Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 11(7): 22576–22597.
DOI: 10.3934/math.2026912
Received: 11 May 2026
Revised: 07 July 2026
Accepted: 16 July 2026
Published: 27 July 2026

Research article

Complexity of a predictor-corrector interior-point method using special kernel function for sufficient WLCPs

Zhuoran Gao[1,3], Hongjin Wei[4], Suobin Zhang[5], and Xiaoni Chi[1,2,3,*]

[1] School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin 541004, China

[2] Guangxi Colleges and Universities Key Laboratory of Data Analysis and Computation, Guilin University of Electronic Technology, Guilin 541004, China

[3] Center for Applied Mathematics of Guangxi (GUET), Guilin 541004, China

[4] School of Mathematics and Computer Engineering, Guangxi Science and Technology Normal University, Laibin Guangxi 546199, China

[5] Institute of Scientific Research and Development, Guilin University of Electronic Technology, Guilin 541004, China

* **Correspondence:** Email: chixiaoni@126.com.

Abstract: A predictor-corrector interior-point method (PC IPM) is developed in this paper to address sufficient weighted linear complementarity problems (WLCPs). Specifically, the analysis is conducted by choosing the parameters of the kernel function $\varphi(t) = \frac{t^{\mathfrak{p}+1}-1}{\mathfrak{p}+1} + \frac{t^{1-\mathfrak{q}}-1}{\mathfrak{q}-1}$ to $\mathfrak{p} = 1$ and $\mathfrak{q} = 3$. Each iteration of the algorithm performs two steps: a predictor step followed by a corrector step. Furthermore, with appropriate parameter selections, we prove that the proposed method remains strictly feasible and converges. A polynomial complexity bound is derived, which is comparable to the best-known iteration bound reported in the literature on PC IPMs for solving sufficient WLCPs. Additionally, preliminary numerical experiments indicate that the proposed algorithm performs effectively.

Keywords: predictor-corrector interior-point method; sufficient weighted linear complementarity problem; full-Newton step; proximity measure; kernel function
Mathematics Subject Classification: 90C33, 90C51

1. Introduction

The weighted complementarity problem (WCP) was originally proposed by Potra [1], which is an extension of the complementarity problem. The weighted linear complementarity problem (WLCP) aims to seek a pair of vectors lying in the intersection of a manifold and a cone, so that, with the specified algebraic product, they yield the given weight vector. In addition, the linear complementarity problem (LCP) becomes a special case of the WLCP when the weight vector ω is the zero vector. However, the presence of a weight vector renders WLCPs more challenging than LCPs, both in terms of theory and algorithm design. For instance, Potra [1] provided a large-step path-following interior-point method (IPM) and a predictor-corrector interior-point method (PC IPM) for monotone WLCPs (MWLCPs). Subsequently, Tang [2] introduced a variant nonmonotone smoothing algorithm, which is designed to solve the MWLCP. Recently, existence and uniqueness of solutions to the weighted horizontal LCP over Euclidean Jordan algebras were proved by Chi et al. [3]. In 2016, Potra [4] proposed the sufficient WLCP and solved it via a PC IPM, showing that the algorithm's iteration complexity matches the best-known results for solving the sufficient LCP using IPMs. In parallel, to solve monotone WLCPs, Zhang [5] developed a smoothing Newton algorithm and established its local as well as global convergence.

IPMs can be extended to solve the WLCP. Since Karmarkar [6] introduced the well-known IPM, it is regarded as a highly effective algorithm for various WLCPs. For instance, quadratic convergence to the target point on the central path for MWLCPs was established by Asadi et al. [7] through an IPM based on full-Newton steps. Furthermore, for a special WLCP [8] on the nonnegative orthant, Chi and Wang [9] proposed an infeasible IPM with full-Newton steps.

In practice, PC IPMs constitute one of the most successful classes within the IPM framework. Mehrotra [10] and Sonnevend [11] introduced the first PC IPM for linear optimization (LO). In most PC IPMs, a typical main iteration comprises one predictor step and multiple corrector steps. Employing only one corrector step per main iteration, Mizuno et al. [12] proposed the first Mizuno-Todd-Ye PC IPM for linear programming (LP). Building upon this line of research, a Mizuno-Todd-Ye-type PC IPM for $P_*(\kappa)$-LCPs was introduced by Potra and Sheng [13]. Subsequently, Darvay [14] introduced PC IPMs for LO based on the algebraic equivalent transformation (AET) technique. The algorithm uses the function $\psi(t) = \sqrt{t}$, which is defined on the domain $D_\psi = \left(\frac{1}{4}, \infty\right)$, to determine the transformed central path and the modified Newton system. Kheirfam introduced two PC IPMs: one for convex quadratic symmetric cone optimization [15] and another for second-order cone optimization [16]. Recently, a PC IPM for $P_*(\kappa)$-WLCP was developed by Zhang [17] employing the kernel function $\varphi(t) = \sqrt{t}$, while the polynomial iteration complexity of this method was proved.

Determining search directions is crucial for IPMs, which could be defined based on kernel functions. For instance, Potra [18] proposed a novel smooth central path and developed two IPMs to solve MWLCPs [1, 19]. The characterization of sufficiency for the WLCP was established by Potra [4], who specifically proposed a CP IPM requiring only one matrix factorization in each iteration. A significant development in deriving search directions was the AET technique, proposed by Darvay [20] for LO. This technique was later extended by Achache [21] to convex quadratic programming, and subsequently applied by Wang [22] to monotone LCPs based on symmetric cones. An infeasible IPM employing a full-Newton step for monotone linear complementarity problems (MLCPs) was developed by Zhang [23], utilizing a simple locally-kernel function. This method yields the same search directions as [20]. In addition, Mansouri and Pirhaji [24] applied Darvay's technique [20] for LO by using $\phi(t) = \sqrt{t}$, which is a continuously differentiable kernel function. Then they developed an IPM for the MLCP.

Building upon the aforementioned work, a predictor-corrector algorithm for solving the sufficient

WLCP that uses a new search direction is presented. Using the kernel function $\varphi(t) = \frac{t^{\mathfrak{p}+1}-1}{\mathfrak{p}+1} + \frac{t^{1-\mathfrak{q}}-1}{\mathfrak{q}-1}$ for $0 \leq \mathfrak{p} \leq 1$ and for $\mathfrak{q} > 1$ [25], we employ the standard kernel-based PC IPM theory to derive the search directions of our algorithm. Furthermore, the iteration bound of the PC IPM is proved, matching the best-known result for the sufficient WLCP. In conclusion, we also report numerical results and provide our conclusions.

The paper is structured as follows. Preliminaries on the sufficient WLCP and the description of search directions are provided in Section 2. A detailed description of the PC IPM is provided in Section 3. The convergence analysis of the PC IPM, along with its iteration bound, is presented in Section 4. Preliminary numerical outcomes related to the PC IPM are showcased in Section 5. We conclude this paper in Section 6.

Some notations used throughout this paper are listed below. Throughout the paper, $\mathbb{R}$ denotes the real numbers, while $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote the nonnegative and positive reals, respectively. We use $\mathbb{R}^{m\times n}$ to denote the space of $m \times n$ matrices with real entries. As usual, $\|\cdot\|_\infty$ and $\|\cdot\|$ denote the maximum norm and Euclidean norm, respectively. Given $x, s \in \mathbb{R}^n$, we write xs for their componentwise (Hadamard) product. Let $x, s \in \mathbb{R}^n$ with $s_\mathfrak{i} \neq 0$ for $\mathfrak{i} = 1, 2, \cdots, n$. Then we define the coordinate-wise division x/s as $x/s = (x_\mathfrak{i}/s_\mathfrak{i})_{1\leq\mathfrak{i}\leq n}$. The smallest element of a vector v is denoted by $\min v = \min\limits_{1\leq\mathfrak{i}\leq n} v_\mathfrak{i}$.

2. Preliminaries

Consider a vector $q \in \mathbb{R}^n$ and a matrix $M \in \mathbb{R}^{n\times n}$. The sufficient WLCP consists of finding vectors $x, s \in \mathbb{R}^n$ satisfying

$$\begin{aligned} -Mx + s = q, x \geq 0, \\ xs = \omega, s \geq 0, \end{aligned} \tag{2.1}$$

for a given weight vector $\omega \in \mathbb{R}^n_+$. Given nonnegative number κ, M is called a $P_*(\kappa)$-matrix [26] if the following condition holds:

$$(1 + 4\kappa) \sum_{\mathfrak{i}\in\mathfrak{I}_+} x_\mathfrak{i} (Mx)_\mathfrak{i} + \sum_{\mathfrak{i}\in\mathfrak{I}_-} x_\mathfrak{i} (Mx)_\mathfrak{i} \geq 0, \forall x \in \mathbb{R}^n, \tag{2.2}$$

with the index sets $\mathfrak{I}_+(x) = \{\mathfrak{i} : x_\mathfrak{i} (Mx)_\mathfrak{i} > 0\}$, $\mathfrak{I}_-(x) = \{\mathfrak{i} : x_\mathfrak{i} (Mx)_\mathfrak{i} < 0\}$ and $\mathfrak{i} = 1, 2, \ldots, n$. The handicap of matrix M is defined as the smallest κ for which inequality (2.2) holds. Väliaho [27] established that matrices are $P_*(\kappa)$-matrices if and only if they are sufficient matrices.

