Electronic Research Archive

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

ERA, 33(12): 7442–7462.
DOI: 10.3934/era.2025328
Received: 13 September 2025
Revised: 06 November 2025
Accepted: 04 December 2025
Published: 10 December 2025

Research article

A relaxation-type accelerating iteration method for large sparse horizontal nonlinear complementarity problems

Liwei Tian[1], **Suping Liu**[1], **Hua Zheng**[2,*] **and Xiaoping Lu**[3]

[1] School of Computer Science, Guangdong University of Science and Technology, Dongguan 523083, China
[2] School of Mathematics and Statistics, Shaoguan University, Shaoguan 512005, China
[3] School of Computer Science and Engineering, Macau University of Science and Technology, Macao, China

* **Correspondence:** Email: hzheng@sgu.edu.cn.

Abstract: In this work, a relaxation modulus-based matrix splitting iteration method for large sparse horizontal nonlinear complementarity problems was established. The convergence analysis was presented, where the proposed conditions were shown to be weaker than the existing result. Furthermore, a practical selection strategy of the relaxation parameter was provided by analyzing the error function in each iteration. Numerical examples were given to verify the theoretical improvement and show the effectiveness of the proposed method with the suggested relaxation parameters.

Keywords: horizontal nonlinear complementarity problem; modulus-based method; relaxation

1. Introduction

The horizontal nonlinear complementarity problem (HNCP) is an extension of various complementarity problems, including the horizontal linear complementarity problem (HLCP), nonlinear complementarity problem (NCP), linear complementarity problem (LCP) [1–3]. Given two matrices A and B in $\mathbb{R}^{n\times n}$, a vector q in $\mathbb{R}^n$, and a nonlinear function $\phi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$, our focus lies in solving HNCPs. This involves the search for two vectors z and r in $\mathbb{R}^n$ such that

$$Az + q + \phi(z, r) = Br, \quad \min\{z, r\} = 0, \tag{1.1}$$

where the minimum operation is taken componentwise.

The HNCP can manifest in various applications, for example, when discretizing differential equations in hydrodynamic lubrication incorporating complementarity constraints and a weak nonlinear source term [4–8]. A recent approach to address the HNCP involves transforming it into an equivalent implicit fixed-point system. The modulus-based matrix splitting (MMS) iteration method, as introduced in [9], represents a significant advancement. This method generalizes the MMS iteration techniques for the LCP [10], NCP [11], and HLCP [12]. It has been shown to be much more effective than the reduction approaches for solving the HNCP. Moreover, the MMS iteration method has evolved into the modulus-based synchronous multisplitting iteration method, detailed in [13], to leverage parallel computation techniques. An Anderson acceleration variation of the MMS iteration method was developed in [14]. Additionally, a nonsmooth Newton method with nearly quadratic convergence, under specific assumptions, was proposed in [15]. For a comprehensive overview of various methods addressing different complementarity problems, readers can explore recent research works such as [16–22], and the references therein.

Notably, within the realm of enhanced techniques related to the MMS iteration method, the relaxation approach has demonstrated success in various problem domains including the LCP [23], NCP [24], circular cone NCP [25], implicit complementarity problems [26], vertical LCP [27], and HLCP [28–30]. The core concept behind the relaxation method involves merging the current iterative vector with the previous one in the MMS iteration using a relaxation parameter. This integration effectively leverages the information from the two most recent steps. By suitably selecting the relaxation parameter, one can anticipate a faster convergence compared to the original MMS iteration technique. Hence, in order to achieve higher computing efficiency, it is interesting to investigate the relaxation MMS iteration method for the HNCP. The primary contributions of this study include:

- Establishing the relaxation MMS iteration method for the HNCP;
- Presenting a convergence theorem for the proposed method, which extends the convergent domain of the parameter matrix beyond that outlined in [9];
- Providing a practical strategy for selecting the relaxation parameter.

Next, we will explain some definitions, notations, and results that have already been existed.

Denote $e = (1, 1, \dots, 1)^T \in \mathbb{R}^n$. Let $A = (a_{ij})$ be an $n \times n$ real matrix, decomposed as $A = D_A - L_A - U_A = D_A - C_A$, where $D_A, -L_A, -U_A$, and $-C_A$ represent the diagonal, the strictly lower-triangular, the strictly upper-triangular, and the nondiagonal matrices of A, correspondingly. The symbol $|A|$ denotes $|A| = (|a_{ij}|)$. The comparison matrix of A is denoted as $\langle A \rangle = (\langle a_{ij} \rangle)$, defined such that $\langle a_{ii} \rangle = |a_{ii}|$ if $i = j$ and $\langle a_{ij} \rangle = -|a_{ij}|$ if $i \neq j$. We classify A as a strictly diagonal dominant (s.d.d.) matrix if $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$ for all $1 \leq i \leq n$; a Z-matrix if $C_A \geq 0$; a nonsingular M-matrix if $C_A \geq 0$ and $A^{-1} \geq 0$; and an H-matrix if $\langle A \rangle$ forms a nonsingular M-matrix; an H_+-matrix if A is an H-matrix with positive diagonal entries (for example, see [31, 32]). If $\langle M \rangle - |N|$ is an M-matrix, we denote $A = M - N$ as an H-splitting. I_n represents the $n \times n$ identity matrix. $\rho(A)$ is the spectral radius of A.

The structure of this paper is as follows. Section 2 introduces the new relaxation method tailored for the HNCP, building upon the foundations laid out in [9]. In Section 3, the convergence theorems of the proposed method are detailed, demonstrating enhancements over those presented in [9]. The strategy for selecting the relaxation parameter is outlined in Section 4. Section 5 showcases numerical examples to illustrate the efficacy of the proposed method. Finally, Section 6 provides concluding remarks.

2. The relaxation method

First, we revisit the MMS iteration method employed for the HNCP.

Let $A = M_A - N_A$ and $B = M_B - N_B$ represent two splittings of A and B, respectively. By introducing $z = \frac{1}{\gamma}(|x| + x)$ and $r = \frac{1}{\gamma}\Omega(|x| - x)$, the HNCP can be equivalently transformed into a system of fixed-point equations

$$(M_B\Omega + M_A)x = (N_B\Omega + N_A)x + (B\Omega - A)|x| - \gamma(q + \phi(z, r)), \tag{2.1}$$

where Ω denotes a positive diagonal parameter matrix and γ represents a positive constant; see [9] for details. Subsequently, the MMS iteration method is formulated based on (2.1) as follows.

Method 2.1. *[9]* **MMS**

1) *Given* $A, B, \Omega \in \mathbb{R}^{n\times n}$; $q, x^{(1)} \in \mathbb{R}$; *a nonlinear function* ϕ, $\gamma, \epsilon > 0$; *a positive integer* $k_{\max}$, *initialize* $k = 1$.
2) *When* $k < k_{\max}$
3) *Compute* $z^{(k)} = \frac{1}{\gamma}(x^{(k)} + |x^{(k)}|)$ *and* $r^{(k)} = \frac{1}{\gamma}\Omega(|x^{(k)}| - x^{(k)})$.
4) *If* $\|Az^{(k)} - Br^{(k)} + q + \phi(z^{(k)}, r^{(k)})\|_\infty < \epsilon$
5) *break and output the results.*
6) *End If*
7) *Solve*
   $(M_B\Omega + M_A)x^{(k+1)} = (N_B\Omega + N_A)x^{(k)} + (B\Omega - A)|x^{(k)}| - \gamma(q + \phi(z^{(k)}, r^{(k)}))$
   to get $x^{(k+1)}$.
8) $k = k + 1$.
9) *End when*

At the k-th iteration of Method 2.1, consider mixing the new approach vector and the previous one before the next iteration, by introducing a positive diagonal matrix $\Lambda^{(k)} \in \mathbb{R}^{n\times n}$. Then, the relaxation modulus-based matrix splitting (RMMS) iteration method can be established as follows.

Method 2.2. RMMS

1) *Given* $A, B, \Omega \in \mathbb{R}^{n\times n}$; $q, x^{(1)} \in \mathbb{R}$; *a nonlinear function* ϕ, $\gamma, \epsilon > 0$; *a positive integer* $k_{\max}$, *initialize* $k = 1$.
2) *When* $k < k_{\max}$
3) *Compute* $z^{(k)} = \frac{1}{\gamma}(x^{(k)} + |x^{(k)}|)$ *and* $r^{(k)} = \frac{1}{\gamma}\Omega(|x^{(k)}| - x^{(k)})$.
4) *If* $\|Az^{(k)} - Br^{(k)} + q + \phi(z^{(k)}, r^{(k)})\|_\infty < \epsilon$
5) *break and output the results.*
6) *End If*
7) *Solve*
   $(M_B\Omega + M_A)x^{(k+\frac{1}{2})} = (N_B\Omega + N_A)x^{(k)} + (B\Omega - A)|x^{(k)}| - \gamma(q + \phi(z^{(k)}, r^{(k)}))$
   to get $x^{(k+\frac{1}{2})}$.
8) *Calculate* $x^{(k+1)} = \Lambda^{(k)}x^{(k+\frac{1}{2})} + (I_n - \Lambda^{(k)})x^{(k)}$.
9) $k = k + 1$.
10) *End when*

If we take $\Lambda^{(k)} = \lambda^{(k)}I_n$, where $\lambda^{(k)}$ is a scalar, then Step 8 of Method 2.2 is simplified to

$$x^{(k+1)} = \lambda x^{(k+\frac{1}{2})} + (1 - \lambda)x^{(k)}.$$

Remark 2.3. *It is evident that Method 2.2 simplifies to Method 2.1 if we set $\Lambda^{(k)} = I_n$ for every k.*

Furthermore, if we take

$$M_A = \frac{1}{\alpha}(D_A - \beta L_A), M_B = \frac{1}{\alpha}(D_B - \beta L_B), \tag{2.2}$$

we can derive the relaxation modulus-based accelerated overrelaxation (RMAOR) iteration method. When setting $\alpha = \beta$, the RMAOR iteration method simplifies to the relaxation modulus-based successive overrelaxation (RMSOR) iteration method.

3. Convergence analysis

In this section, the convergence conditions are presented for Method 2.2. Similar to the existing literatures such as [9, 13], we assume that both of the system matrices of the HNCP are H_+-matrices.

