Journal of Computational Mathematics, Vol.20, No.1, 2002, 97–112.

RELAXED ASYNCHRONOUS ITERATIONS FOR THE LINEAR COMPLEMENTARITY PROBLEM*1)

Zhong-zhi Bai
(*LSEC, ICMSEC, Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing 100080, China*)

Yu-guang Huang
(*Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford OX1 3QD, U.K.*)

Abstract

We present a class of relaxed asynchronous parallel multisplitting iterative methods for solving the linear complementarity problem on multiprocessor systems, and set up their convergence theories when the system matrix of the linear complementarity problem is an H-matrix with positive diagonal elements.

Key words: Linear complementarity problem, Matrix multisplitting, Relaxation method, Asynchronous iteration, Convergence theory.

1. Introduction

Consider the large sparse Linear Complementarity Problem (LCP):

$$Mz + q \geq 0, \quad z \geq 0, \quad z^T(Mz + q) = 0,$$

where $M = (m_{kj}) \in L(R^n)$ is a given real matrix and $q = (q_k) \in R^n$ a given real vector. This problem arises in many areas of scientific computing. For example, it arises from problems in (linear and) convex quadratic programming, the problem of finding a Nash equilibrium point of a bimatrix game (e.g., Cottle and Dantzig[5] and Lemke[13]), and also a number of free boundary problems of fluid mechanics (e.g., Cryer[8]). There have been a lot of researches on the approximate solution of the LCP in the literature, e.g., Cottle and Sacher[7], Cottle, Golub and Sacher[6], Mangasarian[14], Mangasarian and De Leone[15] and De Leone and Mangasarian[9]. These works, besides presenting efficient iterative methods, afforded feasible ways and essential techniques for studying the numerical solution of the LCP.

To solve the LCP on high-speed multiprocessor systems, recently, Bai and Evans[2] and Bai, Evans and Wang[3] presented a class of relaxed parallel iterative methods. These methods are based upon several splittings of the system matrix $M \in L(R^n)$, as well as the equivalence of the LCP to a fixed-point system, and they have many advantages such as high parallelism, strong generality and extensive applicability.

In accordance with the principle of sufficiently using the delayed information, and by making use of both the matrix multisplitting and the successive overrelaxation techniques, in this paper we propose a class of new relaxed asynchronous iterations for solving the LCP on the MIMD

* Received December 13, 1999; Final revised September 12, 2000.
1) Subsidized by The Special Funds For Major State Basic Research Project G1999032803.

systems. These new methods decrease the exchange frequencies of the information among the processors and increase the efficient numerical computations of each processor. Moreover, mutual waits among the processors of the multiprocessor system are thoroughly avoided. In addition, since two relaxation sweeps are induced within each iteration, and each sweep possibly includes its own pair of relaxation parameters, these new methods can cover all the known and a lot of novel practical and efficient relaxed asynchronous parallel methods in the sense of multisplitting. Following suitable adjustments of the relaxation parameters, the convergence properties of this class of new asynchronous parallel multisplitting relaxation methods can be greatly improved. These are the advantages of our new methods over the known ones discussed in [2] and [3]. When the system matrix $M \in L(R^n)$ is an H-matrix with positive diagonal elements, we establish the convergence theories of these new methods under proper constraints on both the multiple splittings and the relaxation parameters.

This paper affords efficient method models and necessary convergence theories for the asynchronous parallel multisplitting relaxation iterations for solving the LCP on MIMD multiprocessor systems. Essentially, it is an extension of the work of Bai and Evans in [2]; and is also a development of those of Bai, Evans and Wang in [3].

2. Preliminaries

We first briefly describe the notations. Let $C = (c_{kj})$ be an $n \times n$ matrix. By $diag(C)$ we denote the $n \times n$ diagonal matrix coinciding in its diagonal with C. For $A = (a_{kj})$, $B = (b_{kj}) \in L(R^n)$, we write $A \leq B$ if $a_{kj} \leq b_{kj}$ holds for all $k, j = 1, 2, \cdots, n$. We call A nonnegative if $A \geq 0$. This definition carries immediately over to vectors by identifying them with $n \times 1$ matrices. In particular, we call the vector $x \in R^n$ positive (writing $x > 0$) if all its entries are positive. By $|A| = (|a_{kj}|)$ we define the absolute value of $A \in L(R^n)$; it is a nonnegative $n \times n$ matrix satisfying $|AB| \leq |A||B|$ for any $B \in L(R^n)$. We denote by $\langle A \rangle = (\alpha_{kj})$ the $n \times n$ comparison matrix of $A \in L(R^n)$ where $\alpha_{kj} = |a_{kk}|$ for $k = j$ and $\alpha_{kj} = -|a_{kj}|$ for $k \neq j$, $k, j = 1, 2, \cdots, n$. We call $A = (a_{kj}) \in L(R^n)$ an M-matrix if it is nonsingular with $a_{kj} \leq 0$ for $k \neq j$ and $A^{-1} \geq 0$. We call it an H-matrix if $\langle A \rangle$ is an M-matrix. Note that an H-matrix is nonsingular, and has the properties that $|A^{-1}| \leq \langle A \rangle^{-1}$ and $\rho(|D_A|^{-1}|B_A|) < 1$, where $D_A = diag(A)$, $B_A = D_A - A$ and $\rho(\bullet)$ the spectral radius of a matrix. If $x \in R^n$, x_+ is used to denote the vector with elements $(x_+)_j = \max\{0, x_j\}$, $j = 1, 2, \cdots, n$. For any $x, y \in R^n$, there hold: (a) $(x + y)_+ \leq x_+ + y_+$; (b) $x_+ - y_+ \leq (x - y)_+$; (c) $|x| = x_+ + (-x)_+$; and (d) $x \leq y$ implies $x_+ \leq y_+$.

It is well-known that the LCP can be equivalently transformed to a fixed-point problem. More concretely, we have the following conclusion:

Lemma 2.1. (*see [2, 14]*) *A vector $z \in R^n$ solves the LCP if and only if it satisfies*

$$z = (z - \Phi(Mz + q))_+ ,$$

where $\Phi = diag(\varphi_1, \varphi_2, \cdots, \varphi_n) \in L(R^n)$ is any positive diagonal matrix.

Since a fixed point equation readily leads to an iterative method, Lemma 2.1 then affords one basis for the establishments of some efficient and practical iterative methods for solving the LCP; see, e.g., [1]-[3], [6]-[9], [13]-[15] and references therein. Moreover, the existence and uniqueness of the solution of the LCP can be directly concluded from Lemma 2.1.

Lemma 2.2. (*see [2]*) *Let $M \in L(R^n)$ be an H-matrix with positive diagonal elements. Then the LCP has a unique solution $z^* \in R^n$.*

3. Establishment of the Methods

Given a positive integer α ($\alpha \le n$), we separate the number set $N := \{1, 2, \cdots, n\}$ into α nonempty subsets $J_i (i = 1, 2, \cdots, \alpha)$ such that $\cup_{i=1}^{\alpha} J_i = N$. Corresponding to this separation, for $i \in \Lambda := \{1, 2, \cdots, \alpha\}$ we introduce matrices

$$L_i = (\mathcal{L}_{kj}^{(i)}) \in L(R^n), \quad \mathcal{L}_{kj}^{(i)} = \begin{cases} l_{kj}^{(i)}, & \text{for } k, j \in J_i \text{ and } k > j, \\ 0, & \text{otherwise,} \end{cases} \qquad k, j \in N,$$

$$U_i = (\mathcal{U}_{kj}^{(i)}) \in L(R^n), \quad \mathcal{U}_{kj}^{(i)} = \begin{cases} u_{kj}^{(i)}, & \text{for } k, j \in J_i \text{ and } k < j, \\ 0, & \text{otherwise,} \end{cases} \qquad k, j \in N,$$

$$W_i = (\mathcal{W}_{kj}^{(i)}) \in L(R^n), \quad \mathcal{W}_{kj}^{(i)} = \begin{cases} 0, & \text{for } k = j, \\ w_{kj}^{(i)}, & \text{otherwise,} \end{cases} \qquad k, j \in N,$$

$$E_i = diag(e_k^{(i)}) \in L(R^n), \quad e_k^{(i)} = \begin{cases} e_k^{(i)} \ge 0, & \text{for } k \in J_i, \\ 0, & \text{otherwise,} \end{cases} \qquad k \in N.$$

Evidently, $L_i (i = 1, 2, \cdots, \alpha)$ are strictly lower triangular matrices, $U_i (i = 1, 2, \cdots, \alpha)$ are strictly upper triangular matrices, $W_i (i = 1, 2, \cdots, \alpha)$ are zero-diagonal matrices, and $E_i (i = 1, 2, \cdots, \alpha)$ are nonnegatively diagonal matrices. For the system matrix $M \in L(R^n)$ of the LCP, let $D = diag(M)$ nonsingular and $M = D + B$. If

(a) $M = D + L_i + U_i + W_i \equiv D + B, \quad i = 1, 2, \cdots, \alpha;$

(b) $\sum_{i=1}^{\alpha} E_i = I$ (the $n \times n$ identity matrix),

then the collection $(D + L_i, D + U_i, W_i, E_i)(i = 1, 2, \cdots, \alpha)$ is called a multisplitting of the matrix $M \in L(R^n)$.