The feasible region of the sufficient WLCP (2.1) is defined as

$$\mathfrak{F}^* := \{(x, s) \mid -Mx + s = 0, x \geq 0, s \geq 0\}.$$

The strictly feasible region is

$$\mathfrak{F}^0 := \{(x, s) \in \mathfrak{F}^* \mid -Mx + s = 0, x > 0, s > 0\}.$$

We assume a strictly feasible starting point $\left(x^0, s^0\right) \in \mathfrak{F}^0$ exists, i.e., $\mathfrak{F}^0 \neq \emptyset$.

Choosing a strictly feasible point $\left(x^0, s^0\right) \in \mathfrak{F}^0$, we define

$$\omega(t) = tx^0 s^0 + (1 - t)\omega, \tag{2.3}$$

where $0 < t \leq 1$. If the second relation in (2.1) is substituted by the parameterized condition $xs = \omega(t)$, we obtain

$$\begin{aligned} -Mx + s = q,\ x \geq 0, \\ xs = \omega(t),\ s \geq 0. \end{aligned} \tag{2.4}$$

Furthermore, if $t \to 0$, this yields a solution of the sufficient WLCP (2.1).

Using Newton's method on system (2.4) for a strictly feasible solution (x, s), we obtain the following system that determines search directions Δ_x and Δ_s:

$$\begin{aligned} -M\Delta_x + \Delta_s = 0,\ x \geq 0, \\ s\Delta_x + x\Delta_s = \omega(t) - xs,\ s \geq 0. \end{aligned} \tag{2.5}$$

By defining the vectors

$$v := \sqrt{\frac{xs}{\omega(t)}},\ d := \sqrt{\frac{x}{s}},\ d_x := \frac{v\Delta_x}{x},\ d_s := \frac{v\Delta_s}{s}, \tag{2.6}$$

we can express system (2.5) as

$$\begin{aligned} -\overline{M}d_x + d_s = 0, \\ d_x + d_s = v^{-1} - v, \end{aligned} \tag{2.7}$$

where $\overline{M} = \sqrt{\mathcal{W}^{-1}(t)}\mathcal{D}M\mathcal{D}\sqrt{\mathcal{W}(t)}$, $\mathcal{W}(t) = \operatorname{diag}(\omega(t))$, and $\mathcal{D} = \operatorname{diag}(d)$. Suppose M is a $P_*(\kappa)$-matrix and $x^0 s^0 > \omega$. Then $\overline{M}$ belongs to the class $P_*(\kappa')$-matrix with [28]

$$\frac{1 + 4\kappa'}{1 + 4\kappa} = \frac{\max x^0 s^0}{\min\{\chi\omega + (e - \chi)\, x^0 s^0\}},$$

where

$$\chi \in \mathbb{R}^n, \chi_{\mathrm{i}} = \begin{cases} 0, \text{if } \mathrm{i} \in \mathcal{J}_0, \\ 1, \text{if } \mathrm{i} \in \mathcal{J}_+, \end{cases} ,\mathcal{J} = \{1, 2, \ldots, n\}, \mathcal{J}_0 = \{\mathrm{i} \in \mathcal{J} : \omega_{\mathrm{i}} = 0\}$$

and $\mathcal{J}_+ = \{\mathrm{i} \in \mathcal{J} : \omega_{\mathrm{i}} > 0\}$.

The second equation of system (2.7) provides the scaled centering equation. The classical logarithmic barrier function is defined as

$$\varphi(v) = \sum_{\mathrm{i}=1}^{n} \left(\frac{v_{\mathrm{i}}^2 - 1}{2} - \log v_{\mathrm{i}}\right),$$

and its negative gradient is precisely $v - v^{-1}$. Then, we consider the second equation in the system (2.7) as follows: $d_x + d_s = -\nabla\varphi(v)$. Let $\varphi(t) : \mathbb{R}_{++} \to \mathbb{R}_+$ be a kernel function and the associated barrier function be defined by $\varphi(v)$, which is twice differentiable and satisfies

$$(i)\ \varphi(1) = \varphi'(1) = 0;$$

$$(ii)\ \varphi''(t) > 0,\ \forall t > 0;$$

$$(iii) \lim_{t\to 0}\varphi(t) = \lim_{t\to\infty}\varphi(t) = +\infty.$$

For $0 \le \mathfrak{p} \le 1$ and $\mathfrak{q} > 1$, we consider a specific kernel function [25] (the tenth function in Table 1)

$$\varphi(t) = \frac{t^{\mathfrak{p}+1}-1}{\mathfrak{p}+1} + \frac{t^{1-\mathfrak{q}}-1}{\mathfrak{q}-1}, \tag{2.8}$$

which is self-regular and eligible.

In this paper, we analyze the PC IPM by setting the kernel parameters in (2.8) to $\mathfrak{p} = 1$ and $\mathfrak{q} = 3$, i.e.,

$$\varphi(t) = \frac{t^2-1}{2} + \frac{t^{-2}-1}{2} = \frac{\left(t-t^{-1}\right)^2}{2}. \tag{2.9}$$

Since $\varphi'(t) = t^{\mathfrak{p}} - t^{-\mathfrak{q}} = t - t^{-3}$, system (2.5) can be reformulated as

$$\begin{aligned} -\overline{M}d_x + d_s &= 0, \\ d_x + d_s &= \frac{v^{-3}-v}{2}. \end{aligned} \tag{2.10}$$

To measure how far the current iteration point (x, s) is from the central path, we define the following proximity measure:

$$\delta(v) = \delta(x, s; \omega(t)) = 2\|d_x + d_s\| = \left\|v^{-3} - v\right\|. \tag{2.11}$$

For all $(x, s) \in \mathfrak{F}^0$ and $0 < t \le 1$, we can easily verify that

$$\delta(x, s; \omega(t)) = 0 \Leftrightarrow v = e \Leftrightarrow xs = \omega(t). \tag{2.12}$$

The following section will focus on establishing the rigorous feasibility and convergence properties of the PC IPM to solve the sufficient WLCP.

3. The predictor and corrector step

In the following, a PC IPM is presented for solving the sufficient WLCP based on the kernel function (2.9). For simplicity, we define $p_v = d_x + d_s,\ q_v = d_x - d_s$, which implies that

$$d_x = \frac{1}{2}(p_v + q_v),\ d_s = \frac{1}{2}(p_v - q_v), \tag{3.1}$$

and

$$d_x d_s = \frac{1}{4}\left(p_v^2 - q_v^2\right)$$

holds.

For the LO case, Roos et al. [29] showed that orthogonality holds between the scaled search directions d_x and d_s, i.e., $d_x^T d_s = 0$. One of the difficulties in analyzing algorithms for sufficient WLCPs arises because this relation does not hold in the sufficient WLCP case.

First, the corrector step is given by $x^+ = x + \Delta_x,\ s^+ = s + \Delta_s$. Define

$$v_+ := \sqrt{\frac{x^+ s^+}{\omega(t)}},\ d^+ := \sqrt{\frac{x^+}{s^+}},\ \mathcal{D}^+ := \operatorname{diag}(d^+),$$

and

$$\overline{M}_+ := \sqrt{\mathcal{W}^{-1}(t)}\mathcal{D}^+ M \mathcal{D}^+ \sqrt{\mathcal{W}(t)}.$$

For the scaled predictor system, we have

$$\begin{aligned} -\overline{M}_+ d_x^p + d_s^p &= 0, \\ d_x^p + d_s^p &= -\frac{1}{2} v_+. \end{aligned} \tag{3.2}$$

Subsequently, we determine the search directions (Δ_x^p, Δ_s^p) from

$$\Delta_x^p = \frac{x^+}{v_+} d_x^p, \ \Delta_s^p = \frac{s^+}{v_+} d_s^p. \tag{3.3}$$

We define the predictor iterate as

$$x^p = x^+ + \vartheta t \Delta_x^p, \ s^p = s^+ + \vartheta t \Delta_s^p, \tag{3.4}$$

with $\vartheta \in (0, 1)$.

The detailed steps of the PC IPM for solving the sufficient WLCP are now described. The algorithm starts from a strictly feasible initial point $\left(x^0, s^0\right)$ with $x^0 s^0 > \omega \geq 0$. In each main iteration, the PC IPM carries out one predictor step and one corrector step, repeating them until $\left\|x^k s^k - \omega\right\| > \varepsilon$. The search direction (Δ_x, Δ_s) is obtained in the corrector step by applying Newton's method to (2.10) together with the relation in (2.6). Then, we calculate the predictor search directions (Δ_x^p, Δ_s^p) from (3.2) and (3.3). We describe the PC IPM in Algorithm 1.

