Mathematics

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

AIMS Mathematics, 9(6): 14358–14375.
DOI:10.3934/math.2024698
Received: 21 February 2024
Revised: 06 April 2024
Accepted: 17 April 2024
Published: 19 April 2024

Research article

A two-step iteration method for solving vertical nonlinear complementarity problems

Wenxiu Guo[1,2], **Xiaoping Lu**[1,*] **and Hua Zheng**[2]

[1] School of Computer Science and Engineering, Macau University of Science and Technology, Macao, China

[2] School of Mathematics and Statistics, Shaoguan University, Shaoguan, China

* **Correspondence:** Email: xplu@must.edu.mo.

Abstract: In this paper, for vertical nonlinear complementarity problems, a two-step modulus-based matrix splitting iteration method is established by applying the two-step splitting technique to the modulus-based matrix splitting iteration method. The convergence theorems of the proposed method are given when the number of system matrices is larger than 2. Numerical results show that the convergence rate of the proposed method can be accelerated compared to the existing modulus-based matrix splitting iteration method.

Keywords: vertical nonlinear complementarity problem; two-step; modulus-based method; H-splitting
Mathematics Subject Classification: 65F10, 90C33

1. Introduction

Let $M_1, M_2 \ldots, M_s \in \mathbb{R}^{n\times n}$, and let $u_1(z), u_2(z), \ldots, u_s(z) : \mathbb{R}^n \to \mathbb{R}^n$, be s nonlinear mappings. The definition of the vertical nonlinear complementarity problem (abbreviated as VNCP) is: Find $z \in \mathbb{R}^n$ such that

$$r_i = M_i z + u_i(z), 1 \le i \le s, \quad \text{with} \quad \min\{z, r_1, \ldots, r_s\} = 0. \tag{1.1}$$

Here, the minimum operation is taken component-wise.

The VNCP has many applications in generalized Leontief input-output models, control theory, nonlinear networks, contact problems in lubrication, and so on; e.g., see [11, 18, 34, 35, 39]. Taking $u_i(z) = u_i \in \mathbb{R}^n, 1 \le i \le s$, the VNCP reduces to the vertical linear complementarity problem (abbreviated as VLCP) [9, 20, 40]. Further, taking $s = 1$, the VLCP reduces to the linear complementarity problem (abbreviated as LCP) [10].

In earlier literatures, there were some iterative methods for such problems. In [39], the Mangasarian's general iterative algorithm was constructed to solve the VLCP. Solving an approximation equation of the nonlinear complementarity problem by a continuation method can be extended to solve the VNCP; see [8]. Furthermore, in [37] the authors approximated the equivalent minimum equation of the VNCP by a sequence of aggregation functions, and found the zero solution of the approximated problems. Moreover, some interior-point methods were applied to solve complementarity problems, such as LCP [24, 29], nonlinear complementarity problems (abbreviated as NCP) [36] and horizontal LCP [46]. In recent years, modulus-based matrix splitting (abbreviated as MMS) iteration methods have gained popularity for numerical solutions to various complementarity problems. References [2, 12, 15, 16] detail their application to the LCP, while [13, 32, 50] focus on the horizontal LCP. The second-order cone LCP is discussed in [27], implicit complementarity problems in [7, 14, 25], quasi-complementarity problem in [38], NCP in [41], and the circular cone NCP is addressed in [28]. Numerical examples have demonstrated that MMS iteration methods often outperform state-of-the-art smooth Newton methods in practical applications. Specifically, for the VLCP, a special case of the VNCP, the MMS iteration method was introduced in [31]. Alternatively, an MMS iteration method without auxiliary variables based on a non-auxiliary-variable equivalent modulus equation was presented in [23] and shown to be more efficient than the method in [31]. Accelerated techniques and improved results for MMS iteration methods in the VLCP are further detailed in [21, 48, 49, 51]. On the other hand, Projected type methods were also used to solve the VLCP; see [6, 33]. For the VNCP, the only literature on the MMS iteration method currently is [42].

To improve the convergence rate of the MMS iteration method for solving the VNCP, in this work, we aim to construct a two-step MMS iteration method. The two-step splitting technique had been successfully used in other complementarity problems, e.g., see [43–45, 52], where the main idea is to change the iteration in the MMS to two iterations based on two matrix splittings of the system matrices, which can make full use of the information of the matrices for acceleration.

In the following, after presenting some required preliminaries, the new two-step MMS iteration method is established in Section 2. The convergence analysis of the proposed method is given in Section 3, which can generalize and improve the results in [42]. Some numerical examples are presented to illustrate the efficiency of the proposed method in Section 4, and concluding remarks of the whole work are presented in Section 5.

2. New method

First, some notations, definitions, and existing results needed in the following discussion are introduced.

Let $M = (m_{ij}) \in \mathbb{R}^{n\times n}$ and $M = D_M - B_M$, where D_M and $-B_M$ are the diagonal and the nondiagonal matrices of M, respectively. For two matrices M and N, the order in inequality $M \geq (>)N$ means $m_{ij} \geq (>)n_{ij}$ for every i, j. Let $|M| = (|m_{ij}|)$ and the comparison matrix of M be denoted by $\langle M\rangle = (\langle m_{ij}\rangle)$, where $\langle m_{ij}\rangle = |m_{ij}|$ if $i = j$ and $\langle m_{ij}\rangle = -|m_{ij}|$ if $i \neq j$. If $m_{ij} \leq 0$ for any $i \neq j$, M is called a Z-matrix. If M is a nonsingular Z-matrix and $M^{-1} \geq 0$, it is called a nonsingular M-matrix. M is called an H-matrix if $\langle M\rangle$ is a nonsingular M-matrix. If $|m_{ii}| > \sum\limits_{j\neq i}|m_{ij}|$ for all $1 \leq i \leq n$, M is called a strictly diagonal dominant (abbreviated as s.d.d.) matrix (see [5]). If M is an H-matrix with all its diagonal entries positive (e.g., see [1]), it is called an H_+-matrix. If M is a nonsingular M-matrix, it is well known that there exists a positive diagonal matrix Λ, which can make $M\Lambda$ be an s.d.d. matrix with all the diagonal entries of $A\Lambda$ positive [5]. $M = F - G$ is called an H-splitting if $\langle F\rangle - |G|$ is a nonsingular M-matrix [19].

Let $\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_s\}$ be a set of $n \times n$ real matrices (or $n \times 1$ real vectors). Denote a mapping φ as follows:

$$\varphi(\mathcal{Y}) = \sum_{i=1}^{s-1} 2^{s-i-1} Y_i + Y_s.$$

Let $\Omega \in \mathbb{R}^{n\times n}$ be a positive diagonal matrix, $\sigma > 0 \in \mathbb{R}$ and $M_i = F_i - G_i$ $(1 \le i \le s)$ be s splittings of M_i. Denote

$$\mathcal{F} = \{F_1, F_2, \ldots, F_s\}, \mathcal{G} = \{G_1, G_2, \ldots, G_s\}, \mathcal{M} = \{M_1, M_2, \ldots, M_s\},$$

$$\mathcal{D}_{\mathcal{F}} = \{D_{F_1}, D_{F_2}, \ldots, D_{F_s}\}, \mathcal{B}_{\mathcal{F}} = \{B_{F_1}, B_{F_2}, \ldots, B_{F_s}\},$$

and

$$U(z) = \{u_1(z), u_2(z), \ldots, u_s(z)\}.$$

By equivalently transforming the VNCP to a fixed-point equation, the MMS iteration method for the VNCP was given in [42].

