Int. J. Appl. Math. Comput. Sci., 2026, Vol. 36, No. 2, 177–194
DOI: 10.61822/amcs-2026-0013

A PRIMAL–DUAL INTERIOR POINT METHOD FOR COMPLEX–VARIABLE OPTIMIZATION PROBLEMS

MOUNIA LAOUAR[a], MAHMOUD BRAHIMI[a] , RAOUF ZIADI[b], MOHAMMED A. SALEH[c*], ABDULGADER Z. ALMAYMUNI[c] , ABDALILAH ALHALANGY[d]

[a] Laboratory of Partial Differential Equations
Department of Mathematics
University of Batna 2
Constantine Road, Fesdis, Batna 05078, Algeria
e-mail: {m.laouar,m.brahimi}@univ-batna2.dz

[b] Laboratory of Fundamental and Numerical Mathematics (LMFN)
Department of Mathematics
University Ferhat Abbas
Setif 1 19000 Setif, Algeria
e-mail: ziadi.raouf@gmail.com

[c] Department of Cybersecurity
College of Computer
Qassim University
Buraydah 52571, Saudi Arabia
e-mail: {m.saleh,almaymuni}@qu.edu.sa

[d] Department of Computer Engineering
College of Computer
Qassim University
Buraydah 52571, Saudi Arabia
e-mail: a.alhalangy@qu.edu.sa

In this paper, we propose a primal-dual interior-point method for solving convex optimization problems with complex variables, relying on a newly defined complex-valued kernel function. We extend classical kernel functions to the complex domain by establishing appropriate differentiability and convexity properties that guarantee the well-posedness and convergence of the proposed algorithm. Our theoretical approach encompasses the formulation of penalized optimality conditions, the definition of a modified Newton direction tailored to complex parametrization, and the design of a central pathtracking algorithm featuring adaptive barrier parameter updating. A rigorous complexity analysis yields polynomial bounds depending on the problem dimension and the desired accuracy. Numerical experiments on large-scale complex-variable problems demonstrate both the effectiveness and robustness of the proposed approach. The results validate the algorithm's dimension-independence property, with iteration counts remaining stable across substantial increases in problem size, and reveal significant computational advantages over state-of-the-art general-purpose solvers including IPOPT (Interior Point Optimizer). This work advances the theoretical foundations of interior-point methods in the complex domain and opens new perspectives for high-dimensional complex optimization.

Keywords: convex optimization, complex variables, optimization problems, complex-valued kernel function, Newton direction.

1. Introduction

Convex optimization is central to modern computational science and engineering, with applications in machine learning (Bauschke and Combettes, 2017; Nesterov, 2004), signal processing (Took and Mandic, 2010), control theory, and quantum information (Watrous, 2018). It involves minimizing a convex function over a convex set, where every local minimum

*Corresponding author

is also global. Interior-point methods (IPMs) are among the most powerful solvers, combining theoretical guaranties in polynomial-time with outstanding practical efficiency (Boyd and Vandenberghe, 2004; Forsgren *et al*., 2002; Nesterov and Nemirovski, 1994; Wright, 1997; Si *et al*., 2021; Matychyn and Onyshchenko, 2021; Srinivasarengan *et al*., 2022; Kaczorek, 2026).

However, a significant limitation of existing IPMs is their near-exclusive focus on real-valued variables. This stands in sharp contrast with many modern applications that are naturally formulated in the complex domain. In quantum computing, for instance, optimizing quantum states and operators requires objective functions defined over complex vector spaces (Viksal, 2023; Luo *et al*., 2007), while in signal processing and deep neural networks, complex variables are essential for accurately modeling phase and frequency phenomena (Hirose, 2013; Trabelsi *et al*., 2018). The standard workaround—decomposing problems into real and imaginary components—often proves analytically and numerically limiting, as it destroys the inherent complex structure and may compromise both convergence behavior and numerical precision or stability.

The extension of a unified IPM theory to the complex domain thus poses significant theoretical challenges, particularly in defining appropriate kernel functions and deriving search directions that are compatible with complex geometry. General-purpose interior-point solvers such as IPOPT (Wächter and Biegler, 2006; Leulmi *et al*., 2024; Maiza *et al*., 2025), while capable of handling complex variables through real reformulation, do not exploit convexity and lack the polynomial-time complexity guarantees that specialized convex methods can achieve. While some research has addressed specific cases, such as problems with real-valued cost functions (Bai *et al*., 2004; Cai *et al*., 2014; Lesaja and Roos, 2010; Peng *et al*., 2002), and recent work has continued to advance kernel function design for primal-dual methods in various settings (Bouafia *et al*., 2016; Darvay *et al*., 2024; Lee *et al*., 2013; Zhang and Xu, 2011), a unified theory for general convex optimization problems over complex variables remains an open area.

This work addresses the general problem formulated as

$$\begin{cases} \min\limits_{\mathbf{z}\in\mathbf{C}^n} & f(\mathbf{z}) \\ \text{subject to} & g_i(\mathbf{z}) \le 0, \quad i = 1, \dots, m, \end{cases} \tag{P}$$

where the cost function $f : \mathbb{C}^n \to \mathbb{R}$ and the constraints $g_i : \mathbb{C}^n \to \mathbb{R}$ are convex and twice continuously differentiable when viewed as functions from $\mathbb{R}^{2n}$ to $\mathbb{R}$ (under the standard identification of $\mathbb{C}^n$ with $\mathbb{R}^{2n}$).

Let $g(\mathbf{z}) = (g_1(\mathbf{z}), \dots, g_m(\mathbf{z}))^\top$. We define the Lagrangian function as

$$\mathcal{L}(\mathbf{z}, \mathbf{s}) := f(\mathbf{z}) + \mathbf{s}^\top g(\mathbf{z}), \quad \mathbf{z} \in \mathbb{C}^n,\ \mathbf{s} \in \mathbb{R}^m.$$

The associated dual function is

$$L(\mathbf{s}) := \inf_{\mathbf{z}\in\mathbb{C}^n} \mathcal{L}(\mathbf{z}, \mathbf{s}) = \inf_{\mathbf{z}\in\mathbb{C}^n} \big(f(\mathbf{z}) + \mathbf{s}^\top g(\mathbf{z})\big).$$

The Lagrange dual problem is then given by

$$\begin{cases} \max\limits_{\mathbf{s}\in\mathbb{R}^m} & L(\mathbf{s}) \\ \text{subject to} & \mathbf{s} \ge 0, \end{cases} \tag{D}$$

where the inequality $\mathbf{s} \ge 0$ holds componentwise.

To efficiently solve Problem (P), this article presents a new primal-dual interior-point method based on a kernel function specifically designed for this class of problems.

Leveraging fundamental work on self-regularizing barriers (Djeffal and Laouar, 2019; Laouar *et al.*, 2023; 2022; Tanksley, 2009; Ye, 1997) as well as recent theoretical advances in kernel-based interior-point algorithms (Darvay *et al*., 2024; Zhang and Xu, 2011), and employing the sophisticated machinery of Cauchy–Riemann (CR) calculus (Wirtinger) calculus tools, we successfully extend these concepts to the complex setting. Our main contribution lies in the design of a new class of complex kernel functions specifically tailored to the nonlinear convex optimization problem in complex variables introduced in Problem (P). These functions are the first to rigorously extend the most powerful real-valued self-regular kernels to the complex domain while preserving, directly in $\mathbb{C}^n$, all the theoretical properties that have made large-update primal-dual methods successful in the real case: exponential convexity, twice continuous differentiability with respect to the complex variable, and strict satisfaction of the classical growth and elasticity conditions.

Thanks to these properties, the proposed kernels can be directly incorporated into a full-Newton primal-dual interior-point framework without falling back on the usual real-isomorphic splitting into $\mathbb{R}^{2n}$. We derive the appropriate complex penalized Karush–Kuhn–Tucker (KKT) system, introduce a modified Newton direction that genuinely exploits the complex structure (using Wirtinger derivatives while maintaining symmetry of the linear systems), and design an adaptive barrier parameter update that ensures the existence and feasibility of a complex central path. This leads to the first primal-dual interior-point algorithm with proven polynomial iteration complexity for large-scale convex optimization over the complex domain.

The remainder of this paper is organized as follows. Section 2 presents the details of the complex primal-dual interior-point algorithm for Problem (P) and introduces

Listing 1. Notational convertions used in the paper.

Symbol	Description
Sets and variables	
$\mathbb{C}^n, \mathbb{R}^n$	n-Dimensional complex and real spaces
$\mathbf{z}, \mathbf{x}, \mathbf{y}$	Vectors in $\mathbb{C}^n$ or $\mathbb{R}^n$ (bold notation)
$\mathbf{g}(\mathbf{z}) \leq 0$	Vector of inequality constraints ($g_j(\mathbf{z}) \leq 0$ for all j)
$\mathbf{s} > 0$	Vector of strictly positive slack variables ($s_j > 0$ for all j)
$\mathbb{R}_{++}$	Set of strictly positive real numbers
t, μ	Step size (scalar) and barrier parameter
Operators and measures	
$\odot$	Hadamard (elementwise) product: $(\mathbf{u} \odot \mathbf{v})_i = u_i v_i$
$\mathbf{A}^\top$	Transpose of a vector or matrix $\mathbf{A}$
$\|\cdot\|$	Euclidean norm (2-norm) unless otherwise specified
$\nabla f, \nabla^2 f$	Gradient and Hessian of a function f
$\delta(\mathbf{v})$	Proximity measure quantifying the distance of the current iterate to the central path
$O(\sqrt{m})$	Big-O notation: theoretical iteration complexity bound
KKT	Karush–Kuhn–Tucker optimality conditions
Solver configurations (IPOPT)	
IPOPT-Def	IPOPT solver with default settings (tolerance 10^{-8})
IPOPT-Tight	High-precision configuration (tolerance 10^{-10})
IPOPT-Loose	Low-precision configuration (tolerance 10^{-6})
IPOPT-Mono	Monotone barrier parameter update strategy
IPOPT-V.T.	Very tight precision (tolerance 10^{-12})
IPOPT-Fast	Optimized for speed with reduced accuracy checks

several new definitions. In Section 3 we investigate the theoretical properties of the proposed complex kernel function and establish the polynomial complexity of the algorithm. Section 4 reports the numerical performance of our approach, and Section 5 concludes the paper.

The main notations, operators, and solver configurations used throughout this paper are presented as Listing 1.

2. Interior-point algorithm based on a complex kernel function

The development of efficient and robust optimization algorithms remains a cornerstone of computational mathematics and operations research. Building upon the established framework of primal-dual interior-point methods (IPMs), this section introduces an algorithm approach that incorporates a complex kernel function designed to enhance the method's performance characteristics. We require the following assumptions, where (A2) is a strengthening of (A1):

(A1) The feasible set $\mathcal{P} := \{\mathbf{z} \in \mathbb{C}^n \mid g(\mathbf{z}) \leq 0\}$ is nonempty.

(A2) Slater's constraint qualification holds: there exists $\tilde{\mathbf{z}} \in \mathbb{C}^n$ such that $g(\tilde{\mathbf{z}}) < 0$.

The discussion then proceeds to define the core components of the IPM, including the Lagrangian, the dual system, and the mechanism for calculating the perturbed KKT conditions through the introduction of a positive barrier parameter μ. This rigorous formulation sets the stage for the subsequent analysis of the proposed algorithm and its defining features.

Throughout this paper, $\mathbb{C}^n$ is identified with $\mathbb{R}^{2n}$ via the standard real–imaginary decomposition, so that all derivatives and convexity notions are understood in the real sense. We assume that f and the functions g_i ($i = 1, \ldots, m$) are twice continuously differentiable on $\mathbb{C}^n$, with f convex and each g_i convex. Consequently, the Lagrangian

$$\mathcal{L}(\mathbf{z}, \mathbf{s}) = f(\mathbf{z}) + \sum_{i=1}^{m} \mathbf{s}_i g_i(\mathbf{z}), \qquad (\mathbf{z}, \mathbf{s}) \in \mathbb{C}^n \times \mathbb{R}^m,$$

is twice continuously differentiable and convex in $\mathbf{z}$ for every fixed $\mathbf{s} \geq 0$. Its gradients are

$$\nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}, \mathbf{s}) = \nabla f(\mathbf{z}) + \sum_{i=1}^{m} \mathbf{s}_i \nabla g_i(\mathbf{z}),$$

$$\nabla_{\mathbf{s}} \mathcal{L}(\mathbf{z}, \mathbf{s}) = g(\mathbf{z}) \in \mathbb{R}^m.$$

The primal-dual strictly feasible set is defined by

$$\mathcal{F}^0 := \mathring{\mathcal{P}} \times \mathbb{R}^m_{++} = \big\{(\mathbf{z}, \mathbf{s}) \in \mathbb{C}^n \times \mathbb{R}^m_{++} \mid g(\mathbf{z}) < 0, \mathbf{s} > 0\big\}.$$

Assumption (A2) immediately implies that $\mathcal{F}^0$ is nonempty.

