ICST Transactions on Scalable Information Systems Research Article ICST.ORG

# Non-stationary Parallel Multisplitting Two-Stage Iterative Methods with Self-Adaptive Weighting Schemes

GuoYan Meng[1], ChuanLong Wang[2,*], XiHong Yan[2], QingShan Zhao[1]

[1]Department of Compute Science, Xinzhou teacher University, Xinzhou 034000, Shanxi Province, P. R. China
[2]Department of mathematics, Taiyuan Normal University, Taiyuan 030012, Shanxi Province, P. R. China

## Abstract

In this paper, we study the non-stationary parallel multisplitting two-stage iterative methods with self-adaptive weighting matrices for solving a linear system whose coefficient matrix is symmetric positive definite. Two choices of Self-adaptive weighting matrices are given, especially, the nonnegativity is eliminated. Moreover, we prove the convergence of the non-stationary parallel multisplitting two-stage iterative methods with self-adaptive weighting matrices. Finally, the numerical comparisons of several self-adaptive non-stationary parallel multisplitting two-stage iterative methods are shown.

Received on 05 January 2013; accepted on 03 August 2013; published on 04 March 2014

**Keywords:** Self-adaptive weighting matrices, non-stationary, multisplitting, two-stage, linear systems



doi:10.4108/sis.1.2.e5

## 1. Introduction

To solve large sparse linear system of equations on multiprocessor systems,

$$Ax = b, \quad A = (a_{ij}) \in R^{n\times n} \text{ nonsingular and } b \in R^n \ . \quad (1)$$

O'Leary and White [14] first proposed parallel methods based on multisplitting of matrices in 1985, after this, combing with two-stage iterative methods (see [2, 4, 10]), the multisplitting two-stage iterative methods [15] were proposed, where several basic convergence results were found. The scheme was proposed as following

$$A = B_i - C_i, \quad B_i = M_i - N_i, \quad i = 1, 2, \cdots, m \ , \quad (2)$$

$$M_i x_i^{(k,l)} = N_i x_i^{(k,l-1)} + C_i x^{(k)} + b \ , \quad (3)$$

$$x^{(k+1)} = \sum_{i=1}^{m} E_i x_i^{(k,q(i,k))} \ , \quad (4)$$

where $E_i \geq 0$, diagonal, and $\sum_{i=1}^{m} E_i = I$. $(M_i, N_i, C_i, E_i)_{i=1}^{m}$ will be unchanged and independent of the iterative number $k$.

Later, many authors studied the methods for the case that $A$ is an $M$-matrix, an $H$-matrix and a symmetric positive definite matrix. When $A$ is an $M$-matrix or an $H$-matrix, many parallel multisplitting two-stage iterative methods (see [3, 5, 6, 12, 15, 17]) were presented, and the weighting matrices $E_i, i = 1, 2, \cdots, m$ were generalized (see [1, 11])

$$\sum_{i=1}^{m} E_i^{(k)} = I (\text{or} \neq I), \quad E_i^{(k)} \geq 0, \ k = 1, 2, \cdots \ , \quad (5)$$

and $E_i^{(k)}$ is diagonal, but these weighting matrices were preset as multi-parameter.

When $A$ is a symmetric positive definite matrix, generally, which require the assumption that the weighting matrices are multiples of the identity matrix, that is $E_i = \alpha_i I, i = 1, 2, \cdots, m$ (see [8, 14]), but these results have little applicability for analysis of parallel processing. In order to improve the weighting matrices, White [19, 20] and Wen [18] presented the multisplitting which had a very special structure,

★This paper is an extended version of [22]. We have added a kind of self-adaptive weighting schemes in Algorithm 1, and also proven the convergence of Algorithm 1 in this condition. In addition, we have added the numerical example and completely recalculated the numerical examples with highly precision and higher size of coefficient matrix.

*Chuan-Long Wang. Email: clwang218@126.com

Chen [21] discussed asynchronous multisplitting, Cao [7] gave a nonstandard multisplitting, Migallón [13] proposed the non-stationary multisplittings, Wang and Bai [17] discussed the non-stationary two-stage multisplitting, but the non-stationary multisplitting usually had a block splitting for parallel processing. Furthermore, as we know, the weighting matrices have important role in parallel multisplitting methods, but the weighting matrices in all above-mentioned methods are determined previously, they are not known to be good or bad, this influences the efficiency of parallel methods. Fortunately, Wang [23] has presented modified parallel multisplitting iterative methods by optimizing the weighting matrices based on the sparsity of the coefficient matrix A. But none has ever studied that how to choose optimal weighting matrices for the parallel multisplitting two-stage iterative algorithms, we will discuss this problem in the paper.

Here, we still use the scalar weighting matrices

$$E_i^{(k)} = \alpha_i^{(k)} I,\ i = 1, 2, \cdots, m,\ k = 1, 2, \cdots \ . \tag{6}$$

in the parallel multisplitting two-stage iterative method, but $\alpha_i^{(k)}$ $(i = 1, 2, \cdots, m,\ \ k = 1, 2, \cdots)$ are chosen by finding the optimal point in the hyperplane $H_k$, where

$$H_k = \left\{ x | x = \sum_{i=1}^{m} \alpha_i^{(k)} x_i^{(k)},\ \ \sum_{i=1}^{m} \alpha_i^{(k)} = 1 \right\},\ \ k = 1, 2, \cdots \ . \tag{7}$$

Thus, $\alpha_i^{(k)}(i = 1, 2, \cdots, m,\ \ k = 1, 2, \cdots)$ are the optimal parameters in $k$-th iteration. In other words, the point $x^{(k)} = \sum_{i=1}^{m} \alpha_i^{(k)} x_i^{(k)}$ generated by the optimal weighting matrices (6) may be the optimal point to the solution of linear systems (1) in $H_k$. Thus, we search the optimal weighting matrices without nonnegative condition. In fact, numerical examples (will be seen in section 4) show that the methods with the weighting matrices (6) are effective.

