Journal of Computational Mathematics, Vol.16, No.3, 1998, 221–238.

A CLASS OF ASYNCHRONOUS MATRIX MULTI-SPLITTING MULTI-PARAMETER RELAXATION ITERATIONS*[1)]

Zhong-zhi Bai
(*ICMSEC, Chinese Academy of Sciences, Beijing 100080, China*)
De-ren Wang
(*Department of Mathematics, Shanghai University* (*Jiading Campus*)
Shanghai 201800, China)
D.J. Evans
(*Department of Computer Studies, Loughborough University of Technology Loughborough, U.K.*)

Abstract

A class of asynchronous matrix multi-splitting multi-parameter relaxation methods, including the asynchronous matrix multisplitting SAOR, SSOR and SGS methods as well as the known asynchronous matrix multisplitting AOR, SOR and GS methods, etc., is proposed for solving the large sparse systems of linear equations by making use of the principle of sufficiently using the delayed information. These new methods can greatly execute the parallel computational efficiency of the MIMD-systems, and are shown to be convergent when the coefficient matrices are H-matrices. Moreover, necessary and sufficient conditions ensuring the convergence of these methods are concluded for the case that the coefficient matrices are L-matrices.

Key words: System of linear equations, asynchronous iteration, matrix multisplitting, relaxation, convergence.

1. Introduction

Multisplitting methods for getting the solution of large sparse system of linear equations

$$Ax = b, \quad A = (a_{mj}) \in L(R^n) \ \text{ nonsingular}, \quad x = (x_m), b = (b_m) \in R^n \tag{1.1}$$

are efficient parallel iterative methods which are based on several splittings of the coefficient matrix $A \in L(R^n)$. Following [1] there has been bounteous literature (see [2-11], [14-29] and references therein) on both synchronous and asynchronous parallel iterative methods in the sense of matrix multisplitting.

In this paper, based on the more recent works of [8-9] and by simultaneously taking into account of both the advantages of the matrix multisplitting and the concrete characterizations of the high-speed MIMD-systems, we further propose a class

* Received January 23, 1995.
1) Project 19601036 supported by the National Natural Science Foundation of China.

of asynchronous matrix multisplitting unsymmetric AOR(UAOR) methods, the multi-parameter extensions of the methods given in [8] and [9], for solving the system of linear equations (1.1). These methods, besides being able to execute greatly the parallel computational efficiency of the MIMD-systems, can cover a series of practical asynchronous matrix multisplitting relaxed methods such as the novel asynchronous matrix multisplitting symmetric AOR(SAOR), unsymmetric SOR(USOR), symmetric SOR(SSOR) and symmetric Gauss-Seidel(SGS) methods as well as the known asynchronous matrix multisplitting AOR method in [8] and [9], etc., and therefore, they have certain generalities. Moreover, speedy convergence rates can be attained with suitably adjusting the relaxation parameters included in the new methods. Through some numerical results, we practically confirm that these new methods are really of distinct superiority. The convergence theories about the new methods are demonstrated in detail under more practical conditions, that is, the coefficient matrix $A \in L(R^n)$ is an H-matrix, or an L-matrix.

2. Asynchronous matrix multisplitting UAOR methods

We first split the number set $\{1, 2, \cdots, n\}$ into $\alpha(\alpha \le n$, an integer) nonempty subsets J_i $(i = 1, 2, \cdots, \alpha)$, i.e., $J_i \subseteq \{1, 2, \cdots, n\}$ $(i = 1, 2, \cdots, \alpha)$ and $\bigcup_{i=1}^{\alpha} J_i = \{1, 2, \cdots, n\}$, where there may be overlappings among these J_i $(i = 1, 2, \cdots, \alpha)$.

For a nonsingular matrix $A = (a_{mj}) \in L(R^n)$, define matrices

$$\begin{cases} D = diag(A), \qquad \det(D) \neq 0 \\ L_i = (\mathcal{L}_{mj}^{(i)}), \quad \mathcal{L}_{mj}^{(i)} = \begin{cases} l_{mj}^{(i)}, & \text{if } j < m \text{ and } m, j \in J_i \\ 0, & \text{otherwise,} \end{cases} \\ U_i = (\mathcal{U}_{mj}^{(i)}), \quad \mathcal{U}_{mj}^{(i)} = \begin{cases} u_{mj}^{(i)}, & \text{if } j > m \text{ and } m, j \in J_i \\ 0, & \text{otherwise} \end{cases} \\ W_i = (\mathcal{W}_{mj}^{(i)}), \quad \mathcal{W}_{mj}^{(i)} = \begin{cases} 0, & \text{if } m = j \\ w_{mj}^{(i)}, & \text{otherwise} \end{cases} \\ m, j = 1, 2, \cdots, n; \quad i = 1, 2, \cdots, \alpha \end{cases}$$

such that

$$A = D - L_i - U_i - W_i, \quad i = 1, 2, \cdots, \alpha. \tag{2.1}$$

Obviously, for $i = 1, 2, \cdots, \alpha$, $L_i, U_i \in L(R^n)$ are strictly lower triangular and strictly upper triangular, respectively, while $W_i \in L(R^n)$ are zero-diagonal.

Additionally, introduce nonnegative diagonal matrices $E_i \in L(R^n)$ $(i = 1, 2, \cdots, \alpha)$,

$$E_i = diag(e_1^{(i)}, e_2^{(i)}, \cdots, e_n^{(i)}), \quad e_m^{(i)} = \begin{cases} e_m^{(i)} \ge 0, & \text{if } m \in J_i \\ 0, & \text{otherwise,} \end{cases} \tag{2.2}$$

such that $\sum_{i=1}^{\alpha} E_i = I(I \in L(R^n)$ is the identity matrix). This type of matrices $E_i(i = 1, 2, \cdots, \alpha)$ are called weighting matrices.

The splittings (2.1) of the matrix $A \in L(R^n)$ with the weighting matrices (2.2) is called a multisplitting of the matrix $A \in L(R^n)$, and is simply denoted as $(D-L_i, D-U_i, W_i, E_i)$ $(i = 1, 2, \cdots, \alpha)$.

Assume that the MIMD-system referred by us is constructed by α CPU's, for the requirements of establishing the new asynchronous matrix multisplitting UAOR methods for the system of linear equations (1.1), we now introduce the following important notations:

(i) for $\forall i \in \{1, 2, \cdots, \alpha\}, \forall p \in N_0 := \{0, 1, 2, \cdots\}, J^{(i)} = \{J_i(p)\}_{p \in N_0}$ is used to denote a subset (may be empty set $\emptyset$) sequence of the set J_i;

(ii) for $\forall m \in \{1, 2, \cdots, n\}, \forall p \in N_0, N_m(p) := \{i \mid m \in J_i(p), i = 1, 2, \cdots, \alpha\}$ is a subset of the set $\{1, 2, \cdots, \alpha\}$;

(iii) for $\forall i \in \{1, 2, \cdots, \alpha\}, S^{(i)} = \{s_1^{(i)}(p), s_2^{(i)}(p), \cdots, s_n^{(i)}(p)\}_{p \in N_0}$ is n infinite sequences of nonnegative integers.

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

(a) for $\forall i \in \{1, 2, \cdots, \alpha\}, \forall m \in \{1, 2, \cdots, n\}$, the set $\{p \in N_0 \mid m \in J_i(p)\}$ is infinite;

(b) for $\forall p \in N_0$, there holds $\bigcup\limits_{i=1}^{\alpha} J_i(p) \neq \emptyset$;

(c) for $\forall i \in \{1, 2, \cdots, \alpha\}, \forall m \in \{1, 2, \cdots, n\}, \forall p \in N_0$, there holds $s_m^{(i)}(p) \leq p$;

(d) for $\forall i \in \{1, 2, \cdots, \alpha\}, \forall m \in \{1, 2, \cdots, n\}$, there holds

$$\lim_{p \to \infty} s_m^{(i)}(p) = \infty.$$

For $\forall p \in N_0$, once we define $s(p) = \min\limits_{\substack{1 \leq m \leq n \\ 1 \leq i \leq \alpha}} s_m^{(i)}(p)$, there hold $s(p) \leq p$ and $\lim\limits_{p \to \infty} s(p) = \infty$.

