Japan Journal of Industrial and Applied Mathematics (2021) 38:1023–1036
https://doi.org/10.1007/s13160-021-00466-y

ORIGINAL PAPER

A two-step iteration method for the horizontal nonlinear complementarity problem

Hua Zheng[1] · **Liang Luo**[1] · **Shao-Yong Li**[1]

Received: 23 November 2020 / Revised: 31 March 2021 / Accepted: 7 April 2021 /
Published online: 13 April 2021

Abstract
In this paper, for solving horizontal nonlinear complementarity problem (HNCP), a two-step modulus-based matrix splitting iteration method is established. The convergence analysis of the proposed method is presented. Numerical examples are reported to show the efficiency of the proposed method.

Keywords Horizontal nonlinear complementarity problem · Modulus-based matrix splitting iteration method · Two-step method

Mathematics Subject Classification 65F10 · 90C33

1 Introduction

The horizontal nonlinear complementarity problem (HNCP) is a generalization of some complementarity problems, such as horizontal linear complementarity problem (HLCP), nonlinear complementarity problem (NCP), linear complementarity problem (LCP) [4, 7, 8]. Given $A, B \in \mathbb{R}^{n\times n}$, $q \in \mathbb{R}^n$ and a nonlinear $\varphi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$, we focus on solving HNCPs with weak nonlinearity, which consist in finding two vectors $z, r \in \mathbb{R}^n$ such that

$$Az - Br + q + \varphi(z, r) = 0,\ z, r \geq 0 \text{ and } z^T r = 0, \tag{1}$$

✉ Liang Luo
liangluo825@163.com

Hua Zheng
hzheng@sgu.edu.cn

Shao-Yong Li
lishaoyongok@sohu.com

[1] School of Mathematics and Statistics, Shaoguan University, Shaoguan, People's Republic of China

where for $F = (f_{ij}), G = (g_{ij}) \in \mathbb{R}^{m\times n}$, the order $F \geq (>)G$ means $f_{ij} \geq (>)g_{ij}$ for any i and j.

The HNCP can arise in some applications, for example from the discretization of the differential equation in hydrodynamic lubrication with complementarity constraints and a weak nonlinear source term [5, 9, 12, 13, 18]. Recently, for solving the HNCP, by reforming the HNCP to an equivalent implicit fixed-point system, the modulus-based matrix splitting (MMS) iteration method was introduced in [16], which generalizes the MMS iteration methods for LCP [2], NCP [11, 19] and HLCP [14, 15, 28, 30], and was shown to be more efficient than the reduction approaches of the HNCP. To see more details on the MMS iteration methods for different complementarity problems, readers can refer to the recent works [17, 21, 24, 26, 27] and the references therein.

In particular, among the existing improved techniques of the MMS, the two-step splittings had been successfully used in LCP [22, 23], NCP [20, 25] and HLCP [29], by making full use of the information contained in the two system matrices. It is interesting to investigate the two-step splittings in the MMS of the HNCP. In this paper, in order to achieve higher computing efficiency, in Sect. 2, we establish the two-step modulus-based matrix splitting (TMMS) iteration method for the HNCP, which directly extends the ones in [16] and [29]. In Sect. 3, the convergence theorems of the proposed method are given, which generalize the existing results. Next, numerical examples are presented to show the efficiency of the proposed method in Sect. 4. Finally, concluding remarks are given in Sect. 5.

Next, we introduce some definitions, notations and existing results.

Let $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ and $A = D_A - L_A - U_A = D_A - C_A$, where $D_A, -L_A, -U_A$ and $-C_A$ denote the diagonal, the strictly lower-triangular, the strictly upper-triangular and the nondiagonal matrices of A, respectively. By $|A|$ we denote $|A| = (|a_{ij}|)$. $\langle A\rangle = (\langle a_{ij}\rangle)$ is the comparison matrix of A, where $\langle a_{ii}\rangle = |a_{ii}|$ if $i = j$ and $\langle a_{ij}\rangle = -|a_{ij}|$ if $i \neq j$. We call A a Z-matrix if $C_A \leq 0$; a nonsingular M-matrix if $C_A \leq 0$ and $A^{-1} \geq 0$; an H-matrix if $\langle A\rangle$ is a nonsingular M-matrix; an H_+-matrix if A is an H-matrix with positive diagonal entries; a strictly diagonal dominant (s.d.d.) matrix if $|a_{ii}| > \sum_{j\neq i} |a_{ij}|$ for all $1 \leq i \leq n$ (e.g., see [1, 3]). If $\langle A\rangle = \langle M\rangle - |N|$, we call $A = M - N$ an H-compatible splitting. Denote the identity matrix of order n and the Kronecker product by I_n and "$\otimes$", respectively.

2 Two-step method

First, the MMS iteration method for solving the HNCP is reviewed.

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

$$(M_B\Omega + M_A)x = (N_B\Omega + N_A)x + (B\Omega - A)|x| - \gamma(q + \varphi(z, r)), \tag{2}$$

where Ω is a positive diagonal parameter matrix and γ is a positive constant; see [16] for more details. The MMS iteration method is presented based on (2) as follows:

Method 1 [16] *Let $\Omega \in \mathbb{R}^{n\times n}$ be a positive diagonal matrix, γ be a positive constant and $A = M_A - N_A, B = M_B - N_B$ be two splittings of the matrix $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times n}$, respectively. Given an initial vector $x^{(0)} \in \mathbb{R}^n$, compute $x^{(k+1)} \in \mathbb{R}^n$ by solving the linear system*

$$(M_B\Omega + M_A)x^{(k+1)} = (N_B\Omega + N_A)x^{(k)} + (B\Omega - A)|x^{(k)}| - \gamma(q + \varphi(z, r)).$$

Then set $z^{(k+1)} = \frac{1}{\gamma}(|x^{(k+1)}| + x^{(k+1)})$ and $r^{(k+1)} = \frac{1}{\gamma}\Omega(|x^{(k+1)}| - x^{(k+1)})$ for $k = 0, 1, 2, \ldots$, until the iteration sequence $\{(z^{(k)}, r^{(k)})\}_{k=1}^{+\infty}$ is convergent.