Algorithm 1. PC IPM for solving sufficient WLCP

Input

An accuracy parameter $\varepsilon > 0$;
A barrier update parameter $\vartheta \in (0, 1)$;
Let $\left(x^0, s^0\right) \in \mathfrak{F}^0$ with $\delta\left(x^0, s^0; \omega(t_0)\right) \leq \frac{t_0}{2(1+4\kappa')}$, where $t_0 = 1$;
Set $k := 0$;
begin
$x := x^0$; $s := s^0$; $t := t_0$;
  while $\left\|x^k s^k - \omega\right\| > \varepsilon$ **do**
  begin
    corrector step:
    Obtain $\left(\Delta_x^k, \Delta_s^k\right)$ from solving (2.10) using (2.6);
    take $(x^+, s^+) := (x + \Delta_x, s + \Delta_s)$;
    predictor step:
    Obtain (Δ_x^p, Δ_s^p) by solving the system (3.2) via (3.3);
    take $(x^p, s^p) := (x^+ + \vartheta t \Delta_x^p, s^+ + \vartheta t \Delta_s^p)$;
    set $t_p := (1 - \vartheta) t$, $\omega^p(t) := \left(1 - \frac{1}{2}\vartheta t\right)\omega(t)$;
    $x^k := x^p$,$s^k := s^p$; $\omega(t) := \omega^p(t)$; $t := t_p$; $k := k + 1$;
  end
end

4. Analysis of PC IPM

In the following, our analysis will show that Algorithm 1 is well-defined. By analyzing the corrector and predictor steps, an upper bound on the proximity measure after a full-Newton step is derived, sufficient conditions for maintaining strict feasibility are given, and the convergence of Algorithm 1 is established. We prove that Algorithm 1 can solve the sufficient WLCP in polynomial complexity.

4.1. Strict feasibility of the algorithm

Now, we analyze that the PC IPM of the $P_*(\kappa)$-WLCP is well-defined. From (2.6), it follows that

$$d_x^p d_s^p = \frac{\Delta_x^p \Delta_s^p}{\omega(t)}.$$

$\overline{M}$ belongs to the class of $P_*(\kappa')$-matrices

$$(1+4\kappa')\sum_{\mathrm{i}\in\Im_+} d_{x_\mathrm{i}}^p d_{s_\mathrm{i}}^p + \sum_{\mathrm{i}\in\Im_-} d_{x_\mathrm{i}}^p d_{s_\mathrm{i}}^p \geq 0, \tag{4.1}$$

where $\Im_+ = \{\mathrm{i} : \Delta_{x_\mathrm{i}}^p \Delta_{s_\mathrm{i}}^p > 0\}$ and $\Im_- = \{\mathrm{i} : \Delta_{x_\mathrm{i}}^p \Delta_{s_\mathrm{i}}^p < 0\}$. Let us introduce the notation $\delta := \delta(v) = \delta(x, s; \omega(t))$.

Lemma 1. *Let $v \in \mathbb{R}_+^n$. If $\delta < 1$, then for all $\mathrm{i} = 1, \ldots, n$,*

$$1 - \delta \leq v_\mathrm{i} \leq 1 + \delta.$$

Proof. By (2.11), we have for each $\mathrm{i} = 1, \cdots, n$,

$$\begin{aligned}\delta(v) &= \left\|v^{-3} - v\right\| = \left\|\frac{e - v^4}{v^3}\right\| \\ &\geq \left\|\frac{v^3 + v^2 + v + e}{v^3}\right\|_\infty \cdot \|e - v\| \\ &\geq \|e - v\|.\end{aligned} \tag{4.2}$$

This leads to the desired result for each $\mathrm{i} = 1, \cdots, n$,

$$1 - \delta \leq v_\mathrm{i} \leq 1 + \delta.$$

□

The following two lemmas estimate the upper bounds of $\|d_x d_s\|_\infty$ and $\|d_x d_s\|$ for LO, which can be extended to analyze the convergence of Algorithm 1 for the WLCP.

Lemma 2. *(Lemma 1 in [28]) We have*

$$-\frac{\kappa'}{4}\delta^2 \leq (d_x)^T d_s \leq \frac{\delta^2}{16},\ \|q_v\|^2 \leq \frac{1+4\kappa'}{4}\delta^2,\ \|d_x d_s\|_\infty \leq \frac{1+4\kappa'}{16}\delta^2.$$

Lemma 3. *(Lemma 2 in [28]) We have*

$$\|d_x d_s\| \leq \frac{\sqrt{1+(1+4\kappa')^2}}{16}\delta^2.$$

Lemma 4. *Under the condition that $\delta < \frac{4}{\sqrt{1+4\kappa'}}$, the iterate (x^+, s^+) is positive.*

Proof. Let β be any number in $[0,1]$. Define

$$(x(\beta), s(\beta)) = (x + \beta\Delta_x, s + \beta\Delta_s).$$

By (2.6) and (2.10),

$$\begin{aligned} \frac{x(\beta)s(\beta)}{\omega(t)} &= v^2 + \beta v p_v + \beta^2 d_x d_s \\ &= v^2 + \frac{1}{2}\beta v\left(v^{-3} - v\right) + \beta^2 d_x d_s \\ &= (1-\beta)v^2 + \frac{1}{2}\left(v^2 + v^{-2}\right) + \beta^2 d_x d_s. \end{aligned} \tag{4.3}$$

From Lemma 2,

$$\begin{aligned} \frac{v^2 + v^{-2}}{2} + \beta^2 d_x d_s &\geq (1 - \|d_x d_s\|_\infty)\, e \\ &\geq \left(1 - \frac{1+4\kappa'}{16}\delta^2\right) e > 0. \end{aligned}$$

The last inequality together with (4.3) implies $x(\beta)s(\beta) > 0$, which holds for every $\beta \in [0,1]$. Note that $x(\beta)$ and $s(\beta)$ are linear functions of β with $x(0) = x^0 > 0, s(0) = s^0 > 0$, so we conclude that $x(\beta) > 0, s(\beta) > 0$. Hence, we have $(x^+, s^+) = (x(1), s(1)) > 0$. □

4.2. Convergence of algorithm

This subsection derives an upper bound on the proximity measure achieved after the corrector and predictor steps. First, for a fixed $\omega(t)$, an upper bound on $\delta(v_+)$ is provided by following a full-Newton step.

Lemma 5. *Let $v_+ = \sqrt{\frac{x^+ s^+}{\omega(t)}}$. Under the condition that $\delta < \frac{4}{\sqrt{1+4\kappa'}}$, we have*

$$\delta(v_+) \leq \left(\frac{1}{\left(1 - \frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}} + \frac{1}{\left(1 - \frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2} + \frac{\sqrt{1+(1+4\kappa')^2}}{16}\right)\delta^2.$$

Proof. From (4.2), one gets

$$\begin{aligned} \delta(v_+) &= \left\|v_+^{-3} - v_+\right\| = \left\|\frac{e - v_+^4}{v_+^3}\right\| \\ &\leq \left\|v_+^{-3} + v_+^{-1}\right\|_\infty \cdot \left\|e - v_+^2\right\| \\ &\leq \left(\frac{1}{\min v_+^3} + \frac{1}{\min v_+}\right) \cdot \left\|e - v_+^2\right\|. \end{aligned}$$

It follows from (4.3) that

$$v_+^2 = \frac{x(1)s(1)}{\omega(t)} = \frac{v^2 + v^{-2}}{2} + d_x d_s. \tag{4.4}$$

Then, we get by Lemmas 1 and 2

$$\begin{aligned} \min v_+ &= \sqrt{\min v_+^2} \geq \sqrt{\frac{v^2 + v^{-2}}{2} + d_x d_s} \\ &\geq \min \sqrt{1 - \|d_x d_s\|_\infty} \\ &\geq \sqrt{1 - \frac{1 + 4\kappa'}{16}\delta^2}. \end{aligned} \tag{4.5}$$

In addition,

$$\begin{aligned} \left\|e - v_+^2\right\| &= \left\|e - \frac{v^2 + v^{-2}}{2} - d_x d_s\right\| \\ &\leq \left\|e - \frac{1}{2}v^2 - \frac{1}{2}v^{-2}\right\| + \|d_x d_s\| \\ &\leq \left\|\frac{e - \frac{1}{2}v^2 - \frac{1}{2}v^{-2}}{\left(v^{-3} - v\right)^2}\right\|_\infty \cdot \left\|v^{-3} - v\right\|^2 + \|d_x d_s\| \\ &\leq \left\|\frac{v^8 - 2v^6 + v^4}{2v^8 - 4v^4 + 2e}\right\|_\infty \cdot \left\|v^{-3} - v\right\|^2 + \|d_x d_s\| \\ &\leq \frac{\delta^2}{2} + \frac{\sqrt{1 + (1 + 4\kappa')^2}}{16}\delta^2 \\ &= \left(\frac{1}{2} + \frac{\sqrt{1 + (1 + 4\kappa')^2}}{16}\right)\delta^2. \end{aligned} \tag{4.6}$$

Hence

$$\delta(v_+) \leq \left(\frac{1}{\left(1 - \frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}} + \frac{1}{\left(1 - \frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2} + \frac{\sqrt{1 + (1 + 4\kappa')^2}}{16}\right)\delta^2.$$

The proof is now complete. □

Let $v^p = \sqrt{\frac{x^p s^p}{\omega^p(t)}}$, where $\omega^p(t) = \left(1 - \frac{1}{2}\vartheta t\right)\omega(t)$. The next lemma gives an upper bound for the proximity measure after updating $\omega(t)$ and carrying out the predictor step.