With the above preparations, we can now set up the asynchronous matrix multisplitting UAOR method for solving the large sparse system of linear equations (1.1) as follows:

Method I: Given initial guess $x^0 \in R^n$ and suppose that we have got the approximations $x^0, x^1, \cdots, x^p$ of the solution $x^* = (x_1^*, x_2^*, \cdots, x_n^*)^T$ of the system of linear equations (1.1). Then the $(p+1)$st approximation $x^{p+1} = (x_1^{p+1}, x_2^{p+1}, \cdots, x_n^{p+1})^T$ of x^* can be calculated through the following processes:

$$\begin{aligned} x_m^{p+1/2,i} =& \frac{1}{a_{mm}} \Bigg\{ r_1 \sum_{\substack{j<m \\ j \in J_i(p)}} l_{mj}^{(i)} x_j^{p+1/2,i} + (\omega_1 - r_1) \sum_{\substack{j<m \\ j \in J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \\ &+ \omega_1 \Bigg(\sum_{\substack{j>m \\ j \in J_i}} u_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum_{j \neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum_{\substack{j<m \\ j \in J_i \setminus J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + b_m \Bigg) \Bigg\} \\ &+ (1 - \omega_1) x_m^{s_m^{(i)}(p)}, \qquad m \in J_i(p) \end{aligned}$$

$$
\begin{aligned}
x_m^{p+1,i} =& \frac{1}{a_{mm}}\Bigg\{ r_2 \sum_{\substack{j>m\\ j\in J_i(p)}} u_{mj}^{(i)} x_j^{p+1,i} + (\omega_2 - r_2) \sum_{\substack{j>m\\ j\in J_i(p)}} u_{mj}^{(i)} x_j^{p+1/2,i} \\
&+ \omega_2 \Bigg(\sum_{\substack{j<m\\ j\in J_i(p)}} l_{mj}^{(i)} x_j^{p+1/2,i} + \sum_{j\neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \Bigg) \\
&+ \omega_2 \Bigg(\sum_{\substack{j<m\\ j\in J_i\setminus J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum_{\substack{j>m\\ j\in J_i\setminus J_i(p)}} u_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + b_m \Bigg) \Bigg\} \\
&+ (1-\omega_2) x_m^{p+1/2,i}, \qquad m \in J_i(p) \\
x_m^{p+1} =& \sum_{i\in N_m(p)} e_m^{(i)} x_m^{p+1,i} + \sum_{i\notin N_m(p)} e_m^{(i)} x_m^p.
\end{aligned}
\tag{2.3}
$$

Evidently, with different choices of the relaxation parameters $r_1, r_2, \omega_1, \omega_2$ in Method I, many special but very practical asynchronous matrix multisplitting relaxed methods can be obtained. For example, corresponding $(r_1, r_2, \omega_1, \omega_2)$ to be (r, r, ω, ω), $(\omega, \bar{\omega}, \omega, \bar{\omega})$, $(\omega, \omega, \omega, \omega)$, $(0, 0, 1, 1)$ and $(r, 0, \omega, 0)$, Method I automatically reduces to the asynchronous matrix multisplitting SAOR, USOR, SSOR, SGS and AOR[8,9] methods, respectively. In this manner, Method I really forms an extensive sequence of asynchronous matrix multisplitting relaxed methods, which therefore makes it much more flexible and convenient in the concrete applications.

In addition, note that for each $m \in J_i(p)$, $i \in \{1, 2, \cdots, \alpha\}$, the terms

$$
\sum_{j\neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)}, \qquad \sum_{\substack{j<m\\ j\in J_i\setminus J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)}, \qquad \sum_{\substack{j>m\\ j\in J_i\setminus J_i(p)}} u_{mj}^{(i)} x_j^{s_j^{(i)}(p)}
$$

need only be computed once in (2.3). However, no additional savings will, in general, be possible when performing Method I. This is in contrast to the usual UAOR method, which can be implemented in such a manner that apart from the first half step it requires almost the same computational work as the AOR method.

On the other hand, when

$$
J_i(p) = J_i, s_m^{(i)}(p) = p, \ \forall m \in \{1, 2, \cdots, n\}, \ \forall i \in \{1, 2, \cdots, \alpha\}, \ \forall p \in N_0,
$$

Method I naturally reduces to an efficient and practical variant of the known synchronous parallel matrix multisplitting UAOR method introduced in [4] for the system of linear equations (1.1), while

$$
\begin{cases}
J_i = \{1, 2, \cdots, n\} \\
(J_i(p) = J_i) \bigvee (J_i(p) = \emptyset) = True, \quad s_m^{(i)}(p) = s_i(p) \in N_0 \\
\forall m \in \{1, 2, \cdots, n\}, \quad \forall i \in \{1, 2, \cdots, \alpha\}, \quad \forall p \in N_0,
\end{cases}
$$

Method I automatically turns to an efficient and practical variant of the existed asynchronous parallel matrix multisplitting unsymmetric AOR method studied in [7].

If we supplement

$$x_m^{p+1,i} = x_m^{p+1/2,i} \equiv x_m^{s_m^{(i)}(p)}, \ m \in J_i \setminus J_i(p), \ i = 1, 2, \cdots, \alpha,$$

from (2.3) we have

$$\begin{cases} a_{mm} x_m^{p+1/2,i} - r_1 \sum\limits_{\substack{j<m \\ j \in J_i}} l_{mj}^{(i)} x_j^{p+1/2,i} = (1-\omega_1) a_{mm} x_m^{s_m^{(i)}(p)} + (\omega_1 - r_1) \sum\limits_{\substack{j<m \\ j \in J_i}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \\ \qquad + \omega_1 \Bigg(\sum\limits_{\substack{j>m \\ j \in J_i}} u_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum\limits_{j \neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \Bigg) + \omega_1 b_m, \quad m \in J_i(p) \\ a_{mm} x_m^{p+1,i} - r_2 \sum\limits_{\substack{j>m \\ j \in J_i}} u_{mj}^{(i)} x_j^{p+1,i} = (1-\omega_2) a_{mm} x_m^{p+1/2,i} + (\omega_2 - r_2) \sum\limits_{\substack{j>m \\ j \in J_i}} u_{mj}^{(i)} x_j^{p+1/2,i} \\ \qquad + \omega_2 \Bigg(\sum\limits_{\substack{j<m \\ j \in J_i}} l_{mj}^{(i)} x_j^{p+1/2,i} + \sum\limits_{j \neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \Bigg) + \omega_2 b_m, \quad m \in J_i(p), \end{cases}$$

or equivalently,

$$\begin{cases} x_m^{p+1/2,i} = e_m^T (D - r_1 L_i)^{-1} \{ [(1-\omega_1) D + (\omega_1 - r_1) L_i + \omega_1 (U_i + W_i)] x^{s^{(i)}(p)} + \omega_1 b \} \\ x_m^{p+1,i} = e_m^T (D - r_2 U_i)^{-1} \{ [(1-\omega_2) D + (\omega_2 - r_2) U_i + \omega_2 L_i] x_m^{p+\frac{1}{2},i} + \omega_2 W_i x^{s^{(i)}(p)} + \omega_2 b \} \\ m \in J_i(p), \quad m = 1, 2, \cdots, n; \quad i = 1, 2, \cdots, \alpha, \end{cases}$$

where

$$x^{s^{(i)}(p)} = (x_1^{s_1^{(i)}(p)}, x_2^{s_2^{(i)}(p)}, \cdots, x_n^{s_n^{(i)}(p)})^T, \quad i = 1, 2, \cdots, \alpha, \quad \forall p \in N_0,$$

$x_m^{p+1/2,i} (m \notin J_i)$ can be evaluated arbitrarily, and e_m denotes the m-th unit vector in R^n.