Assume that the considered multiprocessor system is constructed by α processors. Then, to describe the new asynchronous parallel multisplitting relaxation methods, we introduce the following notations.

(1) for $\forall i \in \Lambda$ and $\forall p \in N_0 := \{0, 1, 2, \cdots\}$, $J_i(p)$ denotes a subset of the set J_i. Write $J^{(i)} = \{J_i(p)\}_{p \in N_0}$;

(2) for $\forall k \in N$ and $\forall p \in N_0$, $N_k(p) = \{i \mid k \in J_i(p), \quad i = 1, 2, \cdots, \alpha\}$;

(3) for $\forall k \in N$ and $\forall i \in \Lambda$, $\{s_k^{(i)}(p)\}_{p \in N_0}$ denotes an infinite nonnegative integer sequence. Write $S^{(i)} = \{s_1^{(i)}(p), s_2^{(i)}(p), \cdots, s_n^{(i)}(p)\}_{p \in N_0}$.

$J^{(i)}$ and $S^{(i)} (i = 1, 2, \cdots, \alpha)$ have the following properties:

(a) for $\forall k \in N$ and $\forall i \in \Lambda$, the set $\{p \in N_0 \mid k \in J_i(p)\}$ is infinite;

(b) for $\forall p \in N_0$, $\cup_{i=1}^{\alpha} J_i(p) \ne \emptyset$;

(c) for $\forall k \in N$, $\forall i \in \Lambda$ and $\forall p \in N_0$, $s_k^{(i)}(p) \le p$;

(d) for $\forall k \in N$ and $\forall i \in \Lambda$, $\lim_{p \to \infty} s_k^{(i)}(p) = \infty$.

For $\forall p \in N_0$, if we define $s(p) = \min_{k \in N, i \in \Lambda} s_k^{(i)}(p)$, then there hold $s(p) \le p$ and $\lim_{p \to \infty} s(p) = \infty$. Here, we remark that α is the number of the processors, J_i is the subset of variables assigned to processor i, $J_i(p)$ is the subset of variables updated by processor i at iteration p, $N_k(p)$ is the subset of processors updating the k-th entry of the global variable at iteration p, and $s_j^{(i)}(p)$ is the iteration index for the j-th entry of the global variable received by processor i from the host processor at iteration p.

Now, based upon the above prerequisites, we can establish the following asynchronous parallel multisplitting relaxation method for solving the LCP.

Method I. Let $z^0 \in R^n$ be an initial vector, and assume that we have got the approximate solutions $z^0, z^1, \cdots, z^p$ of the LCP. Then the $(p+1)$-th approximate solution $z^{p+1} = (z_1^{p+1}, \cdots, z_n^{p+1})^T$ of the LCP can be obtained element by element from

$$z_k^{p+1} = \sum_{i \in N_k(p)} e_k^{(i)} z_k^{p+1,i} + \sum_{i \notin N_k(p)} e_k^{(i)} z_k^p, \qquad k \in N, \tag{3.1}$$

where $z_k^{p+1,i} (k \in J_i(p), i \in \Lambda)$ are successively determined by the formulas

$$\begin{aligned} z_k^{p+1/2,i} =& \Bigg(z_k^{s_k^{(i)}(p)} - \varphi_k^{(1)} \Bigg[\gamma_1 \sum_{\substack{j<k \\ j \in J_i(p)}} l_{kj}^{(i)} (z_j^{p+1/2,i} - z_j^{s_j^{(i)}(p)}) \\ &+ \omega_1 \Bigg(\sum_{j=1}^n m_{kj} z_j^{s_j^{(i)}(p)} + q_k \Bigg) \Bigg] \Bigg)_+, \quad k \in J_i(p); \\ z_k^{p+1,i} =& \Bigg(z_k^{p+1/2,i} - \varphi_k^{(2)} \Bigg[\gamma_2 \sum_{\substack{j>k \\ j \in J_i(p)}} u_{kj}^{(i)} (z_j^{p+1,i} - z_j^{p+1/2,i}) \\ &+ \omega_2 \Bigg(\sum_{j \in J_i(p)} m_{kj} z_j^{p+1/2,i} + \sum_{j \in N \setminus J_i(p)} m_{kj} z_j^{s_j^{(i)}(p)} + q_k \Bigg) \Bigg] \Bigg)_+, \quad k \in J_i(p). \end{aligned} \tag{3.2}$$

Here $\gamma_1, \gamma_2 \in [0, \infty)$ are relaxation factors, $\omega_1, \omega_2 \in (0, \infty)$ are acceleration factors, and $\Phi_k = diag(\varphi_j^{(k)})$, $k = 1, 2$, are positively diagonal matrices.

In Method I, we have stipulated that the multiprocessor system is made up of α processors, and the host processor is undertaken by any one of these α processors. Processor i only needs to solve variables in the variable subset J_i assigned to it through the two relaxation sweeps in (3.2) defined explicitly by the i-th splitting $M = D + L_i + U_i + W_i$ of the system matrix $M \in L(R^n)$. Because the subset $J_i(p)$ of the variables updated by processor i at iteration p is again a subset of J_i, a piece of new local iteration $z^{p+1,i}$ can be contributed frequently by processor i to form the global iteration z^{p+1}. The α splittings $M = D + L_i + U_i + W_i$ $(i = 1, 2, \cdots, \alpha)$ and the α weightings E_i $(i = 1, 2, \cdots, \alpha)$ can be chosen suitably so that the workloads among the processors are well balanced, and Method I achieves high parallel efficiency. Moreover, considerable computational workloads can be saved since the elements of the local variables corresponding to the zero entries of the weightings need not be calculated. In addition, corresponding to suitable choices of the involved arbitrary parameters, a series of practical asynchronous parallel relaxation methods for solving the LCP in the sense of multisplitting can be generated. In particular, this method covers the asynchronous matrix multisplitting AOR method discussed in [2], which is given by choosing $\gamma_2 = \omega_2 = 0$ and

$$J_i(p) = J_i, \quad s_k^{(i)}(p) = s_i(p) \in N_0, \quad \forall i \in \Lambda, \quad \forall k \in N, \quad \forall p \in N_0;$$

the multi-parameter relaxed parallel multisplitting methods investigated in [3], which are resulted by letting

$$J_i(p) = J_i \equiv N, \quad s_k^{(i)}(p) = p, \quad \forall i \in \Lambda, \quad \forall k \in N, \quad \forall p \in N_0;$$

and a class of applicable asynchronous variants of the latter ones, which are yielded by selecting

$$J_i(p) = J_i, \quad s_k^{(i)}(p) = s_i(p) \in N_0, \quad \forall i \in \Lambda, \quad \forall k \in N, \quad \forall p \in N_0.$$

Moreover, reasonable adjustments of the relaxation parameters can considerably improve the convergence property of this method.

On the other hand, it is evident that, in this method, each processor is allowed to update the globally approximated solution, or retrive any subset of the elements of the globally approximated solution residing in the host processor, at any time. Hence, new information can be used on time once it is available.