Lemma 3.4. *[32] Let $A \in \mathbb{R}^{n\times n}$ be a nonsingular M-matrix. Then, there exists a positive diagonal matrix $D \in \mathbb{R}^{n\times n}$ satisfying that AD is an s.d.d. matrix.*

Lemma 3.5. *[33] If A is an H-matrix, then we have $|A^{-1}| \le \langle A\rangle^{-1}$.*

Lemma 3.6. *[34] Let $M \in \mathbb{R}^{n\times n}$ be an s.d.d. matrix. Then for any $N \in \mathbb{R}^{n\times n}$, $\|M^{-1}N\|_\infty \le \max_{1\le i\le n} \frac{(|N|e)_i}{(\langle M\rangle e)_i}$.*

Lemma 3.7. *[32] Let $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times n}$ be a Z-matrix and a nonsingular M-matrix, respectively. If $A \ge B$, then A is a nonsingular M-matrix.*

In the subsequent discussion, following [9, 13, 14], the considered HNCP is always assumed to have a unique solution (z^*, r^*). Moreover, we also assume that $\phi(z, r)$ satisfies the smoothness assumptions outlined in [9] as below: let

$$\phi(z, r) = (\phi_1(z_1, r_1), \phi_2(z_2, r_2), \dots, \phi_n(z_n, r_n))^T$$

be differentiable with

$$0 \le \frac{\partial \phi_i}{\partial z_i} \le \psi_{z_i} \text{ and } 0 \le \frac{\partial \phi_i}{\partial r_i} \le \psi_{r_i},$$

where $\psi_{z_i}, \psi_{r_i} \ge 0, i = 1, 2, \dots, n$.

Then, by the same deduction as those in Section 3 of [9], we have

$$\begin{aligned} &\phi(z^{(k)}, r^{(k)}) - \phi(z^*, r^*) \\ = \ &\frac{1}{\gamma}[(\Phi_z^{(k)} - \Phi_r^{(k)}\Omega)(x^{(k)} - x^*) + (\Phi_z^{(k)} + \Phi_r^{(k)}\Omega)(|x^{(k)}| - |x^*|)], \end{aligned} \tag{3.1}$$

with $\Phi_z^{(k)} \le \Psi_z$ and $\Phi_r^{(k)} \le \Psi_r$, where

$$\Phi_z^{(k)} = \mathrm{diag}(\frac{\partial \phi_1}{\partial z_1}(\eta_1), \frac{\partial \phi_2}{\partial z_2}(\eta_2), \dots, \frac{\partial \phi_n}{\partial z_n}(\eta_n)),$$

$$\Phi_r^{(k)} = \mathrm{diag}(\frac{\partial \phi_1}{\partial r_1}(\eta_1), \frac{\partial \phi_2}{\partial r_2}(\eta_2), \dots, \frac{\partial \phi_n}{\partial r_n}(\eta_n)),$$

$$\Psi_z = \mathrm{diag}(\psi_{z_1}, \psi_{z_2}, \ldots, \psi_{z_n}),$$

$$\Psi_r = \mathrm{diag}(\psi_{r_1}, \psi_{r_2}, \ldots, \psi_{r_n}),$$

and η_i is a convex combination of $(z_i^{(k)}, r_i^{(k)})$ and (z_i^*, r_i^*), $i = 1, 2, \ldots, n$.

Theorem 3.8. *Let $\Omega \in \mathbb{R}^{n\times n}$ be a positive diagonal matrix, $\gamma > 0, \Lambda^{(k)} = \lambda^{(k)} I$, where $\lambda^{(k)} > 0$, and $A, B \in \mathbb{R}^{n\times n}$ be two H_+-matrices. Assume that $A = M_A - N_A$ is an H-splitting, with a positive diagonal matrix D making $(\langle M_A\rangle - |N_A|)D$ be an s.d.d. matrix; $|C_{M_A}| \geq |C_{M_B}|\Omega$; $|N_A| \geq |N_B|\Omega$; and $D_B \geq \Psi_r$. Then, for any $x^{(1)} \in \mathbb{R}^n$, Method 2.2 converges provided (I) or (II) is satisfied.*
(I)

$$(D_{M_B} - \Psi_r)\Omega \geq D_{M_A} + \Psi_z \tag{3.2}$$

and

$$0 < \lambda^{(k)} < \frac{2}{1 + \max\limits_i \frac{\{[\langle M_B\Omega + M_A\rangle - 2(\langle M_A\rangle - |N_A|)]De\}_i}{(\langle M_B\Omega + M_A\rangle De)_i}}. \tag{3.3}$$

(II) $B = M_B - N_B$ is an H-splitting,

$$(D_{M_B} - \Psi_r)^{-1}(\Psi_z + |C_{M_A}| + |N_A|)\tilde{D}e < \Omega\tilde{D}e < D_{M_B}^{-1} D_{M_A}\tilde{D}e, \tag{3.4}$$

*where $\tilde{D}$ is a positive diagonal matrix such that $\langle M_B\rangle\tilde{D}$ is an s.d.d. matrix *, and*

$$0 < \lambda^{(k)} < \frac{2}{1 + \max\limits_i \frac{\{[\langle M_B\Omega + M_A\rangle - 2(D_{M_B}\Omega - \Psi_r\Omega - \Psi_z - |C_{M_A}| - |N_A|)]\Omega^{-1}\tilde{D}e\}_i}{(\langle M_B\Omega + M_A\rangle\Omega^{-1}\tilde{D}e)_i}}. \tag{3.5}$$

Proof. By the assumptions that $|C_{M_A}| \geq |C_{M_B}|\Omega$ and $|N_A| \geq |N_B|\Omega$, one can easily get

$$|C_{M_A} + C_{M_B}\Omega| + |C_{M_A} - C_{M_B}\Omega| = 2|C_{M_A}|, |N_A + N_B\Omega| + |N_A - N_B\Omega| = 2|N_A|. \tag{3.6}$$

Let $\delta^{(k)} = x^{(k)} - x^*$ and $\bar{\delta}^{(k)} = |x^{(k)}| - |x^*|$. By Step 7 of Method 2.2, together with (3.1), we have the error

$$\begin{aligned}
\delta^{(k+\frac{1}{2})} &= (M_B\Omega + M_A)^{-1}[(N_B\Omega + N_A)\delta^{(k)} + (B\Omega - A)\bar{\delta}^{(k)} - (\Phi_z^{(k)} - \Phi_r^{(k)}\Omega)\delta^{(k)} \\
&\quad + (\Phi_z^{(k)} + \Phi_r^{(k)}\Omega)\bar{\delta}^{(k)}] \\
&= (M_B\Omega + M_A)^{-1}\Big\{\big[N_B\Omega + N_A - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega\big]\delta^{(k)} + (B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega)\bar{\delta}^{(k)}\Big\}.
\end{aligned}$$

Then, by Step 8 of Method 2.2, we get

$$\begin{aligned}
\delta^{(k+1)} &= \lambda^{(k)}\delta^{(k+\frac{1}{2})} + (1 - \lambda^{(k)})\delta^{(k)} \\
&= \lambda^{(k)}(M_B\Omega + M_A)^{-1}\Big\{\big[N_B\Omega + N_A - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega\big]\delta^{(k)} \\
&\quad + (B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega)\bar{\delta}^{(k)}\Big\} + (1 - \lambda^{(k)})\delta^{(k)}.
\end{aligned} \tag{3.7}$$

Next, we distinguish the following deductions to two cases.

*Since $B = M_B - N_B$ is an H-splitting, we have $\langle M_B\rangle - |N_B|$ is a nonsingular M-matrix. Then $\langle M_B\rangle$ is also a nonsingular M-matrix due to $\langle M_B\rangle \geq \langle M_B\rangle - |N_B|$, which implies such $\tilde{D}$ exists.

Case (I): By (3.2), we have $D_{M_B}\Omega \geq D_{M_A}$. Together with $|C_{M_A}| \geq |C_{M_B}|\Omega$, we get $\langle M_B\rangle\Omega \geq \langle M_A\rangle$. Therefore, we have

$$\langle M_B\Omega + M_A\rangle \geq \langle M_B\rangle\Omega + \langle M_A\rangle \geq 2\langle M_A\rangle \geq 2(\langle M_A\rangle - |N_A|). \tag{3.8}$$

Since $A = M_A - N_A$ is an H-splitting, one can see that $\langle M_B\Omega + M_A\rangle$ is an M-matrix by Lemma 3.7, which implies that $M_B\Omega + M_A$ is an H-matrix. By (3.7) and Lemma 3.5, we have

$$\begin{aligned} |\delta^{(k+1)}| &\leq [\lambda^{(k)}\langle M_B\Omega + M_A\rangle^{-1}(|N_B\Omega + N_A - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega| \\ &\quad +|B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|) + |1 - \lambda^{(k)}|I_n]|\delta^{(k)}| \\ &\doteq (\lambda^{(k)}\mathcal{M}^{-1}\mathcal{N} + |1 - \lambda^{(k)}|I_n)|\delta^{(k)}|, \end{aligned} \tag{3.9}$$

where

$$\mathcal{M} = \langle M_B\Omega + M_A\rangle, \mathcal{N} = |N_B\Omega + N_A - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega| + |B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|.$$

Note that $(\langle M_A\rangle - |N_A|)D$ is an s.d.d. matrix. By (3.8), it is clear that $\mathcal{M}De = \langle M_B\Omega + M_A\rangle De \geq 2(\langle M_A\rangle - |N_A|)De > 0$, which implies that $\mathcal{M}D$ is an s.d.d. matrix, too. Then, by Lemma 3.6, we have

$$\|(\mathcal{M}D)^{-1}\mathcal{N}D\|_\infty \leq \max_{1\leq i\leq n} \frac{(|\mathcal{N}D|e)_i}{(\langle\mathcal{M}D\rangle e)_i}. \tag{3.10}$$

By (3.6), we directly deduce that

$$\begin{aligned} &\mathcal{M}De - \mathcal{N}De \\ = &(\langle M_B\Omega + M_A\rangle - |N_B\Omega + N_A - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega| - |B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|)De \\ = &(|D_{M_B}\Omega + D_{M_A}| - |C_{M_B}\Omega + C_{M_A}| - |N_B\Omega + N_A - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega| \\ &-|D_{M_B}\Omega - D_{M_A} - C_{M_B}\Omega + C_{M_A} - N_B\Omega + N_A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|]De \\ \geq &(D_{M_B}\Omega + D_{M_A} - |C_{M_B}\Omega + C_{M_A}| - |C_{M_B}\Omega - C_{M_A}| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega \\ &-|D_{M_B}\Omega - D_{M_A} - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega| - |N_B\Omega + N_A| - |N_B\Omega - N_A|]De \\ = &(D_{M_B}\Omega + D_{M_A} - 2|C_{M_A}| - 2|N_A| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega \\ &-|D_{M_B}\Omega - D_{M_A} - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|)De. \end{aligned} \tag{3.11}$$