Under (A1)–(A2) and the above convexity assumptions, strong duality holds and the Karush–Kuhn–Tucker (KKT) conditions are necessary and sufficient for optimality. Specifically, $\mathbf{z}^* \in \mathbb{C}^n$ is an optimal solution of Problem (P) if and only if there exists $\mathbf{s}^* \in \mathbb{R}^m_+$ such that $(\mathbf{z}^*, \mathbf{s}^*)$ satisfies the KKT system

$$\begin{cases} \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}^*, \mathbf{s}^*) = 0, \\ \mathbf{s}_i^* \, g_i(\mathbf{z}^*) = 0, \quad i = 1, \dots, m, \\ g(\mathbf{z}^*) \le 0, \quad \mathbf{s}^* \ge 0. \end{cases} \tag{1}$$

To derive the central path, we relax the complementarity condition $\mathbf{s}_i g_i(\mathbf{z}) = 0$ by introducing a positive barrier parameter $\mu > 0$ and consider the perturbed KKT system

$$\begin{cases} \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}, \mathbf{s}) = 0, \\ -\, \mathbf{s}_i g_i(\mathbf{z}) = \mu, \quad i = 1, \dots, m, \\ g(\mathbf{z}) < 0, \quad \mathbf{s} > 0, \end{cases} \tag{2}$$

where the inequalities are understood componentwise. Equivalently, using the Hadamard (elementwise) product $\odot$ and the vector of ones $\mathbf{1} = (1, \dots, 1)^\top \in \mathbb{R}^m$, the perturbed system can be written as

$$\begin{cases} \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}, \mathbf{s}) = 0, \\ -\, \mathbf{s} \odot g(\mathbf{z}) = \mu \mathbf{1}, \\ g(\mathbf{z}) < 0, \quad \mathbf{s} > 0. \end{cases} \tag{3}$$

For all $\mu > 0$, we assume that the perturbed system (3) admits a unique solution $(\mathbf{z}(\mu), \mathbf{s}(\mu)) \in \mathcal{F}^0$. The mapping $\mu \mapsto (\mathbf{z}(\mu), \mathbf{s}(\mu))$ then defines the so-called central path as $\mu \searrow 0$.

The perturbed system (3) can be recast as finding $(z, \mathbf{s}) \in \mathcal{F}^0$ such that $F_\mu(\mathbf{z}, \mathbf{s}) = 0$, where

$$F_\mu(\mathbf{z}, \mathbf{s}) := \begin{bmatrix} \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}, \mathbf{s}) \\ -\, \mathbf{s} \odot g(\mathbf{z}) - \mu \mathbf{1} \end{bmatrix}, \qquad F_\mu : \mathbb{C}^n \times \mathbb{R}^m \to \mathbb{R}^{2n+m}. \tag{4}$$

2.1. Application of Newton's method. For fixed $\mu > 0$, the primal-dual Newton direction $(\Delta \mathbf{z}, \Delta \mathbf{s})$ at the current iterate $(\mathbf{z}, \mathbf{s}) \in \mathcal{F}^0$ is defined as the solution of the linear system

$$DF_\mu(\mathbf{z}, \mathbf{s}) \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{s} \end{bmatrix} = -F_\mu(\mathbf{z}, \mathbf{s}),$$

where the Jacobian matrix is

$$DF_\mu(\mathbf{z}, \mathbf{s}) = \begin{bmatrix} \nabla^2_{\mathbf{zz}} \mathcal{L}(\mathbf{z}, \mathbf{s}) & \nabla g(\mathbf{z})^\top \\ -S \nabla g(\mathbf{z}) & -G(\mathbf{z}) \end{bmatrix}. \tag{5}$$

Here $\nabla g(\mathbf{z}) \in \mathbb{R}^{m \times 2n}$ is the constraint Jacobian with the i-th row $\nabla g_i(\mathbf{z})^\top$, $S = \operatorname{diag}(\mathbf{s})$, and $G(\mathbf{z}) = \operatorname{diag}\big(g(\mathbf{z})\big)$.

Thus, solving (3) amounts to solving the smooth nonlinear system $F_\mu(\mathbf{z}, \mathbf{s}) = 0$ over the strictly feasible region $\mathcal{F}^0 = \{(\mathbf{z}, \mathbf{s}) : g(\mathbf{z}) < 0, \mathbf{s} > 0\}$. The Newton method maintains strict feasibility at each iteration.

This yields the Newton system

$$\begin{cases} \nabla^2_{\mathbf{zz}} \mathcal{L}(\mathbf{z}, \mathbf{s}) \, \Delta \mathbf{z} + \nabla g(\mathbf{z})^\top \Delta \mathbf{s} = -\nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}, \mathbf{s}), \\ -S \nabla g(\mathbf{z}) \, \Delta \mathbf{z} - G(\mathbf{z}) \Delta \mathbf{s} = \mathbf{s} \odot g(\mathbf{z}) + \mu \mathbf{1}. \end{cases} \tag{6}$$

Reduced KKT system. By eliminating $\Delta \mathbf{s}$ from the second equation and substituting into the first, we obtain

$$M \, \Delta \mathbf{z} = R, \tag{7}$$

where

$$M := \nabla^2_{\mathbf{zz}} \mathcal{L}(\mathbf{z}, s) - \nabla g(\mathbf{z})^\top G(\mathbf{z})^{-1} S \nabla g(\mathbf{z}), \tag{8}$$

$$R := -\nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}, \mathbf{s}) + \nabla g(\mathbf{z})^\top G(\mathbf{z})^{-1} \big(\mathbf{s} \odot g(\mathbf{z}) + \mu \mathbf{1}\big). \tag{9}$$

Remark 1. Since $g(\mathbf{z}) < 0$ and $\mathbf{s} > 0$, the diagonal matrix

$$G(\mathbf{z})^{-1} S = \operatorname{diag}\left(\frac{\mathbf{s}_1}{g_1(\mathbf{z})}, \dots, \frac{\mathbf{s}_m}{g_m(\mathbf{z})} \right)$$

is negative definite. Hence the second term in (8) is negative semidefinite. If, in addition, $\nabla^2_{\mathbf{zz}} \mathcal{L}(\mathbf{z}, \mathbf{s})$ is positive definite (e.g., if $\mathcal{L}(\cdot, \mathbf{s})$ is strongly convex in $\mathbf{z}$), then the reduced Hessian M is positive definite in a neighborhood of the central path, ensuring uniqueness of $\Delta \mathbf{z}$.

2.2. New complex-valued kernel function. We use the notation $t = \Re(t) + \mathrm{i}\Im(t) \in \mathbb{C}$ and identify it with $(\Re(t), \Im(t))^\top \in \mathbb{R}^2$ when computing derivatives (Hjørungnes and Gesbert, 2007; Kreutz-Delgado, 2009). We set $e_1 = (1, 0)^\top \equiv 1 \in \mathbb{C}$ and $\mathbb{0} = (0, 0)^\top \equiv 0 \in \mathbb{C}$.

Definition 1. (*Complex-kernel function*) Let $\psi : \mathbb{C}^*_+ \to \mathbb{R}_+$, where

$$\mathbb{C}^*_+ := \{t \in \mathbb{C} \mid \Re(t) > 0\}.$$

The function ψ is called a *complex-kernel function* if it satisfies the following conditions:

1. ψ is twice continuously differentiable, i.e., $\psi \in C^2(\mathbb{C}^*_+)$;
2. for all $t \in \mathbb{C}^*_+$, the Hessian matrix

$$H_\psi(t) := \begin{pmatrix} \dfrac{\partial^2 \psi}{\partial \Re(t)^2} & \dfrac{\partial^2 \psi}{\partial \Re(t) \partial \Im(t)} \\ \dfrac{\partial^2 \psi}{\partial \Im(t) \partial \Re(t)} & \dfrac{\partial^2 \psi}{\partial \Im(t)^2} \end{pmatrix}$$

is symmetric and positive definite;

3. we have

$$\psi(e_1) = 0, \quad \nabla\psi(e_1) = \mathbb{0},$$

and

$$\lim_{\Re(t)\to 0^+} \psi(t) = +\infty, \quad \lim_{|t|\to\infty} \psi(t) = +\infty.$$

Based on the above definition, we propose the following new complex kernel function:

$$\psi(t) = -\log \Re(t) - \log(1 + \Im(t)^2) + \frac{|t|^2}{2} - \frac{1}{2}, \quad (10)$$

$$t = \Re(t) + i\Im(t) \in \mathbb{C}_+^*.$$

Its gradient and Hessian are

$$\begin{aligned} \frac{\partial\psi}{\partial\Re(t)} &= \Re(t) - \frac{1}{\Re(t)}, \\ \frac{\partial\psi}{\partial\Im(t)} &= \Im(t)\left(1 - \frac{2}{1+\Im(t)^2}\right), \end{aligned} \quad (11)$$

$$\begin{aligned} \frac{\partial^2\psi}{\partial\Re(t)^2} &= 1 + \frac{1}{\Re(t)^2}, \\ \frac{\partial^2\psi}{\partial\Im(t)^2} &= \frac{\Im(t)^4 + 4\Im(t)^2 - 1}{(1+\Im(t)^2)^2}, \\ \frac{\partial^2\psi}{\partial\Re(t)\partial\Im(t)} &= 0. \end{aligned} \quad (12)$$

The third-order derivatives are

$$\begin{aligned} \frac{\partial^3\psi}{\partial\Re(t)^3} &= -\frac{2}{\Re(t)^3}, \\ \frac{\partial^3\psi}{\partial\Im(t)^3} &= \frac{-4\Im(t)(\Im(t)^2 - 3)}{(1+\Im(t)^2)^3}. \end{aligned} \quad (13)$$

All mixed third-order terms vanish. The function ψ clearly satisfies Definition 1.

Remark 2. When $t \in \mathbb{R}_{++}$ (i.e., $\Im(t) = 0$), ψ reduces to the classical self-regular kernel

$$\psi(t) = -\log t + \tfrac{1}{2}t^2 - \tfrac{1}{2},$$

which is known to induce polynomial iteration complexity in real interior-point methods.

Remark 3. (*Laouar* et al.*, 2026*) The scaling vectors $\mathbf{v}$, $\mathbf{dz}$ and $\mathbf{ds}$ are defined by

$$\begin{aligned} \mathbf{v}_i &= \sqrt{-\frac{g_i(\mathbf{z})\mathbf{s}_i}{\mu}}, && i = 1, \dots, m, \\ (d\mathbf{z})_i &= \frac{\mathbf{v}_i}{g_i(\mathbf{z})}(\nabla_{\mathbf{z}} g_i(\mathbf{z}))^H \Delta\mathbf{z}, && i = 1, \dots, m, \\ (d\mathbf{s})_i &= (\mathbf{v} \oslash \mathbf{s} \odot \Delta s)_i, && i = 1, \dots, m, \end{aligned} \quad (14)$$

where $\oslash$ denotes the Hadamard division. We also set

$$d\mathbf{z} = \big((d\mathbf{z})_1, \dots, (d\mathbf{z})_m\big)^\top, \quad d\mathbf{s} = \big((d\mathbf{s})_1, \dots, (d\mathbf{s})_m\big)^\top,$$

where $\mathbf{s}_i > 0$, $g_i(\mathbf{z}) < 0$, and $\mu > 0$. Therefore, $\mathbf{v}_i \in \mathbb{R}_{++}$ for all i.

Using the notations in (8) and (9) as well as Definition 1, the system (6) can be rewritten as

$$\begin{cases} M\Delta\mathbf{z} = R, \\ d\mathbf{z} + d\mathbf{s} = -\nabla\Psi(\mathbf{v}), \end{cases} \quad (15)$$

where

$$\Psi(\mathbf{v}) := \sum_{i=1}^{m} \psi(\mathbf{v}_i)$$

is the total barrier function and

$$\nabla\Psi(\mathbf{v}) := (\nabla\psi(\mathbf{v}_1), \dots, \nabla\psi(\mathbf{v}_m))^\top.$$

Hence, for such real-valued components, the gradient simplifies to

$$\nabla\psi(\mathbf{v}_i) = \begin{pmatrix} \mathbf{v}_i - 1/\mathbf{v}_i \\ 0 \end{pmatrix}.$$

Remark 4. Since $\nabla^2_{\mathbf{zz}}\mathcal{L}(\mathbf{z}, \mathbf{s})$ is positive definite by convexity and $G(\mathbf{z})^{-1}S = \operatorname{diag}(\mathbf{s}_i/g_i(\mathbf{z}))$ is negative definite (because $\mathbf{s}_i > 0$ and $g_i(\mathbf{z}) < 0$), the matrix

$$-\nabla_{\mathbf{z}} g(\mathbf{z})^\top G(\mathbf{z})^{-1} S \nabla_{\mathbf{z}} g(\mathbf{z})$$

is positive semidefinite. Therefore, the matrix M is positive definite, which ensures the uniqueness of the solution to (15).

Definition 2. (*New proximity measure*) Let $\mathbf{v} = (\mathbf{v}_1, \dots, \mathbf{v}_m) \in (\mathbb{C}_+^*)^m$ be a vector whose components $\mathbf{v}_i$ belong to the domain of ψ, as specified in (10). Building upon Definition 1, we define the *normalized proximity measure* $\delta(\mathbf{v})$ for the system (15) as follows:

$$\begin{aligned} \delta(\mathbf{v}) &:= \frac{1}{2}\left|\nabla\Psi(\mathbf{v})\right| = \frac{1}{2}|d\mathbf{z} + d\mathbf{s}| \\ &= \frac{1}{2}\sqrt{\sum_{i=1}^{m} \|\nabla\psi(\mathbf{v}_i)\|^2}. \end{aligned} \quad (16)$$

In Algorithm 1, the proximity measure (16) simplifies since $\mathbf{v}_i \in \mathbb{R}_{++}$ (as defined in (14)), yielding

$$\delta(\mathbf{v}) = \frac{1}{2}\sqrt{\sum_{i=1}^{m} \left(\mathbf{v}_i - \frac{1}{\mathbf{v}_i}\right)^2}. \quad (17)$$

This coincides with the proximity function used in classical self-regular kernel-based interior-point methods, allowing us to leverage well-established real-variable theory for the complexity analysis.