Method 2.1. *[42] (**MMS**) Given $x_1^{(0)} \in \mathbb{R}^n$, for $k = 0, 1, 2, \ldots$, compute $x_1^{(k+1)} \in \mathbb{R}^n$ by*

$$(2^{s-1}\Omega + \varphi(\mathcal{F}))x_1^{(k+1)} = \varphi(\mathcal{G})x_1^{(k)} + (2^{s-1}\Omega - \varphi(\mathcal{M}))|x_1^{(k)}| + \Omega \sum_{i=2}^{s} 2^{s-i+1}|x_i^{(k)}| - \sigma\varphi(U(z^{(k)})),$$

where

$$z^{(k)} = \frac{1}{\sigma}(x_1^{(k)} + |x_1^{(k)}|)$$

and $x_2^{(k)}, \ldots, x_s^{(k)}$ are computed by

$$\begin{cases} x_s^{(k)} & = & \frac{1}{2}\Omega^{-1}[(M_{s-1} - M_s)(|x_1^{(k)}| + x_1^{(k)}) + \sigma u_{s-1}(z^{(k)}) - \sigma u_s(z^{(k)})], \\ x_j^{(k)} & = & \frac{1}{2}\Omega^{-1}[(M_{j-1} - M_j)(|x_1^{(k)}| + x_1^{(k)}) + \Omega(|x_{j+1}^{(k)}| + x_{j+1}^{(k)}) + \sigma u_{j-1}(z^{(k)}) - \sigma u_j(z^{(k)})], \\ & & j = s-1, s-2, \ldots, 2, \end{cases} \tag{2.1}$$

until the iteration is convergent.

Based on Method 2.1, to fully utilize the information of the entries of the matrix set $\mathcal{M}$, for $1 \le i \le s$, consider two matrix splittings of M_i, e.g., $M_i = F_i^{(1)} - G_i^{(1)} = F_i^{(2)} - G_i^{(2)}$. Then, the two-step MMS (abbreviated as TMMS) iteration method can be established as follows:

Method 2.2. *(**TMMS**) Given $x_1^{(0)} \in \mathbb{R}^n$, for $k = 0, 1, 2, \ldots$, compute $x_1^{(k+1)} \in \mathbb{R}^n$ by*

$$\begin{cases} (2^{s-1}\Omega + \varphi(\mathcal{F}^{(1)}))x_1^{(k+\frac{1}{2})} & = & \varphi(\mathcal{G}^{(1)})x_1^{(k)} + (2^{s-1}\Omega - \varphi(\mathcal{M}))|x_1^{(k)}| + \Omega \sum_{i=2}^{s} 2^{s-i+1}|x_i^{(k)}| \\ & & -\sigma\varphi(U(z^{(k)})), \\ (2^{s-1}\Omega + \varphi(\mathcal{F}^{(2)}))x_1^{(k+1)} & = & \varphi(\mathcal{G}^{(2)})x_1^{(k+\frac{1}{2})} + (2^{s-1}\Omega - \varphi(\mathcal{M}))|x_1^{(k+\frac{1}{2})}| + \Omega \sum_{i=2}^{s} 2^{s-i+1}|x_i^{(k+\frac{1}{2})}| \\ & & -\sigma\varphi(U(z^{(k+\frac{1}{2})})), \end{cases} \tag{2.2}$$

where

$$z^{(k)} = \frac{1}{\sigma}(x_1^{(k)} + |x_1^{(k)}|), z^{(k+\frac{1}{2})} = \frac{1}{\sigma}(x_1^{(k+\frac{1}{2})} + |x_1^{(k+\frac{1}{2})}|),$$

where $x_2^{(k)}, \ldots, x_s^{(k)}$ *are computed by (2.1) and*

$$\left\{ \begin{array}{lcl} x_s^{(k+\frac{1}{2})} & = & \frac{1}{2}\Omega^{-1}[(M_{s-1} - M_s)(|x_1^{(k+\frac{1}{2})}| + x_1^{(k+\frac{1}{2})}) + \sigma u_{s-1}(z^{(k+\frac{1}{2})}) - \sigma u_s(z^{(k+\frac{1}{2})})], \\ x_j^{(k+\frac{1}{2})} & = & \frac{1}{2}\Omega^{-1}[(M_{j-1} - M_j)(|x_1^{(k+\frac{1}{2})}| + x_1^{(k+\frac{1}{2})}) + \Omega(|x_{j+1}^{(k+\frac{1}{2})}| + x_{j+1}^{(k+\frac{1}{2})}) \\ & & +\sigma u_{j-1}(z^{(k)}) - \sigma u_j(z^{(k+\frac{1}{2})})], \quad j = s-1, s-2, \ldots, 2, \end{array} \right. \tag{2.3}$$

until the iteration is convergent.

Clearly, if we take $F_i^{(1)} = F_i^{(2)}$, Method 2.2 reduces to Method 2.1 immediately. Furthermore, we can obtain a class of relaxation methods from Method 2.2 by specially choosing the two matrix splittings, similar to those in [43–45, 49, 50, 52]. For example, for $i = 1, 2, \ldots, s$, taking

$$\left\{ \begin{array}{l} F_i^{(1)} = \frac{1}{\alpha}(D_{A_i}^{(1)} - \beta L_{A_i}^{(1)}), G_i^{(1)} = F_i^{(1)} - A_i, \\ F_i^{(2)} = \frac{1}{\alpha}(D_{A_i}^{(2)} - \beta U_{A_i}^{(2)}), G_i^{(2)} = F_i^{(2)} - A_i, \end{array} \right. \tag{2.4}$$

one can get the two-step modulus-based accelerated overrelaxation (abbreviated as TMAOR) iteration method, which can reduce to the two-step modulus-based successive overrelaxation (abbreviated as TMSOR) and Gauss-Seidel (abbreviated as TMGS) methods, when $(\alpha, \beta) = (\alpha, \alpha)$ and $(\alpha, \beta) = (1, 1)$, respectively.

3. Convergence analysis

In this section, the convergence conditions of Method 2.2 are given under the assumption that the VNCP has a unique solution z^*, the same as that in [42]. Furthermore, for $1 \le i \le s$, $u_i(z)$ is assumed to satisfy the locally Lipschitz smoothness conditions: Let

$$u_i(z) = (u_i(z_1), u_i(z_2), \ldots, u_i(z_n))^T$$

be differentiable with

$$0 \le \frac{\partial u_i(z)}{\partial z} \le U_i, \quad i = 1, 2, \ldots, s. \tag{3.1}$$

Then, by Lagrange mean value theorem, there exists ξ_j between $z_j^{(k)}$ and z_j^* such that

$$u_i(z^{(k)}) - u_i(z^*) = U_i^{(k)}(z^{(k)} - z^*) = \frac{1}{\sigma} U_i^{(k)}[(x_1^{(k)} - x_1^*) + (|x_1^{(k)}| - |x_1^*|)], \tag{3.2}$$

where $U_i^{(k)}$ is a nonnegative diagonal matrix with diagonal entries $\left.\frac{\partial u_i(z_j)}{\partial z_j}\right|_{z_j = \xi_j}$, $1 \le j \le n$. Furthermore, by (3.1), we have

$$0 \le U_i^{(k)} \le U_i. \tag{3.3}$$

Denote

$$\mathcal{U}^{(k)} = \{U_1^{(k)}, U_2^{(k)}, \ldots, U_s^{(k)}\}, \mathcal{U} = \{U_1, U_2, \ldots, U_s\}.$$

Lemma 3.1. *Let $M_i, 1 \le i \le s$, be H_+-matrices, $\Omega \in \mathbb{R}^{n\times n}$ be a positive diagonal matrix, and $\sigma > 0 \in \mathbb{R}$. For $t = 1, 2$, assume that:*
(I) $D_{F_i^{(t)}} > 0, i = 1, 2, \ldots, s-1$, and $M_s = F_s^{(t)} - G_s^{(t)}$ be an H-splitting of M_s;
(II)

$$\begin{cases} \langle F_{s-1}^{(t)} \rangle \ge \langle F_s^{(t)} \rangle, \\ 2^{s-j}\langle F_{j-1}^{(t)} \rangle \ge \langle \sum\limits_{i=j}^{s-1} 2^{s-i-1} F_i^{(t)} + F_s^{(t)} \rangle, \quad 2 \le j \le s-1; \end{cases} \tag{3.4}$$

(III) There exists a positive diagonal matrix Λ such that $(\langle F_s^{(t)} \rangle - |G_s^{(t)}|)\Lambda$ is an s.d.d. matrix.
Then, $2^{s-1}\Omega + \varphi(\mathcal{F}^{(t)})$ is an H-matrix.