If we supplement $z_k^{p+1,i} = z_k^{p+1/2,i} = z_k^{s_k^{(i)}(p)} (k \in N \setminus J_i(p), i \in \Lambda)$, then for $\forall k \in J_i(p)$, by the concrete structures of the matrices L_i and U_i $(i = 1, 2, \cdots, \alpha)$ (3.2) can be equivalently expressed in the following form:

$$z_k^{p+1/2,i} = \left(z_k^{s_k^{(i)}(p)} - \varphi_k^{(1)} \left[\gamma_1 \sum_{\substack{j<k \\ j \in J_i}} l_{kj}^{(i)} (z_j^{p+1/2,i} - z_j^{s_j^{(i)}(p)}) + \omega_1 \left(\sum_{j=1}^{n} m_{kj} z_j^{s_j^{(i)}(p)} + q_k \right) \right] \right)_+ ;$$

$$z_k^{p+1,i} = \left(z_k^{p+1/2,i} - \varphi_k^{(2)} \left[\gamma_2 \sum_{\substack{j>k \\ j \in J_i}} u_{kj}^{(i)} (z_j^{p+1,i} - z_j^{p+1/2,i}) + \omega_2 \left(\sum_{j=1}^{n} m_{kj} z_j^{p+1/2,i} + q_k \right) \right] \right)_+ . \tag{3.3}$$

Furthermore, by introducing a new relaxation parameter $\beta \in (0, \infty)$, we can extend Method I to the following extrapolated one.

Method II. Let $z^0 \in R^n$ be an initial vector, and assume that we have got the approximate solutions $z^0, z^1, \cdots, z^p$ of the LCP. Then the $(p+1)$-th approximate solution z^{p+1} of the LCP can be obtained element by element from

$$z_k^{p+1} = \sum_{i \in N_k(p)} e_k^{(i)} \overline{z}_k^{p+1,i} + \sum_{i \notin N_k(p)} e_k^{(i)} z_k^p, \qquad k \in N \tag{3.4}$$

with

$$\overline{z}_k^{p+1,i} = \beta z_k^{p+1,i} + (1-\beta) z_k^{s_k^{(i)}(p)}, \qquad k \in J_i(p), \quad i \in \Lambda, \tag{3.5}$$

where $z_k^{p+1,i}$ and $z_k^{p+1/2,i} (k \in J_i(p), i \in \Lambda)$ are successively determined by (3.2) or (3.3).

Obviously, Method II is a meaningful modification of Method I, and it possesses all the properties and advantages of Method I. Likewise, corresponding to different choices of the arbitrary parameters involved in it, Method II can result in another series of asynchronous parallel multisplitting relaxation methods for solving the LCP on MIMD-systems. This then makes it be possible to implement all the synchronous parallel multisplitting relaxation methods in [2, 3] for solving the LCP in the asynchronous parallel environments.

4. Convergence Theories

To establish the convergence theorems of the new asynchronous parallel multisplitting relaxation methods, we require the following known result, which was proved in [4].

Lemma 4.1. *(see [4]) Given $\bar{x}^* \in R^n$ and $\{\bar{x}^t\}_{t=0}^p \subset R^n (\forall p \in N_0)$. Assume for all $t \in \{0, 1, \cdots, p\}$ that there exist a positive constant δ and a positive vector $v \in R^n$ such that $|\bar{x}^t - \bar{x}^*| \le \delta v$. Then there hold $|\bar{x}^{s^{(i)}(p)} - \bar{x}^*| \le \delta v$ for $\forall i \in \Lambda$, provided $s_k^{(i)}(p) \le p (\forall k \in N, \forall i \in \Lambda)$, where*

$$\bar{x}^{s^{(i)}(p)} = \left(\bar{x}_1^{s_1^{(i)}(p)}, \bar{x}_2^{s_2^{(i)}(p)}, \cdots, \bar{x}_n^{s_n^{(i)}(p)} \right)^T .$$

For all $p \in N_0$ and $k \in N$, let $\mathcal{I}_k^p = \sum_{i \in N_k(p)} e_k^{(i)}$ and $\mathcal{P}_k^p = \mathcal{I}_k^p (\mathcal{I}_k^p)^+$, where $(\mathcal{I}_k^p)^+$ is equal to $(\mathcal{I}_k^p)^{-1}$ if $\mathcal{I}_k^p \neq 0$ and 0 otherwise. Then the following useful relations are evident.

Lemma 4.2. *The following conclusions are true:*

(1) $0 \leq \mathcal{I}_k^p \leq \mathcal{P}_k^p \leq 1$; $(\mathcal{P}_k^p)^2 = \mathcal{P}_k^p$, $\forall p \in N_0$;

(2) $\mathcal{P}_k^p \mathcal{I}_k^p = \mathcal{I}_k^p \mathcal{P}_k^p = \mathcal{I}_k^p$, $\forall k \in N$, $\forall p \in N_0$;

(3) $(1 - \mathcal{P}_k^p)\mathcal{I}_k^p = \mathcal{I}_k^p(1 - \mathcal{P}_k^p) = 0$, $\forall k \in N$, $\forall p \in N_0$;

(4) $(1 - \mathcal{P}_k^p)(1 - \mathcal{I}_k^p) = (1 - \mathcal{I}_k^p)(1 - \mathcal{P}_k^p) = 1 - \mathcal{P}_k^p$, $\forall k \in N$, $\forall p \in N_0$.

Define the positive integer sequence $\{m_t\}_{t \in N_0}$ by induction in accordance with the following rule: $m_0 = 0$, and for $t = 0, 1, 2, \cdots$, m_{t+1} is the least positive integer such that $\bigcup_{m_t \leq s(p) \leq p < m_{t+1}} J_i(p) = J_i$, $i = 1, 2, \cdots, \alpha$. Then by the definitions of the subsets J_i, $J_i(p)$ and the nonnegative integer sequences $\{s_j^{(i)}(p)\}_{p \in N_0}$, $\{s(p)\}_{p \in N_0}$, the positive integer sequence $\{m_t\}_{t \in N_0}$ is well-defined and possesses the following properties:

Lemma 4.3. *For $\forall k \in N$ and $\forall t \in N_0$, there hold*

(1) $\mathcal{Q}_k^{(t)} \equiv \sum_{p=m_t}^{m_{t+1}-1} \mathcal{P}_k^p$ *is positive;*

(2) $\mathcal{S}_k^{(t)} \equiv \prod_{p=m_t}^{m_{t+1}-1} (1 - \mathcal{P}_k^p) = 0$.

Proof. Evidently, for $\forall k \in N$ and $\forall t \in N_0$, there hold $\mathcal{P}_k^p \in \{0, 1\}$, that is, either $\mathcal{P}_k^p = 0$ or $\mathcal{P}_k^p = 1$. Now, we first verify (1). Suppose that for some $k_0 \in N$ and some $t_0 \in N_0$ there hold $\mathcal{Q}_{k_0}^{(t_0)} = \sum_{p=m_{t_0}}^{m_{t_0+1}-1} \mathcal{P}_{k_0}^p = 0$. Then we have $\mathcal{P}_{k_0}^p = 0$, $p = m_{t_0}, m_{t_0} + 1, \cdots, m_{t_0+1} - 1$. Hence, $\mathcal{I}_{k_0}^p = 0$, $p = m_{t_0}, m_{t_0} + 1, \cdots, m_{t_0+1} - 1$, or in other words, $\sum_{i \in N_{k_0}(p)} e_{k_0}^{(i)} = 0$, $p = m_{t_0}, m_{t_0}+1, \cdots, m_{t_0+1}-1$. These equalities and the definition of the integer sequence $\{m_t\}_{t \in N_0}$ straightforwardly imply that $e_{k_0}^{(i)} = 0$, $i = 1, 2, \cdots, \alpha$, or equivalently, $\sum_{i=1}^{\alpha} e_{k_0}^{(i)} = 0$. However, this obviously contradicts with $\sum_{i=1}^{\alpha} e_k^{(i)} = 1 (k = 1, 2, \cdots, n)$, which is required by the weighting matrices $E_i (i = 1, 2, \cdots, \alpha)$.