Lemma 6. *Provided that $\delta < \frac{4}{\sqrt{1+4\kappa'}}$, one has*

$$\left\|d_x^p d_s^p\right\| \leq \frac{n(1 + 2\kappa')}{8}\left[1 + \left(\frac{1}{\left(1 - \frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}} + \frac{1}{\left(1 - \frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2} + \frac{\sqrt{1 + (1 + 4\kappa')^2}}{16}\right)\delta^2\right]^2.$$

Proof. Applying the second equation of (3.2) yields

$$\frac{1}{4}\|v_+\|^2 = \sum_{i=1}^{n}\left(d_{x_i}^p + d_{s_i}^p\right)^2 \geq \sum_{i\in\Im_+}\left(d_{x_i}^p + d_{s_i}^p\right)^2 \geq 4\sum_{i\in\Im_+} d_{x_i}^p d_{s_i}^p.$$

From the above relation, Lemma 1, and Lemma 5, we get

$$
\begin{aligned}
2\left\|d_x^p d_s^p\right\| &\leq \left\|d_x^p\right\|^2+\left\|d_s^p\right\|^2 \leq \left\|d_x^p+d_s^p\right\|^2-2\left(\sum_{\mathrm{i}\in\Im_+} d_{x_\mathrm{i}}^p d_{s_\mathrm{i}}^p+\sum_{\mathrm{i}\in\Im_-} d_{x_\mathrm{i}}^p d_{s_\mathrm{i}}^p\right)\\
&\leq \frac{\|v_+\|^2}{4}+8\kappa'\sum_{\mathrm{i}\in\Im_+} d_{x_\mathrm{i}}^p d_{s_\mathrm{i}}^p \leq \frac{(1+2\kappa')\|v_+\|^2}{4} \leq \frac{n(1+2\kappa')(1+\delta(v_+))^2}{4}\\
&\leq \frac{n(1+2\kappa')}{4}\left[1+\left(\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}}+\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2}+\frac{\sqrt{1+(1+4\kappa')^2}}{16}\right)\delta^2\right]^2.
\end{aligned}
$$

This finishes the proof. □

Lemma 7. *Given that* $(x^+, s^+) > 0$ *and* $\vartheta \leq \frac{1}{\sqrt{n(1+2\kappa')}}$ *with* $n \geq 2$*, and* $\delta \leq \frac{1}{\sqrt{2(1+4\kappa')}}$*, then* (x^p, s^p) *is positive, if* $\varrho(\delta,\vartheta,n) > 0$*, where*

$$
\begin{aligned}
\varrho(\delta,\vartheta,n) = 1-\frac{1+4\kappa'}{16}\delta^2 \\
-\frac{n(1+2\kappa')\vartheta^2t^2}{4(2-\vartheta t)}\left[1+\left(\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}}+\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2}+\frac{\sqrt{1+(1+4\kappa')^2}}{16}\right)\delta^2\right]^2.
\end{aligned}
$$

Proof. Let $0 \leq \beta \leq 1$. Using (3.2)–(3.4), we get

$$
\begin{aligned}
x^p(\beta)s^p(\beta) &= \left(x^+ + \beta\vartheta t\Delta_x^p\right)\left(s^+ + \beta\vartheta t\Delta_s^p\right)\\
&= \left(x^+ + \beta\vartheta t\frac{x^+}{v_+}d_x^p\right)\left(s^+ + \beta\vartheta t\frac{s^+}{v_+}d_s^p\right)\\
&= \frac{x^+s^+}{v_+^2}\left(v_+ + \beta\vartheta t d_x^p\right)\left(v_+ + \beta\vartheta t d_s^p\right)\\
&= \omega(t)\left[v_+^2+\beta\vartheta t v_+\left(d_x^p+d_s^p\right)+\beta^2\vartheta^2t^2d_x^pd_s^p\right]\\
&= \omega(t)\left[\left(1-\frac{1}{2}\beta\vartheta t\right)v_+^2+\beta^2\vartheta^2t^2d_x^pd_s^p\right].
\end{aligned}
\tag{4.7}
$$

From (4.5), (4.7), and Lemma 6, we get

$$
\begin{aligned}
&\min\frac{x^p(\beta)s^p(\beta)}{\omega(t)\left(1-\frac{1}{2}\beta\vartheta t\right)} \geq \min v_+^2-\frac{2\beta^2\vartheta^2t^2}{2-\beta\vartheta t}\left\|d_x^pd_s^p\right\| \geq \min v_+^2-\frac{2\vartheta^2t^2}{2-\vartheta t}\left\|d_x^pd_s^p\right\| \geq 1-\frac{1+4\kappa'}{16}\delta^2\\
&-\frac{n(1+2\kappa')\vartheta^2t^2}{4(2-\vartheta t)}\left[1+\left(\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}}+\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2}+\frac{\sqrt{1+(1+4\kappa')^2}}{16}\right)\delta^2\right]^2\\
&:= \varrho(\delta,\vartheta,n).
\end{aligned}
\tag{4.8}
$$

Suppose $\vartheta \leq \frac{1}{\sqrt{n(1+2\kappa')}}$, $\delta \leq \frac{1}{\sqrt{2(1+4\kappa')}}$, and $n \geq 2$ and we obtain

$$\begin{aligned}
\varrho(\delta, \vartheta, n) &\geq 1 - \frac{1}{32} - \frac{n(1+2\kappa')\vartheta^2 t^2}{4(2-\vartheta t)}\left[1 + \left(\frac{1}{\left(1-\frac{1}{32}\right)^{\frac{1}{2}}} + \frac{1}{\left(1-\frac{1}{32}\right)^{\frac{3}{2}}}\right)\frac{8+\sqrt{1+(1+4\kappa')^2}}{32}\right]^2 \\
&\geq \frac{31}{32} - \frac{25n(1+2\kappa')\vartheta^2 t^2}{36(2-\vartheta t)} \geq \frac{31}{32} - \frac{25n(1+2\kappa')}{36(1-\vartheta)} \cdot \frac{1}{n(1+2\kappa')} \\
&\geq \frac{31}{32} - \frac{25}{36} \cdot \frac{\sqrt{n(1+2\kappa')}}{2\sqrt{n(1+2\kappa')}-1} \geq \frac{31}{32} - \frac{25}{36} \cdot \frac{\sqrt{2}}{2\sqrt{2}-1} > 0.
\end{aligned}$$

Therefore, $\varrho(\delta, \vartheta, n) > 0$ holds for $\kappa' \geq 0$ and $n \geq 2$. It follows that $x^p(\beta)\, s^p(\beta) > 0$ for all $0 \leq \beta \leq 1$ and $0 < \vartheta \leq \frac{1}{\sqrt{n(1+2\kappa')}}$. Consequently, $x^p(\beta) > 0$, $s^p(\beta) > 0$ for all $0 \leq \beta \leq 1$. With $x^p(0) = x^+ > 0$ and $s^p(0) = s^+ > 0$, we conclude that $x^p(1) = x^p > 0$ and $s^p(1) = s^p > 0$. □

The following lemma is devoted to obtaining an upper bound for the proximity measure $\delta(v^p) := \delta(x^p, s^p; \omega^p(t))$ by following a predictor step.

Lemma 8. *Suppose that* $\delta \leq \frac{1}{\sqrt{2(1+4\kappa')}}$ *and* $\omega^p(t) = (1 - \frac{1}{2}\vartheta t)\omega(t)$ *with* $\vartheta \leq \frac{1}{\sqrt{n(1+2\kappa')}}$. *Then it holds that*

$$\delta(v^p) \leq \left(\varrho(\delta, \vartheta, n)^{-\frac{1}{2}} + \varrho(\delta, \vartheta, n)^{-\frac{3}{2}}\right)\left(\frac{9}{16}\delta^2 - \varrho(\delta, \vartheta, n) + 1\right),$$

where $\varrho(\delta, \vartheta, n) = 1 - \frac{1+4\kappa'}{16}\delta^2 - \frac{n(1+2\kappa')\vartheta^2 t^2}{4(2-\vartheta t)}\left[1 + \left(\frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{1}{2}}} + \frac{1}{\left(1-\frac{1+4\kappa'}{16}\delta^2\right)^{\frac{3}{2}}}\right)\left(\frac{1}{2} + \frac{\sqrt{1+(1+4\kappa')^2}}{16}\right)\delta^2\right]^2$.