Proof. Let $e =\in \mathbb{R}^n$ be a vector with all entries being 1. By reusing (3.4) $s-2$ times, we have

$$\begin{aligned} & \langle 2^{s-1}\Omega + \varphi(\mathcal{F}^{(t)}) \rangle \Lambda e \\ > \ & \langle \varphi(\mathcal{F}^{(t)}) \rangle \Lambda e \\ \ge \ & (\langle 2^{s-2} F_1^{(t)} \rangle + \langle \sum_{i=2}^{s-1} 2^{s-i-1} F_i + F_s^{(t)} \rangle)\Lambda e \\ \ge \ & 2\langle \sum_{i=2}^{s-1} 2^{s-i-1} F_i^{(t)} + F_s^{(t)} \rangle \Lambda e \\ \ge \ & 2(\langle 2^{s-3} F_2^{(t)} \rangle + \langle \sum_{i=3}^{s-1} 2^{s-i-1} F_i + F_s^{(t)} \rangle)\Lambda e \\ \ge \ & \ldots \ge 2^{s-1}\langle F_s^{(t)} \rangle \Lambda e \\ \ge \ & 2^{s-1}(\langle F_s^{(t)} \rangle - |G_s^{(t)}|)\Lambda e \\ > \ & 0. \end{aligned}$$

Hence, $\langle 2^{s-1}\Omega + \varphi(\mathcal{F}^{(t)}) \rangle \Lambda$ is an s.d.d matrix, which implies that $2^{s-1}\Omega + \varphi(\mathcal{F}^{(t)})$ is an H-matrix. □

Lemma 3.2. *With the same notations as those in Lemma 3.1, denote $x_i^*, i = 1, 2, \ldots, s$, as the solution of the VNCP, and let*

$$\delta_i^{(k)} = x_i^{(k)} - x_i^*, \bar{\delta}_i^{(k)} = |x_i^{(k)}| - |x_i^*|.$$

Then, we have

$$\sum_{i=2}^{s} 2^{s-i+1}\Omega|\bar{\delta}_i^{(k)}| \le [2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j + \Gamma_{s-1}) + \Theta]|\delta_1^{(k)}|, \tag{3.5}$$

where

$$\Gamma_j = \begin{cases} |M_{s-1} - M_s|, & \text{if} \quad j = 1, \\ |2^{j-1}M_{s-j} - \sum\limits_{t=s-j+1}^{s-1} 2^{s-t-1}M_t - M_s|, & \text{if} \quad 2 \le j \le s-1, \end{cases}$$

and

$$\Theta = \sum_{j=2}^{s-1}(2^s - 2^{s-j})U_j + (2^s - 2)U_s.$$

Proof. By (2.1), we get

$$\begin{cases} \delta_s^{(k)} &= \Omega^{-1}[(M_{s-1}-M_s)+(U_{s-1}^{(k)}-U_s^{(k)})](\bar{\delta}_1^{(k)}+\delta_1^{(k)}), \\ \delta_j^{(k)} &= \frac{1}{2}\Omega^{-1}[(M_{j-1}-M_j)+(U_{j-1}^{(k)}-U_j^{(k)})(\bar{\delta}_1^{(k)}+\delta_1^{(k)})+\Omega(\bar{\delta}_{j+1}^{(k)}+\delta_{j+1}^{(k)})], \\ & \quad j=s-1,s-2,\dots,2. \end{cases} \tag{3.6}$$

By the first equation of (3.6), we have

$$\begin{aligned} & 2^1\Omega|\delta_s^{(k)}| \\ = & |[(M_{s-1}-M_s)+(U_{s-1}^{(k)}-U_s^{(k)})](\bar{\delta}_1^{(k)}+\delta_1^{(k)})| \\ \le & (|M_{s-1}-M_s|+U_{s-1}+U_s)|\delta_1^{(k)}| \\ = & 2(\Gamma_1+U_{s-1}+U_s)|\delta_1^{(k)}|. \end{aligned}$$

Then, when the subscript is $s-1$, with the second equation of (3.6), we can get

$$\begin{aligned} & 2^2\Omega|\delta_{s-1}^{(k)}| \\ = & |[2(M_{s-2}-M_{s-1})+2(U_{s-2}^{(k)}-U_{s-1}^{(k)})](\bar{\delta}_1^{(k)}+\delta_1^{(k)})+2\Omega(\bar{\delta}_s^{(k)}+\delta_s^{(k)})| \\ = & |[(2M_{s-2}-M_{s-1}-M_s)+(2U_{s-2}^{(k)}-U_{s-1}^{(k)}-U_s^{(k)})](\bar{\delta}_1^{(k)}+\delta_1^{(k)})+2\Omega\bar{\delta}_s^{(k)}| \\ \le & 2(\Gamma_2+2U_{s-2}+U_{s-1}+U_s)|\delta_1^{(k)}|+2\Omega|\delta_s^{(k)}| \\ \le & 2[\Gamma_2+\Gamma_1+2(U_{s-2}+U_{s-1}+U_s)]|\delta_1^{(k)}|. \end{aligned}$$

By induction, for $2\le i\le s$, we have

$$2^{s-i+1}\Omega|\delta_i^{(k)}|\le(\sum_{j=1}^{s-i}2^{s-i-j+1}\Gamma_j+2\Gamma_{s-i+1}+2^{s-i+1}\sum_{j=i-1}^{s}U_j)|\delta_1^{(k)}|. \tag{3.7}$$

Then, by (3.7), we get

$$\begin{aligned} & \sum_{i=2}^{s}2^{s-i+1}\Omega|\bar{\delta}_i^{(k)}| \\ \le & \sum_{i=2}^{s}2^{s-i+1}\Omega|\delta_i^{(k)}| \\ \le & \sum_{i=2}^{s}(\sum_{j=1}^{s-i}2^{s-i-j+1}\Gamma_j+2\Gamma_{s-i+1}+2^{s-i+1}\sum_{j=i-1}^{s}U_j)|\delta_1^{(k)}| \\ = & (\sum_{j=1}^{s-2}\sum_{i=2}^{s-j}2^{s-i-j+1}\Gamma_j+2\sum_{i=1}^{s-1}\Gamma_i+\sum_{i=2}^{s}2^{s-i+1}\sum_{j=i-1}^{s}U_j)|\delta_1^{(k)}| \\ = & [\sum_{j=1}^{s-2}(2^{s-j}-2)\Gamma_j+2\sum_{j=1}^{s-1}\Gamma_j+\sum_{j=2}^{s-1}(2^s-2^{s-j})U_j+(2^s-2)U_s)]|\delta_1^{(k)}| \\ = & [2(\sum_{j=1}^{s-2}2^{s-j-1}\Gamma_j+\Gamma_{s-1})+\Theta]|\delta_1^{(k)}|. \end{aligned}$$

□

Theorem 3.1. *With the same notations and assumptions as those in Lemmas 3.1 and 3.2, for $t = 1, 2$, assume that*

$$\begin{cases} |G_{s-1}^{(t)}| \le |G_s^{(t)}|, \\ 2^{s-j}|G_j^{(t)}| \le |\sum_{i=j}^{s-1} 2^{s-i-1} G_i^{(t)} + G_s^{(t)}|, \quad 2 \le j \le s-1, \end{cases} \tag{3.8}$$