To verify (2), we notice that $\mathcal{P}_k^p(1 - \mathcal{P}_k^p) = 0$ holds for $\forall k \in N$ and $\forall p \in N_0$. Then, by direct calculations we immediately obtain

$$\begin{aligned} \mathcal{Q}_k^{(t)} \mathcal{S}_k^{(t)} &= \left(\sum_{p=m_t}^{m_{t+1}-1} \mathcal{P}_k^p \right) \left(\prod_{p=m_t}^{m_{t+1}-1} (1 - \mathcal{P}_k^p) \right) \\ &= \sum_{p=m_t}^{m_{t+1}-1} \left(\mathcal{P}_k^p (1 - \mathcal{P}_k^p) \prod_{\substack{m_t \leq q < m_{t+1} \\ q \neq p}} (1 - \mathcal{P}_k^q) \right) = 0. \end{aligned}$$

From (1) we see that $\mathcal{Q}_k^{(t)} > 0$. Therefore, $\mathcal{S}_k^{(t)} = 0$.

Now, let the system matrix $M \in L(R^n)$ of the LCP be an H-matrix with positive diagonal elements. Then in accordance with Lemma 2.2 we know that the LCP has a unique solution $z^* = (z_1^*, z_2^*, \cdots, z_n^*)^T \in R^n$. Hence, Lemma 2.1 gives the identities

$$z_k^* = \left(z_k^* - \varphi_k \omega \left[\sum_{j=1}^{n} m_{kj} z_j^* + q_k \right] \right)_+, \qquad k \in N. \tag{4.1}$$

Subtracting (4.1) from (3.3) we have for $k \in J_i(p)$ and $i \in \Lambda$ that

$$\begin{aligned}
&z_k^{p+1/2,i} - z_k^* \\
&\quad= \left(z_k^{s_k^{(i)}(p)} - \varphi_k^{(1)} \left[\gamma_1 \sum_{\substack{j<k \\ j \in J_i}} l_{kj}^{(i)} (z_j^{p+1/2,i} - z_j^{s_j^{(i)}(p)}) + \omega_1 \left(\sum_{j=1}^n m_{kj} z_j^{s_j^{(i)}(p)} + q_k \right) \right] \right)_+ \\
&\qquad - \left(z_k^* - \varphi_k^{(1)} \omega_1 \left[\sum_{j=1}^n m_{kj} z_j^* + q_k \right] \right)_+ \\
&\quad\le \left(z_k^{s_k^{(i)}(p)} - z_k^* - \varphi_k^{(1)} \left[\gamma_1 \sum_{\substack{j<k \\ j \in J_i}} l_{kj}^{(i)} (z_j^{p+1/2,i} - z_j^{s_j^{(i)}(p)}) + \omega_1 \sum_{j=1}^n m_{kj} \left(z_j^{s_j^{(i)}(p)} - z_j^* \right) \right] \right)_+
\end{aligned}$$

and

$$\begin{aligned}
&z_k^{p+1,i} - z_k^* \\
&\quad= \left(z_k^{p+1/2,i} - \varphi_k^{(2)} \left[\gamma_2 \sum_{\substack{j>k \\ j \in J_i}} u_{kj}^{(i)} (z_j^{p+1,i} - z_j^{p+1/2,i}) + \omega_2 \left(\sum_{j=1}^n m_{kj} z_j^{p+1/2,i} + q_k \right) \right] \right)_+ \\
&\qquad - \left(z_k^* - \varphi_k^{(2)} \omega_2 \left[\sum_{j=1}^n m_{kj} z_j^* + q_k \right] \right)_+ \\
&\quad\le \left(z_k^{p+1/2,i} - z_k^* - \varphi_k^{(2)} \left[\gamma_2 \sum_{\substack{j>k \\ j \in J_i}} u_{kj}^{(i)} (z_j^{p+1,i} - z_j^{p+1/2,i}) + \omega_2 \sum_{j=1}^n m_{kj} \left(z_j^{p+1/2,i} - z_j^* \right) \right] \right)_+ .
\end{aligned}$$

Therefore, it holds for $k \in J_i(p)$ and $i \in \Lambda$ that

$$\begin{aligned}
&\left(z_k^{p+1/2,i} - z_k^* \right)_+ \\
&\quad\le \left(z_k^{s_k^{(i)}(p)} - z_k^* - \varphi_k^{(1)} \left[\gamma_1 \sum_{\substack{j<k \\ j \in J_i}} l_{kj}^{(i)} (z_j^{p+1/2,i} - z_j^{s_j^{(i)}(p)}) + \omega_1 \sum_{j=1}^n m_{kj} \left(z_j^{s_j^{(i)}(p)} - z_j^* \right) \right] \right)_+
\end{aligned}$$

and

$$\begin{aligned}
&\left(z_k^{p+1,i} - z_k^* \right)_+ \\
&\quad\le \left(z_k^{p+1/2,i} - z_k^* - \varphi_k^{(2)} \left[\gamma_2 \sum_{\substack{j>k \\ j \in J_i}} u_{kj}^{(i)} (z_j^{p+1,i} - z_j^{p+1/2,i}) + \omega_2 \sum_{j=1}^n m_{kj} \left(z_j^{p+1/2,i} - z_j^* \right) \right] \right)_+ .
\end{aligned}$$

Similarly, we can also obtain for $k \in J_i(p)$ and $i \in \Lambda$ that

$$\begin{aligned}
&\left(z_k^* - z_k^{p+1/2,i} \right)_+ \\
&\quad\le \left(z_k^* - z_k^{s_k^{(i)}(p)} - \varphi_k^{(1)} \left[\gamma_1 \sum_{\substack{j<k \\ j \in J_i}} l_{kj}^{(i)} (z_j^{s_j^{(i)}(p)} - z_j^{p+1/2,i}) + \omega_1 \sum_{j=1}^n m_{kj} \left(z_j^* - z_j^{s_j^{(i)}(p)} \right) \right] \right)_+
\end{aligned}$$

and

$$\begin{aligned}&\left(z_k^* - z_k^{p+1,i}\right)_+\\&\quad\leq \left(z_k^* - z_k^{p+1/2,i} - \varphi_k^{(2)}\left[\gamma_2 \sum_{\substack{j>k\\ j\in J_i}} u_{kj}^{(i)}(z_j^{p+1/2,i} - z_j^{p+1,i}) + \omega_2 \sum_{j=1}^{n} m_{kj}\left(z_j^* - z_j^{p+1/2,i}\right)\right]\right)_+.\end{aligned}$$

Hence, for $k \in J_i(p)$ and $i \in \Lambda$ there hold

$$\begin{aligned}&\left|z_k^{p+1/2,i} - z_k^*\right|\\&\quad= \left(z_k^{p+1/2,i} - z_k^*\right)_+ + \left(z_k^* - z_k^{p+1/2,i}\right)_+\\&\quad\leq \left|z_k^{s_k^{(i)}(p)} - z_k^* - \varphi_k^{(1)}\left[\gamma_1 \sum_{\substack{j<k\\ j\in J_i}} l_{kj}^{(i)}(z_j^{p+1/2,i} - z_j^{s_j^{(i)}(p)}) + \omega_1 \sum_{j=1}^{n} m_{kj}\left(z_j^{s_j^{(i)}(p)} - z_j^*\right)\right]\right|\end{aligned} \tag{4.2}$$

and

$$\begin{aligned}&\left|z_k^{p+1,i} - z_k^*\right|\\&\quad= \left(z_k^{p+1,i} - z_k^*\right)_+ + \left(z_k^* - z_k^{p+1,i}\right)_+\\&\quad\leq \left|z_k^{p+1/2,i} - z_k^* - \varphi_k^{(2)}\left[\gamma_2 \sum_{\substack{j>k\\ j\in J_i}} u_{kj}^{(i)}(z_j^{p+1,i} - z_j^{p+1/2,i}) + \omega_2 \sum_{j=1}^{n} m_{kj}\left(z_j^{p+1/2,i} - z_j^*\right)\right]\right|.\end{aligned} \tag{4.3}$$