With (3.2), we can get

$$\begin{aligned} &\mathcal{M}De - \mathcal{N}De \\ \geq &[D_{M_B}\Omega + D_{M_A} - 2|C_{M_A}| - 2|N_A| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega - (D_{M_B}\Omega - D_{M_A} - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega)]De \\ = &(2D_{M_A} - 2|C_{M_A}| - 2|N_A|)De \\ = &2(\langle M_A\rangle - |N_A|)De \\ > &0, \end{aligned} \tag{3.12}$$

which implies that $\|(\mathcal{M}D)^{-1}\mathcal{N}D\|_\infty < 1$ by (3.10). Then, by direct computation, we have

$$\lambda^{(k)}\|D^{-1}\mathcal{M}^{-1}\mathcal{N}D\|_\infty + |1 - \lambda^{(k)}| < 1$$

for $\lambda^{(k)}$ satisfying (3.3). Furthermore, we get

$$\rho((\lambda^{(k)}\mathcal{M}^{-1}\mathcal{N} + |1 - \lambda^{(k)}|I_n))$$

$$
\begin{aligned}
&= \rho(D^{-1}(\lambda^{(k)}\mathcal{M}^{-1}\mathcal{N} + |1-\lambda^{(k)}|I_n)D) \\
&\le \lambda^{(k)}\|D^{-1}\mathcal{M}^{-1}\mathcal{N}D\|_\infty + |1-\lambda^{(k)}| \\
&< 1,
\end{aligned}
$$

which implies that Method 2.2 is convergent by (3.9).

Case (II): By the first inequality of (3.4), we have

$$D_{M_B}\Omega\tilde{D}e > (\Psi_r\Omega + \Psi_z + |C_{M_A}| + |N_A|)\tilde{D}e \ge |C_{M_A}|\tilde{D}e. \tag{3.13}$$

On the other hand, note that $D_{M_B}\Omega < D_{M_A}$ is satisfied according the second inequality of (3.4). Then, with (3.13), we have

$$
\begin{aligned}
& \langle M_B\Omega + M_A\rangle\Omega^{-1}\tilde{D}e \\
\ge\ & (\langle M_B\Omega\rangle + \langle M_A\rangle)\Omega^{-1}\tilde{D}e \\
\ge\ & (\langle M_B\Omega\rangle + D_{M_B}\Omega - |C_{M_A}|)\Omega^{-1}\tilde{D}e \\
\ge\ & \langle M_B\rangle\tilde{D}e \\
>\ & 0,
\end{aligned}
$$

which implies that $M_B\Omega + M_A$ is an H-matrix. Then, by the same deductions as those in **Case (I)**, we can also get the error (3.9). Moreover, we have

$$\|(\mathcal{M}\Omega^{-1}\tilde{D})^{-1}\mathcal{N}\Omega^{-1}\tilde{D}\|_\infty \le \max_{1\le i\le n}\frac{(|\mathcal{N}\Omega^{-1}\tilde{D}|e)_i}{(\langle\mathcal{M}\Omega^{-1}\tilde{D}\rangle e)_i} \tag{3.14}$$

and

$$
\begin{aligned}
& \mathcal{M}\Omega^{-1}\tilde{D}e - \mathcal{N}\Omega^{-1}\tilde{D}e \\
\ge\ & (D_{M_B}\Omega + D_{M_A} - 2|C_{M_A}| - 2|N_A| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega \\
& -|D_{M_B}\Omega - D_{M_A} - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|)\Omega^{-1}\tilde{D}e,
\end{aligned} \tag{3.15}
$$

similar to (3.10) and (3.11), respectively. By (3.4) and (3.15), we get

$$
\begin{aligned}
& \mathcal{M}\Omega^{-1}\tilde{D}e - \mathcal{N}\Omega^{-1}\tilde{D}e \\
\ge\ & [D_{M_B}\Omega + D_{M_A} - 2|C_{M_A}| - 2|N_A| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega \\
& +(D_{M_B}\Omega - D_{M_A} - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega)]\Omega^{-1}\tilde{D}e \\
=\ & 2(D_{M_B}\Omega - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega - |C_{M_A}| - |N_A|)\Omega^{-1}\tilde{D}e \\
\ge\ & 2(D_{M_B}\Omega - \Psi_z - \Psi_r\Omega - |C_{M_A}| - |N_A|)\Omega^{-1}\tilde{D}e \\
>\ & 0.
\end{aligned}
$$

Then, when (3.5) is satisfied, by the same deductions as those following (3.12) in **Case (I)**, we can also get

$$\rho((\lambda^{(k)}\mathcal{M}^{-1}\mathcal{N} + |1-\lambda^{(k)}|I_n)) < 1,$$

which finishes the proof. □

Remark 3.9. *We have obtained the convergence conditions of Method 2.2 in Theorem 3.8. Note that if we take $\lambda^{(k)} \equiv 1$, Theorem 3.8 reduces to the convergence theorem of Method 2.1. Comparing to Theorem 2 in [9], some comments can be given.*

- *Theorem 3.8 presents convergence conditions for two distinct cases, and we hereby state that these two cases are mutually exclusive. Indeed, by (3.2), we have*

$$\Omega \geq (D_{M_B} - \Psi_r)^{-1}(D_{M_A} + \Psi_z) \geq D_{M_B}^{-1} D_{M_A},$$

while inequality (3.4) clearly implies

$$\Omega \tilde{D} e < D_{M_B}^{-1} D_{M_A} \tilde{D} e \Rightarrow \Omega < D_{M_B}^{-1} D_{M_A}.$$

- *The proof techniques between Theorem 3.8 and Theorem 2 of [9] are different from each other.*
- *The assumptions in* **Case (I)** *are the same as those in Theorem 2 of [9].*
- *One can have more choices of the parameter matrix Ω in* **Case (II)** *of Theorem 3.8 than Theorem 2 of [9]. In fact, Ω can be chosen when $\Omega \leq D_{M_B}^{-1} D_{M_A}$ in the former but not in the latter. Hence, the research scope of the convergence of the MMS iteration method for the HNCP has been extended. Numerical examples will be presented in the following section to show the significance of the improvement.*

4. Selection strategy for relaxation parameters

For acceleration methods incorporating relaxation techniques, a natural question arises regarding the selection of optimal relaxation parameters $\lambda^{(k)}$ in application. However, due to the modulus operations present in the equivalent modulus equation of the HNCP, the convergence analysis actually involves approximation when estimating the spectral radius of the iteration matrix of the MMS iteration. This significantly complicates the search for theoretically optimal parameters. In most existing literatures on relaxation-type methods such as [28–30], which investigate relaxation approaches for the HLCP (a simplified case of the HNCP considered in this paper), no theoretical analysis had been conducted regarding the selection strategy for relaxation parameters. Instead, in numerical experiments, the computational performance of the relaxation method was demonstrated merely by exhaustively testing parameters within a predefined interval.

To propose a theoretically grounded strategy for selecting near-optimal relaxation parameters, consider the error $\delta^{(k+1)}$ following the relaxation operation in the k-th iteration. Clearly, we have

$$\delta^{(k+1)} = (1 - \lambda^{(k)})\delta^{(k)} + \lambda^{(k)}\delta^{(k+\frac{1}{2})}.$$

Let

$$f(\lambda) = \|(1 - \lambda)\delta^{(k)} + \lambda\delta^{(k+\frac{1}{2})}\|_2^2.$$

One can easily get that the minimum point of the above quadratic function f is

$$\lambda_{\min} = 1 + \frac{(x^{(k+\frac{1}{2})} - x^*)^T(x^{(k)} - x^{(k+\frac{1}{2})})}{\|x^{(k)} - x^{(k+\frac{1}{2})}\|_2^2}, \tag{4.1}$$

which will make the relaxation operation work. Note that x^* is unknown beforehand. During the iteration, when $k > 1$, replacing x^* by $x^{(k-\frac{1}{2})}$, a feasible choice of the relaxation parameter based on (4.1) can be

$$\lambda^{(k)} \doteq \lambda_0 = 1 + \frac{(x^{(k+\frac{1}{2})} - x^{(k-\frac{1}{2})})^T (x^{(k)} - x^{(k+\frac{1}{2})})}{\|x^{(k)} - x^{(k+\frac{1}{2})}\|_2^2}. \tag{4.2}$$

Furthermore, combining with Theorem 3.8, we can use the upper bounds given in (3.3) and (3.5), denoted by τ_1 and τ_2, respectively, to guarantee the theoretical rationality of the choice of $\lambda^{(k)}$. Specifically,

$$\lambda^{(k)} \doteq \lambda_1 = \begin{cases} \lambda_0, & \text{if } \lambda_0 \leq \tau_1, \\ \tau_1, & \textit{otherwise}, \end{cases} \tag{4.3}$$

for **Case (I)**, while

$$\lambda^{(k)} \doteq \lambda_2 = \begin{cases} \lambda_0, & \text{if } \lambda_0 \leq \tau_2, \\ \tau_2, & \textit{otherwise}, \end{cases} \tag{4.4}$$

for **Case (II)**.

Clearly, the dominant computational cost in calculating the upper bounds given by (3.3) or (3.5) arises from matrix-vector multiplication. This indicates that the required computational effort of (4.3) or (4.3) is $O(n^2)$, which is of the same order as the per-iteration cost of the MMS iteration.

5. Numerical examples

This section presents numerical examples to demonstrate the effectiveness of the proposed method. The computations were run on an Intel(R) Core(TM) i7-10710U. The programming language is MATLAB.

The following two examples are from [9].

Example 5.10. *[9] Let $z(u, v)$, $r(u, v)$, and $q(u, v)$ be three $\mathbb{R}^2 \to \mathbb{R}^2$ mappings. Take into account the following boundary problem:*

$$\triangle z + \frac{\partial^2 r}{\partial^2 u} + \mu z + \nu r - q - \phi(z, r) = 0, \;\; \min\{z, r\} = 0,$$

where μ and ν are real parameters.

