East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.261215.220217a

Vol. 7, No. 2, pp. 363-375
May 2017

Modulus-based Synchronous Multisplitting Iteration Methods for an Implicit Complementarity Problem

Chen-Liang Li* and Jun-Tao Hong

School of Mathematics and Computing Science, Guangxi Colleges and Universities Key Laboratory of Data Analysis and Computation, Guilin and University of Electronic Technology, Guilin, Guangxi, China, 541004

Received 26 December 2015; Accepted (in revised version) 22 February 2017.

Abstract. We construct modulus-based synchronous multisplitting iteration methods to solve a large implicit complementarity problem on parallel multiprocessor systems, and prove their convergence. Numerical results confirm our theoretical analysis and show that these new methods are efficient.

AMS subject classifications: 65Y05, 65H10, 65N55

Key words: Implicit complementarity problem, synchronous, iterative method, modulus-based multisplitting.

1. Introduction

Implicit complementarity problems arise in a variety of applications in economics and engineering such as the Nash equilibrium point of a bimatrix game, contact problems, and the free boundary problem for journal bearings — cf. Ref. [9,13,24] and references therein. The modulus method is a classical iteration procedure to solve linear complementarity problems [12, 18, 26], Hadjidimos & Tzoumas [15, 16] proposed extrapolated modulus algorithms, and Bai [3] presented the modulus-based matrix splitting iteration method.

Given the computational requirements of modern high-speed multiprocessor environments, multisplitting iterative methods are powerful tools for large sparse linear complementarity problems [1, 2, 4–7, 11, 19, 23, 25]. Li & Zeng [20, 21] considered several synchronous and asynchronous multisplitting iteration schemes for solving a class of nonlinear complementarity problem when the system matrix is an H-matrix. Recently, via an equivalent reformulation of the linear complementarity problem as a system of fixed-point equations, Bai & Zhang [7] constructed modulus-based synchronous multisplitting (MSM) iteration methods that are suitable for implementation on multiprocessor systems. As one

*Corresponding author. *Email address:* chenliang_li@hotmail.com (C.-L. Li)

 ©2017 Global-Science Press

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

only needs to solve subsystems of linear equations at each iteration step in these modulus-based synchronous multisplitting (MSM) iteration methods, they are better than the general multisplitting iteration methods since they produce linear complementarity subproblems [8, 10, 15, 22, 27–32]. We extend these methods to solve an implicit complementarity problem, noting that it includes a corresponding linear complementarity problem.

Consider the following implicit complementarity problem that we denote as ICP(q, M):

Find a pair of real vectors w and $u \in R^n$ such that

$$u - m(u) \geq 0, \quad w := Mu + q \geq 0, \quad (u - m(u))^T (Mu + q) = 0 \tag{1.1}$$

where $M \in R^{n\times n}$ is a given sparse matrix and $q \in R^n$ is a real vector, and $m(\cdot)$ is an invertible mapping from R^n into itself.

In this article, we extend the MSM iteration methods to solve the $ICP(q, M)$. In Section 2, we present some necessary notation, definitions and lemmas. In Section 3, our MSM methods for solving the $ICP(q, M)$ are established. We prove the convergence of the MSM iteration methods when the system matrix is an H_+-matrix in Section 4. Numerical results presented in Section 5 show that the MSM iteration methods are computationally more efficient than modulus-based splitting iteration methods.

2. Preliminaries

Suppose that matrices $A = (a_{ij})$, $B = (b_{ij}) \in R^{n\times n}$. We shall write $A \geq B$ $(A > B)$ if $m_{ij} \geq n_{ij}$ $(m_{ij} > n_{ij})$ holds for all $1 \leq i \leq n$, $1 \leq j \leq n$. If O is a null matrix and $A \geq O$, A is called a nonnegative matrix. and $|A|$ denotes the nonnegative matrix with entries $|a_{ij}|$. If $A \in R^{n\times n}$ is a real $n \times n$ matrix, its comparison matrix is $\langle A\rangle = (\langle a_{ij}\rangle) \in R^{n\times n}$ with

$$\langle a\rangle_{ij} = \begin{cases} |a_{ij}|, & i = j \\ -|a_{ij}|, & i \neq j \end{cases}, \quad i, j = 1, 2, \cdots, n\,.$$

The matrix M is called an M-matrix if its off-diagonal entries are all non-positive and $M^{-1} \geq O$; M is called an H-matrix if its comparison matrix $\langle M\rangle$ is an M-matrix; an H-matrix M is called an H^+-matrix if it has a positive diagonal matrix; and if M is an M-matrix and Ω is a positive diagonal matrix, then $M \leq B \leq \Omega$ implies that B is an M-matrix. If F is a non-singular matrixm then $M = F - G$ is called a splitting of the matrix; and if $\langle M\rangle = \langle F\rangle - |G|$, then $M = F - G$ is called an H-compatible splitting.