To achieve high computing efficiency, making use of the information in the matrices A and B by two matrix splittings, the TMMS iteration method for solving the HNCP is established as follows:

Method 2 Two-step modulus-based matrix splitting iteration method for the HNCP *For any given positive diagonal matrix $\Omega \in \mathbb{R}^{n\times n}$ and $\gamma > 0$, let $A = M_{A_1} - N_{A_1} = M_{A_2} - N_{A_2}$ be two splittings of the matrix $A \in \mathbb{R}^{n\times n}$, while $B = M_{B_1} - N_{B_1} = M_{B_2} - N_{B_2}$ be two splittings of the matrix $B \in \mathbb{R}^{n\times n}$. Given an initial vector $x^{(0)} \in \mathbb{R}^n$, compute $x^{(k+1)} \in \mathbb{R}^n$ by solving*

$$\begin{cases} (M_{B_1}\Omega + M_{A_1})x^{(k+\frac{1}{2})} = (N_{B_1}\Omega + N_{A_1})x^{(k)} + (B\Omega - A)|x^{(k)}| \\ \qquad\qquad - \gamma(q + \varphi(z^{(k)}, r^{(k)})), \\ (M_{B_2}\Omega + M_{A_2})x^{(k+1)} = (N_{B_2}\Omega + N_{A_2})x^{(k+\frac{1}{2})} + (B\Omega - A)|x^{(k+\frac{1}{2})}| \\ \qquad\qquad - \gamma(q + \varphi(z^{(k+\frac{1}{2})}, r^{(k+\frac{1}{2})})). \end{cases} \tag{3}$$

Then set $z^{(k+1)} = \frac{1}{\gamma}(|x^{(k+1)}| + x^{(k+1)})$ and $r^{(k+1)} = \frac{1}{\gamma}\Omega(|x^{(k+1)}| - x^{(k+1)})$ for $k = 0, 1, 2, \ldots$, until the iteration sequence $\{(z^{(k)}, r^{(k)})\}_{k=1}^{+\infty}$ is convergent.

Furthermore, if we take

$$\begin{cases} M_{A_1} = \frac{1}{\alpha}(D_A - \beta L_A), M_{A_2} = \frac{1}{\alpha}(D_A - \beta U_A), \\ M_{B_1} = \frac{1}{\alpha}(D_B - \beta L_B), M_{B_2} = \frac{1}{\alpha}(D_B - \beta U_B), \end{cases} \tag{4}$$

we can obtain the two-step modulus-based accelerated overrelaxation (TMAOR) iteration method. Taking $\alpha = \beta$ and $\alpha = \beta = 1$, the TMAOR iteration method reduces to the two-step modulus-based successive overrelaxation (TMSOR) iteration method and the two-step modulus-based Gauss-Seidel (TMGS) iteration method, respectively.

It is noted that Method 2 generalizes some existing methods for various complementarity problems:

- if $M_{A_1} = M_{A_2}, N_{A_1} = N_{A_2}, M_{B_1} = M_{B_2}$ and $N_{B_1} = N_{B_2}$, Method 2 reduces to Method 1.
- if $M_{B_1} = M_{B_2} = I, N_{B_1} = N_{B_2} = 0$ and $\varphi(z, r) = \varphi(z)$, Method 2 reduces to the two-step modulus-based matrix splitting iteration method for the NCP [20, 25].
- if $\varphi = 0$, Method 2 reduces to the two-step modulus-based matrix splitting iteration method for the HLCP [29].
- if $M_{B_1} = M_{B_2} = I, N_{B_1} = N_{B_2} = 0$ and $\varphi = 0$, Method 2 reduces to the two-step modulus-based matrix splitting iteration method for the LCP [22].

3 Convergence analysis

Some useful lemmas are given first.

Lemma 1 [6] *Let A be an H-matrix. Then $|A^{-1}| \leq \langle A \rangle^{-1}$.*

Lemma 2 [10] *Let $B \in \mathbb{R}^{n\times n}$ be an s.d.d. matrix. Then, $\forall C \in \mathbb{R}^{n\times n}$,*

$$||B^{-1}C||_\infty \leq \max_{1\leq i\leq n} \frac{(|C|e)_i}{(\langle B\rangle e)_i}$$

holds, where $e = (1, 1, \ldots, 1)^T \in \mathbb{R}^n$.

Lemma 3 [3] *If A is a nonsingular M-matrix, then there exists a positive diagonal matrix D, such that AD is an s.d.d. matrix with positive diagonal entries.*

In the following discussion, we assume that the HNCP has an unique solution (z^*, r^*). Moreover, we also assume that $\varphi(z, r)$ satisfies the smoothness assumptions as those in [16] as below: let

$$\varphi(z, r) = \big(\varphi_1(z_1, r_1), \varphi_2(z_2, r_2), \ldots, \varphi_n(z_n, r_n)\big)^T$$

be differentiable with

$$0 \leq \frac{\partial \varphi_i}{\partial z_i} \leq \psi_{z_i} \text{ and } 0 \leq \frac{\partial \varphi_i}{\partial r_i} \leq \psi_{r_i},$$

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

Then by the same deduction as that in Sect. 3 of [16], we have

$$\begin{aligned} &\varphi(z^{(k)}, r^{(k)}) - \varphi(z^*, r^*) \\ &\quad = \frac{1}{\gamma}\big[(\boldsymbol{\Phi}_z^{(k)} - \boldsymbol{\Phi}_r^{(k)}\Omega)(x^{(k)} - x^*) + (\boldsymbol{\Phi}_z^{(k)} + \boldsymbol{\Phi}_r^{(k)}\Omega)(|x^{(k)}| - |x^*|)\big], \end{aligned} \tag{5}$$

with $\boldsymbol{\Phi}_z^{(k)} \leq \boldsymbol{\Psi}_z$ and $\boldsymbol{\Phi}_r^{(k)} \leq \boldsymbol{\Psi}_r$, where

$$\begin{aligned}
\boldsymbol{\Phi}_z^{(k)} &= \operatorname{diag}\left(\frac{\partial \varphi_1}{\partial z_1}(\eta_1), \frac{\partial \varphi_2}{\partial z_2}(\eta_2), \ldots, \frac{\partial \varphi_n}{\partial z_n}(\eta_n)\right),\\
\boldsymbol{\Phi}_r^{(k)} &= \operatorname{diag}\left(\frac{\partial \varphi_1}{\partial r_1}(\eta_1), \frac{\partial \varphi_2}{\partial r_2}(\eta_2), \ldots, \frac{\partial \varphi_n}{\partial r_n}(\eta_n)\right),\\
\Psi_z &= \operatorname{diag}(\psi_{z_1}, \psi_{z_2}, \ldots, \psi_{z_n}),\\
\Psi_r &= \operatorname{diag}(\psi_{r_1}, \psi_{r_2}, \ldots, \psi_{r_n}),
\end{aligned}$$