By using a five-point difference scheme along with appropriate boundary conditions to discretize the problem, an HNCP can be formulated. Specially, the matrices A and B take the forms

$$A = \begin{pmatrix} S & -I_m & & & \\ -I_m & S & -I_m & & \\ & \ddots & \ddots & \ddots & \\ & & -I_m & S & -I_m \\ & & & -I_m & S \end{pmatrix}, B = \begin{pmatrix} S & & & & \\ & S & & & \\ & & \ddots & & \\ & & & S & \\ & & & & S \end{pmatrix},$$

where

$$S = \begin{pmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{pmatrix} \in \mathbb{R}^{m\times m},$$

respectively, and $n = m^2$.

Example 5.11. *[9] Let*

$$A = \begin{pmatrix} T & -0.5I_m & & & \\ -1.5I_m & T & -0.5I_m & & \\ & \ddots & \ddots & \ddots & \\ & & -1.5I_m & T & -0.5I_m \\ & & & -1.5I_m & T \end{pmatrix}, B = \begin{pmatrix} T & & & & \\ & T & & & \\ & & \ddots & & \\ & & & T & \\ & & & & T \end{pmatrix},$$

where

$$T = \begin{pmatrix} 4 & -0.5 & & & \\ -1.5 & 4 & -0.5 & & \\ & \ddots & \ddots & \ddots & \\ & & -1.5 & 4 & -0.5 \\ & & & -1.5 & 4 \end{pmatrix} + \mu I \in \mathbb{R}^{m\times m},$$

and $q = Az^* - Br^*$, where $n = m^2$.

In all numerical experiments, let $x^{(1)} = e, \gamma = 1, k_{\max} = 150$, and $\epsilon = 10^{-8}$. The nonlinear functions are chosen as below.

$$\begin{cases} \phi_1 = \frac{1}{2}(r + z + \sin(r)\cos(r) + \sin(z)\cos(z)), \\ \phi_2 = -\frac{1}{1+r+z}, \\ \phi_3 = \arctan(r) + \arctan(z), \\ \phi_4 = \sin(r + z), \\ \phi_5 = \ln(1 + r + z), \\ \phi_6 = z^2, \\ \phi_7 = r^2, \\ \phi_8 = zr, \\ \phi_9 = |sin(r)| + z^2. \end{cases} \tag{5.1}$$

We remark that the first five functions are constructed satisfying the smoothness assumption presented in Section 3, while the remaining four are from [9].

5.1. Experiment 1

We first verify the improvement on the larger convergence range of the positive diagonal parameter matrix Ω, shown in the third item of Remark 3.9. Here, we consider the case of $\mu = \nu = 4$.

Consider the first five functions in (5.1), which satisfy the smoothness assumptions with $\Psi_z = \Psi_r = I_n$. Let

$$\Omega = \theta(D_{M_B} - I_n)^{-1}(D_{M_A} + I_n). \tag{5.2}$$

Consider the SOR splitting. Taking $m = 64$ and $m = 128$, by direct computation, we can get the ranges of θ making Ω given by (5.2) satisfy (3.4). Specifically, for the two examples, we have the following results.

- If $\alpha = 0.8$, then $0.7778 \leq \theta \leq 1$.
- If $\alpha = 0.9$, then $0.7465 \leq \theta \leq 1$.
- If $\alpha = 1.0$, then $0.7143 \leq \theta \leq 1$.
- If $\alpha = 1.1$, then $0.9130 \leq \theta \leq 1$.

To show more behaviors of the methods, we run the numerical experiments with more choices of the parameters. Therefore, we set $0.4 \leq \theta \leq 1$ and $0.8 \leq \alpha \leq 1.2$. Numerical results of the MMS iteration method are shown in Tables 1 and 2. Since the computation cost of each iteration in the MMS are the same with different θ and α, we only shown the iteration steps, which can accurately measure the computational efficiency.

It can be seen from Tables 1 and 2 that the MMS iteration method converges in most cases, with only a few cases where the iteration steps have exceeded $k_{\max}$. By the definition of Ω given in (5.2), θ should satisfy $\theta \geq 1$ according to Theorem 2 of [9]. As the numerical results show, when $\theta < 1$, the MMS iteration method still converges in most cases, and in some cases, the computational efficiencies are better than those of $\theta = 1$. These facts can also be visually observed in Figures 1–4. Therefore, the improvement of the convergence range of Ω is significant.

Table 1. Iteration steps of Method 2.1 for Example 5.10.

		$m = 64$							$m = 128$						
	α \ θ	0.4	0.5	0.6	0.7	0.8	0.9	1.0	0.4	0.5	0.6	0.7	0.8	0.9	1.0
ϕ_1	0.8	45	46	49	52	55	59	63	47	48	50	54	57	61	65
	0.9	37	37	40	42	46	49	53	38	38	41	44	47	51	54
	1.0	70	40	32	35	38	41	44	72	41	33	36	39	42	46
	1.1	150	76	45	32	31	34	37	150	77	46	32	32	35	38
	1.2	150	150	80	48	37	44	51	150	150	82	49	39	46	53
ϕ_2	0.8	88	91	96	101	108	114	121	91	95	99	105	112	119	125
	0.9	72	75	80	85	91	96	102	75	78	83	88	94	100	106
	1.0	60	63	67	72	77	82	88	62	65	70	75	80	85	91
	1.1	58	52	57	61	66	71	75	59	55	59	63	68	73	78
	1.2	119	57	48	52	56	61	65	122	59	49	54	58	63	68
ϕ_3	0.8	49	49	51	54	57	61	64	51	51	53	56	59	63	67
	0.9	39	40	41	44	47	50	54	41	41	43	46	49	52	55
	1.0	71	42	34	36	39	42	45	72	43	35	37	40	43	46
	1.1	150	75	46	33	32	35	38	150	77	47	34	33	36	39
	1.2	150	150	81	49	36	33	37	150	150	83	51	37	34	38
ϕ_4	0.8	45	46	48	51	55	58	62	47	48	50	53	56	60	64
	0.9	40	37	39	42	45	49	52	41	38	41	43	47	50	54
	1.0	82	44	32	35	38	41	44	84	45	33	36	39	42	45
	1.1	150	85	47	32	31	34	37	150	87	48	33	32	35	38
	1.2	150	150	88	50	37	44	49	150	150	90	51	39	46	51
ϕ_5	0.8	56	56	58	61	65	69	73	58	58	60	63	67	71	75
	0.9	45	46	48	50	54	57	61	46	47	49	52	56	59	63
	1.0	61	38	39	42	45	48	51	63	39	40	43	46	50	53
	1.1	165	64	41	34	37	40	43	170	66	42	36	38	42	45
	1.2	150	158	67	43	32	34	37	150	163	69	44	33	35	38

Table 2. Iteration steps of Method 2.1 for Example 5.11.

	θ / α	m = 64							m = 128						
		0.4	0.5	0.6	0.7	0.8	0.9	1.0	0.4	0.5	0.6	0.7	0.8	0.9	1.0
ϕ_1	0.8	32	34	37	40	44	48	52	33	35	38	42	46	50	54
	0.9	26	24	27	31	34	38	42	27	25	28	32	36	39	43
	1.0	45	28	21	23	27	30	33	46	29	21	24	27	31	34
	1.1	104	50	31	25	30	35	40	107	51	31	26	30	36	41
	1.2	150	109	55	51	58	80	101	150	113	56	54	63	88	115
ϕ_2	0.8	65	69	75	81	87	93	100	67	72	78	84	91	98	104
	0.9	49	54	59	64	70	76	82	51	56	61	67	73	79	85
	1.0	36	41	46	51	57	62	67	37	43	48	53	59	65	70
	1.1	35	30	35	41	46	50	55	36	31	37	42	47	53	58
	1.2	60	36	27	31	36	41	45	61	37	28	33	38	42	47
ϕ_3	0.8	35	36	38	42	45	49	53	36	37	40	43	47	51	54
	0.9	29	26	28	32	35	38	42	30	27	29	33	36	40	43
	1.0	50	31	22	23	27	30	33	51	32	23	24	28	31	34
	1.1	113	54	34	24	22	26	29	116	55	35	24	23	27	30
	1.2	150	103	56	37	38	41	45	150	106	58	38	40	43	47
ϕ_4	0.8	32	34	37	40	44	47	51	33	35	38	41	45	49	53
	0.9	29	24	28	31	34	38	41	30	25	28	32	35	39	43
	1.0	46	30	23	23	27	30	33	47	31	24	24	28	31	34
	1.1	109	49	31	25	29	34	40	112	50	32	26	30	35	41
	1.2	150	110	52	53	64	86	112	150	111	53	58	67	100	139
ϕ_5	0.8	40	41	44	48	51	56	60	41	43	46	49	53	58	62
	0.9	29	30	33	37	40	44	48	30	31	34	38	42	46	50
	1.0	41	27	24	28	31	35	38	42	28	25	29	32	36	40
	1.1	83	44	29	21	24	27	30	85	45	29	22	25	28	32
	1.2	150	85	47	31	28	35	40	150	87	48	31	29	36	42

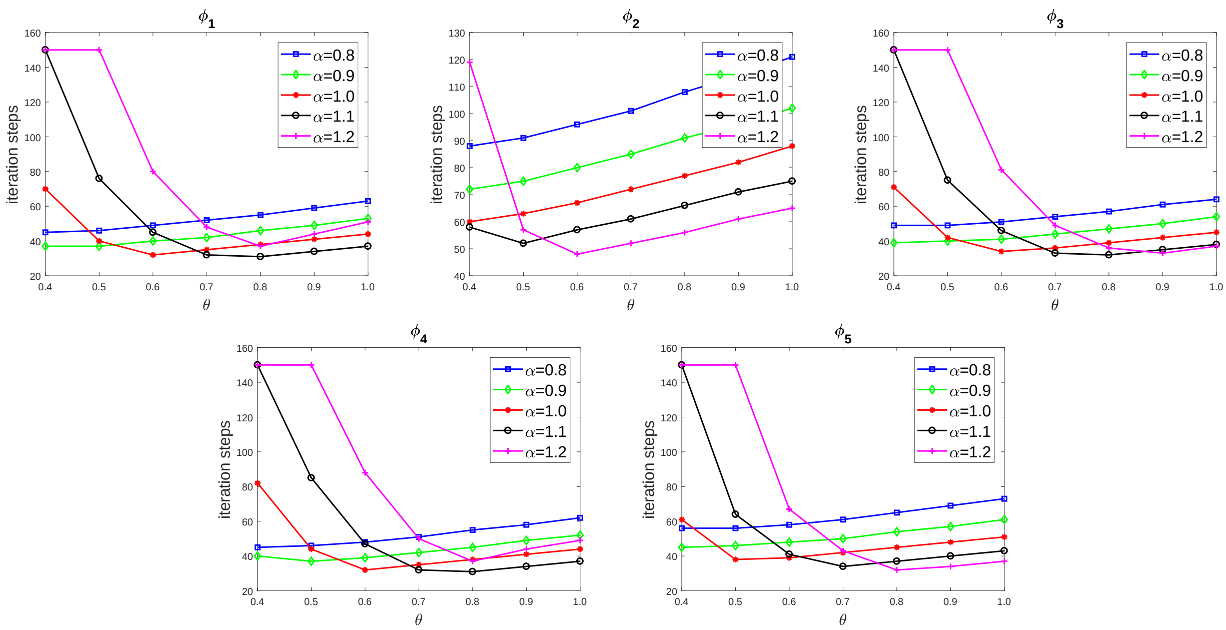

Figure 1. The iteration steps of Method 2.1 for Example 5.10 with m = 64.