The paper is organized as follows. In Section 1, we give some notations and preliminaries. In Section 2, the non-stationary parallel multisplitting two-stage iterative methods with self-adaptive weighting schemes are put forward. In Section 3, the convergence of the new method is established. We provide numerical results in Section 4.

Here are some essential notations and preliminaries. $R^{n\times n}$ is used to denote the $n \times n$ real matrix set, the matrix $A^T$ denotes the transpose of $A$. Similarly the transpose of a vector $x$ is denoted by $x^T$. A matrix $A \in R^{n\times n}$ is called symmetric positive definite(or semidefinite), if it is symmetric and for all $x \in R^n, x \neq 0$, it holds that $x^T Ax > 0$(or $x^T Ax \geq 0$). $A = M - N$ is called a splitting of the matrix $A$ if $M \in R^{n\times n}$ is non-singular; this splitting is called a convergent splitting if $\rho(M^{-1}N) < 1$; a $P$-regular splitting of the symmetric positive definite matrix $A$ if $M^T + N$ is positive definite, a symmetric positive definite splitting if $N$ is symmetric positive semi-definite (see [6, 16]).

## 2. Algorithms

In this section, we give the non-stationary parallel multisplitting two-stage iterative methods with self-adaptive weighting schemes.

Let

$$E_i^{(k)} = \alpha_i^{(k)} I,\ i = 1, 2, \cdots, m,\ \sum_{i=1}^{m} \alpha_i^{(k)} = 1,\ \ k = 1, 2, \cdots \ . \tag{8}$$

It is denoted $\alpha^{(k)} = (\alpha_1^{(k)}, \alpha_2^{(k)}, \cdots, \alpha_m^{(k)})^T$.

**Algorithm 1. (SMTS)** The non-stationary parallel multisplitting two-stage iterative methods with self-adaptive weighting schemes

*Step 0.* *Given the precision* $\epsilon > 0$, *the initial point* $x^{(0)}$ *and set* $k := 0$; *For* $k = 0, 1, \cdots$, *until convergence.*

*Step 1.* *For all processors*

$$x_i^{(k,0)} = x^{(k)}\ ,$$

*Step 2.* *For processor* $i$, *for* $l = 0, 1, \cdots, q(i, k) - 1$

$$M_i x_i^{(k,l+1)} = N_i x^{(k,l)} + C_i x^{(k)} + b,\ \ i = 1, 2, \cdots, m\ . \tag{9}$$

*Step 3.* *Computing* $\alpha_i^{(k)}(i = 1, 2, \cdots, m)$ *by the following quadratic programming models.*

*(a) Let* $x = \sum_{i=1}^{m} \alpha_i x_i^{(k,q(i,k))}$,

$$\min_{\alpha} \frac{1}{2} x^T Ax - x^T b \quad s.t. \sum_{i=1}^{m} \alpha_i = 1\ . \tag{10}$$

*(b) Let* $r_i^{(k,q(i,k))} = Ax_i^{(k,q(i,k))} - b,\ \ r = \sum_{i=1}^{m} \alpha_i r_i^{(k,q(i,k))}$,

$$\min_{\alpha} r^T r \quad s.t. \sum_{i=1}^{m} \alpha_i = 1\ . \tag{11}$$

*Step 4.*

$$x^{(k+1)} = \sum_{i=1}^{m} \alpha_i^{(k)} x_i^{(k,q(i,k))}\ . \tag{12}$$

*Step 5.* *If* $\|Ax^{(k+1)} - b\| < \varepsilon$, *stop; Otherwise, set k:=k+1; Go to Step 1.*

By introducing matrices

$$G(i,k) = \sum_{l=0}^{q(i,k)-1} (M_i^{-1}N_i)^l M_i^{-1} \ , \qquad (13)$$

$$H(i,k) = (M_i^{-1}N_i)^{q(i,k)} + \sum_{l=0}^{q(i,k)-1} (M_i^{-1}N_i)^l M_i^{-1} C_i \ . \qquad (14)$$

We can rewrite the **SMTS** as the following iteration

$$x^{(k+1)} = \sum_{i=1}^{m} E_i^{(k)}(H(i,k)x^{(k)} + G(i,k)b) = H(k)x^{(k)} + G(k)b \ , \qquad (15)$$

where

$$H(k) = \sum_{i=1}^{m} E_i^{(k)} H(i,k), \quad G(k) = \sum_{i=1}^{m} E_i^{(k)} G(i,k) \ . \qquad (16)$$

It follows from straightforward derivation that

$$H(i,k) = I - G(i,k)A, \ \ i = 1,2,\cdots,m, \ \ k = 0,1,\ldots \ , \qquad (17)$$

and the iteration matrix

$$H(k) = I - G(k)A, \ \ k = 0,1,2,\cdots \ . \qquad (18)$$

For the quadratic programming, we have following results (see [9]).