For a given threshold $\tau > 0$, an iterate $(\mathbf{z}, \mathbf{s}) \in \mathcal{F}^0$ (and its corresponding scaling vector $\mathbf{v}$) is said to be

- τ-*central* if $\delta(\mathbf{v}) \leq \tau$;
- *strictly centered* (or a μ-center) if $\delta(\mathbf{v}) = 0$, which is equivalent to $\nabla\psi(\mathbf{v}_i) = \mathbb{0}$ for all $i = 1, \ldots, m$.

Remark 5. The design of the complex kernel function ψ extends classical self-regular kernels to the complex domain while maintaining key theoretical properties:

- *Convexity:* $\nabla^2\psi(t) \succ 0$ for all $t \in \mathbb{C}_+^*$.
- *Barrier property:* $\psi(t) \to +\infty$ as $\Re(t) \to 0^+$ or $|t| \to \infty$.
- *Real restriction:* When restricted to $\mathbb{R}_{++}$, ψ recovers the classical self-regular kernel.

2.2.1. Central path following algorithm. The algorithm employs a two-level scheme to compute an approximate optimal solution $(\mathbf{z}^*, \mathbf{s}^*)$ by following the central path. The main steps are described below.

1. *Outer loop.* Starting from an initial value $\mu^0 > 0$, the barrier parameter is updated according to

$$\mu^{k+1} = (1-\theta)\,\mu^k, \qquad \theta \in (0,1),$$

until the stopping criterion

$$m\,\mu^k < \varepsilon$$

is satisfied, where n is the dimension of the primal variable $\mathbf{z}$ and $\varepsilon > 0$ is the prescribed accuracy. For each value μ^k, the inner loop computes an approximately τ-central point for this barrier parameter (see Definition 2).

2. *Inner loop.* At each inner iteration j, starting from a strictly feasible point $(\mathbf{z}^j, \mathbf{s}^j) \in \mathcal{F}^0$, we compute a primal-dual Newton direction $(\Delta\mathbf{z}^j, \Delta\mathbf{s}^j)$ by solving the linear system (6) with $\mu = \mu^k$. The step size α^j is then chosen, using an Armijo-type backtracking line search on the kernel-based potential function (Armijo, 1966; Nocedal and Wright, 2006; Wächter and Biegler, 2006), so that the next iterate remains strictly feasible and sufficiently away from the boundary.

To this end, we determine the maximum admissible step lengths for the primal and dual variables separately. The maximum primal step $\alpha_{\mathbf{z}}^j$ is the largest step that preserves the linearized feasibility of the inequality constraints, i.e.,

$$g(\mathbf{z}^j + \alpha\,\alpha_{\mathbf{z}}^j\mathbf{z}^j) \approx g(\mathbf{z}^j) + \alpha_{\mathbf{z}}^j\,\nabla g(\mathbf{z}^j)\,\Delta\mathbf{z}^j < 0.$$

It is obtained as

$$\alpha_{\mathbf{z}}^j := \min_{\left\{i \,|\, \left(\nabla g(\mathbf{z}^j)\,\Delta\mathbf{z}^j\right)_i > 0\right\}} \left(\frac{-\,g_i(\mathbf{z}^j)}{\left(\nabla g(\mathbf{z}^j)\,\Delta\mathbf{z}^j\right)_i} \right),$$

with the convention $\alpha_{\mathbf{z}}^j = +\infty$ if the index set is empty (i.e., if all constraints are moving inward or remain inactive).

Similarly, the maximum dual step $\alpha_{\mathbf{s}}^j$ is computed to enforce the nonnegativity of the dual variables:

$$\mathbf{s}^{j+1} = \mathbf{s}^j + \alpha_{\mathbf{s}}^j\,\Delta\mathbf{s}^j > 0$$

and is determined by

$$\alpha_{\mathbf{s}}^j := \min_{\{i \,|\, (\Delta\mathbf{s}^j)_i < 0\}} \left(\frac{-\,\mathbf{s}_i^j}{(\Delta\mathbf{s}^j)_i} \right),$$

with $\alpha_{\mathbf{s}}^j = +\infty$ if all components of Δs^j are nonnegative.

To satisfy both feasibility requirements and keep a safe distance from the boundary, we use a fraction-to-the-boundary rule and define the actual step size as

$$\alpha^j := \eta\,\min\{1, \alpha_{\mathbf{z}}^j, \alpha_{\mathbf{s}}^j\}, \qquad \eta \in (0,1),$$

where η is a safety parameter that controls the minimum distance maintained from the boundary of the feasible region. Typical values are $\eta \in [0.99, 0.9999]$.

The new iterate is then updated by

$$\mathbf{z}^{j+1} = \mathbf{z}^j + \alpha^j\,\Delta\mathbf{z}^j, \qquad \mathbf{s}^{j+1} = \mathbf{s}^j + \alpha^j\,\Delta\mathbf{s}^j.$$

The inner loop terminates once the proximity measure $\delta(\mathbf{v}^j)$ (see Definition 2) falls below the prescribed centrality threshold $\tau > 0$.

In summary, this section successfully lays the theoretical foundation for the proposed primal-dual interior-point algorithm using a novel complex kernel function. By defining the problem space, detailing the application of Newton's method to the system, and introducing the new complex-kernel function, we establish the computational machinery necessary for convergence. The properties of the new kernel, which extend classical self-regular kernels to the complex domain, are rigorously defined. Crucially, the formulation of the central path following algorithm, structured into outer and inner loops, provides a clear procedural mechanism for achieving an approximate optimal solution. The subsequent sections will now leverage this established framework to conduct a comprehensive theoretical analysis and evaluate the numerical performance of the proposed method.

3. Theoretical analysis

A rigorous theoretical foundation is essential to validate the efficiency and reliability of any new interior-point methodology. This section is dedicated to providing a comprehensive theoretical analysis of the proposed primal-dual interior-point algorithm equipped with the

Algorithm 1. Proposed primal-dual interior-point algorithm.

Require: Accuracy parameter $\varepsilon > 0$, centrality threshold $\tau > 0$, step-size safety factor $\eta \in (0,1)$, barrier reduction parameter $\theta \in (0,1)$.

1: **Initialization**
2: Set $k = 0$ and select a strictly feasible initial point $(\mathbf{z}^0, s^0) \in \mathcal{F}^0$.
3: Set $\mu^0 = -\frac{(s^0)^\top g(\mathbf{z}^0)}{m} > 0$.
4: **while** $m\mu^k \geq \varepsilon$ **do**
5: Put $(\mathbf{z}, s) = (\mathbf{z}^k, s^k)$.
6: Set $\mu = (1-\theta)\mu^k$. (Target barrier parameter for this outer iteration)
7: Compute the scaling vector: $v_i = \sqrt{\frac{-s_i g_i(\mathbf{z})}{\mu}},\ i = 1, \ldots, m$.
8: **while** $\delta(v) > \tau$ **do**
9: Solve the Newton system (6) for $(\Delta\mathbf{z}, \Delta s)$.
10: Compute maximum feasible step sizes $\alpha_{\mathbf{z}}$ and $\alpha_{\mathbf{s}}$:

$$\alpha_{\mathbf{z}} = \min\left\{i \mid \left(\nabla g(\mathbf{z})\,\Delta\mathbf{z}\right)_i > 0\right\} \left(\frac{-g_i(\mathbf{z})}{\left(\nabla g(\mathbf{z})\,\Delta\mathbf{z}\right)_i}\right),$$

$$\alpha_{\mathbf{s}} = \min\left\{i \mid (\Delta\mathbf{s})_i < 0\right\} \left(\frac{-s_i}{(\Delta\mathbf{s})_i}\right).$$

11: Set $\alpha = \eta \min\{1, \alpha_{\mathbf{z}}, \alpha_{\mathbf{s}}\}$, $\mathbf{z} = \mathbf{z} + \alpha\,\Delta\mathbf{z}$, and $\mathbf{s} = \mathbf{s} + \alpha\,\Delta\mathbf{s}$.
12: Update the scaling vector: $\mathbf{v}_i = \sqrt{\frac{-\mathbf{s}_i g_i(\mathbf{z})}{\mu}},\ i = 1, \ldots, m$.
13: **end while**
14: Put $(\mathbf{z}^{k+1}, \mathbf{s}^{k+1}) = (\mathbf{z}, \mathbf{s})$ and $\mu^{k+1} = \mu$.
15: Set $k = k + 1$.
16: **end while**

complex kernel function (10), specifically focusing on the sandwich inequalities (20) for the complex kernel potential Ψ and their relation to L_2-type norms, which are critical for deriving subsequent bounds. Following this, Section 3.2 examines the growth rate of the barrier parameter update, a crucial factor influencing the overall complexity.

We identify $\mathbb{C} \simeq \mathbb{R}^2$ via

$$t = \Re(t) + \mathrm{i}\Im(t) \;\longleftrightarrow\; (\Re(t), \Im(t))^\top,\ \ \Re(t) > 0,$$

and denote by $\|\cdot\|$ the Euclidean norm. The total barrier function is

$$\Psi(v) := \sum_{i=1}^{m} \psi(\mathbf{v}_i),$$

$$\nabla\Psi(\mathbf{v}) = \left(\nabla\psi(\mathbf{v}_1), \ldots, \nabla\psi(\mathbf{v}_m)\right)^\top,$$

$$\delta(\mathbf{v}) := \tfrac{1}{2}\|\nabla\Psi(\mathbf{v})\|.$$

3.1. Convexity and kernel properties.

Theorem 1. *For all $t = \Re(t) + \mathrm{i}\Im(t) \in \mathbb{C}_+^* = \{t \in \mathbb{C} \mid \Re(t) > 0\}$, the Hessian of the complex kernel ψ satisfies the following:*

(i) $\inf_{t\in\mathbb{C}_+^*} \lambda_{\min}(\nabla^2\psi(t)) = -1$, *attained at* $t = e_1 = (1,0)$;

(ii) $\nabla^2\psi(t) \succeq 0$ *if and only if* $\Im(t)^2 \geq \sqrt{5} - 2$, *and* $\nabla^2\psi(t) \succeq I_2$ *if and only if* $|\Im(t)| \geq 1$;

(iii) $\lim_{\Re(t)\to+\infty} \nabla^2\psi(t) = I_2$.

Proof. From (12), $\nabla^2\psi(t)$ is diagonal with eigenvalues

$$\lambda_1(t) = 1 + \frac{1}{\Re(t)^2} \geq 1$$

and

$$\lambda_2(t) = \frac{\Im(t)^4 + 4\Im(t)^2 - 1}{(1 + \Im(t)^2)^2}.$$

(i) For

$$\phi(y) := \frac{y^4 + 4y^2 - 1}{(1+y^2)^2},$$

we have

$$\phi'(y) = \frac{4y(3 - y^2)}{(1+y^2)^3}$$

with critical points at $y = 0, \pm\sqrt{3}$. Since $\phi(0) = -1$ and $\lim_{|y|\to\infty} \phi(y) = 1$, we obtain $\inf \lambda_2 = -1$.

(ii) Setting $u = \Im(t)^2$, the condition $u^2 + 4u - 1 \geq 0$ yields $u \geq \sqrt{5} - 2$ (positive root of $u^2 + 4u - 1 = 0$). For $\lambda_2(t) \geq 1$, we get $2\Im(t)^2 \geq 2$.

(iii) Direct from $\lambda_1(t) \to 1$ and $\lambda_2(t) \to 1$ as $\Re(t) \to \infty$. The convergence of $\nabla^2\psi(t)$ to the identity matrix I_2 follows from the fact that the eigenvalues $\lambda_1(t)$ and $\lambda_2(t)$ both tend to 1 as $t \to +\infty$. ∎

Lemma 1. *For all $t \in \mathbb{C}_+^*$,*

$$-\tfrac{1}{2}\|t - e_1\|^2 \leq \psi(t) \leq \tfrac{1}{2}\|\nabla\psi(t)\|^2. \qquad (18)$$

Furthermore, there exists $L > 0$ such that

$$\psi(t) \leq \tfrac{L}{2}\|t - e_1\|^2 \quad \textit{whenever } \|t - e_1\| \leq 1. \qquad (19)$$

Proof.
1. *Lower bound.* By Taylor expansion around e_1 with $\psi(e_1) = \nabla\psi(e_1) = 0$, we have

$$\psi(t) = \int_0^1 (1-\lambda)(t-e_1)^\top \nabla^2\psi(e_1 + \lambda(t-e_1))(t-e_1)\,\mathrm{d}\lambda.$$

From Theorem 1(i), $\lambda_{\min}(\nabla^2\psi(w)) \geq -1$ for all $w \in \mathbb{C}_+^*$, yielding

$$\psi(t) \geq -\int_0^1 (1-\lambda)\|t-e_1\|^2\,\mathrm{d}\lambda = -\tfrac{1}{2}\|t-e_1\|^2.$$

2. *Upper bound.* Let us define

$$\begin{aligned} h(\Re(t),\Im(t)) &:= \tfrac{1}{2}\|\nabla\psi(t)\|^2 - \psi(t) \\ &= h_1(\Re(t)) + h_2(\Im(t)), \end{aligned}$$

where

$$h_1(\Re(t)) = \log\Re(t) - \tfrac{1}{2} + \tfrac{1}{2\Re(t)^2},$$

$$h_2(\Im(t)) = \tfrac{\Im(t)^2(\Im(t)^2-1)^2}{2(1+\Im(t)^2)^2} + \log(1+\Im(t)^2) - \tfrac{1}{2}\Im(t)^2.$$

For h_1 we have

$$h_1(1) = 0$$

and

$$h_1'(\Re(t)) = \frac{1}{\Re(t)} - \frac{1}{\Re(t)^3} = \frac{\Re(t)^2-1}{\Re(t)^3},$$

which changes sign at $\Re(t) = 1$. Thus $\Re(t) = 1$ is a global minimum, giving $h_1(\Re(t)) \geq 0$ with equality if and only if $\Re(t) = 1$.

