BENTERKI, Y. ADNAN: *An efficient parameterized logarithmic kernel function for linear optimization*, Optim. Lett. **12** (2018), 1079-1097.

[19] J.C. PENG, C. ROOS, T. TERLAKY: *New class of polynomial primaldual methods for linear and semide... nite optimization*, European J. Oper. Res. **143**(2) (2002), 234-256.

[20] M. ELGHAMI: *New primal-dual interior point methods based on kernel functions*, Ph. D. thesis, Delft University, Netherland, 2005.

[21] M. EL GHAMI, Z.A. GUENNOUN, S. BOUALI, 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.

[22] http://infohost.nmt.edu/∼sdplib/

DEPARTMENT OF MATHEMATICS
UNIVERSITY OF BATNA 2
BATNA,
ALGERIA.
Email address: guemmazabderrahim@yahoo.com

DEPARTMENT OF MATHEMATICS
UNIVERSITY OF BATNA 2
BATNA,
ALGERIA.
Email address: l.djeffal@univ-batna2.dz

DEPARTMENT OF MATHEMATICS
UNIVERSITY OF BATNA 2
BATNA,
ALGERIA.
Email address: b.bounibane@univ-batna2.dz