Let

$$X(k) = (x_1^{(k,q(1,k))}, \cdots, x_m^{(k,q(m,k))}),$$

$$\alpha = (\alpha_1,\cdots,\alpha_m)^T, \ \ e = (1,\cdots,1)^T \ .$$

**Theorem 2.0.1.** Let $\{x_1^{(k,q(1,k))}, \cdots, x_m^{(k,q(m,k))}\}$ be linear independent, the solution of the quadratic programming (10) is as following

$$\alpha = (X(k)^T A X(k))^{-1}(X(k)^T b + \mu e) \ , \qquad (19)$$

where $\mu = \frac{1-e^T(X(k)^TAX(k))^{-1}X(k)^Tb}{e^T(X(k)^TAX(k))^{-1}e}$ .

**Theorem 2.0.2.** Let $\{r_1^{(k,q(1,k))}, \cdots, r_m^{(k,q(m,k))}\}$ be linear independent, the solution of the quadratic programming (11) is as following

$$\alpha = (R(k)^T R(k))^{-1}(R(k)^T b + \mu e) \qquad (20)$$

where $\mu = \frac{1-e^T(R(k)^TR(k))^{-1}R(k)^Tb}{e^T(R(k)^TR(k))^{-1}e}$.

## 3. Convergence Analysis

In this section, we study the convergence theories for algorithm 1 with self-adaptive weighting matrices.

**Lemma 3.0.3.** [11] Assume that $A$ is a symmetric positive definite matrix, let $A = M - N$ be $P$-regular splitting. Then there exists a positive number $r$ such that

$$\|A^{\frac{1}{2}}(M^{-1}N)A^{-\frac{1}{2}}\|_2 \le r < 1 \ . \qquad (21)$$

**Lemma 3.0.4.** [8] Assume that $A$ is a symmetric positive definite matrix, let $A = F - G$ is a $P$-regular splitting. Given $m \ge 1$, there exists a unique splitting $A = P - Q$ such that $(F^{-1}G)^m = P^{-1}Q$ and $A = P - Q$ is also a $P$-regular splitting.

**Lemma 3.0.5.** Assume that $A$ is a symmetric positive definite matrix. Let $A = B_i - C_i, i = 1,2,\cdots,m$ be symmetric positive definite splittings, and $B_i = M_i - N_i$ be P-regular splittings. If there exists a positive integer $q$ such that the non-stationary iteration number

$$q(i,k) \le q, \ \ k = 1,2,\cdots .$$

Then there exists a positive number $r$ such that

$$\|A^{\frac{1}{2}}H(i,k)A^{-\frac{1}{2}}\|_2 \le r < 1, \ \ i = 1,2,\cdots,m, \ \ k = 1,2,\cdots \ . \qquad (22)$$

*Proof.* We compute $G(i,k)$ directly

$$\begin{aligned} G(i,k) &= \sum_{l=0}^{q(i,k)-1} (M_i^{-1}N_i)^l M_i^{-1} && (23)\\ &= (I - (M_i^{-1}N_i)^{q(i,k)})(I - M_i^{-1}N_i)^{-1}M_i^{-1} \\ &= (I - (M_i^{-1}N_i)^{q(i,k)})B_i^{-1} \ . && (24) \end{aligned}$$

From Lemma 3.0.4, there exists a unique P-regular splitting

$$B_i = P_i(k) - Q_i(k), \quad i = 1,2,\cdots,m, \ k = 1,2,\cdots,$$

such that $P_i^{-1}(k)Q_i(k) = (M_i^{-1}N_i)^{q(i,k)}$. Hence, it is derived that

$$G(i,k) = (I - P_i^{-1}(k)Q_i(k))B_i^{-1} = P_i^{-1}(k),$$
$$i = 1,2,\cdots,m, \ k = 1,2,\cdots \ .$$

and thereby,

$$\begin{aligned} H(i,k) &= I - P_i^{-1}(k)A = P_i^{-1}(k)(P_i(k) - A) \\ &= P_i^{-1}(k)(B_i + Q_i(k) - (B_i - C_i)) \\ &= P_i^{-1}(k)(Q_i(k) + C_i), && (25)\\ & \quad i = 1,2,\cdots,m, \ k = 1,2,\cdots \ . \end{aligned}$$

From the assumptions of Lemma 3.0.5, the splitting

$$A = P_i(k) - (Q_i(k) + C_i), \ \ i = 1,2,\cdots,m, \ k = 1,2,\cdots \ , \qquad (26)$$

are P-regular splittings. Thus, there exist the positive numbers $r(i,k), \ i = 1,2,\cdots,m, \ k = 1,2,\cdots$, such that

$$\|A^{\frac{1}{2}}H(i,k)A^{-\frac{1}{2}}\|_2 \le r(i,k) < 1,$$

$$i = 1, 2, \cdots, m,\ k = 1, 2, \cdots .$$

Because of the $q(i,k) \le q$, $q(i,k) = 1, 2, \cdots, q$ has $q$ different values. Thus, the splittings (26) have at most $q$ different splittings, so are the positive numbers $r(i,k), i = 1, 2, \cdots, m,\ k = 1, 2, \cdots$. Hence, there exists a positive number $r$ such that (22) holds. □

**Theorem 3.0.6.** Assume that $A$ is a symmetric positive definite matrix. Let $A = B_i - C_i, i = 1, 2, \cdots, m$ be symmetric positive definite splitting, and $B_i = M_i - N_i$ be $P$-regular splittings. Suppose that weighting matrices $E_i^{(k)} = \alpha_i^{(k)} I,\ k = 1, 2, \cdots$ are given by (10). If there exists a positive integer $q$ such that the non-stationary iteration number $q(i,k) \le q$. Then $\{x^{(k)}\}$ generated by algorithm 1 converges to the unique solution of the linear system of equations (1).