Now, through combining the above two relations and by considering the structures of the matrices $L_i, U_i (i = 1, 2, \cdots, \alpha)$, Method I can be immediately expressed as the following succint form:

$$\begin{cases} x_m^{p+1} = \sum\limits_{i \in N_m(p)} e_m^{(i)} e_m^T \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2) x^{s^{(i)}(p)} + \sum\limits_{i \notin N_m(p)} e_m^{(i)} x_m^p \\ \qquad\qquad + \sum\limits_{i \in N_m(p)} e_m^{(i)} e_m^T b^{(i)}(r_1, r_2, \omega_1, \omega_2) \\ m = 1, 2, \cdots, n, \end{cases} \tag{2.4}$$

where for $i = 1, 2, \cdots, \alpha$,

$$\begin{cases} \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2) = (D - r_2 U_i)^{-1} \\ \qquad \cdot \{ [(1-\omega_2) D + (\omega_2 - r_2) U_i + \omega_2 L_i] \mathcal{L}^{(i)}(r_1, \omega_1) + \omega_2 W_i \} \\ b^{(i)}(r_1, r_2, \omega_1, \omega_2) = (D - r_2 U_i)^{-1} \\ \qquad \cdot \{ [(1-\omega_2) D + (\omega_2 - r_2) U_i + \omega_2 L_i] (D - r_1 L_i)^{-1} \omega_1 b + \omega_2 b \} \\ \mathcal{L}^{(i)}(r_1, \omega_1) = (D - r_1 L_i)^{-1} [(1-\omega_1) D + (\omega_1 - r_1) L_i + \omega_1 (U_i + W_i)]. \end{cases} \tag{2.5}$$

Furthermore, if we extrapolate Method I by another parameter $\beta > 0$, it can be improved as the following relaxed asynchronous matrix multisplitting UAOR method.

Method II: Given initial guess $x^0 \in R^n$ and suppose that we have got the approximations $x^0, x^1, \cdots, x^p$ of the solution $x^* = (x_1^*, x_2^*, \cdots, x_n^*)^T$ of the system of linear equations (1.1), then the $(p+1)$st approximation $x^{p+1} = (x_1^{p+1}, x_2^{p+1}, \cdots, x_n^{p+1})^T$ of x^* can be calculated through the following processes:

$$\begin{aligned} x_m^{p+1/2,i} =& \frac{1}{a_{mm}} \Big\{ r_1 \sum_{\substack{j<m \\ j \in J_i(p)}} l_{mj}^{(i)} x_j^{p+1/2,i} + (\omega_1 - r_1) \sum_{\substack{j<m \\ j \in J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \\ & + \omega_1 \Big(\sum_{\substack{j>m \\ j \in J_i}} u_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum_{j \neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum_{\substack{j<m \\ j \in J_i \setminus J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + b_m \Big) \Big\} \\ & + (1-\omega_1) x_m^{s_m^{(i)}(p)}, \qquad m \in J_i(p) \\ x_m^{p+1,i} =& \frac{1}{a_{mm}} \Big\{ r_2 \sum_{\substack{j>m \\ j \in J_i(p)}} u_{mj}^{(i)} x_j^{p+1,i} + (\omega_2 - r_2) \sum_{\substack{j>m \\ j \in J_i(p)}} u_{mj}^{(i)} x_j^{p+1/2,i} \\ & + \omega_2 \Big(\sum_{\substack{j<m \\ j \in J_i(p)}} l_{mj}^{(i)} x_j^{p+1/2,i} + \sum_{j \neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \Big) \\ & + \omega_2 \Big(\sum_{\substack{j<m \\ j \in J_i \setminus J_i(p)}} l_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + \sum_{\substack{j>m \\ j \in J_i \setminus J_i(p)}} u_{mj}^{(i)} x_j^{s_j^{(i)}(p)} + b_m \Big) \Big\} \\ & + (1-\omega_2) x_m^{p+1/2,i}, \qquad m \in J_i(p) \\ x_m^{p+1} =& \beta \Big[\sum_{i \in N_m(p)} e_m^{(i)} x_m^{p+1,i} + \sum_{i \notin N_m(p)} e_m^{(i)} x_m^p \Big] + (1-\beta) x_m^p. \end{aligned}$$

Factually, Method II carries on all the properties of Method I stated previously. However, since it includes one more arbitrary parameter than Method I, corresponding to a series of special choices of these relaxation parameters $r_1, r_2, \omega_1, \omega_2$ and β involved in it, another practical and efficient sequence of asynchronous matrix multisplitting relaxed methods can be similarly yielded, too. For the length of this paper, we will not write them here one by one.

Additionally, analogous to Method I, Method II can also be simply written as

$$\begin{aligned} x_m^{p+1} = & \sum_{i \in N_m(p)} e_m^{(i)} e_m^T \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta) x^{s^{(i)}(p)} + \sum_{i \notin N_m(p)} e_m^{(i)} x_m^p \\ & + \sum_{i \in N_m(p)} e_m^{(i)} e_m^T b^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta), \quad m = 1, 2, \cdots, n, \end{aligned} \tag{2.6}$$

where

$$\begin{cases} \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta) = \beta \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2) + (1-\beta) I, \\ b^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta) = \beta b^{(i)}(r_1, r_2, \omega_1, \omega_2), \end{cases} \quad i = 1, 2, \cdots, \alpha \tag{2.7}$$

with $\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)$ and $b^{(i)}(r_1, r_2, \omega_1, \omega_2)(i = 1, 2, \cdots, \alpha)$ being defined by (2.5).

3. Convergence theories

The partial orderings $\leq$, $<$ in R^n and $L(R^n)$ are introduced according to the elements. $\rho(\bullet)$ and $< \bullet >$ are used to denote the spectral radius and the comparison matrix of the corresponding matrices, respectively, and $|\bullet|$ the absolute value of either a vector or a matrix. Also, we will carry on the concepts as well as the essential conclusions used in $[2--4, 6--9]$.

In the following, we assert a general criterion about the convergence of the asynchronous matrix multisplitting iterative methods, which is elementary for the subsequent discussion and will be proved in section 5.

Lemma 3.1. *Let $H_i \in L(R^n)$ $(i = 1, 2, \cdots, \alpha)$ be nonnegative matrices, $E_i (i = 1, 2, \cdots, \alpha)$ be weighting matrices, and the sequence $\{\epsilon^p\}_{p \in N_0}$ be defined by*

$$\begin{cases} \epsilon^p = (\epsilon_1^p, \epsilon_2^p, \cdots, \epsilon_n^p)^T \\ \epsilon_m^{p+1} = \sum\limits_{i \in N_m(p)} e_m^{(i)} e_m^T H_i \epsilon^{s^{(i)}(p)} + \sum\limits_{i \notin N_m(p)} e_m^{(i)} \epsilon_m^p \\ m = 1, 2, \cdots, n, \quad \forall p \in N_0 \end{cases}$$

with

$$\epsilon^{s^{(i)}(p)} = (\epsilon_1^{s_1^{(i)}(p)}, \epsilon_2^{s_2^{(i)}(p)}, \cdots, \epsilon_n^{s_n^{(i)}(p)})^T, \quad i = 1, 2, \cdots, \alpha, \quad \forall p \in N_0.$$

Then, (i) $\lim\limits_{p \to \infty} \epsilon^p = 0$ for any $\epsilon^0 \in R^n$ provided there exist a nonnegative number $\sigma \in [0, 1)$ and a positive vector $v \in R^n$ such that $H_i v \leq \sigma v (i = 1, 2, \cdots, \alpha)$;

(ii) $\underline{\lim}_{p \to \infty} \epsilon^p \neq 0$ for some $\epsilon^0 \in R^n$ provided there exist a positive number $\theta \in [1, \infty)$ and a nonnegative nonzero vector $u \in R^n$ such that $H_i u \geq \theta u (i = 1, 2, \cdots, \alpha)$.

Lemma 3.1 directly implies the following conclusion.

Lemma 3.2. *Let $H_i \in L(R^n)$ $(i = 1, 2, \cdots, \alpha)$ be nonnegative matrices, $E_i (i = 1, 2, \cdots, \alpha)$ be weighting matrices, and the sequence $\{\epsilon^p\}_{p \in N_0}$ be defined by*

$$\epsilon^{p+1} = \sum_{i \in J(p)} E_i H_i \epsilon^{s_i(p)} + \sum_{i \notin J(p)} E_i \epsilon^p, \qquad p = 0, 1, 2, \cdots$$

with $\{J(p)\}_{p \in N_0}$ be a subset sequence of the set $\{1, 2, \cdots, \alpha\}$ and $s_i(p) \in N_0 (i = 1, 2, \cdots, \alpha; p \in N_0)$. If there hold

(1) *the set $\{p \in N_0 \mid i \in J(p)\}$ is infinite;*

(2) *for $\forall p \in N_0$, for $\forall i \in \{1, 2, \cdots, \alpha\}$, $s_i(p) \leq p$; and*

(3) *for $\forall i \in \{1, 2, \cdots, \alpha\}$, $\lim\limits_{p \to \infty} s_i(p) = \infty$,*

then, (i) $\lim\limits_{p \to \infty} \epsilon^p = 0$ for any $\epsilon^0 \in R^n$ provided there exist a nonnegative number $\sigma \in [0, 1)$ and a positive vector $v \in R^n$ such that $H_i v \leq \sigma v (i = 1, 2, \cdots, \alpha)$;

(ii) $\underline{\lim}_{p \to \infty} \epsilon^p \neq 0$ for some $\epsilon^0 \in R^n$ provided there exist a positive number $\theta \in [1, \infty)$ and a nonnegative nonzero vector $u \in R^n$ such that $H_i u \geq \theta u (i = 1, 2, \cdots, \alpha)$.