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

Then, by (2) and straightforward computation, we can get that $x^* = \frac{\gamma}{2}(z^* - \Omega^{-1} r^*)$ satisfies the implicit fixed-point equations

$$\begin{cases}
(M_{B_1}\Omega + M_{A_1})x^* = (N_{B_1}\Omega + N_{A_1})x^* + (B\Omega - A)|x^*| - \gamma(q + \varphi(z^*, r^*)),\\
(M_{B_2}\Omega + M_{A_2})x^* = (N_{B_2}\Omega + N_{A_2})x^* + (B\Omega - A)|x^*| - \gamma(q + \varphi(z^*, r^*)).
\end{cases} \tag{6}$$

By subtracting (6) from (3), we have the error equations:

$$\begin{cases}
(M_{B_1}\Omega + M_{A_1})(x^{(k+\frac{1}{2})} - x^*) = (N_{B_1}\Omega + N_{A_1})(x^{(k)} - x^*)\\
\qquad\qquad + (B\Omega - A)(|x^{(k)}| - |x^*|)\\
\qquad\qquad - \gamma[\varphi(z^{(k)}, r^{(k)}) - \varphi(z^*, r^*)]\\
(M_{B_2}\Omega + M_{A_2})(x^{(k+1)} - x^*) = (N_{B_2}\Omega + N_{A_2})(x^{(k+\frac{1}{2})} - x^*)\\
\qquad\qquad + (B\Omega - A)(|x^{(k+\frac{1}{2})}| - |x^*|)\\
\qquad\qquad - \gamma[\varphi(z^{(k+\frac{1}{2})}, r^{(k+\frac{1}{2})}) - \varphi(z^*, r^*)].
\end{cases} \tag{7}$$

To prove $\lim\limits_{k\to+\infty} z^{(k)} = z^*$ and $\lim\limits_{k\to+\infty} r^{(k)} = r^*$, we need only to prove $\lim\limits_{k\to+\infty} x^{(k)} = x^*$.

Theorem 3.1 *Let $\Omega = (\omega_{jj})$ be an $n \times n$ positive diagonal matrix and $A, B \in \mathbb{R}^{n\times n}$ be two H_+-matrices. Let D be a positive diagonal matrix such that $\langle A\rangle D$ is an s.d.d. matrix. Assume that $A = M_{A_1} - N_{A_1} = M_{A_2} - N_{A_2}$ and $B = M_{B_1} - N_{B_1} = M_{B_2} - N_{B_2}$ are two H-compatible splittings of A and B, respectively; $|b_{ij}|\omega_{jj} \le |a_{ij}|$ $(i \ne j)$ and $\operatorname{sign}(b_{ij}) = \operatorname{sign}(a_{ij})$ $(b_{ij} \ne 0)$, for all i, j; and $D_B \ge \Psi_r$. Then for any initial vector $x^{(0)} \in \mathbb{R}^n$, the iteration sequence $\{(z^{(k)}, r^{(k)})\}_{k=1}^{+\infty}$ generated by Method2 converges to the unique solution (z^*, r^*) of the HNCP provided*

$$(D_B - \Psi_r)\Omega \ge (D_A + \Psi_z) \tag{8}$$

or

$$(|C_A| + \Psi_z)De < (D_B - \Psi_r)\Omega De \ \text{ and } \ (D_B - \min_{1\le i\le n}\psi_{r_i})\Omega \le D_A + \min_{1\le i\le n}\psi_{z_i}. \tag{9}$$

Proof By the assumption of H-compatible splittings, we have

$$\begin{aligned}
&\langle M_{B_1}\Omega + M_{A_1}\rangle De \\
&\quad \geq (\langle M_{B_1}\rangle\Omega + \langle M_{A_1}\rangle)De \\
&\quad = (\langle A\rangle + \langle B\rangle\Omega + |N_{A_1}| + |N_{B_1}|\Omega)De \\
&\quad \geq (D_A + D_B\Omega - |C_A| - |C_B|\Omega)De.
\end{aligned} \tag{10}$$

Since $|b_{ij}|\omega_{jj} \leq |a_{ij}|$ $(i \neq j)$ and $\mathrm{sign}(b_{ij}) = \mathrm{sign}(a_{ij})$ $(b_{ij} \neq 0)$, for all i, j, we get

$$|C_A| \geq |C_B|\Omega \Rightarrow |C_A + C_B\Omega| + |C_A - C_B\Omega| = 2|C_A|. \tag{11}$$

By (10) and (11), we can obtain

$$\begin{aligned}
&\langle M_{B_1}\Omega + M_{A_1}\rangle De \\
&\geq \begin{cases} (2D_A - 2|C_A| + \Psi_z + \Psi_r\Omega)De \geq 2\langle A\rangle De > 0, & \text{if (8) holds;} \\ (D_A - |C_B|\Omega + \Psi_z + \Psi_r\Omega)De > (D_A - |C_A|)De = \langle A\rangle De > 0, & \text{if (9) holds.} \end{cases}
\end{aligned}$$