Lemma 2.1 ([14]). *Let $A \in R^{n\times n}$ be an H - matrix, $D = diag(A)$ and $B = D - A$. Then*

(1) matrix A is non-singular;
(2) $|A^{-1}| \leq \langle A\rangle^{-1}$; and
(3) $|D|$ is non-singular and $\rho(|D|^{-1} \cdot |B|) < 1$.

Lemma 2.2 ([17]). *Let $M = F - G$ be a splitting of the matrix $M \in R^{n\times n}$, r a positive constant, and Ω a positive diagonal matrix. For the problem* (1.1)*, we then have the following:*

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

(a) if (u,w) is a solution of the problem (1), then $x=(r/2)(u-\Omega^{-1}w-m(u))$ satisfies the implicit fixed-point equation

$$(\Omega+F)\,x = Gx+(\Omega-M)\,|x|-r\cdot M\cdot m\left[g^{-1}\left(\frac{1}{r}\,(|x|+x)\right)\right]-r\cdot q\,; \tag{2.1}$$

(b) if x satisfies the implicit fixed-point equation (2.1)*, then*

$$u=\frac{1}{r}(|x|+x)+m(u), \qquad w=\frac{1}{r}\Omega(|x|-x) \tag{2.2}$$

is a solution of the ICP(q,M).

From Lemma 2.2, the ICP(q,M) can be transformed into an equivalent system of fixed-point equations.

3. MSM Iteration Method

Based on the equivalence Lemma 2.2, we first construct the MSM iteration methods for solving ICP(q,M). Let $\varsigma \le n$ be a given positive integer, $M=F_s-G_s$, $s=1,2,\cdots,\varsigma$ be a splitting of the system matrix$M\in R^{n\times n}$, and $E_s\in R^{n\times n}$, $s=1,2,\cdots,\varsigma$ nonnegative diagonal matrices satisfying $\sum_{s=1}^{\varsigma}E_s=I$ (the identity matrix). Then the collection of triples (M_k,F_k,G_k) $(k=1,2,\cdots,\varsigma)$ is called a multisplitting of the matrix M, and the matrices E_s $(s=1,2,\cdots,\varsigma)$ are called weighting matrices. Given a positive constant r and a positive diagonal matrix Ω, we can establish the following iteration method.

Method 3.1 (MSM)

Step 1: Given $\varepsilon>0$ and $u^0\in V$, set $k:=0$.
Step 2: Find the solution u^{k+1}.

(1) Calculate the initial vector

$$x^{0,k}=\frac{r}{2}\left(u^k-\Omega^{-1}w^k-m(u^k)\right). \tag{3.1}$$

(2) Set j:=0.
Parallel compute $x^{k,s}\in R^n$ by solving the equation

$$(\Omega+F_s)\,x^{k,s}=G_s x^{\mathrm{j},k}+(\Omega-M)\left|x^{j,k}\right|-r\cdot M\cdot m(u^k)-r\cdot q\,; \tag{3.2}$$

calculate $x^{\mathrm{j}+1,k}\in R^n$ from

$$x^{\mathrm{j}+1,\mathrm{k}}=\sum_{s=1}^{\varsigma}E_s x^{\mathrm{k},s}\,; \tag{3.3}$$

and if $\|x^{\mathrm{j}+1,\mathrm{k}}$ - $x^{\mathrm{j},\mathrm{k}}\|<\eta$, stop; otherwise, set j:=j+1.

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

(3) Compute

$$u^{k+1} = \frac{1}{\gamma}\left(\left|x^{j+1,k}\right| + x^{j+1,k}\right) + m(u^k). \tag{3.4}$$

Step 3: If $RES(u^{k+1}) = |(Mu^{k+1}+q)^T(u^{k+1} - m(u^{k+1}))| < \varepsilon$, then stop; otherwise, set $k := k+1$ and return to Step 2.