Proof. For $\forall i \in \{1, 2, \cdots, \alpha\}$ and $\forall p \in N_0$, if we take $N_m(p) \equiv J(p)$, $s_m^{(i)}(p) \equiv s_i(p)$, $m = 1, 2, \cdots, n$, in Lemma 3.1, we can immediately know that the conclusions what we are proving are true.

It is deserved to mention that Lemma 3.1 also implies sufficient conditions ensuring the convergence as well as the divergence of all the known synchronous and asynchronous parallel multisplitting iteration methods[1−9,15,18,21−29] for solving the large sparse system of linear equations (1.1). For the length of the paper, we will not give the concrete descriptions here.

Let $x^* = (x_1^*, x_2^*, \cdots, x_n^*)^T \in R^n$ be the unique solution of the system of linear equations (1.1) and define

$$\begin{cases} \varepsilon_m^{s_m^{(i)}(p)} = x_m^{s_m^{(i)}(p)} - x_m^*, \quad \varepsilon^{s^{(i)}(p)} = (\varepsilon_1^{s_1^{(i)}(p)}, \varepsilon_2^{s_2^{(i)}(p)}, \cdots, \varepsilon_n^{s_n^{(i)}(p)})^T \\ \varepsilon_m^{p+1/2,i} = x_m^{p+1/2,i} - x_m^*, \quad \varepsilon_m^{p+1,i} = x_m^{p+1,i} - x_m^*, \quad m \in J_i \\ \varepsilon_m^{p+1} = x_m^{p+1} - x_m^* \\ m = 1, 2, \cdots, n; \quad i = 1, 2, \cdots, \alpha; \quad \forall p \in N_0. \end{cases}$$

Then for all $m = 1, 2, \cdots, n$, it is easy to obtain from (2.4) that for Method I there hold

$$\varepsilon_m^{p+1} = \sum_{i \in N_m(p)} e_m^{(i)} e_m^T \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2) \varepsilon^{s^{(i)}(p)} + \sum_{i \notin N_m(p)} e_m^{(i)} \varepsilon_m^p, \tag{3.1}$$

while for Method II, from (2.6) there hold

$$\varepsilon_m^{p+1} = \sum_{i \in N_m(p)} e_m^{(i)} e_m^T \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta) \varepsilon^{s^{(i)}(p)} + \sum_{i \notin N_m(p)} e_m^{(i)} \varepsilon_m^p. \tag{3.2}$$

Now, we are ready to set up the convergence theories for Method I and Method II.

Theorem 3.1. *Let $A \in L(R^n)$ be an H-matrix, $(D - L_i, D - U_i, W_i, E_i)$ $(i = 1, 2, \cdots, \alpha)$ be a multisplitting of it with $\langle A \rangle = |D| - |L_i| - |U_i| - |W_i| \equiv |D| - |B|$ $(i = 1, 2, \cdots, \alpha)$. Then,*

(i) the sequence $\{x^p\}_{p \in N_0}$ generated by Method I converges to the unique solution $x^* \in R^n$ of the system of linear equations (1.1) *for any starting vector $x^0 \in R^n$ provided the relaxation parameters r_1, r_2 and ω_1, ω_2 are within the region given by*

$$0 \le r_m \le \omega_m, \quad 0 \le \omega_m < 2/(1 + \rho(|D|^{-1}|B|)) \quad (\omega_1 + \omega_2 > 0), \quad m = 1, 2; \tag{3.3}$$

(ii) the sequence $\{x^p\}_{p \in N_0}$ generated by Method II converges to the unique solution $x^* \in R^n$ of the system of linear equations (1.1) *for any starting vector $x^0 \in R^n$ provided the relaxation parameters $r_1, r_2, \omega_1, \omega_2$ and β are within the region given by*

$$\begin{cases} 0 \le r_m \le \omega_m, \quad 0 \le \omega_m < 2/(1 + \rho(|D|^{-1}|B|)) \quad (\omega_1 + \omega_2 > 0), \quad m = 1, 2 \\ 0 < \beta < 2/(1 + \rho(\mathcal{H}(\omega_1, \omega_2))), \end{cases} \tag{3.4}$$

where

$$\mathcal{H}(\omega_1, \omega_2) = \begin{cases} |1 - \omega_1| I + \omega_1 |D|^{-1}|B|, & \textit{if } \omega_2 = 0 \\ |1 - \omega_2| I + \omega_2 |D|^{-1}|B|, & \textit{if } \omega_2 \ne 0. \end{cases} \tag{3.5}$$

Proof. We first prove (i). Because $A \in L(R^n)$ is an H-matrix, we known that $\rho(|D|^{-1}|B|) < 1$. For any $\varepsilon > 0$, write $J_\varepsilon = |D|^{-1}|B| + \varepsilon e e^T$, where $e = (1, 1, \cdots, 1)^T \in R^n$. According to the continuity of the spectral radius and (3.3) there hold

$$\rho_\varepsilon := \rho(J_\varepsilon) < 1, \quad \sigma := \left\{ \begin{array}{ll} |1 - \omega_1| + \omega_1 \rho_\varepsilon, & \text{if } \omega_2 = 0 \\ |1 - \omega_2| + \omega_2 \rho_\varepsilon, & \text{if } \omega_2 \ne 0 \end{array} \right\} < 1 \tag{3.6}$$

provided ε is much small. Now, in accordance with the Perron-Frobenius theorem[12] in the nonnegative matrix theory, there exists a positive vector $v = v^{(\varepsilon)} \in R^n$ such that

$$J_\varepsilon v = \rho_\varepsilon v. \tag{3.7}$$

Since $(D - r_1 L_i)$ and $(D - r_2 U_i)$, $i = 1, 2, \cdots, \alpha$, are all H-matrices, we can obtain

$$\begin{array}{ll} |(D - r_1 L_i)^{-1}| \le (|D| - r_1|L_i|)^{-1}, & \\ & i = 1, 2, \cdots, \alpha. \\ |(D - r_2 U_i)^{-1}| \le (|D| - r_2|U_i|)^{-1}, & \end{array}$$

From (2.5) and by direct estimations, there immediately hold

$$\begin{cases} |\mathcal{L}^{(i)}(r_1, \omega_1)| \le I + (|D| - r_1|L_i|)^{-1}|D|[(|1 - \omega_1| - 1)I + \omega_1 J_\varepsilon] \\ |\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)| \le (|D| - r_2|U_i|)^{-1}\{[|1 - \omega_2||D| + (\omega_2 - r_2)|U_i| \\ \qquad\qquad +\omega_2|L_i|]|\mathcal{L}^{(i)}(r_1, \omega_1)| + \omega_2|W_i|\}. \end{cases} \tag{3.8}$$

Applying (3.6)-(3.7) to the first inequality of (3.8) we easily see that there have

$$|\mathcal{L}^{(i)}(r_1, \omega_1)|v \le (|1 - \omega_1| + \omega_1\rho_\varepsilon)v < v,$$

and thereby,

$$|\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)|v \le |\mathcal{L}^{(i)}(r_1, \omega_1)|v \le (|1 - \omega_1| + \omega_1\rho_\varepsilon)v = \sigma v$$

if $\omega_2 = 0$ and

$$\begin{aligned} |\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)|v &\le (|D| - r_2|U_i|)^{-1}[|1 - \omega_2||D| + (\omega_2 - r_2)|U_i| + \omega_2(|L_i| + |W_i|)]v \\ &\le I + (|D| - r_2|U_i|)^{-1}|D|[(|1 - \omega_2| - 1)I + \omega_2 J_\varepsilon]v \\ &\le (|1 - \omega_2| + \omega_2\rho_\varepsilon)v = \sigma v \end{aligned}$$

if $\omega_2 \ne 0$.