Then, Method 2.1 converges for any initial vector $x_1^{(0)}$ provided that

$$\Omega \ge 2^{1-s}(\varphi(\mathcal{D}_{\mathcal{M}}{}^{(t)}) + \varphi(\mathcal{U}) + \Theta) \tag{3.9}$$

or

$$[2^{1-s}(\varphi(\mathcal{D}_{\mathcal{M}}{}^{(t)}) + \varphi(\mathcal{U})) + 2^{-s}\Theta - (\langle F_s^{(t)}\rangle - |G_s^{(t)}|)]\Lambda e < \Omega\Lambda e < 2^{1-s}(\varphi(\mathcal{D}_{\mathcal{M}}{}^{(t)}) + \Theta)\Lambda e. \tag{3.10}$$

Proof. By (3.2), we get

$$u_i(z^{(k)}) - u_i(z^*) = \frac{1}{\sigma} U_i^{(k)}(\delta_1^{(k)} + \bar{\delta}_1^{(k)}).$$

By the definition of φ, we have

$$\varphi(U(z^{(k)})) - \varphi(U(z^*)) = \varphi(U(z^{(k)}) - U(z^*)) = \frac{1}{\sigma}\varphi(\mathcal{U}^{(k)})(\delta_1^{(k)} + \bar{\delta}_1^{(k)}). \tag{3.11}$$

Combining the first equation of (2.2) and (3.11), we can get

$$\begin{aligned} & (2^{s-1}\Omega + \varphi(\mathcal{F}^{(1)}))\delta_1^{(k+\frac{1}{2})} \\ = \ & \varphi(\mathcal{G}^{(1)})\delta_1^{(k)} + (2^{s-1}\Omega - \varphi(\mathcal{M}))\bar{\delta}_1^{(k)} + \Omega\sum_{i=2}^{s} 2^{s-i+1}\bar{\delta}_i^{(k)} - \sigma[\varphi(U(z^{(k)})) - \varphi(U(z^*))] \\ = \ & \varphi(\mathcal{G}^{(1)})\delta_1^{(k)} + (2^{s-1}\Omega - \varphi(\mathcal{M}))\bar{\delta}_1^{(k)} + \Omega\sum_{i=2}^{s} 2^{s-i+1}\bar{\delta}_i^{(k)} - \varphi(\mathcal{U}^{(k)})(\delta_i^{(k)} + \bar{\delta}_i^{(k)}) \\ = \ & [\varphi(\mathcal{G}^{(1)}) - \varphi(\mathcal{U}^{(k)})]\delta_1^{(k)} + [2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k)})]\bar{\delta}_1^{(k)} + \Omega\sum_{i=2}^{s} 2^{s-i+1}\bar{\delta}_i^{(k)}. \end{aligned}$$

Similarly, by the second equation of (2.2), we have

$$\begin{aligned} & (2^{s-1}\Omega + \varphi(\mathcal{F}^{(2)}))\delta_1^{(k+1)} \\ = \ & [\varphi(\mathcal{G}^{(2)}) - \varphi(\mathcal{U}^{(k+\frac{1}{2})})]\delta_1^{(k+\frac{1}{2})} + [2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k+\frac{1}{2})})]\bar{\delta}_1^{(k+\frac{1}{2})} + \Omega\sum_{i=2}^{s} 2^{s-i+1}\bar{\delta}_i^{(k+\frac{1}{2})}. \end{aligned}$$

Then we have

$$\begin{cases} |\delta_1^{(k+\frac{1}{2})}| & \le & |2^{s-1}\Omega + \varphi(\mathcal{F}^{(1)})|^{-1}\Big(|\varphi(\mathcal{G}^{(1)}) - \varphi(\mathcal{U}^{(k)})||\delta_1^{(k)}| \\ & & +|2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k)})||\bar{\delta}_1^{(k)}| + \sum_{i=2}^{s} 2^{s-i+1}\Omega|\bar{\delta}_i^{(k)}|\Big), \\ |\delta_1^{(k+1)}| & \le & |2^{s-1}\Omega + \varphi(\mathcal{F}^{(2)})|^{-1}\Big(|\varphi(\mathcal{G}^{(2)}) - \varphi(\mathcal{U}^{(k+\frac{1}{2})})||\delta_1^{(k+\frac{1}{2})}| \\ & & +|2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k+\frac{1}{2})})||\bar{\delta}_1^{(k+\frac{1}{2})}| + \sum_{i=2}^{s} 2^{s-i+1}\Omega|\bar{\delta}_i^{(k+\frac{1}{2})}|\Big). \end{cases} \tag{3.12}$$

By Lemma 3.1, we have that $2^{s-1}\Omega + \varphi(\mathcal{F})$ is an H-matrix. By [17], with (3.5) and (3.12), we get

$$|\delta_1^{(k+1)}| \leq \mathcal{P}^{(2)^{-1}}Q^{(2)}\mathcal{P}^{(1)^{-1}}Q^{(1)}|\delta_1^{(k)}|, \tag{3.13}$$

where

$$\mathcal{P}^{(1)} = \langle 2^{s-1}\Omega + \varphi(\mathcal{F}^{(1)})\rangle,$$
$$Q^{(1)} = |\varphi(\mathcal{G}^{(1)}) - \varphi(\mathcal{U}^{(k)})| + |2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k)})| + 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j + \Gamma_{s-1}) + \Theta,$$
$$\mathcal{P}^{(2)} = \langle 2^{s-1}\Omega + \varphi(\mathcal{F}^{(2)})\rangle,$$
$$Q^{(2)} = |\varphi(\mathcal{G}^{(2)}) - \varphi(\mathcal{U}^{(k)})| + |2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k)})| + 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j + \Gamma_{s-1}) + \Theta.$$

First, we estimate $\mathcal{F}^{(1)^{-1}}\mathcal{G}^{(1)}$. By Lemma 3.1 and [26], we have

$$\|\Lambda^{-1}\mathcal{F}^{(1)^{-1}}\mathcal{G}^{(1)}\Lambda\|_\infty = \|(\mathcal{F}^{(1)}\Lambda)^{-1}(\mathcal{G}^{(1)}\Lambda)\|_\infty \leq \max_{1\leq i\leq n} \frac{(\mathcal{G}^{(1)}\Lambda e)_i}{(\mathcal{F}^{(1)}\Lambda e)_i}. \tag{3.14}$$

Let

$$\Gamma_j^{(1)} = \begin{cases} |D_{F_{s-1}^{(1)}} - D_{F_s^{(1)}}|, & \text{if} \quad j = 1, \\ |2^{j-1}D_{F_{s-j}^{(1)}} - \sum\limits_{t=s-j+1}^{s-1} 2^{s-t-1}D_{F_t^{(1)}} - D_{F_s^{(1)}}|, & \text{if} \quad 2 \leq j \leq s-1, \end{cases}$$

$$\Gamma_j^{(2)} = \begin{cases} |B_{F_{s-1}^{(1)}} - B_{F_s^{(1)}}|, & \text{if} \quad j = 1, \\ |2^{j-1}B_{F_{s-j}^{(1)}} - \sum\limits_{t=s-j+1}^{s-1} 2^{s-t-1}B_{F_t^{(1)}} - B_{F_s^{(1)}}|, & \text{if} \quad 2 \leq j \leq s-1, \end{cases}$$

and

$$\Gamma_j^{(3)} = \begin{cases} |G_{s-1}^{(1)} - G_s^{(1)}|, & \text{if} \quad j = 1, \\ |2^{j-1}G_{s-j}^{(1)} - \sum\limits_{t=s-j+1}^{s-1} 2^{s-t-1}G_t^{(1)} - G_s^{(1)}|, & \text{if} \quad 2 \leq j \leq s-1. \end{cases}$$