Method 3.1 has some quite good parallel computational properties. The splitting matrices F_s $(s = 1,2,\cdots,\varsigma)$ and the weighting matrices E_s $(s = 1,2,\cdots,\varsigma)$ can be chosen such that the tasks distributed on the ς processors of the multiprocessor system are well balanced, hence Method 3.1 achieves high parallel computational efficiency. Bai [7] has discussed several triangle splitting forms convenient for practical implementations, and here we adopt so-called Blocking splitting [20]. Consider the system matrix

$$M = \begin{pmatrix} B & -I & O & \cdots & O & O \\ -I & B & -I & \cdots & O & O \\ O & -I & B & \cdots & O & O \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ O & O & \cdots & \cdots & B & -I \\ O & O & \cdots & \cdots & -I & B \end{pmatrix},$$

where $M \in R^{m^2\times m^2}$, $B = tridiag(-1,4,-1) \in R^{m\times m}$ and $I \in R^{m\times m}$. Letting $\varsigma = 2$, we define

$$F_1 = \begin{pmatrix} B & -I & O & O & \cdots & & O \\ -I & B & -I & O & \cdots & & O \\ \vdots & \ddots & \ddots & \ddots & & \ddots & \vdots \\ O & \cdots & -I & B & O & \cdots & O \\ O & \cdots & & O & B & \ddots & \vdots \\ \vdots & & \ddots & & \ddots & \ddots & O \\ O & & \cdots & & & O & B \end{pmatrix} = \begin{pmatrix} M_1 & O \\ O & D_1 \end{pmatrix},$$

$$F_2 = \begin{pmatrix} B & \cdots & O & & \cdots & & O \\ \vdots & \ddots & & & & & \\ O & \cdots & B & O & \ddots & & \vdots \\ O & & O & B & -I & & \\ \vdots & & \ddots & -I & B & \ddots & O \\ & & & & \ddots & \ddots & -I \\ O & & \cdots & & O & -I & B \end{pmatrix} = \begin{pmatrix} D_2 & O \\ O & M_2 \end{pmatrix}.$$

For each splitting

$$M = F_s - G_s\,(s = 1,2), \tag{3.5}$$

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

Eq. (3.2) is equivalent to

$$\begin{cases} (\Omega_1 + M_1)x_1^{k,s} = H_1\,, \\ (\Omega_2 + M_2)x_2^{k,s} = H_2\,, \end{cases} \tag{3.6}$$

where $H_s = G_s x^{j,k} + f_s$, $f_s = [(\Omega - M)|x^{j,k}| - r \cdot M \cdot m(u^k) - rq]_s$, $s = 1, 2$. Thus to solve Eq. (3.2) we can consider several small scale subproblems, which can improve parallel computational efficiency.

4. Convergence Theory

In this section, we prove the convergence of the Method 3.1, when the system matrix $M \in R^{n\times n}$ is an H_+-matrix. We assume that $u^* \in R^n$ is a solution of the ICP(q, M), or equivalently that the vector pair (u^*, w^*) satisfies the ICP(q, M) such that $w^k = Mu^k + q$ and $w^* = Mu^* + q$. Let $x^{(0)}$ be an initial vector of inner iteration and x the solution of Eq. (2.1). Suppose that there exist a nonnegative matrices $N \in R^{n\times n}$ such that

$$\left|m(u) - m(v)\right| \le N\,|u - v|\,, \quad \forall u, v \in R^n\,, \tag{4.1}$$

so we have [17]

$$\left|x^{0,k} - x\right| \le \frac{r}{2}(I + \Omega^{-1}M + N)\left|u^k - u^*\right|\,. \tag{4.2}$$

We now demonstrate the convergence of the MSM iteration method to solve ICP(q, M).

Theorem 4.1. *Let (F_s, G_s, E_s) $(s = 1, 2, \cdots, \varsigma)$ be a multisplitting of the matrix $M \in R^{n\times n}$, r be a positive constant, Ω a positive diagonal matrix,*

$$\delta_1 = \sum_{s=1}^{\varsigma} E_s \left|(\Omega + F_s)^{-1}\right|\big(\,|G_s| + |\Omega - M|\,\big), \qquad \delta_2 = \sum_{s=1}^{\varsigma} E_s \left|(\Omega + F_s)^{-1}\right|\,|M|$$

and

$$\delta = \left(2\sum_{i=0}^{j}\delta_1^i\delta_2 + I\right), \qquad Z = \delta_1^{j+1}(I + \Omega^{-1}M + N) + \delta N\,. \tag{4.3}$$

Then if $\rho(Z) < 1$, for any initial vector u^0 the sequence $\{u^k\}$ generated by Method 3.1 converges to the unique solution of ICP(q, M).