*Proof.* Let $x^*$ be the unique solution of linear system of equations (1), and let $\varepsilon^{(k)} = x^{(k)} - x^*,\ k = 1, 2, \cdots$. From the algorithm 1, we have

$$\varepsilon^{(k+1)} = H(k)\varepsilon^{(k)},\ k = 1, 2, \cdots , \tag{27}$$

where

$$H(k) = \sum_{i=1}^{m} \alpha_i^{(k)} \left( (M_i^{-1}N_i)^{q(i,k)} + \sum_{l=0}^{q(i,k)-1} (M_i^{-1}N_i)^l M_i^{-1} C_i \right), \tag{28}$$
$$k = 1, 2, \cdots .$$

On the other hand, model (10) is equivalent to the following quadratic programming model,

$$\min_{\alpha} \frac{1}{2}(x - x^*)^T A (x - x^*)$$
$$s.t. \sum_{i=1}^{m} \alpha_i = 1. \tag{29}$$

From (29), we have

$$\varepsilon^{(k+1)^T} A \varepsilon^{(k+1)} \le \tilde{\varepsilon}_i^{(k+1)^T} A \tilde{\varepsilon}_i^{(k+1)}, \tag{30}$$
$$i = 1, 2, \cdots, m,\ k = 1, 2, \cdots ,$$

where

$$\tilde{\varepsilon}_i^{(k+1)} = H(i,k)\varepsilon^{(k)}, \tag{31}$$
$$i = 1, 2, \cdots, m,\ k = 1, 2, \cdots .$$

(27) and (28) combine (30) and (31), for $k = 1, 2, \cdots$, it holds that

$$\begin{aligned}
\|A^{\frac{1}{2}}\varepsilon^{(k+1)}\|_2 &= \|A^{\frac{1}{2}} H(k)\varepsilon^{(k)}\|_2 \le \|A^{\frac{1}{2}} H(i,k)\varepsilon^{(k)}\|_2 \\
&= \|A^{\frac{1}{2}} H(i,k) A^{-\frac{1}{2}} A^{\frac{1}{2}} \varepsilon^{(k)}\|_2 \\
&\le \|A^{\frac{1}{2}} H(i,k) A^{-\frac{1}{2}}\|_2 \|A^{\frac{1}{2}}\varepsilon^{(k)}\|_2 \\
&\le \cdots \\
&\le \Pi_{k=0}^{\infty} \|A^{\frac{1}{2}}(H(i,k))A^{-\frac{1}{2}}\|_2 \|A^{\frac{1}{2}}\varepsilon^{(0)}\|_2,
\end{aligned}$$

$$i = 1, 2, \cdots, m .$$

From Lemma 3.0.5, we have

$$\|A^{\frac{1}{2}} H(i,k) A^{-\frac{1}{2}}\|_2 \le r < 1,\ i = 1, 2, \cdots, m .$$

Thus,

$$\lim_{k\to\infty} \varepsilon^{(k+1)^T} A \varepsilon^{(k+1)} = 0 ,$$

which is equivalent to $\lim_{k\to\infty} \varepsilon^{(k+1)} = 0$. □

**Lemma 3.0.7.** Assume that $A$ is a nonsingular matrix, let $A = M - N$ be a convergent splitting. If the matrix $A^T M + M^T A - A^T A$ is symmetric positive definite, then

$$\|(A^T A)^{\frac{1}{2}} (M^{-1}N)((A^T A)^{-\frac{1}{2}}\|_2 < 1 .$$

*Proof.* At first, the matrix $A^T A - (M^{-1}N)^T A^T A (M^{-1}N)$ follows from direct operation that

$$\begin{aligned}
A^T A &- (M^{-1}N)^T A^T A (M^{-1}N) \\
&= A^T A - (I - A^T M^{-T}) A^T A (I - M^{-1}A) \\
&= A^T M^{-T} A^T A + A^T A M^{-1} A - A^T M^{-T} A^T A M^{-1} A \\
&= A^T M^{-T} (A^T M + M^T A - A^T A) M^{-1} A .
\end{aligned}$$

Hence, the matrix $A^T A - (M^{-1}N)^T A^T A M^{-1} N$ is symmetric positive definite if and only if the matrix $A^T M + M^T A - A^T A$ is symmetric positive definite. On the other hand, the matrix $A^T A - (M^{-1}N)^T A^T A M^{-1} N$ is symmetric positive definite if and only if $\|(A^T A)^{\frac{1}{2}} (M^{-1}N)(A^T A)^{-\frac{1}{2}}\|_2 < 1$. □