For the error vector sequence $\{\varepsilon^p\}_{p\in N_0}$ given by (3.1), take $\epsilon^0 = |\varepsilon^0|$ and define $\{\epsilon^p\}_{p\in N_0}$ according to Lemma 3.1 with $H_i = |\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)|$ $(i = 1, 2, \cdots, \alpha)$. Since we have verified these $H_i (i = 1, 2, \cdots, \alpha)$ are nonnegative matrices and $H_i v \le \sigma v$ $(i = 1, 2, \cdots, \alpha)$, in light of Lemma 3.1(i) we see that $\epsilon^p \to 0 (p \to \infty)$.

Moreover, by induction we can get $|\varepsilon^p| \le \epsilon^p$ $(\forall p \in N_0)$, i.e., $\{\epsilon^p\}_{p\in N_0}$ is a majorizing sequence of $\{\varepsilon^p\}_{p\in N_0}$. Therefore, we have $|\varepsilon^p| \to 0 (p \to \infty)$, or in other words, the conclusion (i) is valid.

We now turn to (ii). Considering the continuity of the spectral radius and (3.4)-(3.5), we have

$$\bar{\sigma} := |1 - \beta| + \beta\sigma < 1 \tag{3.9}$$

provided ε is taken to be sufficiently small.

Through (2.7), (3.9) and conclusion (i), we can obtain $|\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta)|v \le \bar{\sigma}v$ $(i = 1, 2, \cdots, \alpha)$ at once.

Presently, similar to (i), we can immediately fulfil the proof of conclusion (ii) by making use of these inequalities and Lemma 3.1(ii), as well as (3.2).

Theorem 3.2. *Let $A \in L(R^n)$ be an L-matrix, $(D - L_i, D - U_i, W_i, E_i)$ $(i = 1, 2, \cdots, \alpha)$ be a multisplitting of it with $L_i \geq 0$, $U_i \geq 0$, $W_i \geq 0$, $i = 1, 2, \cdots, \alpha$. Then either of the following conditions is sufficient and necessary for $\rho(D^{-1}B) < 1$ with $B = D - A$:*

(a) for the starting vector $x^0 \in R^n$, the sequence generated by Method I converges to the unique solution of the system of linear equations (1.1) provided the relaxation parameters r_1, r_2 and ω_1, ω_2 satisfy $0 \leq r_m \leq \omega_m \leq 1 (m = 1, 2)$ and $\omega_1 + \omega_2 > 0$;

(b) for the starting vector $x^0 \in R^n$, the sequence generated by Method II converges to the unique solution of the system of linear equations (1.1) provided the relaxation parameters r_1, r_2, ω_1, ω_2 and β satisfy $0 \leq r_m \leq \omega_m \leq 1 (m = 1, 2)$, $\omega_1 + \omega_2 > 0$ and $0 < \beta \leq 1$.

Proof. The necessities of (a) and (b) are obvious in light of Theorem 3.1. We now deal with their sufficiency separately.

Let $B = D - A$, $J = D^{-1}B$ and suppose $\rho(J) \geq 1$. Simply denote $\rho = \rho(J)$. Then by the Perron-Frobenius theorem in the nonnegative matrix theory again, there exists a nonnegative vector $u \in R^n (\not\equiv 0)$ such that $Ju = \rho u$. Now, we can assert

$$\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)u \geq u, \quad \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta)u \geq u, \quad i = 1, 2, \cdots, \alpha, \tag{3.10}$$

where for $i = 1, 2, \cdots, \alpha$, $\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)$ and $\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta)$ are defined by (2.5) and (2.7), respectively.

In fact, because of

$$\begin{aligned}\mathcal{L}^{(i)}(r_1, \omega_1)u &= u + \omega_1(D - r_1 L_i)^{-1} D(J - I)u \\ &= u + \omega_1(D - r_1 L_i)^{-1} D(\rho - 1)u \geq u,\end{aligned}$$

we have

$$\begin{aligned}\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2)u &\geq (D - r_2 U_i)^{-1}[(1 - \omega_2)D + (\omega_2 - r_2)U_i + \omega_2(L_i + W_i)]u \\ &= u + \omega_2(D - r_2 U_i)^{-1} D(J - I)u \\ &= u + \omega_2(D - r_2 U_i)^{-1} D(\rho - 1)u \geq u,\end{aligned}$$

and hence,

$$\mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta)u = [\beta \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2) + (1 - \beta)I]u \geq [\beta + (1 - \beta)]u = u.$$

Let $\{\varepsilon^p\}_{p \in N_0}$ be given by either (3.1) or (3.2) and take $\epsilon^0 = \varepsilon^0$. Define $\{\epsilon^p\}_{p \in N_0}$ according to Lemma 3.1 again with

$$H_i = \begin{cases} \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2), & \text{for Method I,} \\ \mathcal{L}^{(i)}(r_1, r_2, \omega_1, \omega_2, \beta), & \text{for Method II,} \end{cases} \quad i = 1, 2, \cdots, \alpha,$$

correspondingly. Clearly, $\epsilon^p = \varepsilon^p (p = 0, 1, 2, \cdots)$ hold for both Method I and Method II. Since $H_i (i = 1, 2, \cdots, \alpha)$ are nonnegative matrices and we have demonstrated that $H_i u \geq u (i = 1, 2, \cdots, \alpha)$ hold according to both of these two methods, by Lemma 3.1(ii) we know that $\lim\limits_{p \to \infty} \varepsilon^p = \lim\limits_{p \to \infty} \epsilon^p \neq 0$ for some $\epsilon^0 = \varepsilon^0 \in R^n$, or in other words, for some $x^0 \in R^n$, which results in a contradiction. Therefore, $\rho(D^{-1}B) < 1$.

4. Numerical results

In this section, we are going to imitate the implementations of the previously established asynchronous matrix multi-splitting multi-parameter relaxation methods such as the asynchronous matrix multisplitting SGS method, the asynchronous matrix multisplitting SSOR method and the asynchronous matrix multisplitting SAOR method, which are simply denoted as AMMSGS, AMMSSOR (ω) and AMMSAOR (r,ω), and compare them with their corresponding asynchronous matrix multisplitting relaxation methods[8,9], i.e., the asynchronous matrix multisplitting GS method (AMMGS), the asynchronous matrix multisplitting SOR method (AMMSOR (ω)) and the asynchronous matrix multisplitting AOR method (AMMAOR (r,ω)), by using the following example of the system of linear equations (1.1).

Example 4.1. $A=$ block-tridiag$(-I,B,-I)\in L(R^n)$, $b=(4,4,\cdots,4)^T\in R^n$, where $n=N^2$, $B=tridiag(-1,4+ch^2,-1)\in L(R^N)$, $c=10.0$ and $h=\frac{1}{n+1}$.

For this example, we take $\alpha=2$, $J_1=\{1,2,\cdots,m_1\}$, $J_2=\{m_2,m_2+1,\cdots,n\}$, with m_1,m_2 being positive integers satisfying $1\le m_2\le m_1\le n$, and two particular kinds of multisplittings $(D-L_i,D-U_i,W_i,E_i)(i=1,2)$ of the coefficient matrix $A\in L(R^n)$:

$$\begin{cases} D=\text{diag}\,(A)=\text{diag}\,(4+ch^2)\in L(R^n) \\ L_i=(\mathcal{L}_{mj}^{(i)})\in L(R^n),\quad \mathcal{L}_{mj}^{(i)}=\begin{cases}-a_{mj}, & \text{if } j<m,\quad m,j\in J_i\\ 0, & \text{otherwise}\end{cases} \\ U_i=(\mathcal{U}_{mj}^{(i)})\in L(R^n),\quad \mathcal{U}_{mj}^{(i)}=\begin{cases}-a_{mj}, & \text{if } j>m,\quad m,j\in J_i\\ 0, & \text{otherwise}\end{cases} \\ W_i=(\mathcal{W}_{mj}^{(i)})\in L(R^n),\quad \mathcal{W}_{mj}^{(i)}=\begin{cases}0, & \text{if } j=m\\ -(a_{mj}+\mathcal{L}_{mj}^{(i)}+\mathcal{U}_{mj}^{(i)}), & \text{otherwise}\end{cases} \\ E_i=diag(e_1^{(i)},e_2^{(i)},\cdots,e_n^{(i)})\in L(R^n) \\ e_j^{(1)}=\begin{cases}1, & \text{if } 1\le j<m_2\\ 0.75, & \text{if } m_2\le j\le m_1\\ 0, & \text{if } m_1<j\le n\end{cases},\quad e_j^{(2)}=\begin{cases}0, & \text{if } 1\le j<m_2\\ 0.25, & \text{if } m_2\le j\le m_1\\ 1, & \text{if } m_1<j\le n\end{cases} \\ m,j=1,2,\cdots,n;\quad i=1,2. \end{cases}$$

The starting vector and the positive integers m_1 and m_2 are chosen to be $x^0=(0.5,\cdots,0.5)^T\in R^n$ and $m_1=[4n/5]$, $m_2=[n/5]$, respectively, where $[a]$ is used to denote the integer part of a positive number "a", while the stopping criterion is adopted to be $\dfrac{\|Ax^p-b\|_\infty}{\sqrt{n}\max\{\|x^p\|_\infty,1\}}\le 10^{-6}$ and $\dfrac{\|x^p-x^{p-1}\|_\infty}{\sqrt{n}\max\{\|x^p\|_\infty,1\}}\le 10^{-8}$. Corresponding to different n, or a fixed $n(n=10000)$ but different pairs (r,ω) of the relaxation parameters, we have the following iteration number tables.