Proof. Applying Lemma 5 together with the expression for $\delta(v)$ yields

$$\begin{aligned}
\delta(v^p) &= \left\|(v^p)^{-3} - v^p\right\| \\
&\leq \left\|\frac{e + (v^p)^2}{(v^p)^3} \cdot \left(e - (v^p)^2\right)\right\| \\
&\leq \left(\frac{1}{\min v^p} + \frac{1}{\min(v^p)^3}\right) \cdot \left\|e - (v^p)^2\right\|.
\end{aligned} \tag{4.9}$$

Using (4.6)-(4.8), we derive that

$$\begin{aligned}
\left\|e - (v^p)^2\right\| &= \left\|e - \frac{x^p(1)\, s^p(1)}{\omega^p(t)}\right\| \\
&= \left\|e - v_+^2 - \frac{\vartheta^2 t^2 d_x^p d_s^p}{1 - \frac{1}{2}\vartheta t}\right\| \\
&\leq \left\|e - v_+^2\right\| + \frac{2\vartheta^2 t^2}{2 - \vartheta t}\left\|d_x^p d_s^p\right\| \\
&\leq \left(\frac{1}{2} + \frac{\sqrt{1+(1+4\kappa')^2}}{16}\right)\delta^2 - \frac{1+4\kappa'}{16}\delta^2 - \varrho(\delta, \vartheta, n) + 1 \\
&\leq \frac{9}{16}\delta^2 - \varrho(\delta, \vartheta, n) + 1.
\end{aligned} \tag{4.10}$$

Therefore, by (4.9) and (4.10), it follows that

$$\delta\left(v^{p}\right) \leq \left(\varrho(\delta,\vartheta,n)^{-\frac{1}{2}}+\varrho(\delta,\vartheta,n)^{-\frac{3}{2}}\right)\left(\frac{9}{16}\delta^{2}-\varrho\left(\delta,\vartheta,n\right)+1\right). \tag{4.11}$$

□

Lemma 9. *Let* $t_p=(1-\vartheta)\,t$ *and* $\vartheta\leq\frac{t}{4\sqrt{n}(1+2\kappa')}$ *with* $n\geq 2$*. If* $\delta\left(v\right)\leq\frac{t}{2(1+4\kappa')}$*, then* $\delta\left(v^{p}\right)\leq\frac{t_p}{2(1+4\kappa')}$*.*

Proof. Lemma 8 implies that

$$\delta\left(v^{p}\right)\leq\frac{t_p}{2\left(1+4\kappa'\right)} \tag{4.12}$$

holds, if

$$\delta\left(v^{p}\right) \leq \left(\varrho(\delta,\vartheta,n)^{-\frac{1}{2}}+\varrho(\delta,\vartheta,n)^{-\frac{3}{2}}\right)\left(\frac{9}{16}\delta^{2}-\varrho\left(\delta,\vartheta,n\right)+1\right)\leq\frac{t_p}{2\left(1+4\kappa'\right)}. \tag{4.13}$$

Define

$$H=\left(\varrho(\delta,\vartheta,n)^{-\frac{1}{2}}+\varrho(\delta,\vartheta,n)^{-\frac{3}{2}}\right)\left(\frac{9}{16}\delta^{2}-\varrho\left(\delta,\vartheta,n\right)+1\right)-\frac{t_p}{2\left(1+4\kappa'\right)}.$$

Since $\varrho\left(\delta,\vartheta,n\right)$ is increasing for $t\in(0,1]$ and $\delta\left(v\right)\leq\frac{t}{2(1+4\kappa')}$, we have

$$\begin{aligned}
\varrho\left(\frac{t}{2\left(1+4\kappa'\right)},\vartheta,n\right) &\geq 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{n(1+2\kappa')\vartheta^{2}t^{2}}{4\left(2-\vartheta t\right)}\\
&\quad\times\left[1+\left(\frac{1}{\left(1-\frac{t^{2}}{64(1+4\kappa')}\right)^{\frac{1}{2}}}+\frac{1}{\left(1-\frac{t^{2}}{64(1+4\kappa')}\right)^{\frac{3}{2}}}\right)\frac{8+\sqrt{1+\left(1+4\kappa'\right)^{2}}}{64\left(1+4\kappa'\right)}\right]^{2}\\
&> 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{n(1+2\kappa')\vartheta^{2}t^{2}}{4\left(2-\vartheta t\right)}\times\\
&\quad\left[1+\frac{5}{32}\left(\left(\frac{64\left(1+4\kappa'\right)}{64\left(1+4\kappa'\right)-1}\right)^{\frac{1}{2}}+\left(\frac{64\left(1+4\kappa'\right)}{64\left(1+4\kappa'\right)-1}\right)^{\frac{3}{2}}\right)\right]^{2}\\
&\geq 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{n(1+2\kappa')\vartheta^{2}t^{2}}{4\left(2-\vartheta t\right)}\left[1+\frac{23}{64}\right]^{2}\\
&\geq 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{7569n(1+2\kappa')\vartheta^{2}t^{2}}{16384\left(2-\vartheta t\right)}.
\end{aligned}$$

If $\vartheta\leq\frac{t}{4\sqrt{n}(1+2\kappa')}$ and $n\geq 2$, then

$$\begin{aligned}
\varrho\left(\frac{t}{2\left(1+4\kappa'\right)},\frac{t}{4\sqrt{n}\left(1+2\kappa'\right)},n\right) &\geq 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{7569n(1+2\kappa')t^{2}}{16384\left(2-\vartheta t\right)}\cdot\frac{t^{2}}{16n\left(1+2\kappa'\right)^{2}}\\
&\geq 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{7569t^{2}}{16384\left(2-\frac{t}{4\sqrt{n}(1+2\kappa')}\right)}\cdot\frac{t^{2}}{16\left(1+2\kappa'\right)}\\
&\geq 1-\frac{t^{2}}{64\left(1+4\kappa'\right)}-\frac{7569t^{2}}{262144\left(1+4\kappa'\right)}\\
&\geq 1-\frac{11665t^{2}}{262144\left(1+4\kappa'\right)}.
\end{aligned}$$

Using the fact that $t_p = (1-\vartheta)\,t$, we get

$$
\begin{aligned}
H &= \left(\varrho(\delta,\vartheta,n)^{-\frac{1}{2}} + \varrho(\delta,\vartheta,n)^{-\frac{3}{2}}\right)\left(\frac{9}{16}\delta^2 - \varrho\,(\delta,\vartheta,n) + 1\right) - \frac{t_p}{2\,(1+4\kappa')}\\
&\le \left(\frac{1}{\left(1-\frac{11665t^2}{262144(1+4\kappa')}\right)^{\frac{1}{2}}} + \frac{1}{\left(1-\frac{11665t^2}{262144(1+4\kappa')}\right)^{\frac{3}{2}}}\right)\\
&\times\left(\frac{9t^2}{64\,(1+4\kappa')^2} + \frac{11665t^2}{262144\,(1+4\kappa')}\right) - \frac{(1-\vartheta)\,t}{2\,(1+4\kappa')}\\
&\le \left(\left(\frac{262144\,(1+4\kappa')}{262144\,(1+4\kappa')-11665}\right)^{\frac{1}{2}} + \left(\frac{262144\,(1+4\kappa')}{262144\,(1+4\kappa')-11665}\right)^{\frac{3}{2}}\right)\\
&\times \frac{36864 + 11665\,(1+4\kappa')}{262144\,(1+4\kappa')^2} - \frac{1-\frac{t}{4\sqrt{n}(1+2\kappa')}}{2\,(1+4\kappa')}\\
&\le 2.0938\times 0.1852 - \frac{4\sqrt{2}\,(1+2\kappa')-1}{8\sqrt{2}\,(1+2\kappa')\,(1+4\kappa')}\\
&\le 0.3878 - 0.4116 < 0,
\end{aligned}
$$

which implies that (4.12) holds. □

4.3. Complexity bound

The objective of this subsection is to establish an upper bound for the number of iterations Algorithm 1 needs.