**Lemma 3.0.8.** Assume that $A$ is a nonsingular matrix. Let $A = B_i - C_i, i = 1, 2, \cdots, m$ be convergent splittings, and let $B_i = M_i - N_i,\ i = 1, 2, \cdots, m$ be also convergent splittings. Suppose the induced splitting

$$B_i = P_i(k) - Q_i(k),\ i = 1, 2, \cdots, m,\ k = 1, 2, \cdots ,$$

such that

$$P_i^{-1}(k) Q_i(k) = (M_i^{-1}N_i)^{q(i,k)}, i = 1, 2, \cdots, m,\ k = 1, 2, \cdots ,$$

and

$$A^T P_i(k) + P_i(k)^T A - A^T A, i = 1, 2, \cdots, m,\ k = 1, 2, \cdots .$$

are symmetric positive definite. If there exists a positive integer $q$ such that the non-stationary iteration number $q(i,k) \le q,\ i = 1, 2, \cdots, m,\ k = 1, 2, \cdots$. Then

$$\|(A^T A)^{\frac{1}{2}} H(i,k)((A^T A)^{-\frac{1}{2}})\|_2 < r(i,k) \le r < 1, \tag{32}$$
$$i = 1, 2, \cdots, m,\ k = 1, 2, \cdots ,$$

*Proof.* We apply Lemma 3.0.7 to the splitting

$$A = P_i(k) - (Q_i(k) + C_i),\ i = 1, 2, \cdots, m,\ k = 1, 2, \cdots ,$$

the (32) is obtained directly. □

**Theorem 3.0.9.** Assume that $A$ is a nonsingular matrix. Let $A = B_i - C_i, i = 1, 2, \cdots, m$ be convergent splittings, and let $B_i = M_i - N_i,\ i = 1, 2, \cdots, m,\ k = 1, 2, \cdots$ be also convergent splittings. Suppose that weighting matrices $E_i^{(k)} = \alpha_i^{(k)} I,\ \ i = 1, 2, \cdots, m,\ \ k = 1, 2, \cdots$ are given by (11). If the induced splitting $B_i = P_i(k) - Q_i(k),\ \ i = 1, 2, \cdots, m,\ k = 1, 2, \cdots$ such that

$$P_i^{-1}(k)Q_i(k) = (M_i^{-1}N_i)^{(q(i,k)}, \ i = 1, 2, \cdots, m, \ k = 1, 2, \cdots,$$

and

$$A^T P_i(k) + P_i(k)^T A - A^T A, \ i = 1, 2, \cdots, m, \ k = 1, 2, \cdots \ ,$$

are symmetric positive definite, then $\{x^{(k)}\}$ generated by algorithm 1 converges to the unique solution of the linear system of equations (1).

*Proof.* The model (11) is equivalent to the following quadratic programming model

$$\min_{\alpha} (x - x^*)^T A^T A(x - x^*)$$
$$s.t. \ \sum_{i=1}^{m} \alpha_i = 1 \ . \tag{33}$$

Thus, similar to Theorem 3.0.6, for $i = 1, 2, \cdots, m,\ k = 1, 2, \cdots$, it is derived that

$$\begin{aligned}
&\|(A^T A)^{\frac{1}{2}} \varepsilon^{(k+1)}\|_2 = \|(A^T A)^{\frac{1}{2}} H(k)\varepsilon^{(k)}\|_2 \\
&\le \|(A^T A)^{\frac{1}{2}} H(i,k)\varepsilon^{(k)}\|_2 \\
&= \|(A^T A)^{\frac{1}{2}} H(i,k)(A^T A)^{-\frac{1}{2}} (A^T A)^{\frac{1}{2}} \varepsilon^{(k)}\|_2 \\
&\le \|(A^T A)^{\frac{1}{2}} H(i,k)(A^T A)^{-\frac{1}{2}}\|_2 \|(A^T A)^{\frac{1}{2}} \varepsilon^{(k)}\|_2 \\
&\le \cdots \\
&\le \Pi_{k=0}^{\infty} \|(A^T A)^{\frac{1}{2}} H(i,k)(A^T A)^{-\frac{1}{2}}\|_2 \|(A^T A)^{\frac{1}{2}} \varepsilon^{(0)}\|_2 \ , \\
&\qquad i = 1, 2, \cdots, m.
\end{aligned}$$

From Lemma 3.0.8 we have

$$\|(A^T A)^{\frac{1}{2}} H(i,k)(A^T A)^{-\frac{1}{2}}\|_2 \le r(i,k) \le r < 1$$
$$i = 1, 2, \cdots, m, \ k = 1, 2, \cdots \ .$$

Thus,

$$\lim_{k\to\infty} \varepsilon^{(k+1)^T} (A^T A)\varepsilon^{(k+1)} = 0 \ .$$

so is the sequence $\{\varepsilon^{(k)}\}$. Hence, we have proved this theorem. □

*Remark* 3.0.10. The choice the optimization model of weighting matrices in $k$-th iteration can be various. Here, we only consider two schemes of optimizing weighting matrices for a linear system. In order to obtain self-adaptive weighting matrices, we need to solve the quadratic programming, but it may decrease the iterations largely because of the inequality implied in Theorem 3.0.6 and Theorem 3.0.9. Furthermore, we can parallel compute $\alpha$ as (19) and (20).

## 4. Numerical Experiments

In this section, we give some preliminary computational results. We implement our Algorithm 1 with three splittings (Gauss-Seidel splitting, Relaxation splitting and upper Gauss-Seidel splitting) to solve the linear system (1).