Table (I) AMMGS and AMMSGS methods

n	100	400	900	1600	2500	3600	4900	6400	8100	10000
AMMGS	124	265	444	658	903	1160	1419	1678	1929	2174
AMMSGS	48	128	230	348	476	610	747	883	1017	1147

Table (II) AMMSOR(ω) and AMMSSOR(ω) methods ($n = 10000$)

ω	0.8	0.9	1.1	1.2	1.3	1.4	1.5	1.6
AMMSOR	3234	2645	1785	1461	1185	947	743	586
AMMSSOR	1615	1356	975	829	702	591	493	403

Table (III) AMMAOR(r,ω) and AMMSAOR(r,ω) methods ($n = 10000$)

r	1.5	1.6	1.6	1.62	1.65	1.7	1.7	1.7
ω	0.9	0.8	1.5	1.58	1.55	1.6	1.5	0.9
AMMAOR	1271	1174	613	566	535	460	481	802
AMMSAOR	724	671	418	391	372	328	339	490

Clearly, besides all the asynchronous matrix multisplitting relaxation methods showing better convergence behaviours corresponding to suitably choosing the relaxation parameter(s), the AMMSGS, AMMSSOR and AMMSAOR methods converge faster than the AMMGS, AMMSOR and AMMAOR methods, respectively.

Moreover, Table (I) implies that as a function of n, the iteration number of the AMMSGS method increases very much slower than that of the AMMGS method with n becoming larger. The corresponding facts are also true for the AMMSSOR and AMMSOR methods as well as the AMMSAOR and AMMAOR methods. This shows that the asynchronous matrix multisplitting symmetric relaxation methods in this paper are much less sensible about the relaxation parameters than their corresponding asynchronous matrix multisplitting relaxation methods[8,9]. These good behaviours of the asynchronous matrix multisplitting symmetric relaxation methods deserve further revealing in both theories and experiments.

5. Proof of Lemma 3.1

To prove Lemma 3.1, we first cite the following important facts from [8] and [9].

Lemma 5.1. *Given $\{\bar{\epsilon}^t\}_{t=0}^p \subset R^n (\forall p \in N_0)$. Assume that for all $t \in \{0, 1, \cdots, p\}$, there exist positive number δ and positive vector $v \in R^n$ such that $|\bar{\epsilon}^t| \le \delta v$. Then there identically hold $|\bar{\epsilon}^{s^{(i)}(p)}| \le \delta v (i = 1, 2, \cdots, \alpha)$ provided $s_m^{(i)}(p) \le p (m = 1, 2, \cdots, n; i = 1, 2, \cdots, \alpha)$, where*

$$\bar{\epsilon}^{s^{(i)}(p)} = (\bar{\epsilon}_1^{s_1^{(i)}(p)}, \bar{\epsilon}_2^{s_2^{(i)}(p)}, \cdots, \bar{\epsilon}_n^{s_n^{(i)}(p)})^T, \quad i = 1, 2, \cdots, \alpha, \quad \forall p \in N_0.$$

Lemma 5.2. *Let $\xi_m > 0 (m = 1, 2, \cdots, n)$. Assume that the sequence $\{\epsilon_m^p\}_{p \in N_0} (m = 1, 2, \cdots, n)$ are defined to satisfy $|\epsilon_m^{p+1}| \le i_m^p \xi_m + j_m^p |\epsilon_m^p|$, $p = 0, 1, 2, \cdots$. Then for any nonnegative integer $q \le p - 1$ there hold*

$$\begin{cases} |\epsilon_m^{p+1}| \le \Big(1 - \prod_{k=p-q-1}^{p} j_m^k\Big)\xi_m + \prod_{k=p-q-1}^{p} j_m^k |\epsilon_m^{p-q-1}| \\ m = 1, 2, \cdots, n; \quad p = 0, 1, 2, \cdots \end{cases}$$

where

$$i_m^p = \sum_{i \in N_m(p)} e_m^{(i)}, \quad j_m^p = \sum_{i \notin N_m(p)} e_m^{(i)}, \quad m = 1, 2, \cdots, n, \quad p = 0, 1, 2, \cdots,$$

and $e_m^{(i)}$ *is the* m*-th element of the weighting matrix* E_i.

Lemma 5.3. *Let the sequences* $\{i_m^p\}_{p\in N_0}$, $\{j_m^p\}_{p\in N_0}(m=1,2,\cdots,n)$ *be defined as in Lemma* 5.2 *and the sequences* $\{j_m^{(l)}\}_{l\in N_0}(m=1,2,\cdots,n)$ *be defined as*

$$\begin{cases} j_m^{(0)} = \prod\limits_{p=0}^{m_0-1} j_m^p, \quad j_m^{(l+1)} = \prod\limits_{p=m_l}^{m_{l+1}-1} j_m^p \\ m=1,2,\cdots,n; \quad l=0,1,2,\cdots. \end{cases}$$

Then, there hold $\{j_m^{(l)}\}_{l\in N_0}\subset[0,1)$ $(m=1,2,\cdots,n)$. *Furthermore, if we define the sequence* $\{\gamma^{(l)}\}_{l\in N_0}$ *to be* $\gamma^{(l)}=\max_{1\le m\le n} j_m^{(l)}(l=0,1,2,\cdots)$, *there obviously has* $\{\gamma^{(l)}\}_{l\in N_0}\subset[0,1)$. *Here, the infinite number sequence* $\{m_l\}_{l\in N_0}$ *are defined according to the following rule:* m_0 *is the least positive integer such that* $\bigcup\limits_{0\le s(p)\le p<m_0} J_i(p)=J_i(i=1,2,\cdots,\alpha)$, *in general,* m_{l+1} *is the least positive integer such that* $\bigcup\limits_{m_l\le s(p)\le p<m_{l+1}} J_i(p)=J_i$ $(i=1,2,\cdots,\alpha,l=0,1,2,\cdots)$.

Lemma 5.4. *Let the conditions of Lemma* 5.3 *be satisfied. Then there exist non-negative numbers* $j_m\in[0,1)$ $(m=1,2,\cdots,n)$ *such that* $\{j_m^{(l)}\}_{l\in N_0}\subseteq[0,j_m]$ $(m=1,2,\cdots,n)$. *Moreover, if we define* $\gamma=\max\limits_{1\le m\le n}\{j_m\}$, *then* $\gamma\in[0,1)$ *and there holds* $\{\gamma^{(l)}\}_{l\in N_0}\subseteq[0,\gamma]$.

Proof. Evidently, we only need to prove that there exist $j_m\in[0,1)$ $(m=1,2,\cdots,n)$ such that for $\forall m\in\{1,2,\cdots,n\}$ there hold $0\le j_m^{(l)}\le j_m(l=0,1,2,\cdots)$.