For h_2 we have $h_2(0) = 0$. Since h_2 is even, consider $\Im(t) > 0$. We can verify that h_2 attains its minimum at $\Im(t) = 0$ by noting that, for large $|\Im(t)|$,

$$\begin{aligned} h_2(\Im(t)) &\sim \tfrac{\Im(t)^2}{2} - \tfrac{\Im(t)^2}{2} + 2\log|\Im(t)| \\ &= 2\log|\Im(t)| \to +\infty. \end{aligned}$$

By continuity and the fact that $h_2(0) = 0$, we conclude that $h_2(\Im(t)) \geq 0$ with equality if and only if $\Im(t) = 0$.

Therefore, $h(\Re(t),\Im(t)) \geq 0$ with equality if and only if $t = e_1$, proving that $\psi(t) \leq \frac{1}{2}\|\nabla\psi(t)\|^2$.

3. *Local bound.* The Hessian $\nabla^2\psi$ is continuous on the compact set $K := \{w \in \mathbb{C}_+^* : \|w-e_1\| \leq 1\}$. Thus there exists $L > 0$ such that $\|\nabla^2\psi(w)\|_{\mathrm{op}} \leq L$ for all $w \in K$. The Taylor expansion then gives

$$\psi(t) \leq \int_0^1 (1-\lambda)L\|t-e_1\|^2\,\mathrm{d}\lambda = \tfrac{L}{2}\|t-e_1\|^2.$$

■

An immediate consequence for the composite barrier function is the following.

Corollary 1. *Let* $\mathbf{v} = (\mathbf{v}_1,\dots,\mathbf{v}_m)^\top \in (\mathbb{C}_+^*)^m$ *and let* $\mathbf{1} := (1,\dots,1)^\top \in \mathbb{R}^m$. *Then*

$$\tfrac{1}{2}\|\mathbf{v}-\mathbf{1}\|^2 \leq \Psi(\mathbf{v}) \leq \tfrac{1}{2}\|\nabla\Psi(\mathbf{v})\|^2 = 2\delta(\mathbf{v})^2, \quad (20)$$

where

$$\Psi(\mathbf{v}) := \sum_{i=1}^m \psi(\mathbf{v}_i), \qquad \delta(\mathbf{v}) := \tfrac{1}{2}\|\nabla\Psi(\mathbf{v})\|.$$

3.2. Growth after barrier parameter update.

Lemma 2. *Let* $(\mathbf{z},\mathbf{s})$ *be a* μ*-center* ($\delta(\mathbf{v}) = 0$). *After the barrier parameter update* $\mu^+ = (1-\theta)\mu$ *with* $0 < \theta \leq 1/2$, *the new scaling vector* $\mathbf{v}^+$ *satisfies*

$$\delta(\mathbf{v}^+) = \frac{\theta}{2\sqrt{1-\theta}} \leq \frac{\theta}{2} \leq \frac{1}{4},$$

and

$$\Psi(\mathbf{v}^+) = m\left(\frac{\theta^2}{2(1-\theta)} + \theta\right) \leq \frac{9}{8}\,m\,\theta^2 \leq 2\,m\,\theta.$$

Proof. After the update, $v^+ = 1/\sqrt{1-\theta}\mathbf{1}$. Let $t = 1/\sqrt{1-\theta}$. Then

$$\delta(\mathbf{v}^+) = \frac{1}{2}\Big(t - \frac{1}{t}\Big) = \frac{\theta}{2\sqrt{1-\theta}},$$

which is at most $\theta/2 \leq 1/4$ for $\theta \leq 1/2$.

The proximity function is

$$\Psi(\mathbf{v}^+) = m\,\psi(t), \quad \psi(t) = \frac{t^2-1}{2} - \log t.$$

It is well known (see Peng *et al.*, 2002) that

$$\psi\left(\frac{1}{\sqrt{1-\theta}}\right) = \frac{\theta^2}{2(1-\theta)} + \theta \leq \frac{9}{8}\,\theta^2 \quad \forall\,\theta \in [0,1/2].$$

The desired bounds on $\Psi(\mathbf{v}^+)$ follow immediately. ■

3.3. Decrease of the barrier function during inner iterations.

Lemma 3. (Decrease of the proximity function) *Let* (z,s) *be strictly feasible with* $\delta(\mathbf{v}) \leq 1/2$. *Let also* $(\Delta z,\Delta s)$ *be the primal-dual Newton step for the current* $\mu > 0$, *and let the step size* α *be chosen as*

$$\alpha = \eta\min\{1,\alpha_z,\alpha_s\}, \quad \textit{with } \eta \in (0,1), \quad (21)$$

where α_z *and* α_s *are the maximum step sizes to the boundary in the primal and dual spaces, respectively. Then there exists a constant* $\kappa(\eta) > 0$ *such that the proximity function satisfies*

$$\Psi(\mathbf{v}^+) \leq \Psi(\mathbf{v}) - \kappa(\eta)\,\delta(\mathbf{v})^2. \quad (22)$$

In our numerical implementation (Algorithm 1), we use $\eta = 0.95$.

Proof. Let d_z and d_s be the scaled Newton directions in the v-space, which are related to the primal-dual Newton steps $(\Delta z,\Delta s)$ via the scaling defined in the algorithm. By the definition of the Newton system in the v-space, these directions satisfy

$$d_z + d_s = -\nabla\Psi(\mathbf{v}). \quad (23)$$

The proximity measure is defined as $\delta(\mathbf{v}) = \frac{1}{2}\|d_z+d_s\| = \frac{1}{2}\|\nabla\Psi(\mathbf{v})\|$.

Since we assume $\delta(\mathbf{v}) \leq 1/2$, a fundamental property of self-regular kernel functions is that the unit step ($\alpha = 1$) is strictly feasible. This implies that the distances to the boundaries satisfy $\alpha_z \geq 1$ and $\alpha_s \geq 1$. Consequently, the step size rule (21) simplifies to

$$\alpha = \eta \cdot 1 = \eta.$$

To evaluate the decrease, we use a Taylor expansion of Ψ along the Newton direction. The self-regularity of the kernel implies that the Hessian $\nabla^2\Psi$ is bounded by a constant $L > 0$ in the region $\delta(\mathbf{v}) \leq 1/2$. Thus

$$\Psi(\mathbf{v}^+) - \Psi(\mathbf{v}) \leq \alpha\nabla\Psi(\mathbf{v})^T(d_z+d_s) + \frac{L}{2}\alpha^2\|d_z+d_s\|^2.$$

Using (23), we substitute $\nabla\Psi(\mathbf{v}) = -(d_z+d_s)$ and $\|d_z+d_s\|^2 = 4\delta(\mathbf{v})^2$:

$$\begin{aligned}\Psi(\mathbf{v}^+) - \Psi(\mathbf{v}) &\leq -\alpha\|d_z+d_s\|^2 + \frac{L}{2}\alpha^2\|d_z+d_s\|^2 \\ &= -4\alpha\delta(\mathbf{v})^2 + 2L\alpha^2\delta(\mathbf{v})^2 \\ &= -2\alpha(2-L\alpha)\delta(\mathbf{v})^2.\end{aligned}$$

Substituting $\alpha = \eta$, we obtain

$$\Psi(\mathbf{v}^+) - \Psi(\mathbf{v}) \leq -2\eta(2-L\eta)\delta(\mathbf{v})^2.$$

By setting $\kappa(\eta) = 2\eta(2-L\eta)$, and provided η is chosen such that $L\eta < 2$ (which is satisfied for standard self-regular kernels and $\eta = 0.95$), the required inequality (22) holds. ■

3.4. Iteration complexity.

Theorem 2. *Algorithm 1 with the parameters*

$$\tau = \frac{1}{4}, \quad \eta = 0.99, \quad \theta = \frac{1}{6\sqrt{m}}$$

finds a solution satisfying $m\mu \leq \varepsilon$ in at most

$$O\big(m\log(m\mu^0/\varepsilon)\big)$$

Newton iterations. This matches the best known large-update complexity for classical real self-regular primal-dual methods.

Proof.

1. After each μ-update, Lemma 2 gives

$$\Psi(\mathbf{v}^+) \leq \frac{9}{8}m\theta^2 = \frac{1}{32} < 1$$

and

$$\delta(\mathbf{v}^+) \leq \frac{1}{24} < \tau.$$

2. Each centering step with $\delta(\mathbf{v}) \leq 1/2$ reduces Ψ by at least $\rho > 0$ (Lemma 3). Hence at most $O(\sqrt{m})$ centering steps are needed per outer iteration.

3. The barrier parameter satisfies

$$\mu^k \leq \mu^0 e^{-k\theta},$$

so $m\mu^k \leq \varepsilon$ requires

$$k \geq 6\sqrt{m}\log(m\mu^0/\varepsilon)$$

outer iterations.

The total complexity is

$$O(\sqrt{m}) \times O(\sqrt{m}\log(m\mu^0/\varepsilon)) = O(m\log(m\mu^0/\varepsilon)).$$

■

Remark 6. The choice $\theta = O(1/\sqrt{m})$ corresponds to the large-update regime: practically very efficient while keeping the favourable $O(m\log\varepsilon^{-1})$ worst-case bound of self-regular methods.

3.5. Conclusion of the theoretical analysis. In conclusion, the theoretical analysis presented in this section successfully establishes the convergence guarantees for the proposed interior-point method. By rigorously proving key properties such as the growth bound for the barrier parameter update (Lemma 2) and the sufficient decrease achieved per Newton step (Lemma 3), we were able to quantify the overall iteration complexity of the algorithm. The result, $N_{total} = O(m\log(m\mu^0/\varepsilon))$, demonstrates that the algorithm achieves a competitive polynomial time complexity, which is consistent with the best known complexity results for primal-dual interior-point methods. This theoretical assurance confirms the potential efficiency of the new complex kernel function approach for solving convex optimization problems.

4. Numerical implementation

While theoretical complexity bounds are necessary, the true utility of any optimization algorithm is demonstrated through its practical computational efficiency. This section is concerned with numerically evaluating the performance of the primal-dual interior-point method utilizing the proposed complex kernel. The primary goal is to provide a quantitative assessment of its efficiency against the classical kernel function across a diverse set of optimization instances, including those from the complex domain. We outline the specific experimental setup, detailing the parameter choices and the method for measuring performance metrics (e.g., inner iterations and CPU time). The results, compiled across test instances of increasing problem dimension, will empirically validate the theoretical advancements and highlight the operational benefits of the complex kernel function.

4.1. Experimental setup. The experiments were carried out on laptop with an Intel Core i5-1135G7 11th generation processor, 16 GB RAM, and a Ubuntu 24.04.3 LTS operating system, and the codes are implemented in Python 3.13.7 with NumPy. Complex variables were processed via real decomposition $z = x + iy$. Execution times are measured using `timeit.repeat()` with a warm-up run prior to timing, and averaged over $N = 3$ independent runs reporting mean $\pm$ standard deviation. The stopping criteria are

$$\begin{aligned} m\mu &\leq 10^{-8}, \\ \max\left(\|\nabla_z \mathcal{L}\|_\infty, \|g(z) \odot s\|_\infty\right) &\leq 10^{-6}. \end{aligned} \tag{24}$$

4.2. Kernel functions and test problems. We compare four kernel functions in our numerical experiments, as summarized in Table 1.

For each kernel, results are reported for three values of the barrier reduction parameter: $\theta \in \{0.50, 0.75, 0.95\}$. The reported metrics are the total number of inner iterations and the average CPU time in seconds.

The following examples illustrate the performance of both kernels across diverse problem types, ranging from real quadratic programming to complex nonlinear optimization.

Example 1. We consider the following complex optimization problem ($z_i \in \mathbb{C}$):

$$\begin{cases} \min\limits_{z\in\mathbb{C}^n} & \sum\limits_{i=1}^{n}(|z_i|^4 + |z_i|^2) \\ \text{s.t.} & |z_i|^2 \leq 9, \ \ i = 1, \ldots, m. \end{cases} \tag{25}$$

Example 2. We consider the following real quadratic programming problem over $\mathbb{R}^n$:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} & \frac{1}{2}x^T Q x + b^T x \\ \text{s.t.} & x_i \geq 0, \ i = 1, \ldots, m, \end{cases} \tag{26}$$

where $Q = \text{diag}(0.5 + (i+1)/n)$ and $b_i = 0.1\cos(2\pi(i+1)/n)$ for $i = 1, \ldots, n$.