Proof. Suppose that $\{u^k\}$ is the sequence generated by Method 3.1. Let u^* be the solution of ICP(q, M), and x the solution of Eq. (2.1). From Eq. (3.4) we get

$$u^{k+1} = \frac{1}{r}\left(\left|x^{j+1,k}\right| + x^{j+1,k}\right) + m(u^k)\,,$$
$$u^* = \frac{1}{r}\left(|x| + x\right) + m(u^*)\,,$$

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

hence

$$\begin{aligned}\left|u^{k+1}-u^{*}\right| &= \left|m(u^{k})-m(u^{*})+\frac{1}{r}\left(|x^{j+1,k}|+x^{j+1,k}-(|x|+x)\right)\right| \\ &\le \left|m(u^{k})-m(u^{*})\right|+\frac{1}{r}\left|\left(|x^{j+1,k}|+x^{j+1,k}-|x|-x\right)\right| \\ &\le N\left|u^{k}-u^{*}\right|+\frac{1}{r}\left(|x^{j+1,k}-x|+|x^{j+1,k}-x|\right) \\ &= N\left|u^{k}-u^{*}\right|+\frac{2}{r}\left|x^{j+1,k}-x\right|.\end{aligned}$$

From expression (3.3)

$$x^{j+1,k}=\sum_{s=1}^{\varsigma}E_{s}x^{k,s},$$

and from Eqs. (3.2) and (2.1)

$$\begin{aligned}(\Omega+F_{s})\,x^{k,s} &= G_{s}x^{j,k}+(\Omega-M)\,|x^{j,k}|-r\cdot M\cdot m(u^{k})-r\cdot q,\\ (\Omega+F_{s})x &= G_{s}x+(\Omega-M)\,|x|-r\cdot M\cdot m(u^{*})-r\cdot q,\end{aligned}$$

such that

$$\begin{aligned}x^{k,s} &= (\Omega+F_{s})^{-1}\left[G_{s}x^{j,k}+(\Omega-M)|x^{j,k}|-r\cdot M\cdot m(u^{k})-r\cdot q\right],\\ x &= (\Omega+F_{s})^{-1}\left[G_{s}x+(\Omega-M)|x|-r\cdot M\cdot m(u^{*})-r\cdot q\right],\end{aligned}$$

hence

$$\begin{aligned}\left|x^{j+1,k}-x\right| &= \sum_{s=1}^{\varsigma}E_{s}\left|x^{k,s}-x\right| \\ &= \sum_{s=1}^{\varsigma}E_{s}\Big|(\Omega+F_{s})^{-1}\big[G_{s}(x^{j,k}-x)+\left(\Omega-(F_{s}-G_{s})\right)\left(|x^{j,k}|-|x|\right) \\ &\quad -r\cdot M\cdot\left(m(u^{k})-m(u^{*})\right)\big]\Big| \\ &\le \sum_{s=1}^{\varsigma}E_{s}\Big|(\Omega+F_{s})^{-1}\Big|\Big(2|G_{s}||x^{j,k}-x|+|(\Omega-F_{s})||x^{j,k}-x| \\ &\quad +r|M|N|u^{k}-u^{*}|\Big) \\ &= \sum_{s=1}^{\varsigma}E_{s}\left|(\Omega+F_{s})^{-1}\right|\Big(2|G_{s}|+|\Omega-F_{s}|\Big)|x^{j,k}-x| \\ &\quad +r\sum_{s=1}^{\varsigma}E_{s}\left|(\Omega+F_{s})^{-1}\right||M|N|u^{k}-u^{*}| \\ &= \delta_{1}\left|x^{j,k}-x\right|+r\delta_{2}N\left|u^{k}-u^{*}\right|.\end{aligned}$$

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

Thus

$$\begin{aligned}\left|u^{k+1}-u^*\right| &\le N\left|u^k-u^*\right|+\frac{2}{r}\Big(\delta_1\left|x^{j,k}-x\right|+\delta_2 N\left|u^k-u^*\right|\Big)\\ &=\frac{2}{r}\delta_1\left|x^{j,k}-x\right|+(2\delta_2+I)\cdot N\left|u^k-u^*\right|\\ &\le\frac{2}{r}\delta_1^2\left|x^{j-1,k}-x\right|+(2\delta_1\delta_2+2\delta_2+I)N\left|u^k-u^*\right|\\ &\le\frac{2}{r}\delta_1^{j+1}\left|x^{0,k}-x\right|+\Big(2\delta_1^j\delta_2+\ldots+2\delta_1^2\delta_2+2\delta_1\delta_2+2\delta_2+I\Big)N\left|u^k-u^*\right|,\end{aligned}$$

so from inequality (4.2) we have

$$\begin{aligned}\left|u^{k+1}-u^*\right| &\le \delta_1^{j+1}(I+\Omega^{-1}M+N)\left|u^k-u^*\right|+\delta N\left|u^k-u^*\right|\\ &=\left[\delta_1^{j+1}(I+\Omega^{-1}M+N)+\delta N\right]\left|u^k-u^*\right|\end{aligned}$$

and hence

$$\left|u^{k+1}-u^*\right|\le Z\left|u^k-u^*\right|. \tag{4.4}$$

Since $\rho(Z)<1$, the sequence $\{u^k\}$ is bounded and converges to the solution u^*. □

In particular, when $M\in R^{n\times n}$ is an H_+-matrix, similar to the proof in Ref. [17] Theorem 4.1 immediately gives the following convergence result.