Define the number sets

$$N_m=\{i\mid e_m^{(i)}>0,\quad m\in J_i,\quad i=1,2,\cdots,\alpha\},\quad m=1,2,\cdots,n.$$

Then we easily know that $N_m(p)\subseteq N_m$ and $\sum\limits_{i\in N_m} e_m^{(i)}=1$ for $m=1,2,\cdots,n$ and $\forall p\in N_0$. On the other hand, because if $N_{m'}(p')=\emptyset$ or $N_{m'}(p')\ne\emptyset$ but $e_{m'}^{(i)}=0$ $(\forall i\in N_{m'}(p'))$ holds for some $m'\in\{1,2,\cdots,n\}$ and $p'\in\{m_{l'},m_{l'}+1,\cdots,m_{l'+1}-1\}$ $(l'\in N_0\cup\{-1\},m_{-1}=0)$, then there have $i_{m'}^{p'}=0$, $j_{m'}^{p'}=1$, and

$$j_{m'}^{(l'+1)}=\prod_{p=m_{l'}}^{m_{l'+1}-1} j_{m'}^p=\prod_{p=m_{l'}}^{p'-1} j_{m'}^p\prod_{p=p'+1}^{m_{l'+1}-1} j_{m'}^p.$$

So, without loss of generality, in the remainder of this proof we will assume that $N_m(p)\ne\emptyset$ $(m=1,2,\cdots,n,\ \forall p\in N_0)$, and for $\forall m\in\{1,2,\cdots,n\}$, $\forall p\in\{0,1,2,\cdots\}$ there has at least one $i\in N_m(p)$ such that $e_m^{(i)}>0$.

Now, let $\underline{e_m}=\min\limits_{i\in N_m} e_m^{(i)}$ and $j_m=1-\underline{e_m}$, $m=1,2,\cdots,n$. Then, $0<\underline{e_m}\le 1$ and $0\le j_m<1$, $m=1,2,\cdots,n$. Moreover, we can easily see that

$$\underline{e_m}\le\min\{e_m^{(i)}\mid e_m^{(i)}>0,\quad i\in N_m(p),\quad i=1,2,\cdots,\alpha\},\quad m=1,2,\cdots,n,\quad \forall p\in N_0,$$

and therefore,

$$\begin{cases} i_m^p = \sum\limits_{i\in N_m(p)} e_m^{(i)} = \sum\limits_{\substack{e_m^{(i)}>0 \\ i\in N_m(p)}} e_m^{(i)} \geq \underline{e_m}, \\ j_m^p = \sum\limits_{i\notin N_m(p)} e_m^{(i)} = 1 - i_m^p \leq 1 - \underline{e_m}, \end{cases} \qquad m = 1,2,\cdots,n, \quad p = 0,1,2,\cdots.$$

Noticing from Lemma 5.2 and Lemma 5.3 that there hold $0 \leq j_m^p \leq 1$ and $0 \leq j_m^{(l)} < 1$, $m = 1,2,\cdots,n$, $\forall p \in N_0$, by making use of the inequality

$$\prod_{k=1}^{l} c_k \leq \frac{1}{l}\sum_{k=1}^{l} c_k \quad l = 1,2,\cdots,$$

where $0 \leq c_k \leq 1$ $(k = 1,2,\cdots,l)$, we can obtain for $\forall m \in \{1,2,\cdots,n\}$, $\forall l \in \{0,1,2,\cdots\}$ that

$$\begin{aligned} j_m^{(l)} = \prod_{p=m_{l-1}}^{m_l-1} j_m^p &\leq \frac{1}{m_l - m_{l-1}} \sum_{p=m_{l-1}}^{m_l-1} j_m^p \\ &\leq \frac{1}{m_l - m_{l-1}} \sum_{p=m_{l-1}}^{m_l-1} (1 - \underline{e_m}) = (1 - \underline{e_m}) = j_m. \end{aligned}$$

These are just the inequalities what we are proving.

According to Lemma 3.1, the meaning of the sequence $\{m_l\}_{l\in N_0}$ defined by Lemma 5.3 is that following the working processes of the CPU's, there always exists one moment which guarantees each CPU renewing all the elements with respect to $J_i(i = 1,2,\cdots,\alpha)$ at least once, in other words, the iterative method corresponding to each $H_i(i = 1,2,\cdots,\alpha)$ must fulfil its iteration for all the elements corresponding to $J_i(i = 1,2,\cdots,\alpha)$ at least once.

We are now in the position of proving Lemma 3.1.

Proof of Lemma 3.1. We first prove (i). It is reasonable for us to assume that there exists a $\delta > 0$ such that $|\epsilon^0| \leq \delta v$.

Now, making use of Lemma 5.1 and through induction, we can directly conclude $|\epsilon^p| \leq \delta v(\forall p \in N_0)$. Moreover, we can also assert

$$|\epsilon^p| \leq \Delta_l v, \quad \forall p \geq m_l, \quad \forall l \in N_0, \tag{5.1}$$

where

$$\Delta_0 = (\sigma + (1-\sigma)\gamma^{(0)})\delta, \quad \Delta_{l+1} = (\sigma + (1-\sigma)\gamma^{(l+1)})\Delta_l, \quad l = 0,1,2,\cdots.$$

As a matter of fact, for $l = 0$, by making use of Lemma 5.2 and Lemma 5.3 we have for $m = 1,2,\cdots,n$ that

$$|\epsilon_m^{p+1}| \leq \Big(1 - \prod_{k=0}^{p} j_m^k\Big)\sigma\delta v_m + \prod_{k=0}^{p} j_m^k |\epsilon_m^0| \leq \Big(1 - \prod_{k=0}^{p} j_m^k\Big)\sigma\delta v_m + \prod_{k=0}^{p} j_m^k \delta v_m$$

$$= \Big(\sigma + (1-\sigma)\prod_{k=0}^{p} j_m^k\Big)\delta v_m \le (\sigma + (1-\sigma) j_m^{(0)})\delta v_m$$
$$\le (\sigma + (1-\sigma)\gamma^{(0)})\delta v_m = \Delta_0 v_m.$$

This is just (5.1) for $l = 0$.

Suppose that for $p \ge m_l$, (5.1) is correct. Then, as $p \ge m_{l+1}$, again by Lemma 5.1, for any $m \in \{1, 2, \cdots, n\}$, we have

$$|\epsilon_m^{p+1}| \le \sum_{i \in N_m(p)} e_m^{(i)} e_m^T H_i \Delta_l v + \sum_{i \notin N_m(p)} e_m^{(i)} |\epsilon_m^p| \le \Delta_l \sigma i_m^p v_m + j_m^p |\epsilon_m^p|.$$

Making use of Lemma 5.2, the induction assumption, and Lemma 5.3, for $m = 1, 2, \cdots, n$ we can obtain that

$$\begin{aligned} |\epsilon_m^{p+1}| &\le \Big(1 - \prod_{k=m_l}^{p} j_m^k\Big)\sigma\Delta_l v_m + \prod_{k=m_l}^{p} j_m^k |\epsilon_m^{m_l}| \\ &\le \Big(1 - \prod_{k=m_l}^{p} j_m^k\Big)\sigma\Delta_l v_m + \prod_{k=m_l}^{p} j_m^k \Delta_l v_m \\ &\le (\sigma + (1-\sigma) j_m^{(l+1)})\Delta_l v_m \\ &\le (\sigma + (1-\sigma)\gamma^{(l+1)})\Delta_l v_m = \Delta_{l+1} v_m, \end{aligned}$$

which implies the rightness of (5.1) for $p \ge m_{l+1}$, too. Hence, the induction again guarantees the correctness of (5.1).

Presently, let $\tilde{\beta} = \sigma + (1-\sigma)\gamma$ and $\beta^{(l)} = \sigma + (1-\sigma)\gamma^{(l)} (l = 0, 1, 2, \cdots)$. From Lemma 5.4 there evidently hold $\tilde{\beta} \in [0, 1)$ and $\{\beta^{(l)}\}_{l \in N_0} \subset [0, \tilde{\beta}] (l = 0, 1, 2, \cdots)$. By the definition of the sequence $\{\Delta_l\}_{l \in N_0}$ and by successively regressing, we have

$$\Delta_{l+1} = \beta^{(l+1)}\Delta_l = \cdots = \prod_{k=0}^{l+1} \beta^{(k)}\delta \le \tilde{\beta}^{l+1}\delta \longrightarrow 0 \quad (l \longrightarrow \infty).$$

Taking limit in either side of (5.1), we immediately obtain $\lim\limits_{p \to \infty} |\epsilon^p| = 0$, and therefore, the validity of Lemma 3.1(i) is verified.

We now turn to prove (ii). Take $\epsilon^0 \ge u$. Through induction we can immediately get $\epsilon^p \ge u (p = 0, 1, 2, \cdots)$. Therefore, conclusion (ii) holds.