Example 3. We consider the following real optimization problem ($x_i \in \mathbb{R}$):

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} & \sum\limits_{i=1}^{n}\left(x_i^4 + 0.5x_i^2 + \exp(0.1x_i)\right) \\ \text{s.t.} & x_i^2 + 0.1x_i^3 - \log(1 + x_i^2) - 1 \leq 0, \\ & \qquad i = 1, \ldots, m. \end{cases} \tag{27}$$

Example 4. We consider the following problem:

$$\begin{cases} \min\limits_{z\in\mathbb{C}^n} & \sum\limits_{i=1}^{n}\left(|z_i|^2 - \log(1 + \Re(z_i))\right) \\ \text{s.t.} & (\Re(z_i))^2 + 2(\Im(z_i))^2 - 1 \leq 0, \ \ i = 1, \ldots, m. \end{cases} \tag{28}$$

Example 5. We consider the following problem:

$$\begin{cases} \min\limits_{z\in\mathbb{C}^n} & \sum\limits_{i=1}^{n}|z_i|^4 - 0.05\sum\limits_{i=1}^{n}|z_i|^2 + 0.05\sum\limits_{i=1}^{n-1}\Re(z_i\overline{z_{i+1}}) \\ \text{s.t.} & |z_i| \leq 3, \ \ i = 1, \ldots, m. \end{cases} \tag{29}$$

4.3. Comprehensive performance analysis. The numerical results obtained from Examples 1–5 provide a multifaceted view of the performance of the primal-dual interior-point method based on the proposed complex kernel. This section analyzes these results across four key dimensions: dimensional stability, convergence efficiency, numerical robustness, and algebraic complexity.

4.3.1. Dimensional stability and affine invariance. A fundamental requirement for high-quality IPMs is that the number of outer iterations remains relatively constant regardless of the problem dimension n.

- *Observation.* In Examples 1 and 2, the number of outer iterations for the complex kernel ψ_c (at $\theta = 0.50$) remains strictly stable at 26, even as n scales from 2,000 to 20,000.
- *Analysis.* This confirms that ψ_c successfully preserves the affine invariance property. The convergence path is governed by the underlying geometric structure of the optimization problem rather than its scale, ensuring excellent scalability for large-scale instances.

4.3.2. Algorithmic efficiency in separable problems. Across Examples 1, 2, and 3, the complex kernel consistently outperforms both classical kernels (ψ_1, ψ_2, ψ_3) and the reference IPOPT solver.

- *Iteration reduction.* In Example 3 ($n = 20000$), ψ_c requires only 10 to 14 inner iterations compared to 30 to 65 for classical kernels. This represents approximately a 75% reduction in the total effort required to maintain proximity to the central path.
- *Computational speedup.* In Example 1 ($n = 2000$), ψ_c achieves a maximum speedup of 7.15 times over IPOPT-Default.
- *Geometric correlation.* Figures 1 and 3 display regular bowl and valley landscapes. The complex structure of ψ_c allows the solver to better capture local curvature (Hessian information), resulting in a more direct search trajectory toward the optimal solution.

Table 1. Kernel functions compared in numerical experiments.

Index	Kernel definition	Reference
1	$\psi_1(v) = \frac{v^2-1}{2} - \log v$	Roos, 2006
2	$\psi_c(v) = -\log \Re(v) - \log(1 + \Im(v)^2) + \frac{\lvert v\rvert^2-1}{2}$	Definition 1
3	$\psi_3(v) = \frac{v^{p+1}-1}{p+1} + \frac{(e-1)^2}{e(e^v-1)} - \frac{e-1}{e}$ with $p \in [0,1]$	Darvay *et al.*, 2024
4	$\psi_2(v) = \frac{v^{p+1}-1}{p+1} - \log v$, $p \in [0,1]$	Zhang and Xu, 2011

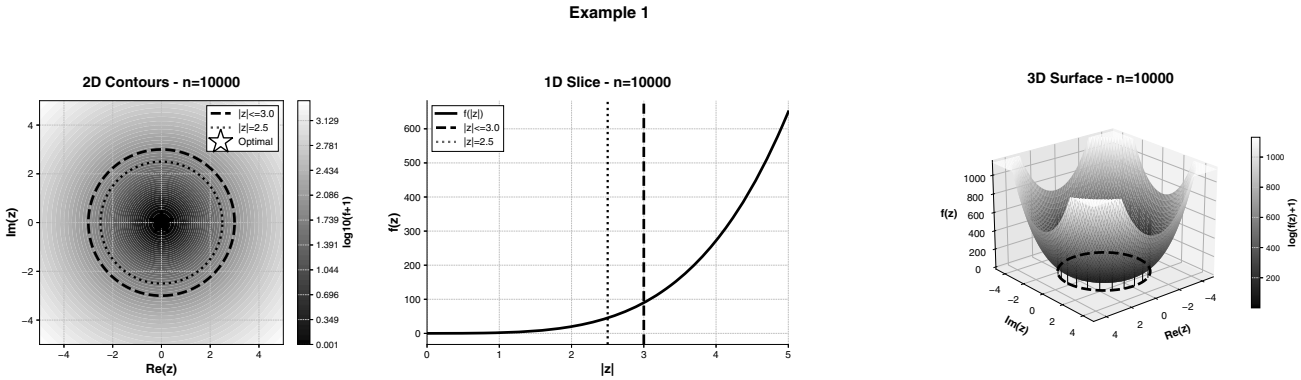

Fig. 1. Comparative visualizations for Example 1: 2D contours with radial symmetry (a), 1D convex slice (b), 3D "bowl" surface (c). The smooth geometry explains the dimensional stability observed in Table 2.

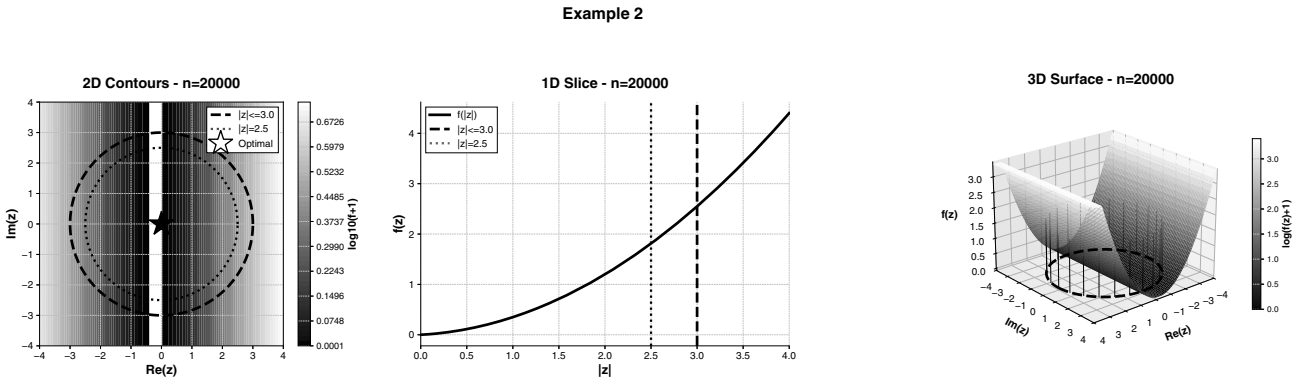

Fig. 2. Comparative visualizations for Example 2: 2D contours with radial symmetry (a), 1D convex slice (b), 3D "bowl" surface (c). The smooth geometry explains the dimensional stability observed in Table3.

4.3.3. Numerical robustness: The asymmetric stress test. Example 4 provides the most compelling evidence of the reliability of ψ_c in challenging environments.

- *Breakdown of classical kernels.* At the intermediate barrier parameter $\theta = 0.75$, all classical kernels (ψ_1, ψ_2, ψ_3) suffer a catastrophic numerical failure. As documented in Table 5, they reach the 180 s timeout limit with KKT errors exploding to 1.85×10^{14} for $n = 20000$.
- *Geometric cause.* Figure 4 identifies a "logarithmic wall" where level sets are extremely compressed. Classical kernels exhibit severe oscillations in these high-curvature zones, preventing convergence.
- *Stability of ψ_c.* Under identical conditions, ψ_c maintains a 100% success rate with a KKT error of 3.51×10^{-7}. This proves that the complex kernel provides intrinsic numerical regularization, stabilizing the algorithm even when the problem geometry is highly asymmetric.

4.3.4. Coupling paradox: Algorithmic vs. algebraic cost. Example 5 highlights a critical trade-off when

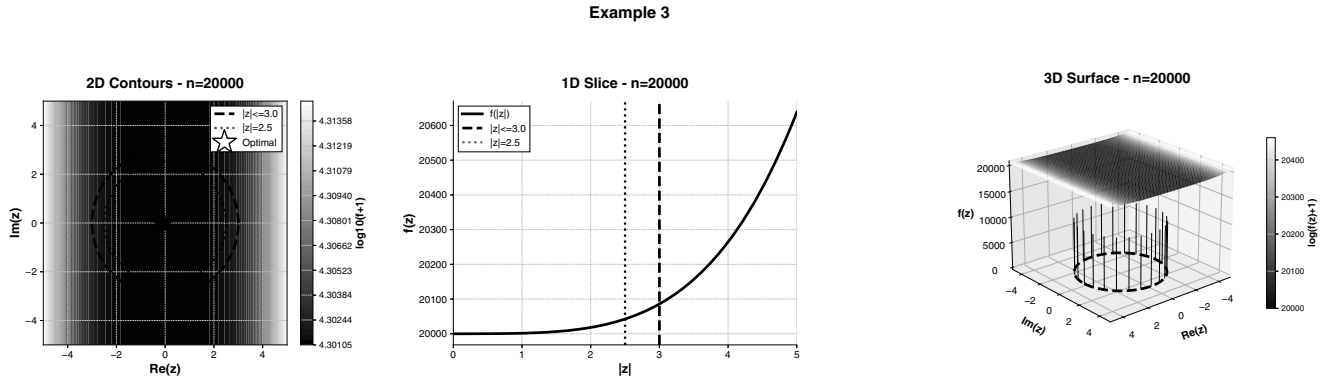

Fig. 3. Comparative visualizations for Example 3: 2D contours with radial symmetry (a), 1D convex slice (b), 3D "bowl" surface (c). The smooth geometry explains the dimensional stability observed in Table 4.

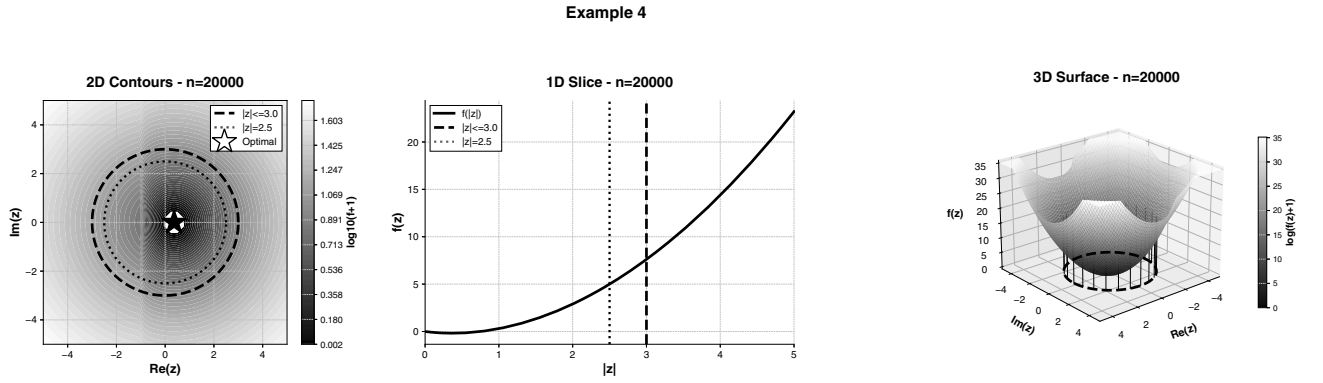

Fig. 4. Comparative visualizations for Example 4: "logarithmic wall" (compressed contours) (a), asymmetric radial growth (b), steep asymmetric bowl causing the breakdown of classical kernels seen in Table 5 (c).