Lemma 10. *If* $\delta\,(v) \le \frac{t}{2(1+4\kappa')}$ *and* $\vartheta \le \frac{t}{4\sqrt{n}(1+2\kappa')}$ *with* $n \ge 2$*, then for* $t \in (0,1]$*, we have*

$$
\|x^p s^p - \omega\| \le \left[\frac{5}{16\,(1+4\kappa')}\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\|\right] t.
$$

Proof. (2.3) and (4.10) imply that

$$
\begin{aligned}
\|x^p s^p - \omega\| &= \|x^p s^p - \omega + \omega^p\,(t) - \omega^p\,(t) + \omega\,(t) - \omega\,(t)\|\\
&\le \|x^p s^p - \omega^p\,(t)\| + \|\omega\,(t) - \omega\| + \|\omega^p\,(t) - \omega\,(t)\|\\
&\le \left\|e - (v^p)^2\right\|\cdot\|\omega^p\,(t)\|_\infty + \left\|x^0 s^0 - \omega\right\| t + \frac{1}{2}\vartheta t\,\|\omega\,(t)\|\\
&\le \left(\frac{9}{16}\delta^2 - \varrho\,(\delta,\vartheta,n) + 1 + \frac{1}{2}\vartheta\sqrt{n}t\right)\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\| t.
\end{aligned}
$$

If $\vartheta_{max} = \frac{t}{4\sqrt{n}(1+2\kappa')}$ with $t \in (0,1]$, and $\delta(v) \le \frac{t}{2(1+4\kappa')}$, we get

$$
\begin{aligned}
\|x^p s^p - \omega\| &\le \left(\frac{9}{16}\delta^2 - \varrho\,(\delta,\vartheta,n) + 1 + \frac{1}{2}\vartheta\sqrt{n}t\right)\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\| t\\
&\le \left(\frac{9t^2}{64(1+4\kappa')^2} + \frac{11665t^2}{262144\,(1+4\kappa')} + \frac{t}{8\,(1+2\kappa')}\right)\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\| t
\end{aligned}
$$

$$\leq \left(\frac{9}{64(1+4\kappa')} + \frac{11665}{262144(1+4\kappa')} + \frac{1}{8}\right)\max\left(x^0 s^0\right) t + \left\|x^0 s^0 - \omega\right\| t$$
$$\leq \left[\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\|\right] t.$$

Then the proof is complete. □

Lemma 11. *Consider x^0 and s^0 as strictly feasible starting points. Assume that $\vartheta \leq \frac{t}{4\sqrt{n}(1+2\kappa')}$ with $n \geq 2$. Moreover, suppose that the iterates after k iterations are denoted by x^k and s^k. Then we have $\left\|x^k s^k - \omega\right\| \leq \varepsilon$ for all*

$$k \geq \left\lceil \vartheta^{-1} \log \frac{\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \eta}{\varepsilon} \right\rceil + 1,$$

where $\eta = \left\|x^0 s^0 - \omega\right\|$.

Proof. Let $t_0 = 1$ and $t_+ = (1-\vartheta)\,t$. From Lemma 10, we obtain the following

$$\left\|x^k s^k - \omega\right\| \leq \left[\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\|\right] t_{k-1}$$
$$\leq \left[\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\|\right] (1-\vartheta)^{k-1}.$$

Then we have $\left\|x^k s^k - \omega\right\| \leq \varepsilon$ whenever

$$\left[\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \left\|x^0 s^0 - \omega\right\|\right] (1-\vartheta)^{k-1} \leq \varepsilon.$$

Applying the logarithm to the last inequality yields

$$k \geq \left\lceil \vartheta^{-1} \log \frac{\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \eta}{\varepsilon} \right\rceil + 1,$$

using the fact that $\log(1-\alpha) \leq -\alpha$ with $\alpha \in (0,1)$, where $\eta = \left\|x^0 s^0 - \omega\right\|$. □

Theorem 1. *Let $\left(x^0, s^0\right) \in \mathcal{F}^0$ and $\vartheta = \frac{t}{4\sqrt{n}(1+2\kappa')}$ with $n \geq 2$. Then Algorithm 1 for solving the sufficient WLCP is well defined, and it requires at most*

$$O\left(\sqrt{n}\,(1+2\kappa')\log \frac{\frac{5}{16(1+4\kappa')}\max\left(x^0 s^0\right) + \eta}{\varepsilon}\right)$$

iterations-, such that the output generated by Algorithm 1 consists of a pair (x, s) satisfying $\|xs - \omega\| \leq \varepsilon$.

Proof. Since $t \in (0, 1]$ and $\vartheta = \frac{t}{4\sqrt{n}(1+2\kappa')}$, the desired result follows from Lemma 11. □

5. Numerical results

The efficiency of Algorithm 1 is illustrated in this section through numerical experiments performed with sufficient WLCPs. The numerical experiments utilized a personal computer with specifications including an Intel (R) Core (TM) i5-1130G7 @ 1.10 GHz CPU and 16 GB of RAM. The operating system used was Windows 11 Enterprise 64-bit, and the algorithm was implemented in MATLAB (R2023a). We use accuracy parameter $\varepsilon = 10^{-5}$ and let $x^0 s^0 > \omega$. In each case, we solve the problems 10 times. The average iteration count (**iter**) and CPU time in seconds (**CPU/s**) are then computed to reduce the experimental error.

Problem 1. *[30] Consider the sufficient WLCP (2.1) given in $\mathbb{R}^{4\times 4}$, where*

$$M = \begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 2 & 0 & 1 \\ 1 & 0 & 1 & 2 \\ -1 & -1 & -2 & 0 \end{pmatrix}, \quad q = \begin{pmatrix} -4 \\ -3 \\ -3 \\ 5 \end{pmatrix}, \quad \omega = \left(x^0 s^0\right)/2.$$

We select $x^0 = s^0 = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}^T$ as the strictly feasible starting point. The update parameter is set to $\vartheta = 0.5$. Algorithm 1 yields a unique solution to Problem 1,

$$x^* = \begin{pmatrix} 0.8596 & 0.9873 & 1.2074 & 0.6749 \end{pmatrix}^T,$$

$$s^* = \begin{pmatrix} 0.5835 & 0.5064 & 0.4141 & 0.7409 \end{pmatrix}^T,$$

with 18 iterations and 0.0013 seconds.

Figures 1–4 illustrate how the values of $x_i s_i$ $(i = 1, 2, 3, 4)$ change with each iteration when Algorithm 1 is applied to solve Problem 1. From Figures 1–4, with the increase of iteration number, the values of $x_i s_i$ gradually decrease from 1 to 0.5, approaching $xs = \omega$. In addition, the value of $x_i s_i$ decreases monotonically with the growth of the iteration number. This result shows that during the iterative process, the iteration point (x, s) approaches the ε-approximate solution of Problem 1, which demonstrates the feasibility and convergence of our algorithm.

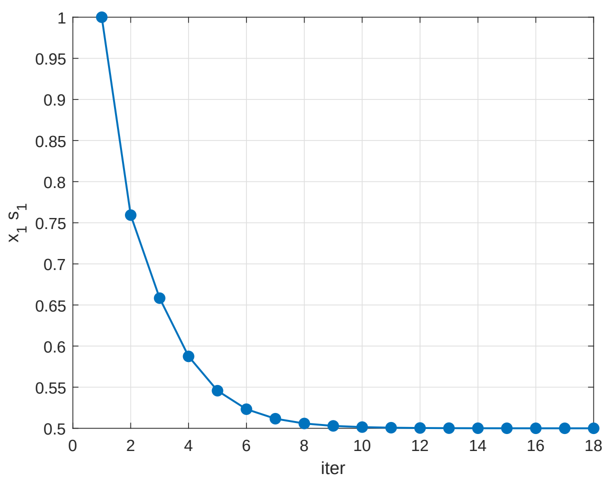

Figure 1. Convergence of $x_1 s_1$ for Problem 1.

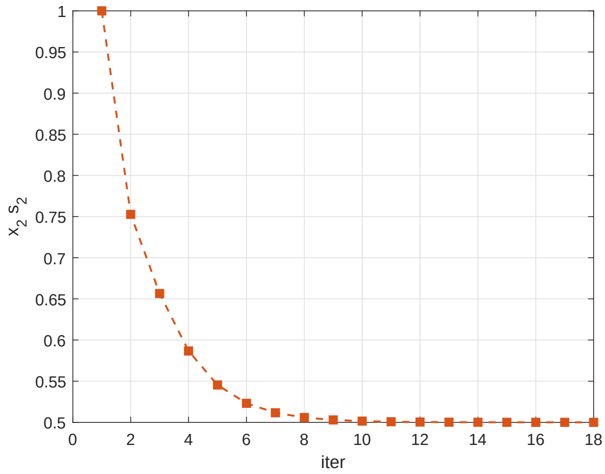

Figure 2. Convergence of x_2s_2 for Problem 1.