6. Conclusions and remarks

In accordance with the concrete characteristics of the high speed multiprocessor systems and on the basis of the principle of sufficiently using the delayed information, we set up in this paper a class of asynchronous matrix multi-splitting multi-parameter relaxation methods for parallely solving the large sparse system of linear equations. This class of methods includes the asynchronous variants of both the symmetric and unsymmetric classical iterative methods in the sense of matrix multisplitting, and thus, it is of great generality and considerable parallelism. Numerical results show that these new methods are much less sensible about their individual relaxation parameters than the ones discussed in [8] and [9]. Based on the existing works[1−11,13−29], we establish general criterions for determining the convergence as well as the divergence of the parallel multisplitting relaxation methods for the L-matrix and the H-matrix classes. These criterions are simple in forms and convenient for applications. With these general criterions, we conveniently give conditions for guaranteeing the convergence as well as the divergence of the new asynchronous matrix multisplitting unsymmetric relaxation methods. All these make the methods as well as their corresponding theories about large sparse systems of linear equations in the sense of multisplitting become more extensive, more deep, and more systematic.

At last, we use the following remarks to end this paper.

Remark I. It may be possible to speed the convergence of Methods I and II by replacing the quantities

$$\sum_{j\neq m} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)} \ (m \in J_i(p),\ i = 1, 2, \cdots, \alpha)$$

in the formulas $x_m^{p+1,i}(m \in J_i(p),\ i = 1, 2, \cdots, \alpha)$ by

$$\sum_{\substack{j\neq m \\ j\in J_i(p)}} w_{mj}^{(i)} x_j^{p+1/2,i} + \sum_{j\notin J_i(p)} w_{mj}^{(i)} x_j^{s_j^{(i)}(p)}, \quad m \in J_i(p), \quad i = 1, 2, \cdots, \alpha,$$

respectively, since the newest information is utilized. However, as costs, additional computational works and savings must be required in these two variants. For their convergence, we can similarly prove that the conclusions corresponding to Theorems 3.1 and 3.2 are still valid for them only with the varying interval of the relaxation parameter β in Theorem 3.1(ii) being replaced by

$$0 < \beta < 2/(1 + \rho(\mathcal{H}(\omega_1))\rho(\mathcal{H}(\omega_2))), \quad \mathcal{H}(\omega) = |1 - \omega|I + \omega|D|^{-1}|B|.$$

Remark II. Evidently, Method I and Method II and their variants just mentioned in Remark I are two different classes of asynchronous matrix multisplitting UAOR methods, each has its advantages and disadvantages. Since the parallel efficiency of a program is determined by not only the method itself, but also the computer architectures it runs on, so which of these two classes of methods is more efficient needs further investigations in practical experiments on genuine multiprocessor systems.

Remark III. It is not difficult to extend all the above methods to parameter-group relaxed forms and establish the corresponding convergence theories for these extensions.

Remark IV. Note that except that $\{\gamma^{(l)}\}_{l\in N_0} \subseteq [0, \gamma]$ holds for some $\gamma \in [0, 1)$, the relation $\{\gamma^{(l)}\}_{l\in N_0} \subseteq [0, 1)$ may be not sufficient for ensuring $\Delta_l \to 0(l \to \infty)$, since at this moment there may have

$$\underline{\lim}_{l\to\infty} \Delta_l = \underline{\lim}_{l\to\infty} \prod_{k=0}^{l} \beta^{(k)} > 0.$$

Therefore, the establishment of Lemma 5.4 is essential for proving the convergence of the asynchronous matrix multisplitting relaxation methods. Based upon this lemma, all the convergence proofs of the existing asynchronous multisplitting relaxation methods in the literature [6-9] and [21-29] can be further modified and exacted correspondingly.

References

[1] D.P. O'Leary, R.E. White, Multi-splittings of matrices and parallel solution of linear systems, *SIAM J. Alg. Disc. Meth.,* **6** (1985), 630-640.

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

[3] D.R. Wang, On the convergence of the parallel multisplitting AOR algorithm, *Linear Algebra Appl.,* **154/156** (1991), 473-486.

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

[5] R. Bru, L. Elsner, M. Neumann, Models of parallel chaotic iteration methods, *Linear Algebra Appl.,* **103** (1989), 175-192.

[6] D.R. Wang, Z.Z. Bai, D.J. Evans, A class of asynchronous parallel matrix multisplitting relaxation methods, *Parallel Algorithms Appl.,* **2** (1994), 173-192.

[7] Z.Z. Bai, D.R. Wang, Asynchronous parallel matrix multi-splitting multi-parameter relaxation methods, *Numer. Math. J. Chinese Univ.(Chinese Ser.),* **16** (1994), 107-115.

[8] Z.Z. Bai, Parallel Iterative Methods for Large-scale Systems of Algebraic Equations, *Ph. D. Thesis, Department of Mathematics, Shanghai University of Science and Technology,* Shanghai, March, 1993.

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

[10] M. Neumann, R.J. Plemmons, Convergence of parallel multisplitting iterative methods for M-matrices, *Linear Algebra Appl.,* **88** (1987), 559-573.

[11] R.E. White, Multisplitting with different weighting schemes, *SIAM J. Matrix Anal. Appl.,* **10** (1989), 481-493.

[12] R.S. Varga, Matrix Iterative Analysis, Englewood Cliffs, N. J.: Prentice-Hall, 1961.

[13] D.J. Evans, D.R. Wang, An asynchronous parallel algorithm for solving a class of nonlinear simultaneous equations, *Parallel Comput.,* **17** (1991), 165-180.

[14] Z.Z. Bai, A note on "Multisplitting with different weighting schemes", *Chinese J. Eng. Math.,* **10** (1993), 107-110.

[15] Z.Z. Bai, D.R. Wang, Generalized matrix multisplitting relaxation methods and their convergence, *Numer. Math. J. Chinese Univ. (English Ser.),* **2** (1993), 87-100.

[16] Z.Z. Bai, Convergence analysis of the matrix multisplitting AOR algorithm, *J. Natural Sci. Heilongjiang Univ.,* **10(special)** (1993), 1-5.

[17] Z.Z. Bai, Comparisons of the convergence and divergence rates of the parallel matrix multisplitting iteration methods, *Chinese J. Eng. Math.,* **11** (1994), 99-102.

[18] Z.Z. Bai, Generalized parallel multisplitting forward and backward relaxed methods, *J. UEST China,* **23** (1994), 207-212.

[19] D.R. Wang, Z.Z. Bai, On the monotone convergence of matrix multisplitting iteration methods, *Proc. '92 Shanghai Intern. Numer. Algebra Appl. Conf., E.X. Jiang eds.,* China Science and Technology Press, Beijing, 1994.

[20] Z.Z. Bai, On the convergence of the generalized matrix multisplitting relaxed methods, *Commun. Numer. Math. Eng.,* **11** (1995), 363-371.

[21] Z.Z. Bai, Variants of the synchronous and asynchronous matrix multisplitting unsymmetric AOR methods, *J. Fudan Univ.(Natural Sci.),* **34** (1995), 139-148.

[22] Z.Z. Bai, Generalized asynchronous parallel multisplitting relaxation methods, *Appl. Math.–JCU,* **10A** (1995), 133-140.

[23] D.R. Wang, Z.Z. Bai, D.J. Evans, A class of asynchronous parallel nonlinear multisplitting relaxation methods, *Parallel Algorithms Appl.,* **2** (1994), 209-228.

[24] Z.Z. Bai, D.R. Wang, Asynchronous multisplitting nonlinear Gauss-Seidel type method, *Appl. Math.–JCU,* **9B** (1994), 189-194.

[25] Z.Z. Bai, Asynchronous multisplitting AOR method for a system of nonlinear algebraic equations, *Intern. J. Computer Math.,* **55** (1995), 223-233.

[26] D.R. Wang, Z.Z. Bai, D.J. Evans, Asynchronous multisplitting relaxed iterations for weakly nonlinear systems, *Intern. J. Computer Math.,* **54** (1994), 57-76.

[27] Z.Z. Bai, D.R. Wang, D.J. Evans, Models of asynchronous parallel nonlinear multisplitting relaxed iterations, *J. Comp. Math.,* **13** (1995), 369-386.

[28] Z.Z. Bai, Asynchronous multisplitting relaxation methods for solving large-scale linear least-squares problems, *Numer. Math. J. Chinese Univ. (Chinese Ser.),* **17** (1995), 127-137.

[29] Z.Z. Bai, Y.M. Wei, Asynchronous parallel nonlinear AOR algorithms, *J. Shanghai Teachers Univ.(Natural Sci.),* **24** (1995), 13-18.