The test PDE problem we are considering in this paper is

$$-\Delta u \equiv -\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = f(x, y) \tag{34}$$

with $(x, y) \in \Omega$, where $\Omega = (0, 1) \times (0, 1)$ is a square region. In all cases, the initial vector $x^{(0)}$ is set to zero and the stopping criterion for Algorithm 1 is

$$\frac{\|b - Ax^{(k)}\|_2}{\|b\|_2} \le 10^{-6} \ .$$

where $\|\cdot\|_2$ refers to $L_2$-norm. In the following Tables, IT stands for the number of iterations satisfying the stopping criterion mentioned above, CPU stands for the parallel execution time of Algorithm 1. All timing results are reported in seconds. For the test problems, only the matrix $A$, which is constructed from finite difference discretization of the given PDE (34), is of importance, so the right-hand side vector $b$ is created artificially. Hence, the right-hand side function $f(x, y)$ in Examples 1 and 2 is not relevant.

*Example 1* This example considers equation $Ax = b$ obtained from nine-point finite difference discretization of the given PDE (34). So the coefficient matrix

$$A = \begin{pmatrix} D_p & G_p & & & \\ G_p & D_p & G_p & & \\ & \ddots & \ddots & \ddots & \\ & & G_p & D_p & G_p \\ & & & G_p & D_p \end{pmatrix}_{q\times q},$$

where

$$D_p = \begin{pmatrix} 20 & -4 & & & \\ -4 & 20 & -4 & & \\ & \ddots & \ddots & \ddots & \\ & & -4 & 20 & -4 \\ & & & -4 & 20 \end{pmatrix}_{p\times p},$$

$$G_p = \begin{pmatrix} -4 & -1 & & & \\ -1 & -4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & -4 & -1 \\ & & & -1 & -4 \end{pmatrix}.$$

and the right-hand side vector b is chosen so that $b = (1, 2, 3, \cdots, n)^T$.

*Example 2* This example considers equation $Ax = b$ from five-point finite difference discretization of the given PDE (34). So the matrix $A$ is constructed as in Example 1, but $D_p$ and $G_p$ are different from Example 1, that is
$$D_p = \begin{pmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{pmatrix}_{p\times p} \text{ and } G_p = -I,$$
and the right-hand side vector is chosen so that $b = (1, 1, \cdots, 1)^T$.

In all our numerical experiments, three splittings of the matrix $A$ are proposed as following. Let

$$A = B_i - C_i,\ i = 1, 2, 3$$

with
$$B_i = \begin{pmatrix} D_{ip} & G_{ip} & & & \\ G_{ip} & D_{ip} & G_{ip} & & \\ & \ddots & \ddots & \ddots & \\ & & G_{ip} & D_{ip} & G_{ip} \\ & & & G_{ip} & D_{ip} \end{pmatrix}.$$

Especially in Examples 1, we chose

$$D_{1p} = \begin{pmatrix} 24 & -4 & & & \\ -4 & 24 & -4 & & \\ & \ddots & \ddots & \ddots & \\ & & -4 & 24 & -4 \\ & & & -4 & 24 \end{pmatrix},$$

$$G_{1p} = \begin{pmatrix} -2 & -1 & & & \\ -1 & -2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & -2 & -1 \\ & & & -1 & -2 \end{pmatrix},$$

$$D_{2p} = \begin{pmatrix} 22 & -4 & & & \\ -4 & 22 & -4 & & \\ & \ddots & \ddots & \ddots & \\ & & -4 & 22 & -4 \\ & & & -4 & 22 \end{pmatrix},$$

$$G_{2p} = \begin{pmatrix} -3 & -1 & & & \\ -1 & -3 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & -3 & -1 \\ & & & -1 & -3 \end{pmatrix},$$

$$D_{3p} = \begin{pmatrix} 26 & -3 & & & \\ -3 & 26 & -3 & & \\ & \ddots & \ddots & \ddots & \\ & & -3 & 26 & -3 \\ & & & -3 & 26 \end{pmatrix},$$

$$G_{3p} = \begin{pmatrix} -4 & & & \\ & -4 & & \\ & & \ddots & \\ & & & -4 \end{pmatrix}.$$

and in Examples 2, we chose

$$D_{1p} = \begin{pmatrix} 10 & -1 & & & \\ -1 & 10 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 10 & -1 \\ & & & -1 & 10 \end{pmatrix},$$

$$G_{1p} = \begin{pmatrix} -3 & & & & \\ & -3 & & & \\ & \ddots & \ddots & \ddots & \\ & & & -3 & \\ & & & & -3 \end{pmatrix},$$

$$D_{2p} = \begin{pmatrix} 8 & -2 & & & \\ -2 & 8 & -2 & & \\ & \ddots & \ddots & \ddots & \\ & & -2 & 8 & -2 \\ & & & -2 & 8 \end{pmatrix},$$

$$G_{2p} = \begin{pmatrix} -2 & & & & \\ & -2 & & & \\ & \ddots & \ddots & \ddots & \\ & & & -2 & \\ & & & & -2 \end{pmatrix},$$

$$D_{3p} = \begin{pmatrix} 12 & -2 & & & \\ -2 & 12 & -2 & & \\ & \ddots & \ddots & \ddots & \\ & & -2 & 12 & -2 \\ & & & -2 & 12 \end{pmatrix},$$

$$G_{3p} = \begin{pmatrix} -2 & -1 & & & \\ -1 & -2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & -2 & -1 \\ & & & -1 & -2 \end{pmatrix}.$$

Let
$$B_i = D_i - L_i - L_i^T, i = 1, 2, 3\ , \tag{35}$$
where $D_i = diag(D_{i,p}, \cdots, D_{i,p}), i = 1, 2, 3.$ and corresponding to the $D_i$ block, $L_i$ is strictly block lower triangular matrix. $M_i$ and $N_i$ of Algorithm 1 are determined by the following three splitting methods.