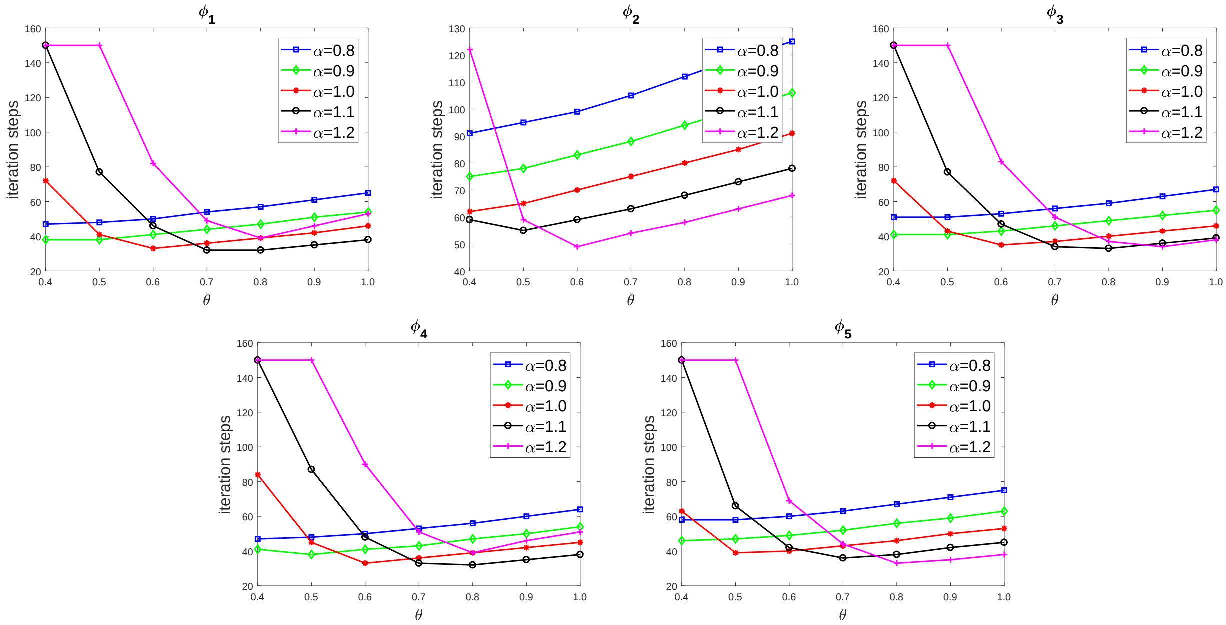

Figure 2. The iteration steps of Method 2.1 for Example 5.10 with $m = 128$.

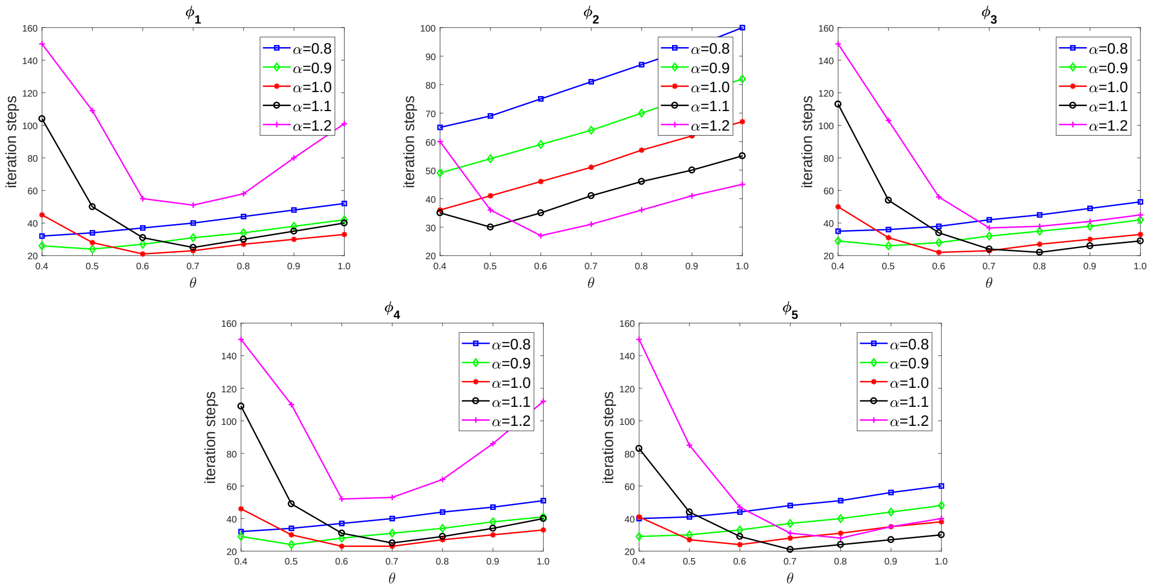

Figure 3. The iteration steps of Method 2.1 for Example 5.11 with $m = 64$.

5.2. Experiment 2

Next, we show the efficiency of the RMMS iteration method.

To illustrate more computational behavior of the method outside the sufficient convergence conditions, consider all nine functions given in (5.1). Let Ω be given in (5.2) with $\theta = 1$, which satisfies the convergence conditions of both MMS and RMMS iteration methods. SOR splitting is also used with $0.8 \leq \alpha \leq 1.2$. When $m = 512$ and $m = 1024$, numerical results are shown in Tables 3 and 4, where "IT" is the iteration step and "T" is the CPU time in seconds. Here, for Method 2.2, two strategies of the relaxation parameter are considered, the first given by (4.2) and the second given by (4.3), correspondingly denoted by "RMMS1" and "RMMS2". When α is given, one can see that the RMMS1 and RMMS2 require significantly fewer iteration steps than the MMS, resulting in significantly less CPU time. Even when α changes from 0.8 to 1.2, the RMMS1 and RMMS2 are more superior to the MMS, which can be seen more clearly in Figures 5 and 6, where the blue (RMMS1) and red (RMMS2) lines are almost always below the black (MMS) lines for a fixed m. Furthermore, the comparisons of the MMS, RMMS1, and RMMS2 for the optimal cases are shown in Figures 7–10. For the comparison between the RMMS1 and RMMS2, the RMMS1 without an upper bound constraint is better than the RMMS2. This implies that the theoretical conditions may be improved. In addition, Method 2.2 is more effective than Method 2.1 with the strategy given in Section 4.

Table 3. Numerical comparisons of Method 2.1 and Method 2.2 for Example 5.10.

		m = 512						m = 1024					
		MMS		RMMS2		RMMS1		MMS		RMMS2		RMMS1	
	α	IT	T	IT	T	IT	T	IT	T	IT	T	IT	T
ϕ_1	0.8	69	2.2883	45	1.2288	32	0.8726	71	9.9271	47	5.7038	32	3.8877
	0.9	57	1.4967	40	1.1027	32	0.8809	59	7.0076	40	4.8522	32	3.9221
	1.0	48	1.2913	37	1.0151	31	0.8533	50	5.6432	38	4.6539	32	3.9143
	1.1	41	1.1736	32	0.8773	30	0.8197	42	5.5499	33	4.0616	30	3.6306
	1.2	56	1.5706	33	0.9113	32	0.8776	58	7.3800	33	4.0741	32	3.9039
ϕ_2	0.8	134	4.1852	91	2.5051	55	1.5275	138	18.5767	93	10.2230	58	6.4235
	0.9	113	3.1663	78	2.1746	50	1.3792	117	13.9116	80	8.7294	53	5.8363
	1.0	97	2.6303	69	1.9052	52	1.4457	100	10.7404	71	7.7700	53	5.8897
	1.1	84	2.4775	64	1.7748	50	1.3910	86	9.0032	65	7.2113	49	5.3462
	1.2	72	1.8699	58	1.5968	47	1.3091	74	7.8276	59	6.4469	50	5.4954
ϕ_3	0.8	71	2.3154	46	1.3557	32	0.9576	73	9.6761	47	5.0806	33	3.5461
	0.9	59	1.7581	40	1.1992	31	0.9132	60	6.5988	42	4.5837	31	3.3804
	1.0	49	1.4023	36	1.0588	30	0.8894	51	5.3953	38	4.0872	31	3.3610
	1.1	41	1.1644	32	0.9558	30	0.8869	42	4.6030	35	3.7707	30	3.2506
	1.2	41	1.3194	30	0.8968	31	0.9264	42	4.5108	31	3.3318	31	3.3318
ϕ_4	0.8	68	2.2003	44	1.2787	33	0.9442	69	9.4393	44	4.6365	34	3.5879
	0.9	57	1.5991	39	1.1282	31	0.8899	58	6.1149	39	4.0794	31	3.2614
	1.0	48	1.3990	36	1.0358	32	0.9165	49	5.1026	37	3.9368	34	3.6174
	1.1	40	1.1395	33	0.9526	31	0.8889	41	4.3585	34	3.6206	31	3.2635
	1.2	55	1.6339	34	0.9792	33	0.9473	57	6.0616	32	3.3770	33	3.4561
ϕ_5	0.8	80	2.5542	53	1.4630	36	0.9809	82	11.1193	54	5.8840	37	4.0263
	0.9	67	1.8655	45	1.2325	35	0.9693	69	7.6724	47	5.1334	36	3.9553
	1.0	56	1.4928	42	1.1470	33	0.9141	58	6.2072	42	4.6304	34	3.7417
	1.1	48	1.3061	37	1.0137	32	0.8797	49	5.3486	38	4.1223	33	3.6256
	1.2	40	1.1276	33	0.8985	33	0.8979	41	4.5427	33	3.5819	34	3.7208
ϕ_6	0.8	53	1.6380	36	1.0336	28	0.7981	54	6.7672	36	4.0446	28	3.1069
	0.9	43	1.2361	31	0.8856	26	0.7438	45	5.3154	32	3.5842	27	3.0081
	1.0	36	0.9499	27	0.7783	26	0.7477	37	4.1743	28	3.1182	26	2.8968
	1.1	31	0.9254	26	0.7412	25	0.7211	32	3.4015	26	2.8925	26	2.9146
	1.2	58	1.7075	25	0.7198	25	0.7111	60	6.4912	26	2.8931	26	2.9133
ϕ_7	0.8	16	0.5270	12	0.3181	8	0.2133	16	2.0114	12	1.3325	8	0.8881
	0.9	13	0.3470	10	0.2637	8	0.2131	13	1.5080	10	1.1125	8	0.8887
	1.0	11	0.2893	9	0.2394	7	0.1852	11	1.2339	9	1.0047	7	0.7767
	1.1	9	0.2380	8	0.2116	6	0.1590	9	0.9707	8	0.8887	6	0.6665
	1.2	7	0.1830	6	0.1581	5	0.1324	7	0.7552	6	0.6748	5	0.5578
ϕ_8	0.8	127	3.8693	86	2.2634	56	1.4936	131	18.0876	89	9.3168	57	5.9101
	0.9	106	2.8132	74	1.9715	51	1.3457	110	11.5652	76	7.9031	52	5.4111
	1.0	90	2.3307	67	1.7676	49	1.2844	93	9.5899	71	7.4337	49	5.1012
	1.1	75	1.9862	61	1.6014	53	1.3934	78	8.1981	63	6.6103	54	5.6336
	1.2	58	1.5616	53	1.4115	50	1.3185	60	6.2761	54	5.6108	50	5.2251
ϕ_9	0.8	70	2.3586	46	1.3090	34	0.9801	73	9.9177	46	4.9512	35	3.7419
	0.9	59	1.6608	43	1.2280	34	0.9754	60	6.3531	44	4.6968	34	3.6390
	1.0	49	1.4065	38	1.0970	34	0.9772	51	5.3157	38	4.0543	34	3.6184
	1.1	41	1.2998	33	0.9380	30	0.8563	42	4.4518	35	3.7126	31	3.2973
	1.2	59	1.6016	35	1.0130	32	0.9175	61	6.7017	34	3.6252	35	3.7691