Clearly, we have $\Gamma_j \leq \Gamma_j^{(1)} + \Gamma_j^{(2)} + \Gamma_j^{(3)}$. By (3.4), we can get

$$\Gamma_j^{(1)} = \begin{cases} D_{F_{s-1}} - D_{F_s}, & \text{if} \quad j = 1, \\ 2^{j-1}D_{F_{s-j}} - \sum\limits_{t=s-j+1}^{s-1} 2^{s-t-1}D_{F_t} - D_{F_s}, & \text{if} \quad 2 \leq j \leq s-1. \end{cases}$$

Then, by direct computation, we can obtain

$$\begin{aligned} \varphi(\mathcal{D}_M^{(1)}) - 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(1)} + \Gamma_{s-1}^{(1)}) &= \sum_{i=1}^{s-1} 2^{s-i-1}D_{F_i^{(1)}} + D_{F_s^{(1)}} - 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(1)} + \Gamma_{s-1}^{(1)}) \\ &= (2^s - 1)D_{F_s^{(1)}} - \sum_{i=1}^{s-1} 2^{s-i-1}D_{F_i^{(1)}}. \end{aligned} \tag{3.15}$$

Moreover, by reusing (3.4) $s-2$ times, we get

$$\begin{aligned}
& 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(2)} + \Gamma_{s-1}^{(2)}) + 2|\sum_{j=1}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} + B_{F_s^{(1)}}| \\
= \; & 2\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(2)} + \Big(2|2^{s-2}B_{F_1^{(1)}} - \sum_{j=2}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} - B_{F_s^{(1)}}| \\
& +2|2^{s-2}B_{F_1^{(1)}} + \sum_{j=2}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} + B_{F_s^{(1)}}|\Big) \\
= \; & 2\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(2)} + 2^2|\sum_{j=2}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} + B_{F_s^{(1)}}| \\
= \; & 2\sum_{j=1}^{s-3} 2^{s-j-1}\Gamma_j^{(2)} + \Big(2^2|2^{s-3}B_{F_2^{(1)}} - \sum_{j=3}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} - B_{F_s^{(1)}}| \\
& +2^2|2^{s-3}B_{F_2} + \sum_{j=3}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} + B_{F_s^{(1)}}|\Big) \\
= \; & 2\sum_{j=1}^{s-3} 2^{s-j-1}\Gamma_j^{(2)} + 2^3|\sum_{j=3}^{s-1} 2^{s-j-1}B_{F_j^{(1)}} + B_{F_s^{(1)}}| \\
= \; & \ldots = 2^s|B_{F_s}|. \qquad (3.16)
\end{aligned}$$

Analogously, by reusing (3.8) $s-2$ times, we get

$$2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(3)} + \Gamma_{s-1}^{(3)}) + 2|\varphi(\mathcal{G}^{(1)})| = 2^s|G_s^{(1)}|. \qquad (3.17)$$

By (3.15)–(3.17), we get

$$\begin{aligned}
& \mathcal{P}^{(1)}\Lambda e - \mathcal{Q}^{(1)}\Lambda e \\
= \; & \Big[\langle 2^{s-1}\Omega + \varphi(\mathcal{F}^{(1)})\rangle - |\varphi(\mathcal{G}^{(1)}) - \varphi(\mathcal{U}^{(k)})| - |2^{s-1}\Omega - \varphi(\mathcal{M}) - \varphi(\mathcal{U}^{(k)})| \\
& -2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j + \Gamma_{s-1}) - \Theta\Big]\Lambda e \\
\geq \; & \Big[2^{s-1}\Omega + \varphi(\mathcal{D_M}^{(1)}) - |\varphi(\mathcal{B_M}^{(1)})| - |\varphi(\mathcal{G}^{(1)}) - \varphi(\mathcal{U}^{(k)})| - |2^{s-1}\Omega - \varphi(\mathcal{D_M}) - \varphi(\mathcal{U}^{(k)})| \\
& -|\varphi(\mathcal{B_M}^{(1)})| - |\varphi(\mathcal{G}^{(1)})| - 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(1)} + \Gamma_{s-1}^{(1)}) - 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(2)} + \Gamma_{s-1}^{(2)}) \\
& -2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(3)} + \Gamma_{s-1}^{(3)}) - \Theta\Big]\Lambda e \\
\geq \; & \Big\{2^{s-1}\Omega + \Big[\varphi(\mathcal{D_M}^{(1)}) - 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(1)} + \Gamma_{s-1}^{(1)})\Big] - \Big[2|\varphi(\mathcal{B_M}^{(1)})| + 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(2)} + \Gamma_{s-1}^{(2)})\Big]
\end{aligned}$$

$$
\begin{aligned}
&-\Big[2|\varphi(\mathcal{G})| + 2(\sum_{j=1}^{s-2} 2^{s-j-1}\Gamma_j^{(3)} + \Gamma_{s-1}^{(3)})\Big] - \varphi(\mathcal{U}^{(k)}) - |2^{s-1}\Omega - \varphi(\mathcal{D_M}^{(1)}) - \varphi(\mathcal{U}^{(k)}) - \Theta|\Big\}\Lambda e \\
= \;& \Big[2^{s-1}\Omega + (2^s-1)D_{F_s^{(1)}} - \sum_{i=1}^{s-1} 2^{s-i-1}D_{F_i^{(1)}} - 2^s|B_{F_s^{(1)}}| - 2^s|G_s^{(1)}| - \varphi(\mathcal{U}^{(k)}) \\
&-|2^{s-1}\Omega - \varphi(\mathcal{D_M}^{(1)}) - \varphi(\mathcal{U}^{(k)}) - \Theta|\Big]\Lambda e.
\end{aligned}
\tag{3.18}
$$

Next, consider the following two cases.
Case 1. When

$$\Omega \geq 2^{1-s}(\varphi(\mathcal{D_M}^{(1)}) + \varphi(\mathcal{U}) + \Theta),$$

by (3.18), we have

$$
\begin{aligned}
&\mathcal{P}^{(1)}\Lambda e - Q^{(1)}\Lambda e \\
\geq \;& \Big[2^{s-1}\Omega + (2^s-1)D_{F_s^{(1)}} - \sum_{i=1}^{s-1} 2^{s-i-1}D_{F_i^{(1)}} - 2^s|B_{F_s^{(1)}}| - 2^s|G_s^{(1)}| - \varphi(\mathcal{U}^{(k)}) \\
&-(2^{s-1}\Omega - \varphi(\mathcal{D_M}) - \varphi(\mathcal{U}^{(k)}) - \Theta)\Big]\Lambda e \\
= \;& [2^s(D_{F_s^{(1)}} - |G_s^{(1)}| - |B_{F_s^{(1)}}|) + \Theta]\Lambda e \\
= \;& [2^s(\langle F_s\rangle - |G_s^{(1)}|) + \Theta]\Lambda e \\
> \;& 0.
\end{aligned}
$$

Case 2. When

$$[2^{1-s}(\varphi(\mathcal{D_M}^{(1)}) + \varphi(\mathcal{U})) + 2^{-s}\Theta - (\langle F_s^{(1)}\rangle - |G_s^{(1)}|)]\Lambda e < \Omega\Lambda e < 2^{1-s}(\varphi(\mathcal{D_M}^{(1)}) + \Theta)\Lambda e,$$

by (3.18), we have

$$
\begin{aligned}
&\mathcal{P}^{(1)}\Lambda e - Q^{(1)}\Lambda e \\
\geq \;& \Big[2^{s-1}\Omega + (2^s-1)D_{F_s^{(1)}} - \sum_{i=1}^{s-1} 2^{s-i-1}D_{F_i^{(1)}} - 2^s|B_{F_s^{(1)}}| - 2^s|G_s^{(1)}| - \varphi(\mathcal{U}^{(k)}) \\
&-(\varphi(\mathcal{D_M}^{(1)}) + \varphi(\mathcal{U}^{(k)}) + \Theta - 2^{s-1}\Omega)\Big]\Lambda e \\
\geq \;& \Big[2^s\Omega - 2\varphi(\mathcal{D_M}^{(1)}) - 2\varphi(\mathcal{U}) - \Theta + 2^s(D_{F_s} - |G_s^{(1)}| - |B_{F_s^{(1)}}|)\Big]\Lambda e \\
= \;& \Big[2^s\Omega - 2\varphi(\mathcal{D_M}^{(1)}) - 2\varphi(\mathcal{U}) - \Theta + 2^s(\langle F_s^{(1)}\rangle - |G_s^{(1)}|)\Big]\Lambda e \\
> \;& 0.
\end{aligned}
$$

Summarizing Cases 1 and 2, we immediately get $\mathcal{P}^{(1)}\Lambda e - Q^{(1)}\Lambda e > 0$, provided that (3.9) or (3.10) holds, which implies that

$$\|\Lambda^{-1}\mathcal{P}^{(1)^{-1}}Q^{(1)}\Lambda\|_\infty < 1$$

by (3.14).