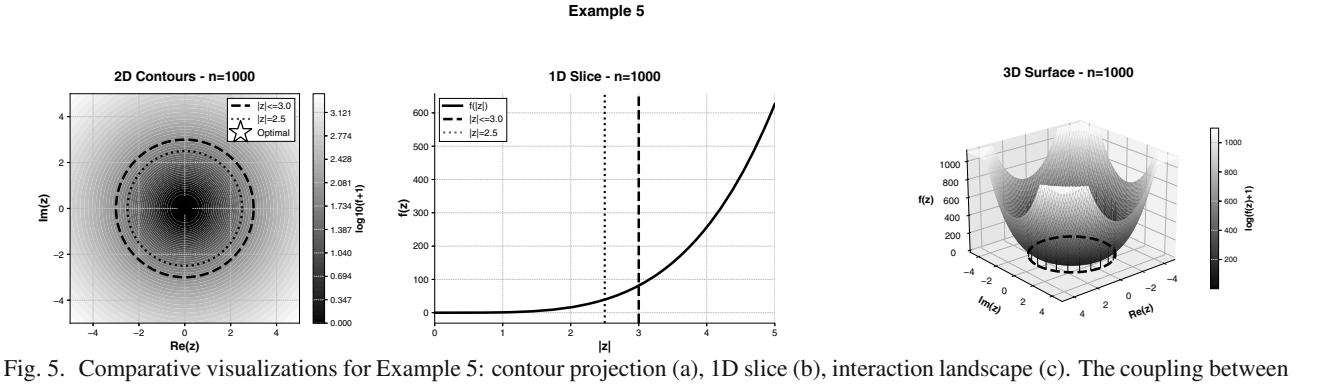

Fig. 5. Comparative visualizations for Example 5: contour projection (a), 1D slice (b), interaction landscape (c). The coupling between variables justifies the higher algebraic cost reported in Table 6.

variables are non-separable (coupled via $\Re(z_i\overline{z_{i+1}})$).

- *Efficiency gain.* ψ_c still requires fewer inner iterations than classical kernels (e.g., 27 vs. 26 for $n = 1000$ at $\theta = 0.75$).
- *Algebraic penalty.* Despite fewer steps, ψ_c is significantly slower in CPU time (275.86 s vs. 7.65 s for ψ_1).
- *Explanation.* Variable coupling generates off-diagonal terms in the Hessian. The specific structure of ψ_c (notably the $\log(1 + \Im(v)^2)$ term) increases the density of the KKT system in non-separable cases. This significantly raises the cost of linear algebra (e.g., Cholesky or LU factorization) per iteration, ultimately outweighing the reduction in total iterations.

Table 2. Comparison of kernel functions for Example 1.

Solver	θ	Out/in	Time (s)	Speed	KKT
$n = m = 2\,000$					
ψ_1	0.50	26/104	1.91 ± 0.23	0.45×	7.45e-07
	0.75	13/52	0.78 ± 0.02	1.11×	7.45e-07
	0.95	6/30	0.44 ± 0.02	1.98×	7.81e-07
ψ_c	0.50	26/25	0.44 ± 0.00	1.98×	7.86e-07
	0.75	13/12	0.21 ± 0.01	4.16×	8.85e-07
	0.95	6/11	0.20 ± 0.01	4.38×	8.22e-07
ψ_2	0.50	26/104	1.54 ± 0.03	0.56×	7.45e-07
	0.75	13/52	0.79 ± 0.01	1.10×	7.45e-07
	0.95	6/30	0.46 ± 0.01	1.87×	7.81e-07
ψ_3	0.50	26/104	1.96 ± 0.25	0.44×	7.45e-07
	0.75	13/52	1.15 ± 0.03	0.75×	7.45e-07
	0.95	6/30	0.65 ± 0.01	1.34×	7.81e-07
IPOPT-Def.	–	–	0.87 ± 0.01	1.00×	6.56e-08
IPOPT-Tight	–	–	0.95 ± 0.02	0.91×	4.77e-09
IPOPT-Loose	–	–	0.83 ± 0.02	1.05×	3.88e-07
IPOPT-Mono.	–	–	1.14 ± 0.02	0.76×	2.77e-08
IPOPT-V.T.	–	–	1.06 ± 0.06	0.82×	4.54e-11
IPOPT-Fast	–	–	0.80 ± 0.01	1.09×	4.78e-07
$n = m = 5\,000$					
ψ_1	0.50	26/104	4.58 ± 0.46	0.63×	7.45e-07
	0.75	13/52	2.25 ± 0.25	1.28×	7.45e-07
	0.95	6/30	1.55 ± 0.10	1.85×	7.81e-07
ψ_c	0.50	26/25	1.40 ± 0.06	2.06×	7.86e-07
	0.75	13/12	0.67 ± 0.02	4.28×	8.85e-07
	0.95	6/11	0.58 ± 0.07	4.93×	8.22e-07
ψ_2	0.50	26/104	4.36 ± 0.08	0.66×	7.45e-07
	0.75	13/52	2.54 ± 0.23	1.13×	7.45e-07
	0.95	6/30	1.51 ± 0.05	1.91×	7.81e-07
ψ_3	0.50	26/104	4.71 ± 0.35	0.61×	7.45e-07
	0.75	13/52	2.23 ± 0.06	1.29×	7.45e-07
	0.95	6/30	1.52 ± 0.03	1.89×	7.81e-07
IPOPT-Def.	–	–	2.88 ± 0.15	1.00×	6.62e-08
IPOPT-Tight	–	–	3.02 ± 0.12	0.95×	3.97e-09
IPOPT-Loose	–	–	2.56 ± 0.10	1.12×	6.78e-07
IPOPT-Mono.	–	–	3.74 ± 0.21	0.77×	6.31e-08
IPOPT-V.T.	–	–	3.50 ± 0.06	0.82×	6.26e-11
IPOPT-Fast	–	–	2.56 ± 0.13	1.12×	4.48e-07
$n = m = 10\,000$					
ψ_1	0.50	26/104	10.86 ± 2.22	0.73×	7.45e-07
	0.75	13/52	7.78 ± 0.39	1.01×	7.45e-07
	0.95	6/30	4.08 ± 0.11	1.93×	7.81e-07
ψ_c	0.50	26/25	3.90 ± 0.11	2.02×	7.86e-07
	0.75	13/12	2.00 ± 0.08	3.94×	8.85e-07
	0.95	6/11	1.76 ± 0.11	4.48×	8.22e-07
ψ_2	0.50	26/104	15.65 ± 1.65	0.50×	7.45e-07
	0.75	13/64	8.70 ± 0.15	0.91×	7.45e-07
	0.95	6/30	4.28 ± 0.13	1.84×	7.81e-07
ψ_3	0.50	26/104	14.89 ± 0.42	0.53×	7.45e-07
	0.75	13/52	7.02 ± 0.36	1.12×	7.45e-07
	0.95	6/30	4.26 ± 0.05	1.85×	7.81e-07
IPOPT-Def.	–	–	7.88 ± 0.30	1.00×	3.14e-08
IPOPT-Tight	–	–	8.20 ± 0.52	0.96×	6.18e-09
IPOPT-Loose	–	–	7.01 ± 0.23	1.12×	7.28e-07
IPOPT-Mono.	–	–	10.53 ± 0.41	0.75×	6.06e-08
IPOPT-V.T.	–	–	9.48 ± 0.29	0.83×	3.06e-11
IPOPT-Fast	–	–	7.11 ± 0.31	1.11×	5.82e-07

Table 3. Comparison of kernel functions for Example 2.

Solver	θ	Out/in	Time (s)	Speed	KKT
$n = m = 5\,000$					
ψ_1	0.50	26/119	2.21 ± 0.08	0.42×	7.45e-07
	0.75	13/67	1.21 ± 0.09	0.76×	7.45e-07
	0.95	6/37	0.60 ± 0.01	1.54×	7.81e-07
ψ_c	0.50	26/28	0.46 ± 0.01	2.00×	9.72e-07
	0.75	15/21	0.34 ± 0.01	2.70×	3.01e-07
	0.95	7/16	0.26 ± 0.01	3.56×	3.13e-07
ψ_2	0.50	26/120	1.94 ± 0.08	0.48×	7.45e-07
	0.75	13/67	1.16 ± 0.10	0.80×	7.45e-07
	0.95	6/37	0.50 ± 0.02	1.85×	7.81e-07
ψ_3	0.50	26/117	1.85 ± 0.01	0.50×	7.45e-07
	0.75	13/67	1.02 ± 0.05	0.91×	7.45e-07
	0.95	6/37	0.58 ± 0.03	1.60×	7.81e-07
IPOPT-Def.	–	–	0.93 ± 0.06	1.00×	5.69e-08
IPOPT-Tight	–	–	0.96 ± 0.03	0.97×	3.59e-09
IPOPT-Loose	–	–	0.88 ± 0.04	1.06×	5.28e-07
IPOPT-Mono.	–	–	1.26 ± 0.02	0.74×	5.50e-08
IPOPT-V.T.	–	–	1.15 ± 0.01	0.81×	8.28e-11
IPOPT-Fast	–	–	0.84 ± 0.03	1.10×	3.12e-07
$n = m = 10\,000$					
ψ_1	0.50	26/122	4.15 ± 0.13	0.40×	7.45e-07
	0.75	13/67	2.05 ± 0.24	0.81×	7.45e-07
	0.95	6/37	0.99 ± 0.05	1.68×	7.81e-07
ψ_c	0.50	26/28	0.80 ± 0.06	2.08×	9.72e-07
	0.75	15/21	0.54 ± 0.01	3.10×	3.01e-07
	0.95	7/16	0.40 ± 0.01	4.14×	3.13e-07
ψ_2	0.50	26/123	3.56 ± 0.27	0.47×	7.45e-07
	0.75	13/67	2.11 ± 0.19	0.79×	7.45e-07
	0.95	6/38	0.96 ± 0.03	1.72×	7.81e-07
ψ_3	0.50	26/121	3.21 ± 0.08	0.52×	7.45e-07
	0.75	13/67	1.80 ± 0.15	0.92×	7.45e-07
	0.95	6/37	1.71 ± 0.48	0.97×	7.81e-07
IPOPT-Def.	–	–	1.66 ± 0.11	1.00×	6.79e-08
IPOPT-Tight	–	–	1.61 ± 0.07	1.03×	5.67e-09
IPOPT-Loose	–	–	1.51 ± 0.20	1.10×	1.81e-07
IPOPT-Mono.	–	–	2.14 ± 0.17	0.78×	4.70e-08
IPOPT-V.T.	–	–	1.95 ± 0.18	0.85×	5.45e-11
IPOPT-Fast	–	–	1.54 ± 0.05	1.08×	2.24e-07
$n = m = 20\,000$					
ψ_1	0.50	26/124	8.16 ± 0.40	0.33×	7.45e-07
	0.75	13/72	4.34 ± 0.21	0.62×	7.45e-07
	0.95	6/38	1.88 ± 0.03	1.44×	7.81e-07
ψ_c	0.50	26/28	1.40 ± 0.01	1.94×	9.72e-07
	0.75	15/21	1.03 ± 0.01	2.64×	3.01e-07
	0.95	7/16	0.79 ± 0.04	3.44×	3.13e-07
ψ_2	0.50	26/125	7.02 ± 0.53	0.39×	7.45e-07
	0.75	13/72	4.09 ± 0.42	0.66×	7.45e-07
	0.95	6/38	2.26 ± 0.06	1.20×	7.81e-07
ψ_3	0.50	26/123	8.20 ± 1.02	0.33×	7.45e-07
	0.75	13/67	3.57 ± 0.14	0.76×	7.45e-07
	0.95	6/38	2.17 ± 0.05	1.25×	7.81e-07
IPOPT-Def.	–	–	2.71 ± 0.12	1.00×	6.40e-08
IPOPT-Tight	–	–	2.93 ± 0.19	0.92×	4.92e-09
IPOPT-Loose	–	–	2.53 ± 0.08	1.07×	4.75e-07
IPOPT-Mono.	–	–	3.86 ± 0.17	0.70×	2.91e-08
IPOPT-V.T.	–	–	3.55 ± 0.14	0.76×	5.33e-11
IPOPT-Fast	–	–	2.65 ± 0.13	1.02×	7.48e-07

Table 4. Comparison of kernel functions for Example 3.