The Gauss-Seidel splitting method
$$M_1 = D_1 - L_1, \quad N_1 = L_1^T\ ; \tag{36}$$

**Table 1.** Comparison of computational results for Example 1

| p | | SMTS with (11) | SMTS with (10) | old Alg with (i) | old Alg with (ii) | old Alg with (iii) |
|---|---|---|---|---|---|---|
| 20 | IT | 49 | 53 | 538 | 382 | 284 |
| | CPU | 0.541037 | 0.537481 | 3.817864 | 3.772102 | 2.799893 |
| 30 | IT | 99 | 111 | 1152 | 819 | 636 |
| | CPU | 2.417201 | 2.775316 | 27.843393 | 19.953243 | 15.499852 |
| 40 | IT | 186 | 193 | 2003 | 1424 | 1003 |
| | CPU | 9.365693 | 9.571924 | 100.408128 | 70.903005 | 50.194703 |
| 50 | IT | 294 | 274 | 3089 | 2196 | 1902 |
| | CPU | 23.093113 | 21.421537 | 258.096561 | 173.782059 | 151.376797 |
| 60 | IT | 424 | 377 | 4412 | 3136 | 2233 |
| | CPU | 50.778948 | 44.560516 | 531.147234 | 396.874754 | 267.580311 |
| 70 | IT | 582 | 391 | 5970 | 4244 | 3448 |
| | CPU | 103.469121 | 68.018041 | 1067.300887 | 748.579123 | 609.459062 |
| 80 | IT | 769 | 463 | 7765 | 5520 | 4126 |
| | CPU | 308.959919 | 149.931800 | 2107.084033 | 1363.542538 | 1039.778497 |

**Table 2.** Comparison of computational results for Example 2

| p | | SMTS with (11) | SMTS with (10) | old Alg with (i) | old Alg with (ii) | old Alg with (iii) |
|---|---|---|---|---|---|---|
| 20 | IT | 14 | 20 | 162 | 136 | 112 |
| | CPU | 0.132377 | 0.194257 | 1.552457 | 1.284103 | 1.080881 |
| 40 | IT | 40 | 44 | 529 | 440 | 424 |
| | CPU | 1.929001 | 2.143014 | 25.336636 | 20.901150 | 20.130860 |
| 60 | IT | 89 | 67 | 1130 | 940 | 851 |
| | CPU | 10.163563 | 7.775480 | 129.333241 | 107.352299 | 105.052453 |
| 80 | IT | 161 | 110 | 1967 | 1637 | 1671 |
| | CPU | 40.565789 | 26.947623 | 496.462925 | 389.863508 | 398.499981 |
| 100 | IT | 251 | 175 | 3039 | 2531 | 2469 |
| | CPU | 105.032853 | 73.645276 | 1177.090124 | 1046.563264 | 1021.525050 |
| 120 | IT | 363 | 244 | 4346 | 3621 | 3467 |
| | CPU | 224.098729 | 151.474390 | 2630.780502 | 2307.731365 | 2101.069915 |

The SOR splitting method

$$M_2 = \frac{1}{\omega}(D_2 - \omega L_2), \quad N_2 = \frac{1}{\omega}((1-\omega)D_2 + \omega L_2^T) \ ; \quad (37)$$

The upper Gauss-Seidel splitting method

$$M_3 = D_3 - L_3^T, \quad N_3 = L_3 \ . \quad (38)$$

In addition, the weighting matrices $E_i^{(k)} = \alpha_i^{(k)} I, \quad i = 1,2,3, \quad k = 1,2,\cdots$.

In order to compare old algorithm with the fixed weighting matrices, we propose the fixed weighting matrices as following,

(i) $E_i = \alpha_i I, \ i = 1,2,3,$ with $\alpha_1 = 0.2, \ \alpha_2 = 0.2 \ \alpha_3 = 0.6$;

(ii) $E_i = \alpha_i I, \ i = 1,2,3,$ with $\alpha_1 = 0.4, \ \alpha_2 = 0.3, \ \alpha_3 = 0.3$;

(iii) $E_1 = diag(\alpha_1 I_p, \alpha_2 I_p, \cdots, \alpha_q I_p)$,
$E_2 = diag(\beta_1 I_p, \beta_2 I_p, \cdots, \beta_q I_p)$,
$E_3 = diag(\gamma_1 I_p, \gamma_2 I_p, \cdots, \gamma_q I_p)$,
where $\alpha_i$ and $\beta_i (i = 1,2\cdots q)$ are generated **randomly** in (0,1), and $\gamma_i = 1 - \alpha_i - \beta_i$.

In all our tests we take $p = q, \omega = 1.5, q(i,k) = 5$. Numerical results for Example 1 and Example 2 are listed in Tables 1 and Tables 1, respectively.

In Example 2, the coefficient matrix $A$ itself contains more zero entries than the matrix of Example 1. So we choose larger $p$. From Table 1 and Table 2 we see that the iteration counts and the CPU times of SMTS with (11) grow rapidly than SMTS with (10) with problem size, but they are much less than the usual old algorithm with fixed weighting matrices. The reason is that the nonnegativity of weighting matrices are deleted, the range for finding the optimal weighting matrices is extended. The iteration counts and the CPU times of the old algorithm with (iii) is not stable because of randomly, so we have chosen lesser iteration number than the old algorithms with (i) and (ii). Numerical experiments have been presented showing the effectiveness of the self-adaptive strategy for weighting matrices.