By similar deductions, we can also get

$$\|\Lambda^{-1}\mathcal{P}^{(2)^{-1}}\mathcal{Q}^{(2)}\Lambda\|_\infty < 1,$$

when (3.9) or (3.10) is satisfied.

In summary, the spectral radius of the iteration matrix given by (3.13) can be estimated as follows:

$$\begin{aligned} & \rho(\mathcal{P}^{(2)^{-1}}\mathcal{Q}^{(2)}\mathcal{F}^{(1)^{-1}}\mathcal{Q}^{(1)}) \\ = & \ \rho(\Lambda^{-1}\mathcal{P}^{(2)^{-1}}\mathcal{Q}^{(2)}\Lambda\Lambda^{-1}\mathcal{P}^{(1)^{-1}}\mathcal{Q}^{(1)}\Lambda) \\ \le & \ \|\Lambda^{-1}\mathcal{P}^{(2)^{-1}}\mathcal{Q}^{(2)}\Lambda\|_\infty\|\Lambda^{-1}\mathcal{P}^{(1)^{-1}}\mathcal{Q}^{(1)}\Lambda\|_\infty \\ < & \ 1. \end{aligned}$$

Hence, we can see that $\{x_1^{(k)}\}_{k=0}^{\infty}$ converges by (3.13). □

Based on Theorem 3.1, we have the next theorem for the TMAOR method.

Theorem 3.2. *With the same notations and assumptions as those in Theorem 3.1, for* $t = 1, 2$, *let* $F_i^{(t)}$ *and* $G_i^{(t)}$ *be given by (2.4),* $1 \le i \le s$. *Assume that*

$$0 < \beta \le \alpha < \frac{2}{1 + \rho(D_{M_s}^{-1}|B_{M_s}|)}. \tag{3.19}$$

Then, the TMAOR converges to the solution of the VNCP.

Proof. By (2.4), with direct computation, we can get

$$\langle F_s^{(1)}\rangle - |G_s^{(1)}| = \langle F_s^{(2)}\rangle - |G_s^{(2)}| = \frac{1 - |1 - \alpha|}{\alpha} D_{M_s} - |B_{M_s}|,$$

if $0 < \beta \le \alpha$. Since M_s is an H_+-matrix, by [5] we have $\rho(D_{M_s}^{-1}|B_{M_s}|) < 1$. Since (3.19) holds, we can easily have that $\frac{1-|1-\alpha|}{\alpha} D_{M_s} - |B_{M_s}|$ is a nonsingular M-matrix. Note that all the assumptions of Theorem 3.1 hold. Hence, the TMAOR is convergent by Theorem 3.1. □

4. Numerical examples

Next, some numerical examples are given to shown the efficiency of Method 2.2 compared to Method 2.1.

Example 4.1. *[42] Consider a VNCP* ($s = 2$), *where the two system matrices are given by*

$$M_1 = \begin{pmatrix} R & -I_m & & & \\ -I_m & R & -I_m & & \\ & \ddots & \ddots & \ddots & \\ & & -I_m & R & -I_m \\ & & & -I_m & R \end{pmatrix} \in \mathbb{R}^{n\times n} \ \textit{and} \ M_2 = \begin{pmatrix} R & & & & \\ & R & & & \\ & & \ddots & & \\ & & & R & \\ & & & & R \end{pmatrix} \in \mathbb{R}^{n\times n},$$

where

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

Example 4.2. *[42] Consider a VNCP ($s = 2$), where the two system matrices are given by*

$$M_1 = \begin{pmatrix} T & -0.5I_m & & & \\ -1.5I_m & T & -0.5I_m & & \\ & \ddots & \ddots & \ddots & \\ & & -1.5I_m & T & -0.5I_m \\ & & & -1.5I_m & T \end{pmatrix} \in \mathbb{R}^{n\times n} \text{ and } M_2 = \begin{pmatrix} T & & & & \\ & T & & & \\ & & \ddots & & \\ & & & T & \\ & & & & T \end{pmatrix} \in \mathbb{R}^{n\times n},$$

where

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

Example 4.3. *Consider the VNCP ($s = 2$) whose system matrices $M_1, M_2 \in \mathbb{R}^{m^2\times m^2}$ come from the discretization of Hamilton-Jacobi-Bellman (HJB) equation [4]*

$$\begin{cases} \max\limits_{i=1,2}\{L_i + f_i\} = 0 & in \quad \Gamma, \\ z = 0 & on \quad \partial\Gamma, \end{cases}$$

with $\Gamma = \{(x, y) \mid 0 < x < 2, 0 < y < 1\}$,

$$\begin{cases} L_1 = 0.002z_{xx} + 0.001z_{yy} - 20u_1(z), f_1 = 1, \\ L_2 = 0.001z_{xx} + 0.001z_{yy} - 10u_2(z), f_2 = 1. \end{cases}$$

Same as those in [42], all the nonlinear functions in Examples 4.1–4.3 are set to $u_1(z) = \frac{z}{1+z}$ and $u_2(z) = arctan(z)$.

The programming language is MATLAB, whose codes are run on a PC with a 12th Gen Intel(R) Core(TM) i7-12700 2.10 GHz and 32G memory. Consider the Gauss-Seidel (abbreviated as GS), SOR and AOR splittings, where the SOR splitting is

$$F_s^{(t)} = \frac{1}{\alpha}(D_{M_s^{(t)}} - \alpha L_{M_s^{(t)}}), G_s^{(t)} = F_s^{(t)} - M_s^{(t)}, \quad t = 1, 2,$$

with α being the relaxation parameter and starting from 0.8 to 1.2 with a step size of 0.1, while the parameters of AOR splitting given by (2.4) satisfy $\alpha = \beta + 0.1$, and β starts from 0.8 to 1.2 with a step size of 0.1.

All initial vectors are set to $x_1^{(0)} = e$. The stopping criterion in the iteration of both Methods 2.1 and 2.2 is taken as

$$\| \min\{z^{(k)}, M_1z^{(k)} + u_1(z^{(k)}), M_2z^{(k)} + u_2(z^{(k)})\}\|_2 < 10^{-10},$$

and Ω is chosen as

$$\Omega = \frac{1}{2\alpha}(D_{M_1} + D_{M_2}) + I.$$

Let $m = 256, 512, 1024$ for the three examples. For fair comparison, when the relaxation parameters are chosen as the experimentally optimal ones for each size of the three examples, the results of the

MMS and TMMS methods are shown in Tables 1–3, where "IT" is the iteration steps, "CPU" is the elapsed CPU time, and

$$SAVE = \frac{CPU_{MMS} - CPU_{TMMS}}{CPU_{MMS}} \times 100\%.$$

It is found from the numerical results that Method 2.2 always converges faster than Method 2.1 for all cases. Specifically, the iteration steps of Method 2.1 take almost twice as much time as those of Method 2.2 due to the fact that there are two equations needed to solve in each iteration in Method 2.2 versus only one in Method 2.1. Focusing on the CPU time, the cost of Method 2.2 is 14%–23% less than that of Method 2.1 for Example 4.1, while the percentages of Examples 4.2 and 4.3 are 21%–38% and 7%–18%, respectively. It is clear that the two-step splitting technique works for accelerating the convergence of the MMS iteration method.