Therefore, $\langle M_{B_1}\Omega + M_{A_1}\rangle D$ is an s.d.d. matrix, which implies that $M_{B_1}\Omega + M_{A_1}$ is an H-matrix. Then, by Lemma 1, (5) and the first equality of (7), we have

$$\begin{aligned}
&|x^{(k+\frac{1}{2})} - x^*| \\
&\quad = \Big|(M_{B_1}\Omega + M_{A_1})^{-1}\big\{(N_{B_1}\Omega + N_{A_1})(x^{(k)} - x^*) + (B\Omega - A)(|x^{(k)}| - |x^*|) \\
&\qquad -\gamma[\varphi(z^{(k)}, r^{(k)}) - \varphi(z^*, r^*)]\big\}\Big| \\
&\quad = \Big|(M_{B_1}\Omega + M_{A_1})^{-1}\big[(N_{B_1}\Omega + N_{A_1} - \boldsymbol{\Phi}_z^{(k)} + \boldsymbol{\Phi}_r^{(k)}\Omega)(x^{(k)} - x^*) \\
&\qquad +(B\Omega - A - \boldsymbol{\Phi}_z^{(k)} - \boldsymbol{\Phi}_r^{(k)}\Omega)(|x^{(k)}| - |x^*|)\big]\Big| \\
&\quad \leq \langle M_{B_1}\Omega + M_{A_1}\rangle^{-1}(|N_{B_1}\Omega + N_{A_1} - \boldsymbol{\Phi}_z^{(k)} + \boldsymbol{\Phi}_r^{(k)}\Omega| \\
&\qquad + |B\Omega - A - \boldsymbol{\Phi}_z^{(k)} - \boldsymbol{\Phi}_r^{(k)}\Omega|)|x^{(k)} - x^*| \\
&\quad \doteq \mathcal{P}_1|x^{(k)} - x^*| \\
&\quad \doteq \mathcal{M}_1^{-1}\mathcal{N}_1|x^{(k)} - x^*|,
\end{aligned}$$

where

$$\mathcal{M}_1 = \langle M_{B_1}\Omega + M_{A_1}\rangle$$

and

$$\mathcal{N}_1 = |N_{B_1}\Omega + N_{A_1} - \boldsymbol{\Phi}_z^{(k)} + \boldsymbol{\Phi}_r^{(k)}\Omega| + |B\Omega - A - \boldsymbol{\Phi}_z^{(k)} - \boldsymbol{\Phi}_r^{(k)}\Omega|.$$

Similarly, by the second equality of (7), we have

$$|x^{(k+1)} - x^*| \leq \mathcal{P}_2|x^{(k+\frac{1}{2})} - x^*| = \mathcal{M}_2^{-1}\mathcal{N}_2|x^{(k+\frac{1}{2})} - x^*|,$$

where

$$\mathcal{M}_2 = \langle M_{B_2}\Omega + M_{A_2}\rangle$$

and

$$\mathcal{N}_2 = |N_{B_2}\Omega + N_{A_2} - \Phi_z^{(k+\frac{1}{2})} + \Phi_r^{(k+\frac{1}{2})}\Omega| + |B\Omega - A - \Phi_z^{(k+\frac{1}{2})} - \Phi_r^{(k+\frac{1}{2})}\Omega|.$$

By Lemma 2, we have

$$||D^{-1}\mathcal{P}_1 D||_\infty = ||(\mathcal{M}_1 D)^{-1}(\mathcal{N}_1 D)||_\infty \le \max_{1\le i\le n} \frac{(\mathcal{N}_1 De)_i}{(\mathcal{M}_1 De)_i}. \tag{12}$$

If (8) holds, we obtain

$$\begin{aligned}
&\mathcal{M}_1 De - \mathcal{N}_1 De\\
&= (\langle M_{B_1}\Omega + M_{A_1}\rangle - |N_{B_1}\Omega + N_{A_1} - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega|\\
&\quad - |B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|)De\\
&\ge (\langle M_{B_1}\Omega\rangle + \langle M_{A_1}\rangle - |N_{B_1}\Omega| - |N_{A_1}| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega - D_B\Omega + D_A\\
&\quad + \Phi_z^{(k)} + \Phi_r^{(k)}\Omega - |C_A| + |C_B\Omega|)De\\
&= 2\langle A\rangle De\\
&> 0.
\end{aligned} \tag{13}$$

If (9) holds, we obtain

$$\begin{aligned}
&\mathcal{M}_1 De - \mathcal{N}_1 De\\
&= (\langle M_{B_1}\Omega + M_{A_1}\rangle - |N_{B_1}\Omega + N_{A_1} - \Phi_z^{(k)} + \Phi_r^{(k)}\Omega|\\
&\quad - |B\Omega - A - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega|)De\\
&\ge (\langle M_{B_1}\Omega\rangle + \langle M_{A_1}\rangle - |N_{B_1}\Omega| - |N_{A_1}| - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega + D_B\Omega - D_A\\
&\quad - \Phi_z^{(k)} - \Phi_r^{(k)}\Omega - |C_A| + |C_B\Omega|)De\\
&\ge 2[(D_B - \Psi_r)\Omega - |C_A| - \Psi_z]De\\
&> 0.
\end{aligned} \tag{14}$$

Then, by (12), (13) and (14), we have $||D^{-1}\mathcal{P}_1 D||_\infty < 1$. Similarly, we can get

$$||D^{-1}\mathcal{P}_2 D||_\infty < 1$$

if (8) or (9) holds. Hence the next inequality holds:

$$\begin{aligned}
&\rho(\mathcal{P}_2\mathcal{P}_1)\\
&\quad = \rho(D^{-1}\mathcal{P}_2\mathcal{P}_1 D)\\
&\quad \le ||D^{-1}\mathcal{P}_2 D D^{-1}\mathcal{P}_1 D||_\infty\\
&\quad \le ||D^{-1}\mathcal{P}_2 D||_\infty ||D^{-1}\mathcal{P}_1 D||_\infty\\
&\quad < 1,
\end{aligned}$$

which implies that $\lim_{k\to+\infty} x^{(k)} = x^*$, proving the claim. □

Remark 1 If we take $\varphi(z, r) = 0$, which implies $\Psi_z = \Psi_r = 0$, then Theorem 3.1 reduces to Theorem 3.1 of [29].