Theorem 4.2. *Let $M\in R^{n\times n}$ be an H_+-matrix,, $M=F-G$ an H-compatible splitting of the matrix M (i.e. $\langle M\rangle=\langle F\rangle-|G|$), Ω a positive diagonal matrix, and r a positive constant. Let $\lambda=\rho(N)$, ρ denote the spectral radius, $\phi_1=\sum_{s=1}^{\varsigma}E_s|(\Omega+\langle F_s\rangle)^{-1}|(|G_s|+|\Omega-M)$ and $\phi_2=\sum_{s=1}^{\varsigma}E_s|(\Omega+\langle F_s\rangle)^{-1}||M|$. If the parameter λ, $\rho(\phi_1)$ and $\rho(\phi_2)$ satisfy*

$$\left(1+\frac{2\rho(\phi_2)}{1-\rho(\phi_1)}\right)\lambda<1,$$

then the iteration sequence $\{u^k\}_{k=0}^{\infty}$ generated by Method 3.1 converges to the unique solution of $ICP(q,M)$.

Proof. As $M=F_s-G_s$, $(s=1,2,\cdots,\varsigma)$ are H-compatible splittings of the matrix M and Ω is a positive diagonal matrix, $\Omega+M_s$,$s=1,2,\cdots,\varsigma$ are H_+-matrices. From Lemma 2.1 (2) we have

$$\left|(\Omega+F_s)^{-1}\right|\le\langle\Omega+F_s\rangle^{-1}=\big(\Omega+\langle F_s\rangle\big)^{-1},\quad s=1,2,\cdots,\varsigma, \tag{4.5}$$

therefore

$$\begin{aligned}&\left|x^{j+1,k}-x\right|\\ &\le\sum_{s=1}^{\varsigma}E_s\left|(\Omega+F_s)^{-1}\right|\big(|G_s|+|\Omega-M|\big)\left|x^{j,k}-x\right|+r\sum_{s=1}^{\varsigma}E_s\left|(\Omega+F_s)^{-1}\right||M|N\left|u^k-u^*\right|\\ &\le\sum_{s=1}^{\varsigma}E_s\big(\Omega+\langle F_s\rangle\big)^{-1}\big(|G_s|+|\Omega-M|\big)\left|x^{j,k}-x\right|+r\sum_{s=1}^{\varsigma}E_s\left|(\Omega+\langle F_s\rangle)^{-1}\right||M|N\left|u^k-u^*\right|,\end{aligned}$$

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

and similar to the proof of Theorem 4.1 we have

$$\begin{aligned}\left|u^{k+1}-u^*\right| &\le \frac{2}{r}\phi_1^{j+1}\left|x^{0,k}-x\right| + \left(2\phi_1^j\phi_2+\cdots+2\phi_1^2\phi_2+2\phi_1\phi_2+2\phi_2+I\right)N\left|u^k-u^*\right| \\ &\le \left(\phi_1^{j+1}+\phi_1^{j+1}\Omega^{-1}M+\phi_1^{j+1}N+\phi N\right)\left|u^k-u^*\right| .\end{aligned}$$

Letting $\phi = (2\sum_{i=0}^{j}\phi_1^i\phi_2 + I)$ we obtain

$$\left|u^{k+1}-u^*\right| \le \left[\phi_1^{j+1}(I+\Omega^{-1}M+N)+\phi N\right]\left|u^k-u^*\right| = Z\left|u^k-u^*\right| ;$$

and from Theorem 4.1 and its proof [7] we have $\rho(\phi_1) < 1$ so $\phi_1^{j+1} \to 0$ when $j \to \infty$, and in the matrix Z we can omit the term $\phi_1^{j+1}(I+\Omega^{-1}M+N)$ to only consider $\rho(\phi N)$. According to the condition

$$\left(1+\frac{2\rho(\phi_2)}{1-\rho(\phi_1)}\right)\lambda < 1 ,$$

we then have

$$\begin{aligned}\rho(Z) \approx \rho(\phi N) &= \rho\left(\left(2\sum_{i=0}^{j}\phi_1^i\phi_2+I\right)N\right) \\ &\le \rho\left(2(\phi_1^j\phi_2+\cdots+\phi_1^2\phi_2+\phi_1\phi_2+\phi_2)+I\right)\cdot\rho(N) \\ &\le \rho\left(2(1-\phi_1)^{-1}\phi_2+I\right)\cdot\lambda \\ &\le \left(1+\frac{2\rho(\phi_2)}{1-\rho(\phi_1)}\right)\lambda < 1 ,\end{aligned}$$