Table 1. Numerical results of Example 4.1.

m	splitting	Method 2.1		Method 2.2		
		IT	CPU	IT	CPU	SAVE
256	GS	98	0.6945	50	0.5976	14%
	SOR	94	0.6679	46	0.5652	15%
	AOR	97	0.7469	47	0.5902	21%
512	GS	100	3.7686	51	3.1709	16%
	SOR	94	3.6039	46	3.0237	16%
	AOR	101	4.0266	47	3.1373	22%
1024	GS	102	22.7605	52	18.9571	17%
	SOR	134	21.1283	47	17.7199	16%
	AOR	97	24.4213	47	18.8083	23%

Table 2. Numerical results of Example 4.2.

m	splitting	Method 2.1		Method 2.2		
		IT	CPU	IT	CPU	SAVE
256	GS	86	0.5939	34	0.3792	36%
	SOR	69	0.4937	31	0.3571	27%
	AOR	71	0.5252	32	0.4136	21%
512	GS	89	3.4883	35	2.1793	38%
	SOR	70	2.6886	31	1.9244	28%
	AOR	71	4.5891	32	3.3323	27%
1024	GS	91	19.1887	36	12.4029	35%
	SOR	70	15.3799	32	11.5026	25%
	AOR	72	17.7342	33	13.3755	24%

Table 3. Numerical results of Example 4.3.

m	splitting	Method 2.1		Method 2.2		
		IT	CPU	IT	CPU	SAVE
256	GS	76	0.6216	39	0.5478	12%
	SOR	59	0.4853	30	0.4394	9%
	AOR	54	0.4732	28	0.4103	13%
512	GS	288	17.4591	145	14.2667	18%
	SOR	253	12.6084	101	11.6480	7%
	AOR	204	14.9972	103	12.6497	16%
1024	GS	1137	172.0573	569	144.5840	16%
	SOR	885	137.8101	443	116.6376	15%
	AOR	805	130.3913	403	111.1060	15%

5. Conclusions

By the two-step matrix splitting technique, we have constructed the TMMS iteration method for solving the VNCP. We also present the convergence theorems of the TMMS iteration method for an arbitrary s, which can extend the research scope of the modulus-based methods for the VNCP.

Note that to show the effectiveness of the proposed two-step method, the choice of the two-step splittings in Section 4 is common in existing literatures. However, since the iteration matrix in (3.13) is essentially an extended bound containing absolute operation, it is very hard to minimize its spectral radius. How to choose an optimal two-step splitting in general or based some special structure is still an open question. On the other hand, it is noted that there were some other accelerated techniques for the MMS iteration method, such as relaxation, precondition, two-sweep and so on. One can also mix a certain technique with the two-step splittings for acceleration, e.g., [3, 22, 30, 47]. More techniques to improve the convergence of the MMS iteration method are worth studying in the future.

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 Basic and Applied Basic Research Foundation of Guangdong (No. 2024A1515011822), Scientific Computing Research Innovation Team of Guangdong Province (No. 2021KCXTD052), Science and Technology Development Fund, Macau SAR (No. 0096/2022/A), Guangdong Key Construction Discipline Research Capacity Enhancement Project (No. 2022ZDJS049) and Technology Planning Project of Shaoguan (No. 230330108034184).

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Z. Bai, On the convergence of the multisplitting methods for the linear complementarity problem, *SIAM J. Matrix Anal. Appl.*, **21** (1999), 67–78. http://dx.doi.org/10.1137/S0895479897324032
2. Z. Bai, Modulus-based matrix splitting iteration methods for linear complementarity problems, *Numer. Linear Algebr.*, **17** (2010), 917–933. http://dx.doi.org/10.1002/nla.680
3. M. Bashirizadeh, M. Hajarian, Two-step two-sweep modulus-based matrix splitting iteration method for linear complementarity problems, *Numer. Math. Theor. Meth. Appl.*, **15** (2022), 592–619. http://dx.doi.org/10.4208/nmtma.OA-2021-0131
4. A. Bensoussan, J. Lions, *Applications of variational inequalities in stochastic control*, Amsterdam: Elsevier, 1982.
5. A. Berman, R. Plemmons, *Nonnegative matrix in the mathematical sciences*, Philadelphia: SIAM Publisher, 1994. http://dx.doi.org/10.1137/1.9781611971262
6. Y. Cao, G. Yang, Q. Shen, Convergence analysis of projected SOR iteration method for a class of vertical linear complementarity problems, *Comp. Appl. Math.*, **42** (2023), 191. http://dx.doi.org/10.1007/s40314-023-02334-6
7. Y. Cao, A. Wang, Two-step modulus-based matrix splitting iteration methods for implicit complementarity problems, *Numer. Algor.*, **82** (2019), 1377–1394. http://dx.doi.org/10.1007/s11075-019-00660-7
8. B. Chen, P. Harker, Smooth approximations to nonlinear complementarity problems, *SIAM J. Optim.*, **7** (1997), 403–420. http://dx.doi.org/10.1137/S1052623495280615
9. R. Cottle, G. Dantzig, A generalization of the linear complementarity problem, *J. Comb. Theory*, **8** (1970), 79–90. http://dx.doi.org/10.1016/S0021-9800(70)80010-2
10. R. Cottle, J. Pang, R. Stone, *The linear complementarity problem*, SanDiego: SIAM Publisher, 1992. http://dx.doi.org/10.1137/1.9780898719000
11. A. Ebiefung, M. Kostreva, The generalized Leontief input-output model and its application to the choice of the new technology, *Ann. Oper. Res.*, **44** (1993), 161–172. http://dx.doi.org/10.1007/BF02061065
12. X. Fang, General fixed-point method for solving the linear complementarity problem, *AIMS Mathematics*, **6** (2021), 11904–11920. http://dx.doi.org/10.3934/math.2021691
13. X. Fang, The convergence of the modulus-based Jacobi (MJ) iteration method for solving horizontal linear complementarity problems, *Comp. Appl. Math.*, **41** (2022), 134. http://dx.doi.org/10.1007/s40314-022-01842-1
14. X. Fang, The convergence of modulus-based matrix splitting iteration methods for implicit complementarity problems, *J. Comput. Appl. Math.*, **411** (2022), 114241. http://dx.doi.org/10.1016/j.cam.2022.114241
15. X. Fang, Z. Gu, Z. Qiao, Convergence of the two-point modulus-based matrix splitting iteration method, *J. Appl. Anal. Comput.*, **13** (2023), 2504–2521. http://dx.doi.org/10.11948/20220400

16. X. Fang, Z. Zhu, The modulus-based matrix double splitting iteration method for linear complementarity problems, *Comput. Math. Appl.*, **78** (2019), 3633–3643. http://dx.doi.org/10.1016/j.camwa.2019.06.012

17. A. Frommer, G. Mayer, Convergence of relaxed parallel multisplitting methods, *Linear Algebra Appl.*, **119** (1989), 141–152. http://dx.doi.org/10.1016/0024-3795(89)90074-8

18. T. Fujisawa, E. Kuh, Piecewise-linear theory of nonlinear networks, *SIAM J. Appl. Math.*, **22** (1972), 307–328. http://dx.doi.org/10.1137/0122030

19. A. Frommer, D. Szyld, H-splittings and two-stage iterative methods, *Numer. Math.*, **63** (1992), 345–356. http://dx.doi.org/10.1007/BF01385865

20. M. Seetharama Gowda, R. Sznajder, The generalized order linear complementarity problem, *SIAM J. Matrix Anal. Appl.*, **15** (1994), 779–795. http://dx.doi.org/10.1137/S0895479892237859