Remark 2 If $\psi_{r_i} = \psi_{r_j}$ for any i, j, then (9) can be simplified to

$$(|C_A| + \Psi_z)De < (D_B - \Psi_r)\Omega De \le (D_A + \min_{1\le i\le n} \psi_{z_i})De.$$

Note that the assumption on the matrix splittings in Theorem 3.1 are H-compatible splittings. It is known that the TMAOR with $\alpha > 1$ does not belong to the cases. Next, we present the convergence results for the TMAOR.

Lemma 4 [29] *Let A, B be two H_+-matrices. If*

$$0 < \alpha < \frac{1}{\rho\left[(D_A + D_B\Omega)^{-1}(D_B\Omega + |C_A|)\right]},$$

there exists a positive diagonal matrix $\bar{D}$, such that

$$\left[\frac{1+\alpha-|1-\alpha|}{\alpha}D_A + \frac{1-\alpha-|1-\alpha|}{\alpha}D_B\Omega - 2|C_A|\right]\bar{D}$$

is an s.d.d. matrix.

Theorem 3.2 *Let $A, B \in \mathbb{R}^{n\times n}$ be two H_+-matrices and $\Omega \in \mathbb{R}^{n\times n}$ be a positive diagonal matrix satisfying $(D_B - \Psi_r)\Omega \ge (D_A + \Psi_z)$. Furthermore, for $i, j = 1, 2, \ldots, n$, let $|b_{ij}|\omega_{jj} \le |a_{ij}|(i \ne j)$ and $\mathrm{sign}(b_{ij}) = \mathrm{sign}(a_{ij})(b_{ij} \ne 0)$. Then, the iteration sequence $(z^{(k)}, r^{(k)})_{k=1}^{+\infty}$ generated by the TMAOR iteration method converges to the unique solution (z^*, r^*) of (1) for any initial vector $x^{(0)} \in \mathbb{R}^n$ provided*

$$0 < \beta \le \alpha < \frac{1}{\rho\left[(D_A + D_B\Omega)^{-1}(D_B\Omega + |C_A|)\right]}. \tag{15}$$

Proof With the same notations and discussion as the proof of Theorem 3.1, let $\bar{D}$ be the positive diagonal matrix given by Lemma 4. If $(D_B - \Psi_r)\Omega \ge (D_A + \Psi_z)$, by Lemma 4, (4) and (15), we have

$$
\begin{aligned}
&\mathcal{M}_1\bar{D}e - \mathcal{N}_1\bar{D}e \\
&= \left[\langle M_{A_1} + M_{B_1}\Omega\rangle - |N_{A_1} + N_{B_1}\Omega - \boldsymbol{\Phi}_z^{(k)} + \boldsymbol{\Phi}_r^{(k)}\Omega| \right.\\
&\quad \left. -|B\Omega - A - \boldsymbol{\Phi}_z^{(k)} - \boldsymbol{\Phi}_r^{(k)}\Omega|\right]\bar{D}e \\
&= \left[\frac{1}{\alpha}(D_A + D_B\Omega) - \frac{\beta}{\alpha}|L_A + L_B\Omega| - |\frac{|1-\alpha|}{\alpha}(D_A + D_B\Omega) - \boldsymbol{\Phi}_z^{(k)} + \boldsymbol{\Phi}_r^{(k)}\Omega| \right.\\
&\quad - \frac{|\beta-\alpha|}{\alpha}|L_A + L_B\Omega| - |U_A + U_B\Omega| - |D_B\Omega - D_A - \boldsymbol{\Phi}_z^{(k)} - \boldsymbol{\Phi}_r^{(k)}\Omega| \\
&\quad \left. -|C_B\Omega - C_A|\right]\bar{D}e \\
&\geq \left[\frac{1+\alpha-|1-\alpha|}{\alpha}D_A + \frac{1-\alpha-|1-\alpha|}{\alpha}D_B\Omega - |C_B\Omega + C_A| \right.\\
&\quad \left. -|C_B\Omega - C_A|\right]\bar{D}e \\
&= \left[\frac{1+\alpha-|1-\alpha|}{\alpha}D_A + \frac{1-\alpha-|1-\alpha|}{\alpha}D_B\Omega - 2|C_A|\right]\bar{D}e \\
&> 0.
\end{aligned}
$$

Hence $\mathcal{M}_1\bar{D}$ is an s.d.d. matrix. Then by Lemma 1, we have $||\bar{D}^{-1}\mathcal{P}_1\bar{D}||_\infty < 1$. Similarly, we can get $||\bar{D}^{-1}\mathcal{P}_2\bar{D}||_\infty < 1$ too. Therefore we also have $\rho(\mathcal{P}_2\mathcal{P}_1) < 1$, which implies the TMAOR iteration method is convergent. □

Remark 3 Clearly, Theorem 3.2 reduces to Theorem 3.2 of [29] when $\varphi(z, r) = 0$.

By the proof of Theorem 3.2, it is easy to have the convergence results for the MAOR iteration method which was not considered in [16].

Corollary 1 *With the same notations and assumptions as Theorem 3.2, if (15) holds, the MAOR iteration method for the HNCP converges globally.*

4 Numerical examples

In this section, numerical examples are given to show the efficiency of the proposed method. The computations were run on an Intel(R) Core(TM) (2.50 GHz CPU and 4.00 GB RAM).

Consider the following two examples in [16].

Example 1 [16] Consider the 2-D boundary problem:

$$\triangle z + \frac{\partial^2 r}{\partial^2 u} + \mu z + \nu r - q - \varphi(z, r) = 0, z \geq 0, r \geq 0 \text{ and } z^T r = 0,$$

where $z(u, v)$, $r(u, v)$, $q(u, v)$ are three 2-D mapings and μ, ν are real parameters.

By discretizing the problem using five-point difference scheme with suitable boundary conditions, one can get the HNCP. More concretely, the matrices A and B are given by $A = \hat{A} + \mu I_n$ and $B = \hat{B} + \nu I_n$, respectively, where $\hat{A} = \text{blktridiag}(-I_m, S, -I_m) \in \mathbb{R}^{n\times n}, \hat{B} = I_m \otimes S \in \mathbb{R}^{n\times n}, S = \text{tridiag}(-1, 4, -1) \in \mathbb{R}^{m\times m}$ and $n = m^2$.