Table 4. Numerical comparisons of Method 2.1 and Method 2.2 for Example 5.11.

		m = 512						m = 1024					
		MMS		RMMS2		RMMS1		MMS		RMMS2		RMMS1	
	α	IT	T	IT	T	IT	T	IT	T	IT	T	IT	T
ϕ_1	0.8	57	1.9219	40	1.1633	28	0.8180	59	8.0153	41	4.9187	28	3.4131
	0.9	46	1.2619	35	1.0353	28	0.8192	47	5.3114	36	4.3665	28	3.4122
	1.0	36	0.9976	31	0.9175	29	0.8543	37	4.0196	31	3.7481	30	3.6156
	1.1	44	1.2036	28	0.8268	26	0.7576	45	5.1346	31	3.7694	29	3.5206
	1.2	127	3.9634	29	0.8444	32	0.9435	132	17.0945	30	3.6432	34	4.0864
ϕ_2	0.8	112	3.4504	82	2.1965	42	1.1382	115	15.5008	84	8.8244	47	4.9360
	0.9	91	2.4893	69	1.8749	43	1.1655	94	10.0291	71	7.4673	45	4.8031
	1.0	75	1.9977	59	1.5954	41	1.1166	77	7.8956	61	6.4990	42	4.4593
	1.1	61	1.6049	54	1.4596	43	1.1699	63	6.7663	55	5.7600	43	4.5461
	1.2	50	1.3984	48	1.3085	41	1.1104	52	5.5457	49	5.1794	37	3.9095
ϕ_3	0.8	58	2.0714	41	1.2043	27	0.7937	59	8.0961	42	4.5845	28	3.0438
	0.9	46	1.4162	34	1.0112	26	0.7754	47	5.2527	34	3.7009	26	2.8252
	1.0	36	1.0806	31	0.9102	25	0.7455	37	3.9823	32	3.5148	25	2.7379
	1.1	32	0.9239	26	0.7639	23	0.6846	33	3.6066	26	2.8375	24	2.5897
	1.2	50	1.4392	24	0.7115	25	0.7357	51	5.4382	24	2.6271	25	2.7003
ϕ_4	0.8	56	1.8652	39	1.2268	29	0.9164	58	7.8559	40	4.7293	29	3.4160
	0.9	45	1.3360	35	1.1058	29	0.9187	46	4.9973	36	4.2683	29	3.4343
	1.0	36	1.0894	30	0.9562	27	0.8535	37	3.9345	31	3.6449	29	3.4223
	1.1	44	1.3459	29	0.9238	26	0.8258	45	4.8882	30	3.5963	27	3.2050
	1.2	156	5.1261	31	0.9729	36	1.1428	162	20.6109	28	3.3022	33	3.9054
ϕ_5	0.8	66	2.4429	47	1.4877	29	0.9083	68	10.2677	48	5.7117	29	3.4761
	0.9	53	1.6208	38	1.2021	28	0.8792	54	6.3875	39	4.6780	29	3.4423
	1.0	42	1.4092	34	1.0570	27	0.8545	43	4.9386	36	4.3152	27	3.2017
	1.1	33	1.0262	30	0.9456	24	0.7516	34	4.0715	32	3.8260	25	3.0165
	1.2	44	1.3393	27	0.8431	23	0.7285	46	5.7485	28	3.3584	25	2.9671
ϕ_6	0.8	43	1.4620	30	0.9742	23	0.7519	44	6.1519	32	3.7727	24	2.7999
	0.9	34	0.8934	27	0.8861	23	0.7533	34	3.7129	27	3.1662	23	2.7086
	1.0	25	0.7257	23	0.7597	21	0.6950	27	2.8439	23	2.6871	22	2.5721
	1.1	45	1.3760	23	0.7565	24	0.7813	46	4.9030	23	2.6999	24	2.8148
	1.2	141	5.0358	23	0.7493	25	0.8253	146	18.2818	23	2.7028	24	2.8260
ϕ_7	0.8	16	0.5624	12	0.3062	8	0.2071	16	2.1887	12	1.3245	8	0.8675
	0.9	13	0.3256	11	0.2809	7	0.1803	13	1.4851	11	1.2135	7	0.7652
	1.0	10	0.2560	9	0.2313	6	0.1534	10	1.0754	9	0.9916	6	0.6544
	1.1	7	0.1835	6	0.1531	5	0.1284	7	0.7487	6	0.6538	5	0.5437
	1.2	5	0.1325	4	0.1032	4	0.1018	5	0.5228	4	0.4388	4	0.4351
ϕ_8	0.8	105	3.3092	78	2.1830	47	1.3227	109	16.1149	80	8.8254	48	5.3416
	0.9	83	2.3341	64	1.7794	43	1.2086	86	9.1724	66	7.2990	44	4.8540
	1.0	69	1.9003	58	1.6290	45	1.2496	71	7.3787	60	6.6028	45	4.9439
	1.1	61	1.6912	55	1.5408	38	1.0560	63	6.8008	57	6.3245	43	4.7034
	1.2	125	3.5487	53	1.4914	43	1.2009	129	15.1253	53	5.8688	51	5.6352
ϕ_9	0.8	58	1.9737	41	1.3027	30	0.9438	60	8.0158	42	5.0516	30	3.6197
	0.9	46	1.3948	36	1.1358	28	0.8837	48	5.1084	38	4.6094	28	3.3523
	1.0	35	1.0867	32	1.0028	27	0.8536	36	3.9552	32	3.8581	27	3.2575
	1.1	45	1.3312	32	1.0064	30	0.9471	47	5.0703	32	3.8827	30	3.6211
	1.2	127	4.2831	33	1.0355	30	0.9550	132	17.6775	29	3.4687	33	3.9334

Remark 5.12. *The numerical results presented in Section 5 reveal that Method 2.2 performs better with the relaxation parameter given in (4.2) than in (4.3), which implies that there may exist potential improvements for the choice of relaxation parameter.*

6. Conclusions

We have applied the relaxation technique to the MMS iteration method for solving the HNCP. When presenting the convergence conditions of the relaxation method, we also enlarge the convergence range of the positive diagonal parameter matrix. Moreover, by minimizing the error in the k-th iteration, we provide the strategy of the relaxation parameter. Finally, the improved convergence range and the effectiveness of the relaxation strategy are validated by numerical examples.

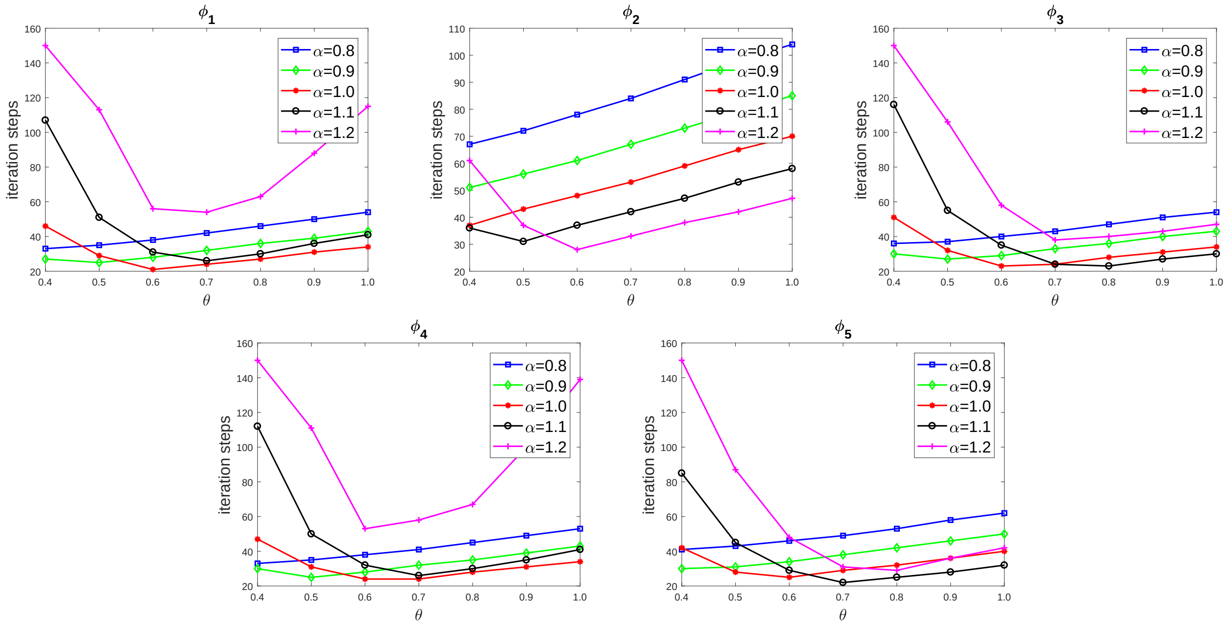

Figure 4. The iteration steps of Method 2.1 for Example 5.11 with $m = 128$.