21. W. Guo, H. Zheng, X. Peng, New convergence results of the modulus-based methods for vertical linear complementarity problems, *Appl. Math. Lett.*, **135** (2023), 108444. http://dx.doi.org/10.1016/j.aml.2022.108444

22. W. Guo, H. Zheng, X. Lu, Y. Zhang, S. Vong, A relaxation two-step parallel modulus method without auxiliary variable for solving large sparse vertical linear complementarity problems, *Numer. Algor.*, in press. http://dx.doi.org/10.1007/s11075-024-01800-4

23. J. He, S. Vong, A new kind of modulus-based matrix splitting methods for vertical linear complementarity problems, *Appl. Math. Lett.*, **134** (2022), 108344. http://dx.doi.org/10.1016/j.aml.2022.108344

24. M. Haddou, M. Tangi, O. J*é*r*é*my, A generalized direction in interior point method for monotone linear complementarity problems, *Optim. Lett.*, **13** (2019), 35–53. http://dx.doi.org/10.1007/s11590-018-1241-2

25. J. Hong, C. Li, Modulus-based matrix splitting iteration methods for a class of implicit complementarity problems, *Numer. Linear Algebr.*, **23** (2016), 629–641. http://dx.doi.org/10.1002/nla.2044

26. J. Hu, Estimates of $\|B^{-1}C\|_\infty$ and their applications (Chinese), *Mathematica Numerica Sinica*, **3** (1982), 272–282.

27. Y. Ke, C. Ma, H. Zhang, The modulus-based matrix splitting iteration methods for second-order cone linear complementarity problems, *Numer. Algor.*, **79** (2018), 1283–1303. http://dx.doi.org/10.1007/s11075-018-0484-4

28. Y. Ke, C. Ma, H. Zhang, The relaxation modulus-based matrix splitting iteration methods for circular cone nonlinear complementarity problems, *Comp. Appl. Math.*, **37** (2018), 6795–6820. http://dx.doi.org/10.1007/s40314-018-0687-2

29. M. Kojima, S. Susumu, H. Shinji, Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices, *SIAM J. Optim.*, **7** (1997), 86–125. http://dx.doi.org/10.1137/S1052623494269035

30. D. Li, L. Wang, Y. Liu, A relaxation general two-sweep modulus-based matrix splitting iteration method for solving linear complementarity problems, *J. Comput. Appl. Math.*, **409** (2022), 114140. http://dx.doi.org/10.1016/j.cam.2022.114140

31. F. Mezzadri, A modulus-based formulation for the vertical linear complementarity problems, *Numer. Algor.*, **90** (2022), 1547–1568. http://dx.doi.org/10.1007/s11075-021-01240-4
32. F. Mezzadri, E. Galligani, Modulus-based matrix splitting methods for horizontal linear complementarity problems, *Numer. Algor.*, **83** (2020), 201–219. http://dx.doi.org/10.1007/s11075-019-00677-y
33. F. Mezzadri, E. Galligani, Projected splitting methods for vertical linear complementarity problems, *J. Optim. Theory Appl.*, **193** (2022), 598–620. http://dx.doi.org/10.1007/s10957-021-01922-y
34. T. Nagae, T. Akamatsu, A generalized complementarity approach to solving real option problems, *J. Econ. Dyn. Control.*, **32** (2008), 1754–1779. http://dx.doi.org/10.1016/j.jedc.2007.04.010
35. K. Oh, The formulation of the mixed lubrication problem as a generalized nonlinear complementarity problem, *J. Tribol.*, **108** (1986), 598–603. http://dx.doi.org/10.1115/1.3261274
36. F. Potra, Y. Ye, Interior-point methods for nonlinear complementarity problems, *J. Optim. Theory Appl.*, **88** (1996), 617–642. http://dx.doi.org/10.1007/BF02192201
37. H. Qi, L. Liao, Z. Lin, Regularized smoothing approximations to vertical nonlinear complementarity problems, *J. Math. Anal. Appl.*, **230** (1999), 261–276. http://dx.doi.org/10.1006/jmaa.1998.6205
38. Q. Shi, Q. Shen, T. Tang, A class of two-step modulus-based matrix splitting iteration methods for quasi-complementarity problems, *Comp. Appl. Math.*, **39** (2020), 11. http://dx.doi.org/10.1007/s40314-019-0984-4
39. M. Sun, Monotonicity of Mangasarian's iterative algorithm for generalized linear complementarity problems, *J. Math. Anal. Appl.*, **144** (1989), 474–485. http://dx.doi.org/10.1016/0022-247X(89)90347-8
40. R. Sznajder, M. Gowda, Generalizations of P_0- and P-properties; extended vertical and horizontal linear complementarity problems, *Linear Algebra Appl.*, **223-224** (1995), 695–715. http://dx.doi.org/10.1016/0024-3795(93)00184-2
41. Z. Xia, C. Li, Modulus-based matrix splitting iteration methods for a class of nonlinear complementarity problem, *Appl. Math. Comput.*, **271** (2015), 34–42. http://dx.doi.org/10.1016/j.amc.2015.08.108
42. S. Xie, Z. Yang, H. Xu, A modulus-based matrix splitting method for the vertical nonlinear complementarity problem, *J. Appl. Math. Comput.*, **69** (2023), 2987–3003. http://dx.doi.org/10.1007/s12190-023-01866-8
43. S. Xie, H. Xu, J. Zeng, Two-step modulus-based matrix splitting iteration method for a class of nonlinear complementarity problems, *Linear Algebra Appl.*, **494** (2016), 1–10. http://dx.doi.org/10.1016/j.laa.2016.01.002
44. L. Zhang, Two-step modulus based matrix splitting iteration for linear complementarity problems, *Numer. Algor.*, **57** (2011), 83–99. http://dx.doi.org/10.1007/s11075-010-9416-7
45. L. Zhang, Two-step modulus-based synchronous multisplitting iteration methods for linear complementarity problems, *J. Comput. Math.*, **33** (2015), 100–112. http://dx.doi.org/10.4208/jcm.1403-m4195

46. Y. Zhang, On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem, *SIAM J. Optim.*, **4** (1994), 208–227. http://dx.doi.org/10.1137/0804012

47. Y. Zhang, W. Guo, H. Zheng, S. Vong, A relaxed two-step modulus-based matrix synchronous multisplitting iteration method for linear complementarity problems, *Comp. Appl. Math.*, **43** (2024), 33. http://dx.doi.org/10.1007/s40314-023-02563-9

48. Y. Zhang, H. Zheng, X. Lu, S. Vong, Modulus-based synchronous multisplitting iteration methods without auxiliary variable for solving vertical linear complementarity problems, *Appl. Math. Comput.*, **458** (2023), 128248. http://dx.doi.org/10.1016/j.amc.2023.128248

49. Y. Zhang, H. Zheng, X. Lu, S. Vong, A two-step parallel iteration method for large sparse horizontal linear complementarity problems, *Appl. Math. Comput.*, **438** (2023), 127609. http://dx.doi.org/10.1016/j.amc.2022.127609

50. H. Zheng, X. Lu, S. Vong, A two-step modulus-based matrix splitting iteration method without auxiliary variable for solving vertical linear complementarity problems, *Commun. Appl. Math. Comput.*, in press. http://dx.doi.org/10.1007/s42967-023-00280-y

51. H. Zheng, Y. Zhang, X. Lu, S. Vong, Modulus-based synchronous multisplitting iteration methods for large sparse vertical linear complementarity problems, *Numer. Algor.*, **93** (2023), 711–729. http://dx.doi.org/10.1007/s11075-022-01436-2

52. H. Zheng, S. Vong, A two-step modulus-based matrix splitting iteration method for horizontal linear complementarity problems, *Numer. Algor.*, **86** (2021), 1791–1810. http://dx.doi.org/10.1007/s11075-020-00954-1

AIMS Press