Example 2 [16] Let $A = \hat{A} + \mu I_n, B = \hat{B} + \nu I_n$ and $q = Az^* - Bw^*$, where $n = m^2$, $\hat{A} = \text{blktridiag}(-1.5I_m, S, -0.5I_m) \in \mathbb{R}^{n\times n}, \hat{B} = \text{blktridiag}(-\tau I_m, S, -\tau I_m) \in \mathbb{R}^{n\times n}$, $S = \text{tridiag}(-1.5, 4, -0.5) \in \mathbb{R}^{m\times m}$ and μ, ν, τ are real parameters.

Consider the nonlinear functions as below:

$$\begin{aligned}
\varphi_1(z, r) &= \frac{z + r + \sin(z)\cos(z) + \sin(r)\cos(r)}{2}, \\
\varphi_2(z, r) &= -\frac{1}{1 + z + r}, \\
\varphi_3(z, r) &= \frac{\arctan(z) + \arctan(r)}{2}, \\
\varphi_4(z, r) &= \sin(z + r), \\
\varphi_5(z, r) &= \ln(1 + z + r).
\end{aligned}$$

Note that all these five functions satisfy the smoothness assumptions given in Sect. 3 with $\Psi_z = \Psi_r = I$.

Next, we compare the TMSOR$_\alpha$ with the MSOR$_\alpha$ for Example 1 and Example 2, where the subscript α denotes the relaxation parameter. Let $\gamma = 1$, $\Omega = (D_B - \Psi_r)^{-1}(D_A + \Psi_z)$, all initial iteration vectors be $x^{(0)} = e$ and the tolerance be set at 10^{-10}. In order to compare with the theoretical result, the upper bounds of α in (15) are presented in Table 1 for Example 1 and Example 2. Note that the upper bounds of α in Example 1 listed in Table 1 are all equal to 1 with 2 decimal digit accuracy. However, the exact bounds are all larger than 1, guaranteed by the proof of Lemma 4 in [29]. On the other hand, we also show the numerical results when $\alpha = 1.2$ to analyze the behavior of the methods in more critical cases.

Numerical results are reported in Tables 2 and 3, where "time" and "iter" denote the CPU time in seconds and the iteration steps, respectively. Specially, the percentages of CPU time saved by the TMSOR iteration method from the MSOR iteration method (denoted by "save") are also presented:

Table 1 Upper bounds of α in (15) when $\Omega = (D_B - \Psi_r)^{-1}(D_A + \Psi_z)$

m	Example 1	Example 2
128	1.00	1.06
256	1.00	1.06
512	1.00	1.06

Table 2 Numerical results of Example 1 when $\mu = 0$ and $\nu = 4$

$\varphi(z, r)$	Method	$m = 128$			$m = 256$			$m = 512$		
		Iter	Time	Save (%)	Iter	Time	Save (%)	Iter	Time	Save (%)
φ_1	$\text{MSOR}_{0.8}$	60	0.2423		61	0.8765		63	4.8783	
	$\text{TMSOR}_{0.8}$	27	0.1300	46.35	28	0.5041	42.49	28	2.7360	43.91
	$\text{MSOR}_{1.0}$	41	0.1825		42	0.6194		43	3.3158	
	$\text{TMSOR}_{1.0}$	19	**0.0998**	45.32	19	**0.3662**	40.88	20	**2.0279**	38.84
	$\text{MSOR}_{1.2}$	53	0.2146		54	0.8215		56	6.5095	
	$\text{TMSOR}_{1.2}$	32	0.1409	34.34	33	0.6391	22.20	33	5.1911	20.25
φ_2	$\text{MSOR}_{0.8}$	82	0.2394		85	0.9349		87	5.3506	
	$\text{TMSOR}_{0.8}$	41	0.1832	23.48	42	0.6808	27.18	43	3.9220	26.70
	$\text{MSOR}_{1.0}$	59	0.1739		60	0.6708		62	3.8524	
	$\text{TMSOR}_{1.0}$	29	**0.1190**	31.57	29	**0.4925**	26.58	30	**2.6932**	30.09
	$\text{MSOR}_{1.2}$	46	0.1503		47	0.5851		48	3.2401	
	$\text{TMSOR}_{1.2}$	35	0.1549	− 3.06	35	0.5908	− 0.97	36	3.3655	− 3.87
φ_3	$\text{MSOR}_{0.8}$	75	0.2669		77	1.0292		79	5.7006	
	$\text{TMSOR}_{0.8}$	36	0.1645	38.37	37	0.7566	26.49	38	3.6788	35.47
	$\text{MSOR}_{1.0}$	53	0.2105		54	0.6600		56	3.8958	
	$\text{TMSOR}_{1.0}$	25	**0.1125**	46.56	26	**0.4451**	32.56	26	**2.4416**	37.33
	$\text{MSOR}_{1.2}$	49	0.2031		51	0.7009		52	3.6796	
	$\text{TMSOR}_{1.2}$	34	0.1464	27.92	35	0.6040	13.83	35	3.4294	6.80
φ_4	$\text{MSOR}_{0.8}$	57	0.2091		59	0.7031		60	3.9675	
	$\text{TMSOR}_{0.8}$	25	0.1242	40.60	26	0.5347	23.95	26	2.3594	40.53
	$\text{MSOR}_{1.0}$	39	0.1520		40	0.4974		41	2.9125	
	$\text{TMSOR}_{1.0}$	19	**0.0898**	40.92	19	**0.4081**	17.95	19	**1.9100**	34.42
	$\text{MSOR}_{1.2}$	48	0.1789		49	0.5989		51	3.5076	
	$\text{TMSOR}_{1.2}$	33	0.1529	14.53	33	0.5776	3.56	34	3.3970	3.15
φ_5	$\text{MSOR}_{0.8}$	63	0.2332		65	0.9272		67	5.2885	
	$\text{TMSOR}_{0.8}$	29	0.1332	42.88	30	0.5257	43.30	31	3.0959	41.46
	$\text{MSOR}_{1.0}$	44	0.1755		45	0.6151		46	3.4869	
	$\text{TMSOR}_{1.0}$	20	**0.0996**	43.25	21	**0.3946**	35.85	21	**2.0735**	40.53
	$\text{MSOR}_{1.2}$	45	0.2176		46	0.6261		47	3.8525	
	$\text{TMSOR}_{1.2}$	34	0.1627	25.23	34	0.5732	8.45	35	3.6354	5.64

$$\text{save} = \frac{\text{time}_{\text{MSOR}} - \text{time}_{\text{TMSOR}}}{\text{time}_{\text{MSOR}}} \times 100\%.$$

It is also noted that the "best" time for each set of experiments is highlighted by the bold font in Tables 2 and 3.