to complete the proof. □

5. Numerical Results

We now give discuss some numerical experiments to test the effectiveness of the Method 3.1, referring to the number of iteration steps (denoted by 'IT'), the elapsed CPU time in seconds (denoted by 'CPU') and the residual errors (denoted by 'RES') defined as

$$RES(u^k) = abs\left((Mu^k+q)^T(u^k-m(u^k))\right)$$

where u^k is the kth approximate solution to the ICP(q, M). We recall that ς is the multisplitting number, and in passing note that our MSM iteration method reduces to the modulus-based splitting iteration method when $\varsigma = 1$ (to be compared). The initial vectors were chosen to be $u^0 = (0, \cdots, 0)^T \in R^n$, and all runs were performed in MATLAB with double machine precision. The inner iteration was terminated when $\|x^{j+1,k} - x^{j,k}\| < 10^{-4}$, and the outer iteration when $RES(u^k) \le 10^{-5}$.

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

Table 1: Testing methods.

MSMJ($\alpha = 1, \beta = 0$)	Multisplitting Modulus Jacobi Method
MSMAOR($\alpha = 1, \beta = 0.05$)	Multisplitting Modulus AOR Method
MSMSOR($\alpha = 0.9, \beta = 0.01$)	Multisplitting Modulus SOR Method

Table 2: Numerical results for Example 5.1.

MSMJ	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
$\varsigma = 1$	6	1.13	1.1e-6	6	1.99	1.8e-6	7	4.62	2.8e-6	17	6.13	2.8e-6
$\varsigma = 2$	7	0.03	1.0e-6	6	0.45	6.2e-6	5	1.69	9.7e-6	5	2.09	1.8e-6

Table 3: Numerical results for Example 5.1.

MSMAOR	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
$\varsigma = 1$	4	1.18	4.5e-6	5	2.25	1.5e-6	6	4.88	3.8e-6	17	6.63	2.7e-6
$\varsigma = 2$	4	0.14	2.1e-6	5	0.40	6.3e-6	5	1.30	9.6e-6	5	1.68	1.8e-6

Table 4: Numerical results for Example 5.1.

MSMSOR	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
$\varsigma = 1$	4	1.23	5.5e-6	5	1.96	1.0e-6	9	5.04	4.6e-6	18	6.83	2.9e-6
$\varsigma = 2$	4	0.13	9.6e-6	4	0.30	5.5e-6	5	1.25	8.4e-6	5	2.05	4.9e-6

Example 5.1. We considered the ICP(q, M) where $M \in R^{m^2 \times m^2}$ and $q \in R^{m^2}$ are

$$M = \begin{pmatrix} B & -I & O & \cdots & O & O \\ -I & B & -I & \cdots & O & O \\ O & -I & B & \cdots & O & O \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ O & O & \cdots & \cdots & B & -I \\ O & O & \cdots & \cdots & -I & B \end{pmatrix}, \quad q = \begin{pmatrix} -1 \\ 1 \\ -1 \\ \vdots \\ (-1)^{n-1} \\ (-1)^n \end{pmatrix} \text{ and } m(u) = \begin{pmatrix} u_1^3 \\ u_2^3 \\ u_3^3 \\ \vdots \\ u_{n-1}^3 \\ u_n^3 \end{pmatrix},$$

where $B = tridiag(-1, 4, -1) \in R^{m \times m}$ and $I \in R^{m \times m}$ is a unit matrix.

We used the multisplittings of the system matrix M as in Eq. (3.5) (Blocking splitting), instead of the triangular multisplittings in Ref. [7]. Tables 5-7 show that our methods ($\varsigma = 2$) produce better numerical results.

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

Table 5: Numerical comparison of different multisplittings for Example 5.1.

MSMJ	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
Triangular splitting	5	0.39	5.3e-6	4	0.89	5.3e-6	6	1.98	2.2e-6	8	2.86	2.7e-6
Block splitting	4	0.03	1.0e-6	6	0.45	6.2e-6	5	1.69	9.7e-6	5	2.09	1.8e-6

Table 6: Numerical comparison of different multisplittings for Example 5.1.

MSMAOR	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
Triangular splitting	4	0.53	2.3e-6	4	0.97	7.1e-6	5	2.19	1.9e-6	11	2.29	1.4e-6
Block splitting	4	0.14	2.1e-6	5	0.40	6.3e-6	5	1.30	9.6e-6	5	1.68	1.8e-6

Table 7: Numerical comparison of different multisplittings for Example 5.1.