*Acknowledgement* This work is supported by NSF of China (11071184) and NSF of Shanxi Province (201001006, 2012011015-6).

## References

[1] Bai Z. Z. and Sun J. C. and Wang D. R. (1996) A unified framework for the construction of various matrix multisplitting iterative methods for large sparse system of linear equations. *Comput. Math. Appl.* **32**(12):51–76. doi:10.1016/S0898-1221(96)00207-6.

[2] Bai Z. Z. (1997) A class of two-stage iterative methods for systems of weakly nonlinear equations. *Numer. Alg.* **14**(4): 295–319. doi:10.1023/A:1019125332723.

[3] Bai Z. Z. (1997) Parallel multisplitting two-stage iterative methods for large sparse systems of weakly nonlinear equations. *Numer. Alg.* **15**(3–4):(347–372). doi:10.1023/A:1019110324062.

[4] Bai Z. Z. (1998) The convergence of the two-stage iterative method for hermitian positive definite linear systems. *Appl. Math. Lett.* **11**(2): 1–5. doi:10.1016/S0893-9659(98)00001-9.

[5] Bai Z. Z. (1999) Convergence analysis of two-stage multisplitting method. *Calcolo* **36**(63–74).

[6] Bai Z. Z. and Wang C. L. (2003) Convergence theorems for parallel multisplitting two-stage iterative methods for mildly nonlinear systems. *Lin .Alg. Appl.* **362**: 237–250. doi:10.1016/S0024-3795(02)00518-9.

[7] Cao Z. H. and Liu Z. Y. (1998) Symmetric multisplitting of a symmetric positive definite matrix. *Lin .Alg. Appl.* **285**(1): 309–319. doi:10.1016/S0024-3795(98)10151-9.

[8] Castel M. J. and Migallón V. and Penadés J. (1998) Convergence of non-stationary parallel multisplitting methods

for Hermitian positive definite matrices. *Math. Comput.* **67**(221): 209–220. doi:10.1090/S0025-5718(98)00893-X.
[9] Fletcher R. (2000) Practical Methods of Optimization. *Americ: John Wiley and Sons Inc.* 2th ed.
[10] Frommer A. and Szyld D. B. (1992) *H*-splitting and two-stage iterative methods. *Numer. Math.* **63**(1): 345–356. doi:10.1007/BF01385865.
[11] Frommer A. and Szyld D. B. (1999) Weighted max norms, splitttngs,and overlapping additive Schwarz iterations. *Numer. Math.* **83**(2): 259–278. doi:10.1007/s002110050449.
[12] Gu T. X. and Liu X. P. and Shen L. J. (2000) Relaxed parallel two-stage multisplitting methods. *Int. J. Computer Math.* **75**(3): 351–363. doi:10.1080/00207160008804990.
[13] Migallón V. and Penadés J. and Szyld D. B. (2001) Nonstationary multisplittings with general weighting matrices. *SIAM J. Matrix Anal. Appl.* **22**(4): 1089–1094. doi:10.1137/S0895479800367038.
[14] O'Leary D. P. and White R. E. (1985) Multi-splittings of matrices and parallel solutions of linear systems. *SIAM J. Alg. Disc. Meth.* **6**(4): 630–640. doi:10.1137/0606062.
[15] Szyld D. B. and Jones M. T. (1992) Two-stage and multisplitting methods for the parallel solution of linear systems. *SIAM J. Matrix Anal. Appl.* **13**(2): 671–679. doi:10.1137/0613042.
[16] Varga R. S. (2000) Matrix Iterative Analysis. *Germany:Springer Berlin Heidelberg*. 2th ed.
[17] Wang C. L. and Bai Z. Z. (2002) On the convergence of nonstationary multisplitting two-stage iteration methods for Hermitian positive definite linear systems. *J. Comput. Appl. Math.* **138**(2): 287–296. doi:10.1016/S0377-0427(01)00376-4.
[18] Wen R. P. and Wang C. L. and Meng G. Y. (2007) Convergence theorems for block splitting iterative methods for linear systems. *J. Comput.Appl. Math.* **202**(2): 540–547. doi:10.1016/j.cam.2006.03.006.
[19] White R. E. (1989) Multisplitting with different weighting schemes. *SIAM J. Matrix Anal. Appl.* **10**(4): 481–493. doi:10.1137/0610034.
[20] White R. E. (1990) Multisplitting of a symmetric positive definite matrix. *SIAM J. Matrix Anal. Appl.* **11**(1): 69–82. doi:10.1137/0611004
[21] Chen F. (2010) Asynchronous multisplitting iteration with different weighting schemes. *Applied Mathematics and Computation.* **216**(6): 1771-1776. doi:10.1016/j.amc.2009.12.027.
[22] Meng G. Y. and Wang C. L. Yan X. H. (2012) Self-Adaptive Non-stationary Parallel Multisplitting Two-Stage Iterative Methods for Linear Systems. *In Proceedings of Third International conference ICDKE2012, Wuyishan, Fujian, China* (Germany:Springer Berlin Heidelberg), LNCS 7696, 38–47. doi:10.1007/978-3-642-34679-8-4.
[23] Wang C. L. and Meng G. Y. and Yong X. R. (2011) Modified parallel multisplitting iterative methods for non-Hermitian positive definite systems. *Adv. Comput. Math.* 1–14. doi: 10.1007/s10444-011-9262-8.