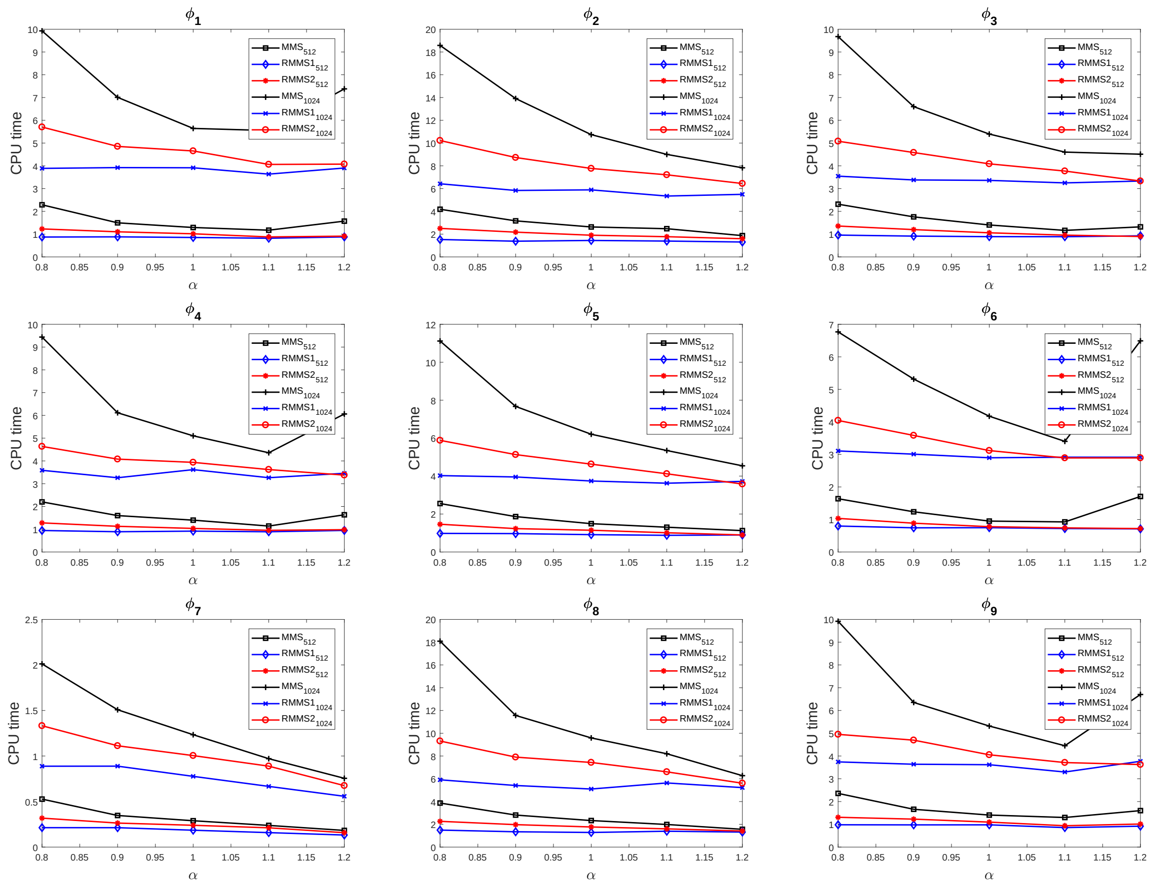

Figure 5. Comparison of CPU times between Method 2.1 and Method 2.2 for Example 5.10.

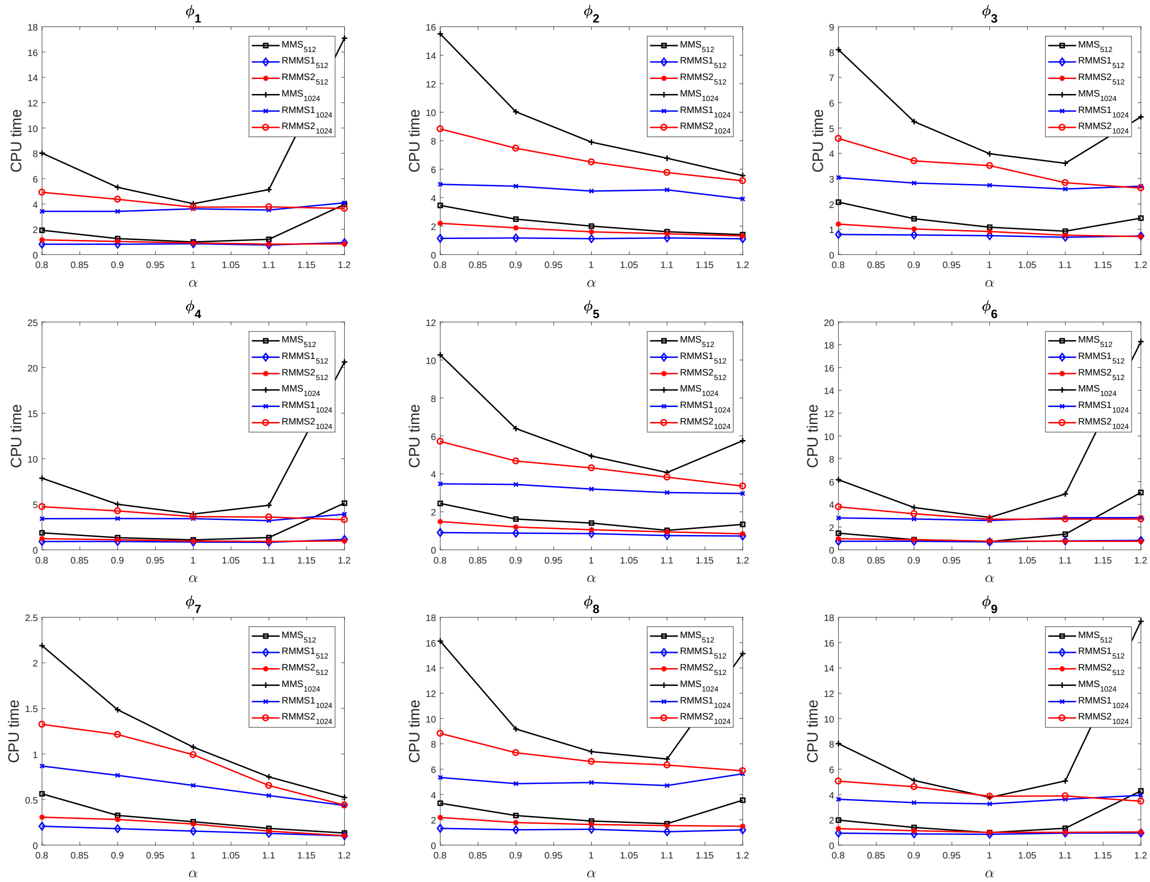

Figure 6. Comparison of CPU times between Method 2.1 and Method 2.2 for Example 5.11.

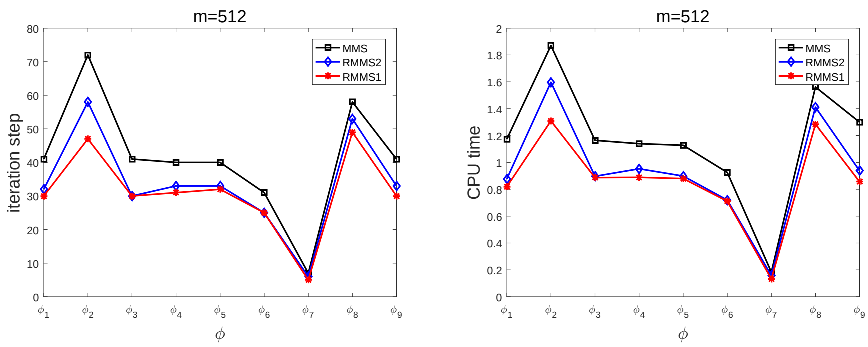

Figure 7. Optimal comparisons between Method 2.1 and Method 2.2 for Example 5.10 when $m = 512$.

Additionally, it is essential to acknowledge that, besides relaxation, there are various techniques that have been employed to improve the convergence speed of MMS iteration methods across different complementarity problems, as reviewed in Section 1. It is conceivable that combining one or more of these techniques with the relaxation method could lead to even better performance in solving large sparse HNCPs. This avenues represent promising directions for future research.

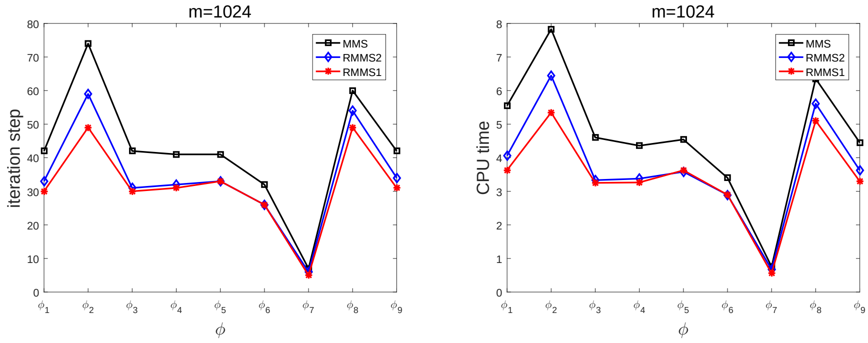

Figure 8. Optimal comparisons between Method 2.1 and Method 2.2 for Example 5.10 when $m = 1024$.