MSMSOR	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
Triangular splitting	4	0.53	6.0e-6	4	1.02	9.4e-6	5	2.18	1.2e-6	11	3.17	1.4e-6
Block splitting	4	0.13	9.6e-6	4	0.30	5.5e-6	5	1.25	8.4e-6	5	2.05	4.9e-6

Example 5.2. We also considered the ICP(q, M) where $M \in R^{m^2\times m^2}$ and $q \in R^{m^2}$ are

$$M = \begin{pmatrix} B & 0.5I & O & \cdots & O & O \\ -1.5I & B & 0.5I & \cdots & O & O \\ O & -1.5I & B & \cdots & O & O \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ O & O & \cdots & \cdots & B & 0.5I \\ O & O & \cdots & \cdots & -1.5I & B \end{pmatrix}, \qquad q = \begin{pmatrix} -1 \\ 1 \\ -1 \\ \vdots \\ (-1)^{n-1} \\ (-1)^n \end{pmatrix},$$

and

$$m(u) = \left(u_1^3, u_2^3, u_3^3, \cdots, u_n^3\right),$$

where $B = tridiag(-1.5, 4, 0.5) \in R^{m\times m}$, $I \in R^{m\times m}$ is a unit matrix.

The numerical results in Tables 8-10 show that the MSM iteration methods are more efficient than the general modulus-based matrix splitting method.

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

Table 8: Numerical results for Example 5.2.

MSMJ	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
$\varsigma = 1$	3	0.33	3.0e-6	4	0.75	1.0e-6	2	3.45	6.0e-6	3	20.2	2.9e-6
$\varsigma = 2$	3	0.13	1.5e-6	3	0.27	5.3e-6	4	1.20	3.0e-6	8	8.1	3.2e-6

Table 9: Numerical results for Example 5.2.

MSMAOR	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
$\varsigma = 1$	4	0.24	4.3e-6	4	0.72	1.3e-6	5	3.46	7.2e-6	9	22.5	8.0e-6
$\varsigma = 2$	4	0.14	2.1e-6	4	0.28	6.9e-6	4	1.36	3.6e-6	7	8.84	6.6e-6

Table 10: Numerical results for Example 5.2.

MSMJ	n=121			n=225			n=529			n=1024		
	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES	IT	CPU	RES
$\varsigma = 1$	4	0.36	6.7e-6	5	0.87	2.3e-6	5	3.95	5.9e-6	8	20.2	6.8e-6
$\varsigma = 2$	4	0.12	3.4e-6	4	0.36	1.2e-6	5	1.52	6.6e-6	8	10.1	3.3e-6

6. Conclusion

We have constructed modulus-based synchronous multisplitting iteration methods to solve an implicit complementarity problem on parallel multiprocessor systems, with the relevant convergence theory. Numerical experiments confirm our theoretical analysis and show that our methods are more efficient than modulus-based splitting iteration methods.

Acknowledgments

The authors thank the referees for helpful suggestions. This work was supported by the Natural Science Foundation of China (Grant No. 11661027), the grant of China Scholarship Council, the Guangxi Natural Science Foundation (Grant Nos. 2015GXNSFAA139014 and 2014GXNSFAA118004), and the Innovation Project of GUET Graduate Education (Grant No. YJCXS201556).

References

[1] Z-Z. Bai, *The convergence of parallel iteration algorithms for linear complementarity problems*, Computers and Mathematics with Applications **32**, 1-17 (1996).

[2] Z-Z. Bai, *On the convergence of the multisplitting methods for the linear complementarity problem*, SIAM Journal of Matrix Analysis and Applications **21**, 67-78 (1999).