By the numerical results presented in Tables 2 and 3, the MSOR and the TMSOR iteration methods converge in all cases.

Table 3 Numerical results of Example 2 when $\mu = 0$ and $\nu = 4$

$\varphi(z, r)$	Method	$m = 128$			$m = 256$			$m = 512$		
		Iter	Time	Save (%)	Iter	Time	Save (%)	Iter	Time	Save (%)
φ_1	$\text{MSOR}_{0.8}$	51	0.2180		53	0.8037		54	4.2784	
	$\text{TMSOR}_{0.8}$	26	0.1281	41.24	26	0.4883	39.24	27	2.7175	36.48
	$\text{MSOR}_{1.0}$	33	0.1563		34	0.5037		34	2.7800	
	$\text{TMSOR}_{1.0}$	19	**0.0993**	36.47	19	**0.3519**	30.14	20	**2.0050**	27.88
	$\text{MSOR}_{1.2}$	87	0.3391		90	1.3156		93	7.3145	
	$\text{TMSOR}_{1.2}$	32	0.1571	53.67	33	0.6965	47.06	33	3.9030	46.64
φ_2	$\text{MSOR}_{0.8}$	72	0.2071		74	0.8252		75	4.9977	
	$\text{TMSOR}_{0.8}$	39	0.1727	16.61	40	0.6648	19.44	41	3.8966	22.03
	$\text{MSOR}_{1.0}$	48	0.1579		49	0.5364		51	3.2683	
	$\text{TMSOR}_{1.0}$	25	**0.1058**	33.00	26	**0.4122**	23.15	26	**2.4139**	26.14
	$\text{MSOR}_{1.2}$	67	0.2086		68	0.7480		70	4.6535	
	$\text{TMSOR}_{1.2}$	35	0.1554	25.50	35	0.5919	20.87	36	3.4527	25.80
φ_3	$\text{MSOR}_{0.8}$	65	0.2803		67	0.9268		68	5.3540	
	$\text{TMSOR}_{0.8}$	35	0.1864	33.50	36	0.6484	30.04	37	3.6413	31.99
	$\text{MSOR}_{1.0}$	43	0.1659		44	0.5543		45	3.2330	
	$\text{TMSOR}_{1.0}$	23	**0.1066**	35.74	23	**0.4014**	27.58	23	**2.2149**	31.49
	$\text{MSOR}_{1.2}$	78	0.2644		80	1.0101		83	6.0045	
	$\text{TMSOR}_{1.2}$	34	0.1451	45.12	35	0.6294	37.69	35	3.4899	41.88
φ_4	$\text{MSOR}_{0.8}$	49	0.1761		50	0.6115		51	3.5076	
	$\text{TMSOR}_{0.8}$	23	0.1101	37.48	24	0.4270	30.17	25	2.3959	31.69
	$\text{MSOR}_{1.0}$	31	0.1176		32	0.4072		32	2.2810	
	$\text{TMSOR}_{1.0}$	19	**0.0946**	19.56	19	**0.3297**	19.03	19	**1.8088**	20.70
	$\text{MSOR}_{1.2}$	72	0.2356		74	0.8973		76	5.2996	
	$\text{TMSOR}_{1.2}$	33	0.1582	32.85	33	0.5724	36.21	34	3.3270	37.22
φ_5	$\text{MSOR}_{0.8}$	54	0.2068		56	0.7699		57	4.3207	
	$\text{TMSOR}_{0.8}$	28	0.1334	35.49	29	0.5128	33.39	29	2.8660	33.67
	$\text{MSOR}_{1.0}$	35	0.1504		36	0.5016		37	2.8037	
	$\text{TMSOR}_{1.0}$	20	**0.1009**	32.91	21	**0.3604**	28.15	21	**2.0409**	27.21
	$\text{MSOR}_{1.2}$	65	0.2263		67	0.9407		68	5.2197	
	$\text{TMSOR}_{1.2}$	34	0.1478	34.69	34	0.6196	34.13	35	3.4937	33.07

- When $\alpha = 0.8$ and $\alpha = 1.0$, satisfying (15), due to the two-step iteration, the iteration step of the TMSOR is about half of those of the MSOR for each case. Furthermore, the TMSOR always converges faster than the MSOR and the TMSOR can save 16%-53% CPU time from those of the MSOR for a given α.
- When $\alpha = 1.2$, outside (15), the CPU time of the TMSOR are also less than those of the MSOR except for φ_2 in Example 1.

In summary, by the two-step technique, the TMSOR can significantly accelerate the convergence rate of the MSOR with α in (15) and the TMSOR performs the best when $\alpha = 1$.

5 Conclusions

We have established the TMMS iteration method for the HNCP by employing two-step matrix splittings, which generalizes the MMS iteration method for the HNCP in [16] and the TMMS iteration method for HLCP in [29]. The convergence results include and extend some existing results. The effectiveness of the TMMS iteration method is shown by numerical experiments. It is worth noticing by the numerical results that the upper bound of α in (15) may be enlarged. How to improve the convergence analysis is the future work.

On the other hand, it is known that multisplitting methods are well-suited for parallel computations. The two-step methods had been coupled with multisplitting techniques in modulus-based methods for LCPs [23]. In the recent work [17], the multisplitting techniques had been applied to HLCPs successfully. It is also an interesting topic to examine the two-step multisplitting methods for HNCP as well based on similar techniques for LCPs and using the multisplittings of the HLCPs.