From (4.2) and (4.3) we know that for $k \in J_i(p)$ and $i \in \Lambda$ the estimates

$$\begin{aligned}\left|z_k^{p+1/2,i} - z_k^*\right| &\leq \left|-\varphi_k^{(1)}\gamma_1 \sum_{\substack{j<k\\ j\in J_i}} l_{kj}^{(i)}(z_j^{p+1/2,i} - z_j^*)\right|\\&\quad+ \left|\left(z_k^{s_k^{(i)}(p)} - z_k^*\right) - \varphi_k^{(1)}\left[\omega_1 \sum_{j=1}^{n} m_{kj}\left(z_j^{s_j^{(i)}(p)} - z_j^*\right) - \gamma_1 \sum_{\substack{j<k\\ j\in J_i}} l_{kj}^{(i)}\left(z_j^{s_j^{(i)}(p)} - z_j^*\right)\right]\right|\end{aligned}$$

and

$$\begin{aligned}\left|z_k^{p+1,i} - z_k^*\right| &\leq \left|-\varphi_k^{(2)}\gamma_2 \sum_{\substack{j>k\\ j\in J_i}} u_{kj}^{(i)}(z_j^{p+1,i} - z_j^*)\right|\\&\quad+ \left|\left(z_k^{p+1/2,i} - z_k^*\right) - \varphi_k^{(2)}\left[\omega_2 \sum_{j=1}^{n} m_{kj}\left(z_j^{p+1/2,i} - z_j^*\right) - \gamma_2 \sum_{\substack{j>k\\ j\in J_i}} u_{kj}^{(i)}\left(z_j^{p+1/2,i} - z_j^*\right)\right]\right|\end{aligned}$$

hold. Therefore, we have

$$\begin{cases}\left|z_k^{p+1/2,i} - z_k^*\right| \leq e_k^T(I - \gamma_1\Phi_1|L_i|)^{-1}|I - \Phi_1(\omega_1 M - \gamma_1 L_i)||z^{s^{(i)}(p)} - z^*|,\\ \left|z_k^{p+1,i} - z_k^*\right| \leq e_k^T(I - \gamma_2\Phi_2|U_i|)^{-1}|I - \Phi_2(\omega_2 M - \gamma_2 U_i)||z_k^{p+1/2,i} - z_k^*|,\\ k \in J_i(p), \quad i \in \Lambda, \quad p \in N_0,\end{cases}$$

where e_k denotes the k-th unit basis vector in R^n, and $z^{s^{(i)}(p)} = \left(z_1^{s_1^{(i)}(p)}, \cdots, z_n^{s_n^{(i)}(p)}\right)^T$, $i \in \Lambda$. Represent by

$$\begin{cases} \mathcal{B}_i(\gamma_1,\omega_1) = I - \gamma_1\Phi_1|L_i|, & \mathcal{C}_i(\gamma_1,\omega_1) = |I - \Phi_1(\omega_1 M - \gamma_1 L_i)|, \\ \mathcal{B}'_i(\gamma_2,\omega_2) = I - \gamma_2\Phi_2|U_i|, & \mathcal{C}'_i(\gamma_2,\omega_2) = |I - \Phi_2(\omega_2 M - \gamma_2 U_i)|, \end{cases} \quad i \in \Lambda. \tag{4.4}$$

Then the above estimates can be briefly expressed as

$$\begin{cases} \left|z_k^{p+1/2,i} - z_k^*\right| \le e_k^T \mathcal{B}_i(\gamma_1,\omega_1)^{-1}\mathcal{C}_i(\gamma_1,\omega_1)|z^{s^{(i)}(p)} - z^*|, \\ \left|z_k^{p+1,i} - z_k^*\right| \le e_k^T \mathcal{B}'_i(\gamma_1,\omega_1)^{-1}\mathcal{C}'_i(\gamma_1,\omega_1)|z_k^{p+1/2,i} - z_k^*|, \\ k \in J_i(p), \quad i \in \Lambda, \quad p \in N_0. \end{cases} \tag{4.5}$$

Based on (4.4) and (4.5), we can prove the convergence of Method I and Method II under proper conditions.

Theorem 4.1. *Let $M = (m_{kj}) \in L(R^n)$ be an H-matrix with positive diagonal elements. Let $(D + L_i, D + U_i, W_i, E_i)(i = 1, 2, \cdots, \alpha)$ be its multisplitting with*

$$\langle M \rangle = D - |L_i| - |U_i| - |W_i| \equiv D - |B|, \quad i = 1, 2, \cdots, \alpha. \tag{4.6}$$

Then for any initial approximation $z^0 \in R^n$, the sequence $\{z^p\}_{p\in N_0}$ generated by Method I converges to the unique solution of the LCP provided

$$0 < \varphi_j^{(k)} \le 1/m_{jj}, \qquad j = 1, 2, \cdots, n; \quad k = 1, 2, \tag{4.7}$$

and the relaxation parameters γ_k and $\omega_k (k = 1, 2)$ satisfy

$$0 \le \gamma_k \le \omega_k, \quad 0 < \omega_k \le \frac{1}{\min\limits_{1\le j\le n}\{\varphi_j^{(k)} m_{jj}\}}, \qquad k = 1, 2. \tag{4.8}$$

Proof. From the definition of Method I, we easily see that

$$|z_k^{p+1} - z_k^*| \le \sum_{i\in N_k(p)} e_k^{(i)}|z_k^{p+1,i} - z_k^*| + \sum_{i\notin N_k(p)} e_k^{(i)}|z_k^p - z_k^*|, \quad k \in N. \tag{4.9}$$

On the other hand, from the proof of Theorem 4.1 in [3], we know that under our hypotheses there exist a positive vector $u \in R^n$ and two nonnegative constants $\theta_1, \theta_2 \in [0, 1)$ such that

$$\begin{cases} \mathcal{B}_i(\gamma_1,\omega_1)^{-1}\mathcal{C}_i(\gamma_1,\omega_1)u \le \theta_1 u, \\ \mathcal{B}'_i(\gamma_1,\omega_1)^{-1}\mathcal{C}'_i(\gamma_1,\omega_1)u \le \theta_2 u, \end{cases} \quad i \in \Lambda. \tag{4.10}$$

Let $\theta = \theta_1\theta_2$, and let $\delta > 0$ be such that $|z^0 - z^*| \le \delta u$. Then by applying (4.5), (4.9) and (4.10), we can directly verify the validity of the inequalities $|z^p - z^*| \le \delta u (p = 0, 1, 2, \cdots)$ through induction. Moreover, we can assert that there hold

$$|z^p - z^*| \le \Gamma^t \delta u, \qquad \forall p \ge m_t, \quad \forall t \in N_0, \tag{4.11a}$$

where

$$\Gamma = 1 - (1 - \theta)e_{\min}, \quad e_{\min} = \min\{e_k^{(i)} > 0 \mid k \in N, \quad i \in \Lambda\}. \tag{4.11b}$$

In fact, (4.11) is trivial when $t=0$. Suppose that (4.11) is true for some $t \geq 1$, we now prove that it is also true for $t+1$. From Lemma 4.1 we can easily get $|z^{s^{(i)}(p)}-z^*| \leq \Gamma^t \delta u (\forall i \in \Lambda)$. Thereby, (4.5) and (4.10) immediately give the estimates $|z_k^{p+1,i}-z_k^*| \leq \theta\Gamma^t\delta u_k$ $(\forall k \in N, \forall i \in \Lambda)$. Now, by making use of (4.9), we obtain for $\forall k \in N$ that

$$\begin{aligned} |z_k^{p+1}-z_k^*| &\leq \sum_{i\in N_k(p)} e_k^{(i)}\theta\Gamma^t\delta u_k + \sum_{i\notin N_k(p)} e_k^{(i)}|z_k^p - z_k^*| \\ &= \mathcal{I}_k^p\theta\Gamma^t\delta u_k + (1-\mathcal{I}_k^p)|z_k^p - z_k^*|. \end{aligned} \tag{4.12}$$

Moreover, because for $\forall k \in N$ and $\forall p \geq m_{t+1}$, from (4.12) and Lemma 4.2 we have

$$\begin{aligned} (1-\mathcal{P}_k^p)|z_k^{p+1}-z_k^*| &\leq (1-\mathcal{P}_k^p)\left[\mathcal{I}_k^p\theta\Gamma^t\delta u_k + (1-\mathcal{I}_k^p)|z_k^p - z_k^*|\right] \\ &= (1-\mathcal{P}_k^p)|z_k^p - z_k^*| \end{aligned} \tag{4.13}$$

and

$$\begin{aligned} \mathcal{P}_k^p|z_k^{p+1}-z_k^*| &\leq \mathcal{P}_k^p\left[\mathcal{I}_k^p\theta\Gamma^t\delta u_k + (1-\mathcal{I}_k^p)|z_k^p - z_k^*|\right] \\ &= \mathcal{I}_k^p\theta\Gamma^t\delta u_k + (\mathcal{P}_k^p - \mathcal{I}_k^p)|z_k^p - z_k^*| \\ &\leq (\mathcal{P}_k^p - (1-\theta)\mathcal{I}_k^p)\Gamma^t\delta u_k, \end{aligned} \tag{4.14}$$

where the last inequality is got by the induction hypothesis.