Solver	θ	Out/in	Time (s)	Speed	KKT
$n = m = 5\,000$					
ψ_1	0.50	26/105	7.81 ± 0.33	0.53×	7.45e-07
	0.75	13/53	4.09 ± 0.28	1.02×	7.45e-07
	0.95	6/30	2.18 ± 0.07	1.91×	7.81e-07
ψ_c	0.50	26/26	2.09 ± 0.08	2.00×	7.86e-07
	0.75	13/14	1.19 ± 0.03	3.49×	8.85e-07
	0.95	6/10	0.75 ± 0.04	5.56×	8.22e-07
ψ_2	0.50	26/105	7.88 ± 0.17	0.53×	7.45e-07
	0.75	13/53	3.90 ± 0.38	1.07×	7.45e-07
	0.95	6/30	2.09 ± 0.06	2.00×	7.81e-07
ψ_3	0.50	26/105	7.26 ± 0.38	0.57×	7.45e-07
	0.75	13/53	3.55 ± 0.19	1.17×	7.45e-07
	0.95	6/30	1.97 ± 0.01	2.12×	7.81e-07
IPOPT-Def.	–	–	4.17 ± 0.14	1.00×	7.34e-08
IPOPT-Tight	–	–	4.47 ± 0.36	0.93×	4.61e-09
IPOPT-Loose	–	–	3.87 ± 0.20	1.08×	4.57e-07
IPOPT-Mono.	–	–	5.69 ± 0.24	0.73×	4.40e-08
IPOPT-V.T.	–	–	5.05 ± 0.22	0.83×	4.85e-11
IPOPT-Fast	–	–	4.07 ± 0.26	1.02×	3.90e-07
$n = m = 10\,000$					
ψ_1	0.50	26/105	14.35 ± 0.49	0.79×	7.45e-07
	0.75	13/53	7.05 ± 0.21	1.60×	7.45e-07
	0.95	6/30	3.90 ± 0.28	2.89×	7.81e-07
ψ_c	0.50	26/26	5.61 ± 1.64	2.01×	7.86e-07
	0.75	13/14	3.83 ± 0.12	2.94×	8.85e-07
	0.95	6/10	2.56 ± 0.13	4.41×	8.22e-07
ψ_2	0.50	26/105	25.82 ± 0.10	0.44×	7.45e-07
	0.75	13/65	16.29 ± 0.21	0.69×	7.45e-07
	0.95	6/30	7.39 ± 0.18	1.53×	7.81e-07
ψ_3	0.50	26/105	24.68 ± 0.78	0.46×	7.45e-07
	0.75	13/53	4.08 ± 0.12	2.76×	7.45e-07
	0.95	6/30	3.67 ± 1.94	3.07×	7.81e-07
IPOPT-Def.	–	–	11.27 ± 3.56	1.00×	3.98e-08
IPOPT-Tight	–	–	12.33 ± 3.75	0.91×	3.87e-09
IPOPT-Loose	–	–	10.27 ± 3.36	1.10×	5.60e-07
IPOPT-Mono.	–	–	15.51 ± 5.43	0.73×	5.19e-08
IPOPT-V.T.	–	–	13.74 ± 3.76	0.82×	6.38e-11
IPOPT-Fast	–	–	10.95 ± 2.95	1.03×	3.61e-07
$n = m = 20\,000$					
ψ_1	0.50	26/105	52.27 ± 2.48	0.32×	7.45e-07
	0.75	13/65	14.83 ± 6.86	1.11×	7.45e-07
	0.95	6/30	5.07 ± 0.10	3.25×	7.81e-07
ψ_c	0.50	26/26	8.07 ± 4.37	2.04×	7.86e-07
	0.75	13/14	9.10 ± 0.35	1.81×	8.85e-07
	0.95	6/10	6.43 ± 0.09	2.56×	8.22e-07
ψ_2	0.50	26/105	53.51 ± 1.10	0.31×	7.45e-07
	0.75	13/65	10.77 ± 2.04	1.53×	7.45e-07
	0.95	6/30	4.53 ± 0.03	3.64×	7.81e-07
ψ_3	0.50	26/105	16.04 ± 0.26	1.03×	7.45e-07
	0.75	13/65	9.94 ± 0.06	1.66×	7.45e-07
	0.95	6/30	4.66 ± 0.07	3.54×	7.81e-07
IPOPT-Def.	–	–	16.48 ± 9.10	1.00×	5.27e-08
IPOPT-Tight	–	–	16.88 ± 8.98	0.98×	3.89e-09
IPOPT-Loose	–	–	14.34 ± 7.40	1.15×	6.75e-07
IPOPT-Mono.	–	–	21.08 ± 10.78	0.78×	6.51e-08
IPOPT-V.T.	–	–	19.50 ± 11.29	0.84×	3.63e-11
IPOPT-Fast	–	–	14.84 ± 7.29	1.11×	7.19e-07

Table 5. Comparison of kernel functions for Example 4.

Solver	θ	Out/in	Time (s)	Speed	KKT
$n = m = 5\,000$					
ψ_1	0.50	26/113	4.34 ± 0.03	2.32×	7.45e-07
	0.75	–	180.11 ± 0.04	0.06×	**9.14e+06**
	0.95	6/30	3.78 ± 0.16	2.67×	7.81e-07
ψ_c	0.50	27/36	4.90 ± 0.35	2.06×	7.39e-07
	0.75	15/25	3.66 ± 0.19	2.75×	3.51e-07
	0.95	7/12	1.70 ± 0.15	5.92×	1.36e-07
ψ_2	0.50	26/113	14.18 ± 0.73	0.71×	7.45e-07
	0.75	–	180.12 ± 0.06	0.06×	**6.60e+06**
	0.95	6/30	4.20 ± 0.13	2.40×	7.81e-07
ψ_3	0.50	26/112	16.23 ± 0.06	0.62×	7.45e-07
	0.75	–	180.15 ± 0.04	0.06×	**1.09e+07**
	0.95	6/30	4.26 ± 0.21	2.37×	7.81e-07
IPOPT-Def.	–	–	10.08 ± 0.43	1.00×	1.86e-08
IPOPT-Tight	–	–	10.26 ± 0.74	0.98×	4.09e-09
IPOPT-Loose	–	–	8.93 ± 0.54	1.13×	4.81e-07
IPOPT-Mono.	–	–	13.65 ± 1.11	0.74×	7.32e-08
IPOPT-V.T.	–	–	11.55 ± 0.83	0.87×	6.90e-11
IPOPT-Fast	–	–	9.28 ± 0.91	1.09×	3.48e-07
$n = m = 10\,000$					
ψ_1	0.50	26/114	32.83 ± 0.46	0.59×	7.45e-07
	0.75	–	180.20 ± 0.01	0.11×	**7.08e+11**
	0.95	6/30	7.40 ± 0.27	2.62×	7.81e-07
ψ_c	0.50	27/36	9.29 ± 0.44	2.09×	7.39e-07
	0.75	15/25	6.38 ± 0.29	3.04×	3.51e-07
	0.95	7/12	3.04 ± 0.15	6.38×	1.36e-07
ψ_2	0.50	26/115	28.48 ± 0.75	0.68×	7.45e-07
	0.75	–	180.13 ± 0.08	0.11×	**2.85e+12**
	0.95	6/30	2.14 ± 0.04	9.08×	7.81e-07
ψ_3	0.50	26/114	7.84 ± 0.06	2.47×	7.45e-07
	0.75	–	180.13 ± 0.06	0.11×	**1.01e+12**
	0.95	6/30	4.40 ± 0.04	4.41×	7.81e-07
IPOPT-Def.	–	–	19.39 ± 0.50	1.00×	5.22e-08
IPOPT-Tight	–	–	19.75 ± 1.13	0.98×	6.92e-09
IPOPT-Loose	–	–	17.15 ± 0.58	1.13×	7.89e-07
IPOPT-Mono.	–	–	24.57 ± 1.00	0.79×	8.77e-08
IPOPT-V.T.	–	–	23.14 ± 1.29	0.84×	6.67e-11
IPOPT-Fast	–	–	17.61 ± 0.69	1.10×	5.73e-07
$n = m = 20\,000$					
ψ_1	0.50	26/115	34.92 ± 1.39	0.54×	7.45e-07
	0.75	–	180.16 ± 0.06	0.11×	**1.85e+14**
	0.95	6/30	7.85 ± 0.33	2.42×	7.81e-07
ψ_c	0.50	27/36	9.57 ± 0.45	1.99×	7.39e-07
	0.75	15/25	6.94 ± 0.30	2.74×	3.51e-07
	0.95	7/12	3.47 ± 0.12	5.48×	1.36e-07
ψ_2	0.50	26/115	27.64 ± 2.70	0.69×	7.45e-07
	0.75	–	180.08 ± 0.02	0.11×	**2.18e+14**
	0.95	6/30	4.87 ± 0.14	3.91×	7.81e-07
ψ_3	0.50	26/115	18.58 ± 0.28	1.02×	7.45e-07
	0.75	–	180.17 ± 0.02	0.11×	**2.38e+14**
	0.95	6/30	4.91 ± 0.05	3.87×	7.81e-07
IPOPT-Def.	–	–	19.02 ± 1.61	1.00×	3.48e-08
IPOPT-Tight	–	–	20.92 ± 1.18	0.91×	5.22e-09
IPOPT-Loose	–	–	17.36 ± 0.96	1.10×	7.34e-07
IPOPT-Mono.	–	–	25.14 ± 0.26	0.76×	6.67e-08
IPOPT-V.T.	–	–	22.51 ± 1.72	0.85×	6.17e-11
IPOPT-Fast	–	–	18.04 ± 1.43	1.05×	4.20e-07

Table 6. Comparison of kernel functions for Example 5.

Solver	θ	Out/in	Time (s)	Speed	KKT
$n = 50, m = 30$					
ψ_1	0.50	49/345	0.82 ± 0.21	$1.72\times$	$2.46e-10$
	0.75	26/493	1.19 ± 0.03	$1.19\times$	$1.27e-10$
	0.95	12/244	0.87 ± 0.01	$1.63\times$	$2.06e-10$
ψ_c	0.50	52/50	0.71 ± 0.03	$2.00\times$	$1.97e-10$
	0.75	27/26	0.92 ± 0.05	$1.54\times$	$1.04e-10$
	0.95	13/24	0.93 ± 0.04	$1.52\times$	$1.17e-10$
ψ_2	0.50	49/345	1.03 ± 0.04	$1.38\times$	$2.46e-10$
	0.75	26/494	1.22 ± 0.04	$1.16\times$	$1.27e-10$
	0.95	12/244	0.94 ± 0.00	$1.51\times$	$2.06e-10$
ψ_3	0.50	49/345	0.92 ± 0.03	$1.54\times$	$2.46e-10$
	0.75	26/473	1.23 ± 0.03	$1.15\times$	$1.33e-10$
	0.95	12/244	0.98 ± 0.03	$1.44\times$	$2.06e-10$
IPOPT-Def	–	–	1.41 ± 0.05	$1.00\times$	$5.03e-08$
IPOPT-Tight	–	–	1.51 ± 0.10	$0.94\times$	$5.80e-09$
IPOPT-Loose	–	–	1.25 ± 0.04	$1.13\times$	$6.70e-07$
IPOPT-Mono	–	–	1.96 ± 0.08	$0.72\times$	$4.53e-08$
IPOPT-V.T.	–	–	1.72 ± 0.08	$0.82\times$	$7.23e-11$
IPOPT-Fast	–	–	1.30 ± 0.04	$1.08\times$	$6.14e-07$
$n = 500, m = 500$					
ψ_1	0.50	27/109	2.82 ± 0.01	$25.72\times$	$5.96e-07$
	0.75	14/56	2.58 ± 0.00	$28.12\times$	$2.98e-07$
	0.95	9/45	35.60 ± 0.38	$2.04\times$	$2.26e-10$
ψ_c	0.50	32/30	35.36 ± 0.47	$2.05\times$	$4.92e-09$
	0.75	27/26	36.76 ± 0.24	$1.97\times$	$4.42e-10$
	0.95	13/24	54.33 ± 1.36	$1.33\times$	$2.24e-10$
ψ_2	0.50	27/109	2.74 ± 0.04	$26.41\times$	$5.96e-07$
	0.75	14/57	2.58 ± 0.01	$28.12\times$	$2.98e-07$
	0.95	9/45	35.33 ± 0.23	$2.05\times$	$2.26e-10$
ψ_3	0.50	27/109	2.77 ± 0.04	$26.21\times$	$5.96e-07$
	0.75	14/56	2.53 ± 0.02	$28.70\times$	$2.98e-07$
	0.95	9/45	35.99 ± 0.29	$2.01\times$	$2.26e-10$
IPOPT-Def	–	–	72.49 ± 0.56	$1.00\times$	$3.95e-08$
IPOPT-Tight	–	–	71.87 ± 2.22	$1.01\times$	$3.31e-09$
IPOPT-Loose	–	–	67.61 ± 2.24	$1.07\times$	$6.07e-07$
IPOPT-Mono	–	–	95.75 ± 4.06	$0.76\times$	$4.25e-08$
IPOPT-V.T.	–	–	89.78 ± 2.60	$0.81\times$	$4.45e-11$
IPOPT-Fast	–	–	63.50 ± 3.66	$1.14\times$	$4.18e-07$
$n = m = 1000$					
ψ_1	0.50	27/109	8.29 ± 0.07	$69.24\times$	$5.96e-07$
	0.75	14/57	7.65 ± 0.05	$75.09\times$	$2.98e-07$
	0.95	9/45	277.08 ± 11.68	$2.07\times$	$2.26e-10$
ψ_c	0.50	32/30	271.22 ± 1.33	$2.12\times$	$4.92e-09$
	0.75	27/26	275.86 ± 0.35	$2.08\times$	$4.42e-10$
	0.95	13/24	446.26 ± 30.63	$1.29\times$	$2.24e-10$
ψ_2	0.50	27/109	10.38 ± 0.16	$55.31\times$	$5.96e-07$
	0.75	14/57	9.73 ± 0.16	$59.01\times$	$2.98e-07$
	0.95	9/45	318.27 ± 1.85	$1.80\times$	$2.26e-10$
ψ_3	0.50	27/109	10.31 ± 0.20	$55.70\times$	$5.96e-07$
	0.75	14/57	9.86 ± 0.27	$58.20\times$	$2.98e-07$
	0.95	9/45	313.03 ± 12.66	$1.83\times$	$2.26e-10$
IPOPT-Def	–	–	574.15 ± 38.98	$1.00\times$	$6.24e-08$
IPOPT-Tight	–	–	567.04 ± 20.75	$1.01\times$	$7.39e-09$
IPOPT-Loose	–	–	518.41 ± 41.75	$1.11\times$	$5.80e-07$
IPOPT-Mono	–	–	740.18 ± 20.09	$0.78\times$	$5.65e-08$
IPOPT-V.T.	–	–	658.16 ± 25.83	$0.87\times$	$6.47e-11$
IPOPT-Fast	–	–	503.26 ± 27.02	$1.14\times$	$4.06e-07$

4.3.5. Selection guidelines and strategic summary. Table 7 provides a decision framework for kernel selection based on our empirical findings.