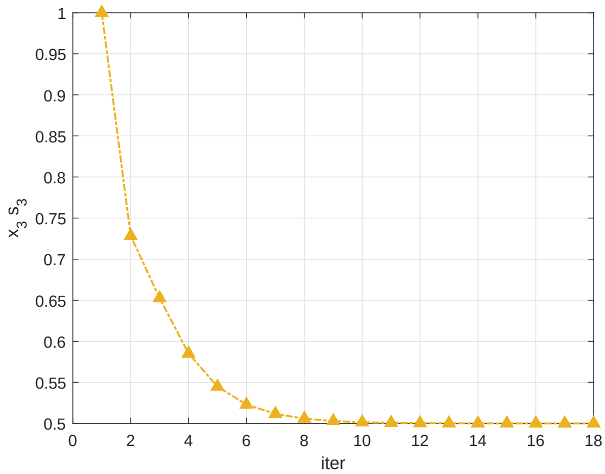

Figure 3. Convergence of x_3s_3 for Problem 1.

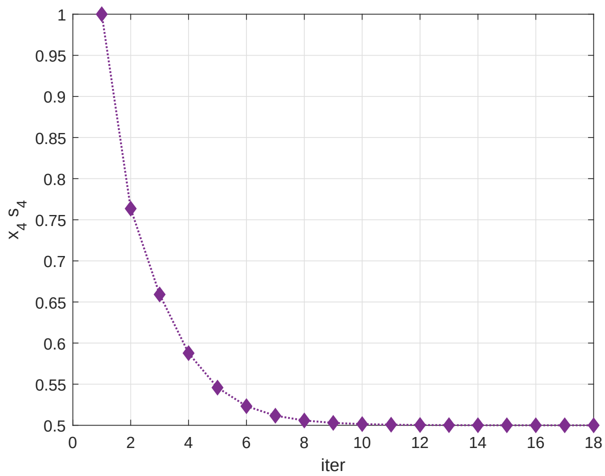

Figure 4. Convergence of x_4s_4 for Problem 1.

Problem 2. *Consider the sufficient WLCP (2.1) with random matrices. Choose a strictly feasible starting point* $x^0 = e$ *and* $s^0 = e$*. Let the weight vector be* $\omega = 0.5x^0s^0$*. The parameter* ϑ *is considered in the set* $\vartheta \in \{0.2, 0.4, 0.6, 0.8\}$*, and the dimensions of this problem are considered in the set* $n \in \{50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1100, 1200\}$*. The obtained results of Problem 2 for different* θ *and* n *are shown in Table 1.*

Table 1. Numerical results for Problem 2 with various ϑ and n.

n	$\vartheta = 0.2$		$\vartheta = 0.4$		$\vartheta = 0.6$		$\vartheta = 0.8$	
	CPU/s	**iter**	**CPU/s**	**iter**	**CPU/s**	**iter**	**CPU/s**	**iter**
50	0.0153	59	0.0067	27	0.0036	16	0.0022	10
100	0.0686	61	0.0237	28	0.0148	17	0.0097	11
200	0.3769	62	0.1450	29	0.0789	17	0.0560	11
300	0.7443	63	0.3163	29	0.1966	17	0.1110	11
400	1.4597	64	0.6414	29	0.3653	17	0.2396	11
500	2.3126	65	1.0803	30	0.6627	18	0.3675	11
600	3.9485	65	1.7597	30	1.0654	18	0.6598	11
700	5.4829	65	2.4240	30	1.5515	18	0.9229	11
800	7.3330	66	3.2855	30	2.2712	18	1.5867	11
900	9.6274	66	5.5304	30	3.5139	18	2.0710	11
1000	12.6135	66	6.8066	30	4.7147	18	2.4683	11
1100	15.7445	66	7.7856	30	5.3340	18	2.9523	11
1200	18.1110	67	10.6828	30	6.6961	18	3.2859	11

Table 1 presents numerical results with different ϑ *and* n *for Problem 2. Experimental results show that the iteration count and runtime required by the algorithm increase with the expansion of dimensions* n*. Meanwhile, as the parameter* ϑ *increases, both iterations and CPU time exhibit a decreasing trend.*

Problem 3. *Consider the sufficient WLCP (2.1) with*

$$M = \begin{pmatrix} 3 & -2 & 1 & 0 & \cdots & 0 \\ -2 & 3 & 2 & 1 & \cdots & 0 \\ 1 & -2 & 3 & -2 & \cdots & 0 \\ 0 & 1 & -2 & 3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 3 \end{pmatrix}, \quad \omega = \left(x^0 s^0\right)/2.$$

Let $x^0 = s^0 = e$ *be a strictly feasible starting point, and let* $q = -Mx^0 + s^0$*. The update parameter is* $\vartheta \in \{0.3, 0.5, 0.7\}$*, and the dimension is* $n \in \{50, 100, 200, 300, 500, 700, 900, 1200, 1500\}$*. The obtained results are presented in Table 2. In addition, with* $\omega = 0.15x^0s^0$*, the update parameter* $\vartheta = 0.1$*, and the dimensions* $n \in \{100, 200, 300, 400, 500, 600\}$*, the numerical results are illustrated in Figures 5 and 6.*

It is obvious from Table 2 that the iteration count and runtime depend on ϑ *and* n*. With the increase of* n *in Problem 3, the CPU time grows, while the iteration count grows slowly. As the parameter* ϑ

increases, the iteration count and CPU time needed by the algorithm both decrease. Clearly, a smaller update parameter ϑ leads to a larger iteration count. In addition, the choice of ω affects the value of η. When $\omega = 0.5x^0s^0$, the algorithm yields a larger η than in the case of $\omega = 0.8x^0s^0$, which results in slightly more iterations and CPU time. As ϑ increases, the difference in iteration count and runtime between the two cases of ω gradually decreases. Meanwhile, the case of $\omega = 0.8x^0s^0$ in our test cases exhibits fewer iterations and less runtime.

Figure 5 displays the change in the proximity measure over different iterations for Algorithm 1. This figure shows that the proximity measure $\delta(v)$ approaches zero after peaking, which demonstrates the convergence of Algorithm 1. Figure 6 shows the values of duality gap in Problem 3 and illustrates the convergence of our algorithm for different n. When t tends to zero, the values of gap reduce to 0, indicating that Algorithm 1 delivers an ε-approximate solution for the sufficient WLCP.

Table 2. Numerical results for Problem 3 with different ϑ and n.

n	ϑ	$\omega = 0.5x^0s^0$		$\omega = 0.8x^0s^0$	
		CPU/s	**iter**	**CPU/s**	**iter**
	0.3	0.0124	38	0.0098	34
50	0.5	0.0055	21	0.0047	19
	0.7	0.0035	13	0.0023	12
	0.3	0.0351	39	0.0309	35
100	0.5	0.0102	21	0.0091	19
	0.7	0.0084	13	0.0071	12
	0.3	0.2366	40	0.2616	36
200	0.5	0.1180	22	0.1001	20
	0.7	0.0758	14	0.0594	12
	0.3	0.4185	41	0.3741	37
300	0.5	0.2472	22	0.2155	20
	0.7	0.1335	14	0.1180	13
	0.3	1.8109	41	1.6008	37
500	0.5	0.7348	22	0.6508	20
	0.7	0.4925	14	0.4098	13
	0.3	4.3486	42	3.9600	38
700	0.5	2.2611	23	1.7568	21
	0.7	1.3212	14	1.0088	13
	0.3	7.3212	42	6.6226	38
900	0.5	3.6966	23	3.1083	21
	0.7	2.2720	14	1.9161	13
	0.3	13.0031	42	12.1331	39
1200	0.5	7.6780	23	6.1110	21
	0.7	4.6128	14	3.8178	13
	0.3	22.6456	43	21.1768	39
1500	0.5	14.0065	23	12.8879	21
	0.7	7.6881	14	6.1934	13

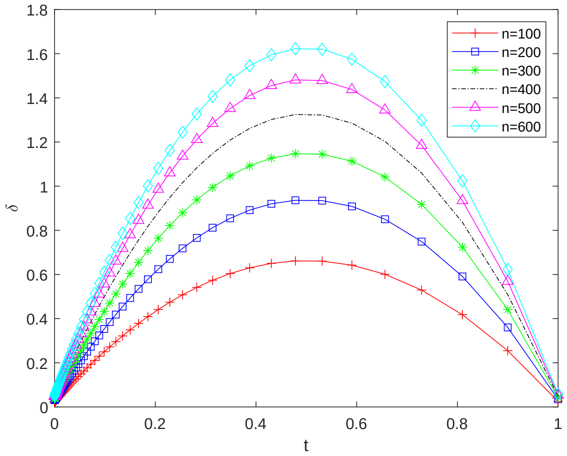

Figure 5. The value of $\delta(v)$ for Problem 3.