(4.13) and (4.14) immediately give the estimate

$$\begin{aligned} |z_k^{p+1}-z_k^*| &= \mathcal{P}_k^p|z_k^{p+1}-z_k^*| + (1-\mathcal{P}_k^p)|z_k^{p+1}-z_k^*| \\ &\leq (\mathcal{P}_k^p - (1-\theta)\mathcal{I}_k^p)\Gamma^t\delta u_k + (1-\mathcal{P}_k^p)|z_k^p - z_k^*|. \end{aligned}$$

Since for $\forall k \in N$ and $\forall p \in N_0$, $\mathcal{P}_k^p = 0$ if and only if $\mathcal{I}_k^p = 0$, by using Lemma 4.2(1)-(2) we see that $(\mathcal{P}_k^p - (1-\theta)\mathcal{I}_k^p) \leq \Gamma\mathcal{P}_k^p$ $(\forall k \in N,\ \forall p \in N_0)$. Therefore,

$$|z_k^{p+1}-z_k^*| \leq \mathcal{P}_k^p\Gamma^{t+1}\delta u_k + (1-\mathcal{P}_k^p)|z_k^p - z_k^*|, \qquad \forall k \in N, \quad \forall p \geq m_{t+1}. \tag{4.15}$$

Let $\tilde{p}_k = \max\{\tilde{p} \mid \mathcal{P}_k^{\tilde{p}} = 1, \tilde{p} = m_t, m_{t+1}, \cdots, p\}$. Because $\mathcal{Q}_k^{(t)}$ is a positive integer by Lemma 4.3(1), we know that the positive integer $\tilde{p}_k$ is well-defined. In addition, remembering that $\mathcal{P}_k^p \in \{0,1\}$ $(\forall k \in N,\ \forall p \in N_0)$, based on (4.15) we have

$$|z_k^{p+1}-z_k^*| \leq \mathcal{P}_k^{\tilde{p}_k}\Gamma^{t+1}\delta u_k + (1-\mathcal{P}_k^{\tilde{p}_k})|z_k^{\tilde{p}_k} - z_k^*| = \Gamma^{t+1}\delta u_k$$

for all $k \in N$ and for all $p \geq m_{t+1} - 1$. This shows that (4.11) is also valid for $t+1$. By induction, we have confirmed (4.11).

Because of $\Gamma \in [0,1)$, by (4.11) we finally get $|z^p - z^*| \leq \Gamma^t\delta u \to 0 (p \to \infty)$, or in other words, $z^p \to z^* (p \to \infty)$.

Theorem 4.2. *Let the conditions of Theorem 4.1 be satisfied. Then for any initial approximation $z^0 \in R^n$, the sequence $\{z^p\}_{p\in N_0}$ generated by Method II converges to the unique solution of the LCP provided $\varphi_j^{(k)} (j \in N; k = 1,2)$ satisfy (4.7), the relaxation parameters γ_k and $\omega_k (k=1,2)$ satisfy (4.8), and the extrapolation parameter β satisfies*

$$0 < \beta < \frac{2}{1+\sigma(\omega_1,\Phi_1)\sigma(\omega_2,\Phi_2)}, \tag{4.16}$$

where for $\Phi = diag(\varphi_1, \varphi_2, \cdots, \varphi_n)$,

$$\sigma(\omega,\Phi) = \max_{1\leq j\leq n}\{1 - \omega\varphi_j m_{jj}(1-\rho(D^{-1}|B|))\}.$$

Proof. In accordance with the definition of Method II, we easily see that

$$|z_k^{p+1} - z_k^*| \le \sum_{i \in N_k(p)} e_k^{(i)} |\overline{z}_k^{p+1,i} - z_k^*| + \sum_{i \notin N_k(p)} e_k^{(i)} |z_k^p - z_k^*|, \quad k \in N,$$

and

$$|\overline{z}_k^{p+1,i} - z_k^*| \le \beta |z_k^{p+1,i} - z_k^*| + |1-\beta||z_k^{s_k^{(i)}(p)} - z_k^*|, \qquad k \in J_i(p), \quad i \in \Lambda.$$

On the other hand, for any $\eta > 0$, let $J_\eta = D^{-1}|B| + \eta e e^T$ with $e = (1, 1, \cdots, 1)^T \in R^n$ and let $\rho_\eta = \rho(J_\eta) < 1$. Then from the proof of Theorem 4.3 in [3], we know that under our assumptions there exists a positive vector $u_\eta \in R^n$ such that

$$\begin{cases} \mathcal{B}_i(\gamma_1, \omega_1)^{-1} \mathcal{C}_i(\gamma_1, \omega_1) u_\eta \le [I - \omega_1 \Phi_1 D(1-\rho_\eta)] u_\eta := \theta_\eta(\omega_1, \Phi_1) u_\eta \\ \mathcal{B}'_i(\gamma_2, \omega_2)^{-1} \mathcal{C}'_i(\gamma_2, \omega_2) u_\eta \le [I - \omega_2 \Phi_2 D(1-\rho_\eta)] u_\eta := \theta'_\eta(\omega_2, \Phi_2) u_\eta \end{cases}$$

hold for $i = 1, 2, \cdots, \alpha$. Define

$$\sigma_\eta(\omega, \Phi) = \max_{1 \le j \le n} \{1 - \omega \varphi_j m_{jj}(1-\rho_\eta)\}.$$

Then the above estimates straightforwardly show that

$$\begin{cases} \mathcal{B}_i(\gamma_1, \omega_1)^{-1} \mathcal{C}_i(\gamma_1, \omega_1) u_\eta \le \sigma_\eta(\omega_1, \Phi_1) u_\eta, \\ \mathcal{B}'_i(\gamma_2, \omega_2)^{-1} \mathcal{C}'_i(\gamma_2, \omega_2) u_\eta \le \sigma_\eta(\omega_2, \Phi_2) u_\eta, \end{cases} \qquad i = 1, 2, \cdots, \alpha.$$

In addition, considering the restriction (4.16) on the extrapolation parameter β, we can take η small enough such that

$$|1-\beta| + \beta \sigma_\eta(\omega_1, \Phi_1) \sigma_\eta(\omega_2, \Phi_2) < 1.$$

Now, quite analogously to the proof of Theorem 4.1, we can deduce that the iterative sequence $\{z^p\}_{p \in N_0}$ generated by Method II converges to the unique solution $z^* \in R^n$ of the LCP.

From the above discussions we see that one reasonable and practical choice of the positive diagonal matrices $\Phi_k (k = 1, 2)$ in both Method I and Method II is $\Phi_1 = \Phi_2 = D^{-1}$. This naturally leads to two multi-parameter relaxed asynchronous parallel multisplitting methods, referred to Method III and Method IV, corresponding to Method I and Method II, respectively, for solving the LCP. Analogously to Theorem 4.1 and Theorem 4.2, we can discuss the convergence properties of the latter two methods. Since the convergence of these two concrete methods can be demonstrated in a similar way to the proofs of Theorem 4.1 and Theorem 4.2, we will only state the conclusions but omit their proofs.