Acknowledgements The authors would like to thank the reviewer for the helpful suggestions. This work was supported by the Major Projects of Guangdong Education Department for Foundation Research and Applied Research (No. 2018KZDXM065), Guangdong provincial Natural Science Foundation (No. 2018A0303100015), Characteristic innovation project of Guangdong Provincial Department of Education (No. 2018KTSCX204), and the Science Foundation of Shaoguan University (No. SZ2017KJ07, SZ2020ZK0).

References

1. Bai, Z.-Z.: On the convergence of the multisplitting methods for the linear complementarity problem. SIAM J. Matrix Anal. Appl. **21**, 67–78 (1999)
2. Bai, Z.-Z.: Modulus-based matrix splitting iteration methods for linear complementarity problems. Numer. Linear Algebra Appl. **17**, 917–933 (2010)
3. Berman, A., Plemmons, R.J.: Nonnegative Matrix in the Mathematical Sciences. SIAM Publisher, Philadelphia (1994)
4. Cottle, R.W., Pang, J.-S., Stone, R.E.: The Linear Complementarity Problem. Academic, San Diego (1992)
5. Delprete, C., Razavykia, A.: Piston ring-liner lubrication and tribological performance evaluation: a review. Proc. Inst. Mech. Eng. J-J Eng. **232**(2), 193–209 (2018)
6. Frommer, A., Mayer, G.: Convergence of relaxed parallel multisplitting methods. Linear Algebra Appl. **119**, 141–152 (1989)
7. Ferris, M.C., Mangasarian, O., Pang, J.-S.: Complementarity: Applications. Algorithms and Extensions. Springer, New York (2011)
8. Facchinei, F., Pang, J.-S.: Finite-Dimensional Variational Inequalities and Complementarity Problems I and II. Springer, New York (2003)
9. Giacopini, M., Fowell, M., Dini, D., Strozzi, A.: A mass-conserving complementarity formulation to study lubricant films in the presence of cavitation. J. Tribol. **132**, 041702 (2010)
10. Hu, J.-G.: Estimates of $||B^{-1}C||_\infty$ and their applications. Math. Numer. Sin. **4**, 272–282 (1982)

11. Huang, N., Ma, C.-F.: The modulus-based matrix splitting algorithms for a class of weakly nonlinear complementarity problems. Numer. Linear Algebra Appl. **23**, 558–569 (2016)
12. Kostreva, M.: Elasto-hydrodynamic lubrication: a nonlinear complementarity problem. Int. J. Numer. Meth. Fl. **4**, 377–397 (1984)
13. Meyer, G.H.: Free boundary problems with nonlinear source terms. Numer. Math. **43**, 463–482 (1984)
14. Mezzadri, F., Galligani, E.: Modulus-based matrix splitting methods for horizontal linear complementarity problems. Numer. Algor. **83**, 201–219 (2020)
15. Mezzadri, F., Galligani, E.: On the convergence of modulus-based matrix splitting methods for horizontal linear complementarity problems in hydrodynamic lubrication. Math. Comput. Simulat. **176**, 226–242 (2020)
16. Mezzadri F., Galligani E.: Modulus-based matrix splitting methods for a class of horizontal nonlinear complementarity problems. Numer. Algorithms (In press) https://doi.org/10.1007/s11075-020-00983-w
17. Mezzadri, F.: Modulus-based synchronous multisplitting methods for solving horizontal linear complementarity problems on parallel computers. Numer. Linear Algebra Appl. **27**, e2319 (2020)
18. Oh, K.: The numerical solution of dynamically loaded elastohydrodynamic contact as a nonlinear complementarity problem. ASME J. Tribol. **106**, 88–95 (1984)
19. Xia, Z.-C., Li, C.-L.: Modulus-based matrix splitting iteration methods for a class of nonlinear complementarity problem. Appl. Math. Comput. **271**, 34–42 (2015)
20. Xie, S.-L., Xu, H.-R., Zeng, J.-P.: Two-step modulus-based matrix splitting iteration method for a class of nonlinear complementarity problems. Linear Algebra Appl. **494**, 1–10 (2016)
21. Xu, W.-W., Zhu, L., Peng, X.-P., Liu, H., Yin, J.-F.: A class of modified modulus-based synchronous multisplitting iteration methods for linear complementarity problems. Numer. Algor. **85**, 1–21 (2020)
22. Zhang, L.-L.: Two-step modulus based matrix splitting iteration for linear complementarity problems. Numer. Algor **57**, 83–99 (2011)
23. Zhang, L.-L.: Two-step modulus-based synchronous multisplitting iteration methods for linear complementarity problems. J. Comput. Math. **33**, 100–112 (2015)
24. Zheng, H., Li, W., Vong, S.: A relaxation modulus-based matrix splitting iteration method for solving linear complementarity problems. Numer. Algor. **74**, 137–152 (2017)
25. Zheng, H., Liu, L.: A two-step modulus-based matrix splitting iteration method for solving nonlinear complementarity problems of H_+-matrices. Comput. Appl. Math. **37**(4), 5410–5423 (2018)
26. Zheng, H., Vong, S.: A modified modulus-based matrix splitting iteration method for solving implicit complementarity problems. Numer. Algor. **82**, 573–592 (2019)
27. Zheng, H., Li, W., Vong, S.: An iteration method for nonlinear complementarity problems. J. Comput. Appl. Math. **372**, 112681 (2020)
28. Zheng, H., Vong, S.: On convergence of the modulus-based matrix splitting iteartion method for horizontal linear complementarity problem of H_+-matrices. Appl. Math. Comput. **369**, 124890 (2020)
29. Zheng, H., Vong, S.: A two-step modulus-based matrix splitting iteration method for horizontal linear complementarity problems. Numer. Algor **86**, 1791–1810 (2021)
30. Zheng, H., Vong, S.: On the modulus-based successive overrelaxation iteration method for horizontal linear complementarity problems arising from hydrodynamic lubrication. Appl. Math. Comput. **402**, 126165 (2021)