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

[3] Z-Z. Bai, *Modulus-based matrix splitting iteration methods for linear complementarity problems*, Numerical Linear Algebra with Applications **17**, 917-933 (2010).
[4] Z-Z. Bai and D.J. Evans, *Matrix multisplitting relaxation methods for linear complementarity problems*, International Journal of Computer Mathematics **63**, 309-326 (1997).
[5] Z-Z, Bai and D.J. Evans, *Matrix multisplitting methods with applications to linear complementarity problems: Parallel synchronous and chaotic methods*, Reseaux et Systemes Repartis: Calculateurs Paralleles **13**, 125-154 (2001).
[6] Z-Z. Bai, J-C. Sun and D-R. Wang, *A unified framework for the construction of various matrix multisplitting iterative methods for large sparse system of linear equations*, Computers and Mathematics with Applications **32**, 51-76 (1996).
[7] Z-Z, Bai and L-L. Zhang, *Modulus-based synchronous multisplitting iteration methods for linear complementarity problems*, Numerical Linear Algebra with Applications **20**, 425-439 (2013).
[8] Z-Z. Bai and L-L. Zhang, *Modulus-based synchronous two-stage multisplitting iteration methods for linear complementarity problems*, Numerical Algorithms **62**, 59-77 (2013).
[9] R.W, Cottle, J.-S. Pang and R.E. Stone, *The Linear Complementarity Problem*, Academic Press (1992).
[10] C.L. Cvetkov and V, Kostic, *A note on the convergence of the MSMAOR method for linear complementarity problems*, Numerical Linear Algebra with Applications **21**, 534-539 (2014).
[11] J-L. Dong, *Inexact multisplitting methods for linear complementarity problems*, Journal of Computational and Applied Mathematics **223**, 714-724 (2009).
[12] J-L. Dong and M-Q. Jiang, *A modified modulus method for symmetric positive-definite linear complementarity problems*, Numerical Linear Algebra with Applications **16**, 129-143 (2009).
[13] M.C, Ferris and J-S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Review **39**, 669-713 (1997).
[14] A. Frommer and D.B. Szyld, *H-splittings and two-stage iterative methods*, Numerische Mathematik **63**, 345-356 (1992).
[15] A. Hadjidimos, M. Lapidakis and M. Tzoumas, *On iterative solution for linear complementarity problems with an H_+matrix*, SIAM Journal of Matrix Analysis and Applications **33**, 97-110 (2012).
[16] A. Hadjidimos and M. Tzoumas, *Nonstationary extrapolated modulus algorithms for the solution of the linear complementarity problem*, Linear Algebra and Its Applications **431**, 197-210 (2009).
[17] J-T. Hong and C-L. Li, *Modulus-based matrix splitting iteration methods for a class of implicit complementarity problems*, Numerical Linear Algebra with Applications **23**, 629-641 (2016).
[18] N.W. Kapple and L.T. Waston, *Iterative algorithms for the linear complementarity problems*, International Journal of Computer Mathematics **19**, 273-297 (1986).
[19] D.P. O'Leary and R.E. White, *Multisplitting of matrices and parallel solution of linear system*, SIAM Journal on Algebraic Discrete Methods **6**, 630-640 (1985).
[20] C-L. Li, *Some New Methods for Variational Inequalities and Complementarity Problems*, PhD Thesis, Hunan University, 44-50 (2004)
[21] C-L. Li and J-P. Zeng, *Multisplitting Iteration Schemes for solving a class of nonlinear complementarity problems*, Acta Mathematicae Applicatae Sinca **1**, 79-90 (2007).
[22] LI W, *A general modulus-based matrix splitting iteration methods for linear complementarity problems of H-matrices*, Applied Mathematics Letters **26**, 1159-1164 (2013).
[23] N. Machida, N. Fukushima and T. Ibaraki, *A multisplitting method for symmetric linear complementarity problems*, Journal of Computational and Applied Mathematics **62**, 217-227 (1995).
[24] K. Murty, *Linear Complementarity, Linear and Nonlinear Programming*, Heldermann (1988).
[25] M. Naoki, F. Masao and E. ToshideI, *A multisplitting method for symmetric linear complemen-*

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press

tarity problems, Journal of Computational and Applied Mathematics **62**, 217-227 (1995).

[26] W.M.G. Van Bokhoven, *Piecewise-linear Modelling and Analysis*, Proefschrift: Eindhoven. (1981).

[27] L-L. Zhang, *Two-step modulus-based matrix splitting iteration methods for linear complementarity problems*, Numerical Algorithms **57**, 83-99 (2011).

[28] L-L. Zhang, *Two-stage multisplitting iteration methods using modulus-based matrix splitting as inner iteration for linear complementarity problems*, Journal of Optimization Theory and Applications **160**, 189-203 (2014).

[29] L-L. Zhang, *Two-step modulus-based synchronous multisplitting iteration methods for linear complementarity problems*, Journal of Computational and Applied Mathematics **33**, 100-112 (2015)

[30] L-L. Zhang and Z-R. Ren, *Improved convergence theorems of modulus-based matrix splitting iteration methods for linear complementarity problems*, Applied Mathematics Letters **26**, 638-642 (2013).

[31] N, Zheng and J-F. Yin, *Accelerated modulus-based matrix splitting iteration methods for linear complementarity problem*, Numerical Algorithms **64**, 245-262 (2013).

[32] N, Zheng and J-F. Yin, *Convergence of accelerated modulus-based matrix splitting iteration methods for linear complementarity problem with an H_+-matrix*, Journal of Computational and Applied Mathematics **260**, 281-293 (2014).

https://doi.org/10.4208/eajam.261215.220217a Published online by Cambridge University Press