Theorem 4.3. *Let the conditions of Theorem 4.1 be satisfied. Then for any initial guess $z^0 \in R^n$,*

(1) *the sequence $\{z^p\}_{p \in N_0}$ generated by Method III converges to the unique solution of the LCP provided the relaxation parameters γ_k and $\omega_k (k = 1, 2)$ satisfy*

$$0 \le \gamma_k \le \omega_k, \quad 0 < \omega_k \le \frac{2}{1 + \rho(D^{-1}|B|)}; \tag{4.17}$$

(2) *the sequence $\{z^p\}_{p \in N_0}$ generated by Method IV converges to the unique solution of the LCP provided the relaxation parameters γ_k and $\omega_k (k = 1, 2)$ satisfy (4.17), and the extrapolation parameter β satisfies $0 < \beta < \frac{2}{1 + \bar{\sigma}(\omega_1)\bar{\sigma}(\omega_2)}$, where $\bar{\sigma}(\omega) = |1-\omega| + \omega \rho(D^{-1}|B|) < 1$.*

5. Numerical Results

We consider the linear complementarity problem with the system matrix and given vector

$$M = \begin{pmatrix} R & -I & & & \\ -I & R & -I & & \\ & \ddots & \ddots & \ddots & \\ & & -I & R & -I \\ & & & -I & R \end{pmatrix} \in L(R^n), \qquad q = \begin{pmatrix} c \\ -c \\ \vdots \\ (-c)^{n-1} \\ (-c)^n \end{pmatrix} \in R^n,$$

respectively, where $R = tridiag(-1, 4, -1) \in L(R^{\tilde{n}})$, $I \in L(R^{\tilde{n}})$ is the identity matrix, $n = \tilde{n}^2$, and c is a parameter.

The test methods used in our numerical experiments are:

(a) the sequential relaxation methods in [6, 7, 14], including SOR, SSOR, AOR and SAOR;

(b) the synchronous parallel multisplitting relaxation methods in [2, 3], including MSOR, MSSOR, MAOR and MSAOR; as well as

(c) the new asynchronous parallel multisplitting relaxation methods, including AMSOR, AMSSOR, AMAOR and AMSAOR.

The parallel machine used in our computations is an SGI Power Challenge multiprocessor computer. It consists of four 75 MHz TFP 64-bit RISC processors. These CMOS processors each delivers a peak theoretical performance of 0.3 GFLOPS. The data cache size is 16 Kbytes.

In our computations, for $i = 1, 2, \cdots, \alpha$ we take $J_i = \{\tilde{n}_{i-1}\tilde{n}+1, \tilde{n}_{i-1}\tilde{n}+2, \cdots, \tilde{n}_{i+1}\tilde{n}\}$, with $\tilde{n}_i = Int\left(\frac{i\tilde{n}}{\alpha+1}\right)$, and

$$L_i = (\mathcal{L}_{kj}^{(i)}) \in L(R^n), \quad \mathcal{L}_{kj}^{(i)} = \begin{cases} m_{kj}, & \text{for } k, j \in J_i \text{ and } k > j, \\ 0, & \text{otherwise}, \end{cases} \qquad k, j \in N,$$

$$U_i = (\mathcal{U}_{kj}^{(i)}) \in L(R^n), \quad \mathcal{U}_{kj}^{(i)} = \begin{cases} m_{kj}, & \text{for } k, j \in J_i \text{ and } k < j, \\ 0, & \text{otherwise}, \end{cases} \qquad k, j \in N,$$

$$W_i = (\mathcal{W}_{kj}^{(i)}) \in L(R^n), \quad \mathcal{W}_{kj}^{(i)} = \begin{cases} 0, & \text{for } k = j, \\ 0, & \text{for } k, j \in J_i, \qquad k, j \in N, \\ m_{kj}, & \text{otherwise}, \end{cases}$$

$$E_i = diag(e_k^{(i)}) \in L(R^n), \quad e_k^{(i)} = \begin{cases} 1, & \text{for } 1 \le k \le \tilde{n}_1\tilde{n}, \quad i = 1, \\ 0.5, & \text{for } \tilde{n}_{i-1}\tilde{n}+1 \le k \le \tilde{n}_i\tilde{n}, \quad 2 \le i \le \alpha, \\ 0.5, & \text{for } \tilde{n}_i\tilde{n}+1 \le k \le \tilde{n}_{i+1}\tilde{n}, \quad 1 \le i \le \alpha-1, \\ 1, & \text{for } \tilde{n}_\alpha\tilde{n}+1 \le k \le n, \quad i = \alpha. \end{cases}$$

The LCP is solved by the aforementioned test methods corresponding to various problem sizes, right-hand sides, and relaxation parameters when the processor number α is respectively taken to be $\alpha = 2$, $\alpha = 3$ and $\alpha = 4$. From the numerical computations we see that in the sense of CPU time and the parallel efficiency, the asynchronous parallel multisplitting relaxation methods are superior to the corresponding synchronous parallel multisplitting relaxation methods, the multisplitting accelerated overrelaxation methods are superior to the corresponding multisplitting successive overrelaxation methods, and the multisplitting symmetric relaxation methods are superior to the corresponding multisplitting relaxation methods. In particular, the advantages of the AMAOR and AMSAOR methods over the AMSOR and AMSSOR methods, respectively, are that (i) when the latter ones diverge, the former ones can still converge; (ii) when the latter ones converge, the former ones converge faster with higher parallel efficiency; and (iii) the former ones are less sensitive to the relaxation parameters and they have larger convergence domains than the latter ones. Therefore, we can conclude that the new asynchronous parallel multisplitting relaxation methods have better numerical properties than both their corresponding synchronous and sequential alternatives.

For $n = 6400$, $c = 1.0$ and $\alpha = 2$, some of the numerical results are listed in Tables 1-6 and plotted in Figures 1-4. Moreover, the optimal CPU timings and the corresponding speed-ups of the AOR-type methods are listed in Table 7 for various problem sizes. All these computations are started from an initial vector having all components equal to 40.0, and terminated once the current iterations z^p obey

$$\frac{|(z^p)^T(Mz^p + q)|}{|(z^0)^T(Mz^0 + q)|} \le 10^{-7}.$$

Here and in the following, we use CPU to denote the CPU time required for an iteration to attain the above stopping criterions, ∞ to denote that an iteration does not satisfy the stopping criterions after 5000 iterations, and SP to denote the speed-up of a parallel synchronous or asynchronous multisplitting relaxation method, which is defined to be the ratio of the CPU times of the sequential relaxation method to the corresponding parallel multisplitting relaxation method. The numerical results for three and four processor cases are much similar to the two processor case.

Table 1. CPUs of SOR and SSOR methods

ω	1	1.1	1.2	1.3	1.4	1.5
SOR	50.70	41.48	33.85	27.30	21.72	16.89
SSOR	25.34	20.68	16.88	13.65	54.55	∞

Table 2. CPUs and SPs of MSOR and MSSOR methods

ω		1	1.1	1.2	1.3	1.4	1.5
MSOR	CPU	38.99	32.38	26.23	21.05	16.78	13.09
	SP	1.46	1.46	1.46	1.44	1.43	1.43
MSSOR	CPU	19.60	16.10	13.08	10.59	∞	∞
	SP	1.45	1.42	1.41	1.42	-	-

Table 3. CPUs and SPs of AMSOR and AMSSOR methods

ω		1	1.1	1.2	1.3	1.4	1.5
AMSOR	CPU	34.73	28.42	23.25	18.99	15.16	11.82
	SP	1.30	1.28	1.29	1.30	1.29	1.29
AMSSOR	CPU	17.51	14.56	11.84	9.60	∞	∞
	SP	1.29	1.28	1.29	1.29	-	-

Table 4. CPUs of AOR and SAOR methods

γ	1.1	1.2	1.4	1.6	1.8	1.9	1.95	1.97	1.99	2.1
ω	1.3	1.3	1.3	1.2	1.1	1.0	1.0	1.0	1.0	1.0
AOR	35.08	31.19	23.39	16.95	9.17	4.85	2.66	2.01	2.26	∞
SAOR	17.63	15.57	12.07	8.92	5.00	2.54	1.71	2.53	5.36	∞

Table 5. CPUs and SPs of MAOR and MSAOR methods

γ		1.1	1.2	1.4	1.6	1.8	1.9	1.95	1.97	1.99	2.1
ω		1.3	1.3	1.3	1.2	1.1	1.0	1.0	1.0	1.0	1.0
MAOR	CPU	27.14	24.08	18.12	13.14	7.25	4.01	2.16	1.91	1.99	∞
	SP	1.29	1.30	1.29	1.29	1.26	1.21	1.23	1.05	1.14	-
MSAOR	CPU	13.65	12.51	9.35	6.95	3.91	2.09	1.28	1.75	2.65	∞
	SP	1.29	1.24	1.29	1.28	1.28	1.21	1.33	1.44	2.02	-