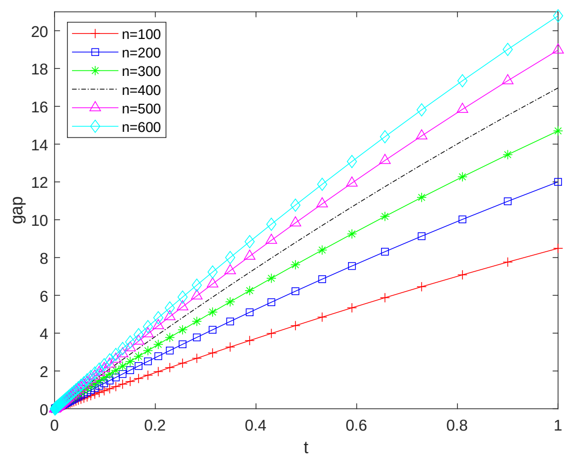

Figure 6. The value of gap for Problem 3.

6. Conclusions

We introduce a new PC IPM to solve the sufficient WLCP. We first employ the kernel function

$$\varphi(t) = \frac{t^{\mathfrak{p}+1} - 1}{\mathfrak{p} + 1} + \frac{t^{1-\mathfrak{q}} - 1}{\mathfrak{q} - 1}$$

with $0 \leq \mathfrak{p} \leq 1$ and $\mathfrak{q} > 1$. Specifically, using the parameters $\mathfrak{p} = 1$ and $\mathfrak{q} = 3$ in the kernel function, we propose a PC IPM for solving the sufficient WLCP. Compared to the sufficient LCP, the analysis of our sufficient WLCP is more complicated due to the presence of a weight vector $\omega \neq 0$. The proposed algorithm exhibits strict feasibility and convergence, which are proven through the selection of suitable initial points and parameters. The derivation of Algorithm 1 shows it has polynomial complexity, which is on par with the best-known complexity bound for IPMs. Ultimately, numerical results provide evidence of our algorithm's efficiency and reliability.

Author contributions

Zhuoran Gao: Methodology, Conceptualization, Formal analysis, Writing–original draft ; Hongjin Wei: Validation, Formal analysis; Suobin Zhang: Writing–review and editing; Xiaoni Chi: Methodology, Conceptualization, Supervision. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (No. 12361064), the Science and Technology Project of Guangxi of China (Guike AD25069086) and Guangxi Natural Science Foundation (No. 2021GXNSFAA220034). The second author's work is supported by Scientific Research Foundation of Guangxi Science and Technology Normal University of China (GXKS2024YB029).

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. F. A. Potra, Weighted complementarity problems-a new paradigm for computing equilibria, *SIAM J. Optim.*, **22** (2012), 1634–1654. http://doi.org/10.1137/110837310
2. J. Y. Tang, A variant nonmonotone smoothing algorithm with improved numerical results for large-scale LWCPs, *Comput. Appl. Math.*, **37** (2018), 3927–3936. http://doi.org/10.1007/s40314-017-0554-6
3. X. N. Chi, M. S. Gowda, J. Y. Tao, The weighted horizontal linear complementarity problem on a Euclidean Jordan algebra, *J. Glob. Optim.*, **73** (2019), 153–169. http://doi.org/10.1007/s10898-018-0689-z
4. F. A. Potra, Sufficient weighted complementarity problems, *Comput. Optim. Appl.*, **64** (2016), 467–488. http://doi.org/10.1007/s10589-015-9811-z
5. J. Zhang, A smoothing Newton algorithm for weighted linear complementarity problem, *Optim. Lett.*, **10** (2016), 499–509. http://doi.org/10.1007/s11590-015-0877-4
6. N. Karmarkar, A new polynomial-time algorithm for linear programming, *Combinatorica*, **4** (1984), 373–395. http://doi.org/10.1007/BF02579150
7. S. Asadi, Z. Darvay, G. Lesaja, N. Mahdavi-Amiri, F. A. Potra, A full-Newton step interior-point method for monotone weighted linear complementarity problems, *J. Optim. Theory Appl.*, **186** (2020), 864–878. http://doi.org/10.1007/s10957-020-01728-4

8. X. N. Chi, Z. P. Wan, Z. J. Hao, A full-modified-Newton step $O(n)$ infeasible interior-point method for the special weighted linear complementarity problem, *J. Ind. Manag. Optim.*, **18** (2022), 2579–2598. http://doi.org/10.3934/jimo.2021082

9. X. N. Chi, G. Q. Wang, A full-Newton step infeasible interior-point method for the special weighted linear complementarity problem, *J. Optim. Theory Appl.*, **190** (2021), 108–129. http://doi.org/10.1007/s10957-021-01873-4

10. S. Mehrotra, On the implementation of a primal-dual interior point method, *SIAM J Optim.*, **2** (1992), 575–601. http://doi.org/10.1137/0802028

11. G. Sonnevend, J. Stoer, G. Zhao, On the complexity of following the central path of linear programs by linear extrapolation II, *Math. Program.*, **52** (1991), 527–553. http://doi.org/10.1007/BF01582904

12. S. Mizuno, M. J. Todd, Y. Ye, On adaptive-step primal-dual interior-point algorithms for linear programming, *Math. Oper. Res.*, **18** (1993), 964–981. http://doi.org/10.1287/moor.18.4.964

13. F. A. Potra, R. Sheng, Predictor-corrector algorithm for solving $P_*(\kappa)$-matrix LCP from arbitrary positive starting points, *Math. Program.*, **76** (1996), 223–244. http://doi.org/10.1007/bf02614385

14. Z. Darvay, A predictor-corrector algorithm for linearly constrained convex optimization, *Stud. univ. babes-Bolyai Inform.*, **54** (2009), 121–138.

15. B. Kheirfam, A corrector-predictor path-following method for Convex quadratic symmetric cone optimization, *J. Optim. Theory Appl.*, **164** (2015), 246–260. https://doi.org/10.1007/s10957-014-0554-2

16. B. Kheirfam, A corrector-predictor path-following method for second-order cone optimization, *Int. J. Comput. Math.*, **93** (2016), 2064–2078. https://doi.org/10.1080/00207160.2015.1085028

17. L. Zhang, X. N. Chi, S. B. Zhang, Y. P. Yang, A predictor-corrector interior-point algorithm for $P_*(\kappa)$-weighted linear complementarity problems, *AIMS Math.*, **8** (2023), 2064–2078. http://doi.org/10.3934/math.2023462

18. F. A. Potra, Y. Y. Ye, Interior-point methods for nonlinear complementarity problems, *J. Optim. Theory Appl.*, **88** (1996), 617–647. http://doi.org/10.1007/BF02192201

19. X. R. He, J. Y. Tang, A smooth Levenberg-Marquardt method without nonsingularity condition for wLCP, *AIMS Math.*, **7** (2022), 8914–8932. http://doi.org/10.3934/math.2022497

20. Z. Darvay, New interior point algorithms in linear programming, *Adv. Model. Optim.*, **5** (2003), 51–92.

21. M. Achache, A new primal-dual path-following method for convex quadratic programming, *Comput. Appl. Math.*, **25** (2006), 97–110. http://doi.org/10.1590/S0101-82052006000100005

22. G. Q. Wang, A new polynomial interior-point algorithm for the monotone linear complementarity problem over symmetric cones with full NT-steps, *Asia-Pacific J. Oper. Res.*, **29** (2012), 1250015. http://doi.org/10.1142/S0217595912500157

23. L. P. Zhang, Y. Q. Bai, Y. H. Xu, A full-Newton step infeasible interior-point algorithm for monotone LCP based on a locally-kernel function, *Numer. Algor.*, **61** (2012), 57–81. http://doi.org/10.1007/s11075-011-9530-1

24. H. Mansouri, M. Pirhaji, A polynomial interior-point algorithm for monotone linear complementarity problems, *J. Optim. Theory Appl.*, **157** (2013), 451–461. http://doi.org/10.1007/s10957-012-0195-2

25. G. Lesaja, C. Roos, Unified analysis of Kernel-based interior-point methods for $P_*(\kappa)$-linear complementary problems, *SIAM J Optim.*, **20** (2010), 3014–3039. https://doi.org/10.1137/090766735

26. M. Kojima, N. Megiddo, T. Noma, A. Yoshise, A unified approach to interior point algorithms for linear complementarity problems, *Oper. Res. Lett.*, **10** (1991), 247–254. https://doi.org/10.1007/3-540-54509-3

27. H. Väliaho, Criteria for sufficient matrices, *Linear Algebra Appl.*, **233** (1996), 109–129. http://doi.org/10.1016/0024-3795(94)00058-1

28. X. N. Chi, L. Gan, Z. R. Gao, J-S Chen, A feasible interior-point method with full-Newton step for $P_*(\kappa)$-weighted linear complementarity problem via the algebraically equivalent transformation, *Appl. Numer. Math.*, **220** (2026), 144–166. http://doi.org/10.1016/j.apnum.2025.10.003

29. C. Roos, T. Terlaky, J. P. Vial, *Theory and Algorithm for Linear Optimization–An Interior Point Approach*, New York: John Wiley and Sons Inc, 1997.

30. W. Hock, K. Shittkowski, Test examples for nonlinear programming codes, *J. Optim. Theory Appl.*, **30** (1980), 127–129. https://doi.org/10.1007/BF00934594