In conclusion, the complex kernel ψ_c is highly recommended for structured, nonlinear, or geometrically difficult problems where numerical stability is paramount. For problems characterized by massive variable interaction, classical kernels remain more efficient due to their simpler algebraic footprint. Overall, ψ_c represents a significant advancement in the reliability and efficiency of primal-dual IPMs in the complex domain.

5. Conclusion

We presented the first primal-dual interior-point framework natively designed for large-scale convex optimization in complex variables. By introducing a new family of complex self-regular kernel functions, exemplified by

$$\psi_c(v) = -\log(\Re v) - \log(1 + (\Im v)^2) + \tfrac{1}{2}|v|^2 - \tfrac{1}{2},$$

defined directly on the right half-plane $\Re v > 0$, we preserve all the key properties (strong convexity, exponential convexity, and eligibility for large updates) that underpin the best real-valued methods.

Theoretical analysis established that the resulting algorithm achieves the currently best-known iteration complexity for large-update primal-dual methods, namely, $O\big(m \log(m\mu^0/\varepsilon)\big)$. Remarkably, numerical evidence revealed that the practical performance significantly surpasses this theoretical bound: the outer iteration count remains constant at 26 iterations as dimensions scale from $m = 2000$ to $m = 20000$, suggesting an empirical complexity of $O(1)$ with respect to m. This striking dimension-independence empirically validates the affine-invariance properties of the proposed framework. The implementation seamlessly reduced to a highly competitive real-variable method, incurring virtually no overhead.

Extensive numerical experiments on problems ranging from $n = 50$ to $n = 20\,000$ revealed a structure-dependent performance hierarchy. The complex kernel ψ_c demonstrated exceptional efficiency in separable problems, achieving up to 9 times speedup over IPOPT and reducing inner iterations by 70–77%. Most notably, ψ_c exhibited unique numerical robustness in asymmetric landscapes (Example 4), maintaining 100% convergence with KKT residuals below 10^{-7} where classical kernels suffer complete breakdown (KKT errors exceeding 10^{14}). For non-separable coupled problems (Example 5), classical kernels ψ_1, ψ_2, ψ_3 proved more efficient due to lower algebraic cost per iteration, achieving spectacular speedups up to 70 times versus IPOPT.

Table 7. Strategic framework for kernel selection.

Problem feature	Kernel	Primary advantage
Separable	ψ_c	Up to 7× faster than IPOPT
Asymmetric	ψ_c	Guaranteed robustness
Coupled variables	$\psi_{1,2,3}$	Lower algebraic cost/iter
Large-scale	All	Scale-free iterations

The parameter choice $\theta = 0.75$ emerges as optimal across all problem types, balancing convergence speed with numerical stability. While aggressive updates ($\theta = 0.95$) can reduce outer iterations in regular problems, they risk numerical instability in coupled or asymmetric settings.

This work lays the theoretical and algorithmic foundation for interior-point methods in complex optimization, demonstrating that kernel selection should be guided by the problem structure: ψ_c for separable and asymmetric problems and classical kernels for coupled systems. Immediate extensions include infeasible-start variants, warm-starting strategies, and the treatment of nonconvex complex problems directions that promise to deliver efficient, structure-exploiting solvers for emerging large-scale applications in phase retrieval, quantum state tomography, complex-valued deep learning, and AC optimal power flow.

Acknowledgment

The authors would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support (QU-APC-2026).

References

Armijo, L. (1966). Minimization of functions having Lipschitz continuous first partial derivatives, *Pacific Journal of Mathematics* **16**(1): 1–3.

Bai, Y.Q., El Ghami, M. and Roos, C. (2004). A comparative study of kernel functions for primal-dual interior-point algorithms in linear optimization, *SIAM Journal on Optimization* **15**(1): 101–128, DOI: 10.1137/S1052623403423114.

Bauschke, H.H. and Combettes, P.L. (2017). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, Cham, DOI: 10.1007/978-3-319-4.

Bouafia, M., Benterki, D. and Yassine, A. (2016). An efficient primal-dual interior-point algorithm for linear programming problems based on a new kernel function with a trigonometric barrier term, *Journal of Optimization Theory and Applications* **170**(2): 528–554, DOI: 10.1007/s10957-016-0895-0.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*, Cambridge University Press, Cambridge.

Cai, X., Wu, L., Yue, Y., Li, M. and Wang, G. (2014). Kernel-function-based primal-dual interior-point methods for convex quadratic optimization over symmetric cones, *Journal of Inequalities and Applications* **2014**: 308, DOI: 10.1186/1029-242X-2014-308.

Darvay, Z., Nagy, E.M., Lesaja, G., Rigó, P.R. and Varga, A. (2024). Comprehensive analysis of kernel-based interior-point methods for $P_*(\kappa)$-LCP, *Working paper,* Corvinus University of Budapest, Budapest, `https://ideas.repec.org/p/cvh/coecwp/2024-03.html`.

Djeffal, E. and Laouar, M. (2019). A primal-dual interior-point method based on a new kernel function for linear complementarity problem, *Asian-European Journal of Mathematics* **13**(1): 2050001, DOI: 10.1142/S1793557120500011.

Forsgren, A., Gill, P.E. and Wright, M.H. (2002). Interior methods for nonlinear optimization, *SIAM Review* **44**(4): 525–597, DOI: 10.1137/S0036144502414942.

Hirose, A. (2013). *Complex-Valued Neural Networks: Advances and Applications*, Wiley/IEEE Press, Hoboken.

Hjørungnes, A. and Gesbert, D. (2007). Complex-valued matrix differentiation: Techniques and key results, *IEEE Transactions on Signal Processing* **55**(6): 2740–2746, DOI: 10.1109/TSP.2007.893762.

Kaczorek, T. (2026). On the global stability of fractional feedback nonlinear systems with interval matrices of positive linear parts and application to electrical circuits, *International Journal of Applied Mathematics and Computer Science* **36**(1): 5–11, DOI: 10.61822/amcs-2026-0001.

Kreutz-Delgado, K. (2009). The complex gradient operator and the CR-Calculus, *arXiv* 0906.4835, DOI: 10.48550/arXiv.0906.4835.

Laouar, M., Brahimi, M. and Djeffal, E.A. (2022). A method projective for solution of a quadratic semi-definite problem with a new potential function, *Journal of Mathematical and Computational Science* **12**: 193, DOI: 10.28919/jmcs/7646.

Laouar, M., Brahimi, M. and Lakhdari, I.E. (2023). Kernel function with BFGS quasi-Newton methods for solving nonlinear semi-definite problems, *Journal of Mathematics and Computer Science* **33**(1): 1–17, DOI: 10.22436/jmcs.033.01.01.

Laouar, M., Brahimi, M., Ziadi, R., Saleh, M.A., Almaymouni, A.Z. and Benmessaoud, C. (2026). A generalized self-regular kernel function for large-scale nonlinear optimization problems, *AIMS Mathematics* **11**(2): 4935–4965, DOI: 10.3934/math.2026202.

Lee, Y., Cho, Y. and Cho, G. (2013). Kernel function based interior-point methods for horizontal linear complementarity problems, *Journal of Inequalities and Applications* **2013**(215): 1–15, DOI: 10.1186/1029-242X-2013-215.

Lesaja, G. and Roos, C. (2010). Unified analysis of kernel-based interior-point methods for conic optimization,

SIAM Journal on Optimization **20**(6): 2613–2638, DOI:10.1137/090766735.

Leulmi, A., Ziadi, R., Souli, C., Saleh, M.A. and Almaymouni, A.Z. (2024). An interior point algorithm for quadratic programming based on a new step-length, *International Journal of Analysis and Applications* **22**(233): 1–12, DOI: 10.28924/2291-8639-22-2024-233.

Luo, Z.D.N., Sidiropoulos, N.D., Tseng, P. and Zhang, S. (2007). Approximation bounds for quadratic optimization with homogeneous quadratic constraints, *SIAM Journal on Optimization* **18**(1): 1–28, DOI: 10.1137/050642691.

Maiza, A., Ziadi, R., Saleh, M.A. and Almaymouni, A.Z. (2025). A combination of two conjugate gradient methods under a new line search with its application, *International Journal of Applied Mathematics and Computer Science* **35**(2): 267–280, DOI: 10.61822/amcs-2025-0019.

Matychyn, I. and Onyshchenko, V. (2021). Time-optimal control of linear fractional systems with variable coefficients, *International Journal of Applied Mathematics and Computer Science* **31**(3): 375–386, DOI: 10.34768/amcs-2021-0025.

Nesterov, Y. (2004). *Introductory Lectures on Convex Optimization: A Basic Course*, Springer, Boston.

Nesterov, Y. and Nemirovski, A. (1994). *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia.

Nocedal, J. and Wright, S., J. (2006). *Numerical Optimization*, 2nd Edn, Springer, New York.

Peng, J., Roos, C. and Terlaky, T. (2002). Self-regular functions and new search directions for linear and semidefinite optimization, *Mathematical Programming* **93**(1): 129–171, DOI: 10.1007/s101070200296.

Roos, C. (2006). A full-Newton step $O(n)$ infeasible interior-point algorithm for linear optimization, *SIAM Journal on Optimization* **16**(4): 1110–1136, DOI: 10.1137/050623917.

Si, X., Yang, H. and Ivanov, I.G. (2021). Conditions and a computation method of the constrained regulation problem for a class of fractional-order nonlinear continuous-time systems, *International Journal of Applied Mathematics and Computer Science* **31**(1): 17–28, DOI: 10.34768/amcs-2021-0002.

Srinivasarengan, K., Ragot, J., Aubrun, C. and Maquin, D. (2022). Parameter identifiability for nonlinear LPV models, *International Journal of Applied Mathematics and Computer Science* **32**(2): 255–269, DOI: 10.34768/amcs-2022-0019.

Tanksley, L.Y. (2009). *Interior Point Methods and Kernel Functions of a Linear Program*, Master's thesis, Georgia Southern University, Statesboro, https://digitalcommons.georgiasouthern.edu/etd/650/.

Took, C.C. and Mandic, D.P. (2010). A quaternion widely linear adaptive filter, *IEEE Transactions on Signal Processing* **58**(8): 4427–4431, DOI: 10.1109/TSP.2010.2048323.

Trabelsi, C., Bilaniuk, O., Zhang, Y., Serdyuk, D., Subramanian, S., Santos, J.F., Mehri, S., Rostamzadeh, N., Bengio, Y. and Pal, C.J. (2018). Deep complex networks, *Proceedings of the International Conference on Learning Representations (ICLR), Vancouver, Canada*.

Viksal, P. (2023). *The Quantum Approximate Optimization Algorithm: Optimization Problems and Implementations*, PhD thesis, Chalmers University of Technology, Göteborg, https://research.chalmers.se/publication/536821/file/536821_Fulltext.pdf.

Watrous, J. (2018). *The Theory of Quantum Information*, Cambridge University Press, Cambridge.

Wright, S.J. (1997). *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia.

Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, *Mathematical Programming* **106**(1): 25–57, DOI: 10.1007/s10107-004-0559-y.

Ye, Y. (1997). *Interior-Point Algorithms: Theory and Analysis*, John Wiley & Sons, New York.

Zhang, L. and Xu, Y. (2011). A new infeasible interior-point algorithm with full-Newton steps for linear programming based on a simple function, *International Journal of Computer Mathematics* **88**(15): 3163–3185, DOI: 10.1080/00207160.2011.597503.

Mounia Laouar is an associate professor in the Department of Mathematics at the University of Batna 2, Algeria. She earned her PhD in mathematics from that university in 2022. Her research focuses on nonlinear optimization, and wavelet theory.

Mahmoud Brahimi is an associate professor in the Department of Mathematics at the University of Mostafa Benboulaid Batna 2, Algeria. He obtained his PhD in applied mathematics in 2020. His research focuses on wavelet theory, optimization, and their applications in applied mathematics.

Raouf Ziadi is an associate professor in the Department of Mathematics, University Setif 1 Ferhat Abbas, Algeria. He received his PhD degree in applied mathematics from Ferhat Abbas University in 2017. His research interests are in optimization and related fields.

Mohammed A. Saleh is an associate professor at the Department of Cybersecurity, College of Computer, Qassim University, Saudi Arabia. He holds a BS in Mathematical and Computer Science from the Faculty of mathematical and computer science, University of Gezira, Sudan, an MS in information security from the Faculty of Computer Science and Information Systems, University of Technology (UTM), Malaysia, and a PhD in information security from the Faculty of Computing, UTM. His research interests include malware analysis and artificial intelligence in cybersecurity.

Abdulgader Z. Almaymuni is an associate professor at the Department of Cybersecurity, College of Computer, Qassim University, Saudi Arabia. He received a PhD from Demont Fort University (UK) in 2013 with a specialization in computer security and smart systems. His research interests include cybersecurity, formal method and smart systems.

Abdalilah Alhalangy is an associate professor of computer engineering, College of Computer at Qassim University, Saudi Arabia. He holds a PhD in information systems, and his research spans AI, intelligent systems, machine learning, computer vision, cypersecurity, digital transformation, wireless networks, cloud computing, GIS, and e-learning. He has published widely in international venues and is active in supervising and mentoring students, leading collaborative projects, and serving the academic community.

Received: 12 December 2025
Revised: 26 March 2026
Accepted: 14 April 2026