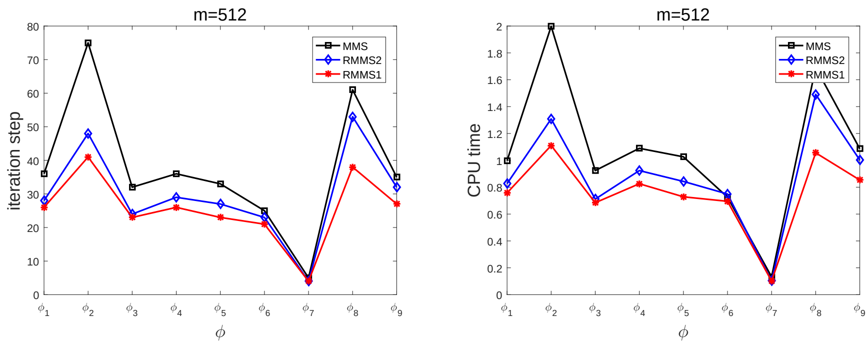

Figure 9. Optimal comparisons between Method 2.1 and Method 2.2 for Example 5.11 when $m = 512$.

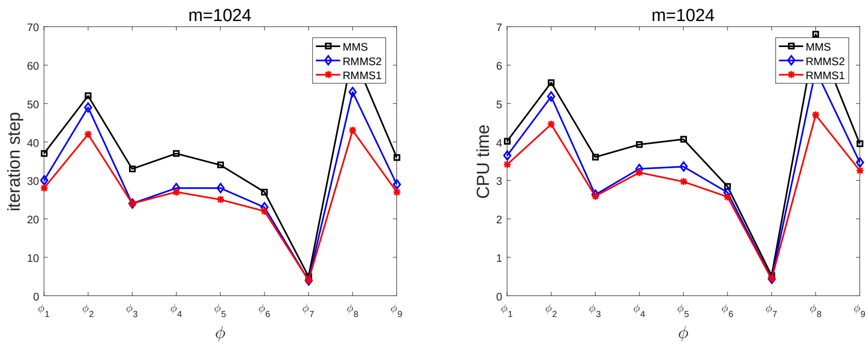

Figure 10. Optimal comparisons between Method 2.1 and Method 2.2 for Example 5.11 when $m = 1024$.

Use of AI tools declaration

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

Acknowledgement

This work was supported by Key Laboratory of Kunpeng+Intelligent Manufacturing in Guangdong Province (No. 2023KSYS012), Research on Key Technologies of Industrial Internet Collaborative Manufacturing Based on Blockchain (No. 2022ZDJS146), Basic and Applied Basic Research Foundation of Guangdong (No. 2024A1515011822), Guangdong Provincial Ordinary University Characteristic Innovation Project (No. 2025KTSCX138), and Science and Technology Development Fund, Macau SAR (No. 0026/2025/RIA1).

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. R. W. Cottle, J. S. Pang, R. E. Stone, *The Linear Complementarity Problem*, Society for Industrial and Applied Mathematics, 1992.
2. M. C. Ferris, O. Mangasarian, J. S. Pang, *Complementarity: Applications, Algorithms and Extensions*, Springer Science & Business Media, 2013.
3. F. Facchinei, J. S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer, New York, 2023.
4. C. Delprete, A. Razavykia, Piston ring-liner lubrication and tribological performance evaluation: A review, *Proc. Inst. Mech. Eng. Part J*, **232** (2018), 193–209. https://doi.org/10.1177/1350650117706269
5. M. Giacopini, M. Fowell, D. Dini, A. Strozzi, A mass-conserving complementarity formulation to study lubricant films in the presence of cavitation, *J Tribol*, **132** (2010), 041702. https://doi.org/10.1115/1.4002215
6. M. Kostreva, Elasto-hydrodynamic lubrication: A nonlinear complementarity problem, *Int J Numer Methods Fluids*, **4** (1984), 377–397. https://doi.org/10.1002/fld.1650040407
7. G. H. Meyer, Free boundary problems with nonlinear source terms, *Numer. Math.*, **43** (1984), 463–482. https://doi.org/10.1007/BF01390185
8. K. Oh, The numerical solution of dynamically loaded elastohydrodynamic contact as a nonlinear complementarity problem, *J. Tribol.*, **106** (1984), 88–95. https://doi.org/10.1115/1.3260872
9. F. Mezzadri, E. Galligani, Modulus-based matrix splitting methods for a class of horizontal nonlinear complementarity problems, *Numer. Algorithms*, **87** (2021), 667–687. https://doi.org/10.1007/s11075-020-00983-w
10. Z. Z. Bai, Modulus-based matrix splitting iteration methods for linear complementarity problems, *Numer. Linear Algebra Appl.*, **17** (2010), 917–933. https://doi.org/10.1002/nla.680
11. Z. C. Xia, C. L .Li, Modulus-based matrix splitting iteration methods for a class of nonlinear complementarity problem, *Appl. Math. Comput.*, **271** (2015), 34–42. https://doi.org/10.1016/j.amc.2015.08.108

12. F. Mezzadri, E. Galligani, Modulus-based matrix splitting methods for horizontal linear complementarity problems, *Numer. Algorithms*, **83** (2020), 201–219. https://doi.org/10.1007/s11075-019-00677-y

13. F. Bu, S. Vong, H. Zheng, Modulus-based synchronous multisplitting method for horizontal nonlinear complementarity problem, *J. Appl. Math. Comput.*, **70** (2024), 2405–2426. https://doi.org/10.1007/s12190-024-02059-7

14. Z. Li, H. Zhang, Anderson acceleration of the modulus based matrix splitting algorithms for horizontal nonlinear complementarity systems, *Numer. Linear Algebra Appl.*, **29**(2022), e2438. https://doi.org/10.1002/nla.2438

15. X. H. Shao, Z. Wang, The nonsmooth Newton's method for the horizontal nonlinear complementarity problem, *Numer. Algorithms*, **96** (2024), 75–103. https://doi.org/10.1007/s11075-023-01640-8

16. X. M. Fang, The convergence of the modulus-based Jacobi (MJ) iteration method for solving horizontal linear complementarity problems, *Comput. Appl. Math.*, **41** (2022), 134. https://doi.org/10.1007/s40314-022-01842-1

17. X. M. Fang, The convergence of a modulus-based matrix splitting iteration method for solving the implicit complementarity problems, *J. Appl. Math. Comput.*, **69** (2023), 853–870. https://doi.org/10.1007/s12190-022-01773-4

18. Z. F. Fu, S. L. Wu, L. Li, New modulus-based matrix splitting method for the vertical nonlinear complementarity problem, *J. Comput. Appl. Math.*, **457** (2025), 116251. https://doi.org/10.1016/j.cam.2024.116251

19. J. He, S. Vong, Newton-type methods for solving vertical linear complementarity problems, *J. Comput. Appl. Math.*, **460** (2025), 116418. https://doi.org/10.1016/j.cam.2024.116418

20. D. Hussain, K. J. Pan, B. Kumar, Numerical exploration of two generalized iteration methods for solving nonlinear complementarity problems, *J. Appl. Math. Comput.*, **71** (2025), 3381–3397. https://doi.org/10.1007/s12190-024-02334-7

21. F. Mezzadri, E. Galligani, Modulus-based matrix splitting algorithms for generalized complex-valued horizontal linear complementarity problems, *J. Comput. Appl. Math.*, **460** (2025), 116440. https://doi.org/10.1016/j.cam.2024.116440

22. Y. J. Xie, Y. F. Ke, Neural network approaches based on new NCP-functions for solving tensor complementarity problem, *J. Appl. Math. Comput.*, **67** (2021), 833–853. https://doi.org/10.1007/s12190-021-01509-w

23. H. Zheng, W. Li, S. Vong, A relaxation modulus-based matrix splitting iteration method for solving linear complementarity problems, *Numer. Algorithms*, **74** (2017), 137–152. https://doi.org/10.1007/s11075-016-0142-7

24. H. Zheng, S. Vong, L. Liu, The relaxation modulus-based matrix splitting iteration method for solving a class of nonlinear complementarity problems, *Intern. J. Comput. Math.*, **96** (2019), 1648–1667. https://doi.org/10.1080/00207160.2018.1504928

25. Y. F. Ke, C. F. Ma, H. Zhang, The relaxation modulus-based matrix splitting iteration methods for circular cone nonlinear complementarity problems, *Comput. Appl. Math.*, **37** (2018), 6795–6820. https://doi.org/10.1007/s40314-018-0687-2
26. N. Li, J. Ding, J. F. Yin, Modified relaxation two-sweep modulus-based matrix splitting iteration method for solving a class of implicit complementarity problems, *J. Comput. Appl. Math.*, **413** (2022), 114370. https://doi.org/10.1016/j.cam.2022.114370
27. D. Wang, J. Li, Relaxation modulus-based matrix splitting iteration method for vertical linear complementarity problem, *J. Comput. Appl. Math.*, **437** (2024), 115430. https://doi.org/10.1016/j.cam.2023.115430
28. Z. G. Huang, J. J. Cui, The relaxation modulus-based matrix splitting iteration method for horizontal linear complementarity problems, *Bull. Iran. Math. Soc.*, **48** (2022), 3285–3319. https://doi.org/10.1007/s41980-022-00695-y
29. Z. G. Huang, J. J. Cui, A relaxation two-sweep modulus-based matrix splitting iteration method for horizontal linear complementarity problems, *Jpn. J. Ind. Appl. Math.*, **40** (2023), 141–182. https://doi.org/10.1007/s13160-022-00514-1
30. D. Wang, J. Li, General double-relaxation two-sweep modulus-based matrix splitting iteration methods for horizontal linear complementarity problem, *Numer. Algorithms*, **98** (2025), 1965–1983. https://doi.org/10.1007/s11075-024-01860-6
31. Z. Z. Bai, On the convergence of the multisplitting methods for the linear complementarity problem, *SIAM J. Matrix Anal. Appl.*, **21** (1999), 67–78. https://doi.org/10.1137/S0895479897324032
32. A. Berman, R. J. Plemmons, *Nonnegative Matrix in the Mathematical Sciences*, Society for Industrial and Applied Mathematics, Philadelphia, 1994. https://doi.org/10.1137/1.9781611971262
33. A. Frommer, G. Mayer, Convergence of relaxed parallel multisplitting methods, *Linear Algebra Appl.*, **119** (1989), 141–152. https://doi.org/10.1016/0024-3795(89)90074-8
34. J. G. Hu, Estimates of $\|B^{-1}C\|_\infty$ and their applications, *Math. Numer. Sin.*, **4** (1982), 272–282.

AIMS Press