Table 6. CPUs and SPs of AMAOR and AMSAOR methods

γ		1.1	1.2	1.4	1.6	1.8	1.9	1.95	1.97	1.99	2.1
ω		1.3	1.3	1.3	1.2	1.1	1.0	1.0	1.0	1.0	1.0
AMAOR	CPU	24.04	21.52	16.26	11.87	6.63	3.79	2.04	1.77	1.77	∞
	SP	1.46	1.45	1.44	1.43	1.38	1.28	1.30	1.14	1.28	-
AMSAOR	CPU	12.40	10.87	9.49	6.26	3.62	2.08	0.86	1.19	1.45	∞
	SP	1.42	1.43	1.27	1.42	1.38	1.22	1.99	2.13	3.69	-

Table 7. CPUs and SPs of the AOR-type methods

n		900	1600	2500	3600	4900	6400	8100	10000
AOR	CPU	0.11	0.28	0.55	0.97	1.51	2.27	3.29	4.60
SAOR	CPU	0.09	0.20	0.43	0.73	1.18	1.72	2.93	4.09
MAOR	CPU	0.08	0.17	0.34	0.58	0.95	1.44	2.10	2.94
	SP	1.38	1.65	1.62	1.67	1.59	1.58	1.57	1.56
MSAOR	CPU	0.04	0.09	0.14	0.30	0.55	0.91	1.38	2.05
	SP	2.25	2.22	3.07	2.43	2.15	1.89	2.12	2.00
AMAOR	CPU	0.07	0.15	0.31	0.55	0.92	1.30	1.99	2.74
	SP	1.57	1.87	1.77	1.76	1.64	1.75	1.65	1.68
AMSAOR	CPU	0.04	0.07	0.16	0.27	0.50	0.74	1.06	1.58
	SP	2.25	2.86	2.69	2.7	2.36	2.32	2.76	2.59

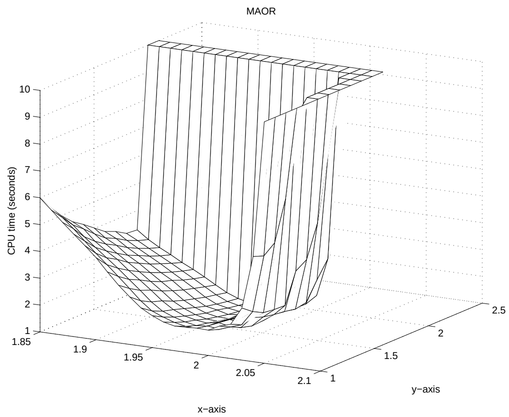

Fig. 1. The behaviour of MAOR method for the problem with $n = 6400$. The x and y axes are corresponding to γ and ω, respectively. The divergent points are represented in the graph by CPU time being 10.

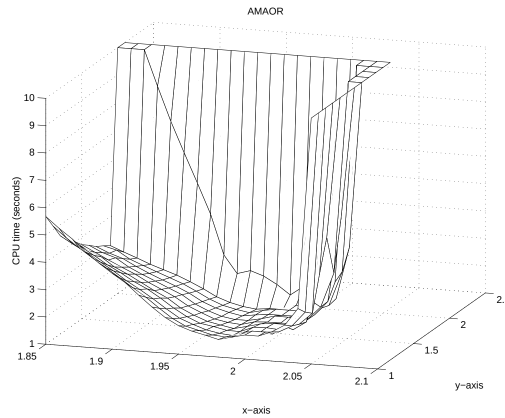

Fig. 2. The behaviour of AMAOR method for the problem with $n = 6400$. The x and y axes are corresponding to γ and ω, respectively. The divergent points are represented in the graph by CPU time being 10.

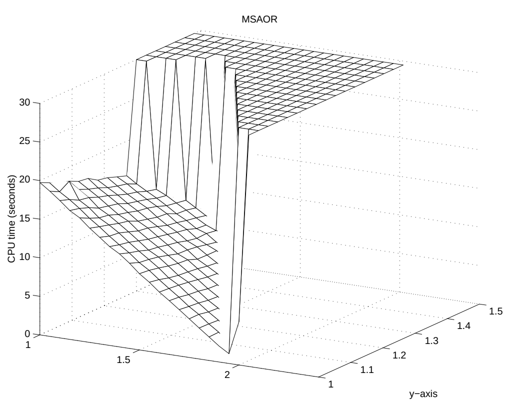

Fig. 3. The behaviour of MSAOR method for the problem with $n = 6400$. The x and y axes are corresponding to γ and ω, respectively. The divergent points are represented in the graph by CPU time being 30.

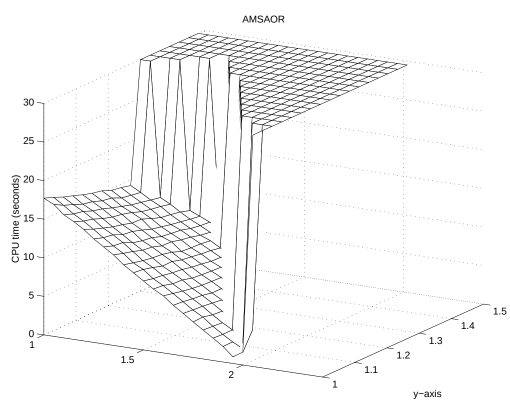

Fig. 4. The behaviour of AMSAOR method for the problem with $n = 6400$. The x and y axes are corresponding to γ and ω, respectively. The divergent points are represented in the graph by CPU time being 30.

References

[1] Z.Z. Bai, On the monotone convergence of the projected iteration methods for linear complementarity problem, *Numer. Math. J. Chinese Univ. (English Ser.),* **5** (1996), 228-233.

[2] Z.Z. Bai, D.J. Evans, Matrix multisplitting relaxation methods for linear complementarity problems, *Intern. J. Computer Math.,* **63** (1997), 309-326.

[3] Z.Z. Bai, D.J. Evans, D.R. Wang, A class of multi-parameter relaxed parallel multisplitting methods for large sparse linear complementarity problems, *Parallel Algorithms Appl.,* **11** (1997), 113-127.

[4] Z.Z. Bai, D.R. Wang, D.J. Evans, Models of asynchronous parallel matrix multisplitting relaxed iterations, *Parallel Comput.,* **21** (1995), 565-582.

[5] R.W. Cottle, G.B. Dantzig, Complementarity pivot theory of mathematical programming, *Linear Algebra Appl.,* **1** (1968), 103-125.

[6] R.W. Cottle, G.H. Golub, R.S. Sacher, On the solution of large structured linear complementarity problems: The block partitioned case, *Appl. Math. Optim.,* **4** (1978), 347-363.

[7] R.W. Cottle, R.S. Sacher, On the solution of large structured linear complementarity problems: The tridiagonal case, *Appl. Math. Optim.,* **4** (1977), 321-340.

[8] C.W. Cryer, The solution of a quadratic programming problem using systematic overrelaxation, *SIAM J. Control,* **9** (1971), 385-392.

[9] R. De Leone, O.L. Mangasarian, Asynchronous parallel successive overrelaxation for the symmetric linear complementarity problem, *Math. Programming,* **42** (1988), 347-361.

[10] D.J. Evans, D.R. Wang, Z.Z. Bai, Matrix multi-splitting multi-parameter relaxation methods, *Intern. J. Computer Math.,* **43** (1992), 173-188.

[11] A. Frommer, G. Mayer, Convergence of relaxed parallel multisplitting methods, *Linear Algebra Appl.,* **119** (1989), 141-152.

[12] G.H. Golub, C.F. Van Loan, Matrix Computations (The Third Edition), The Johns Hopkins University Press, Baltimore and London, 1996.

[13] C.E. Lemke, Bimatrix equilibrium points and mathematical programming, *Management Sci.,* **11** (1965), 681-689.

[14] O.L. Mangasarian, Solution of symmetric linear complementarity problems by iterative methods, *J. Optim. Theory Appl.,* **22** (1977), 465-485.

[15] O.L. Mangasarian, R. De Leone, Parallel successive overrelaxation methods for the symmetric linear complementarity problems, *J. Optim. Theory Appl.,* **54** (1987), 437-446.