World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:7, No:4, 2013

Some Preconditioners for Block Pentadiagonal Linear Systems Based on New Approximate Factorization Methods

Xian Ming Gu, Ting Zhu Huang, and Hou Biao Li

Abstract—In this paper, getting an high-efficiency parallel algorithm to solve sparse block pentadiagonal linear systems suitable for vectors and parallel processors, stair matrices are used to construct some parallel polynomial approximate inverse preconditioners. These preconditioners are appropriate when the desired target is to maximize parallelism. Moreover, some theoretical results about these preconditioners are presented and how to construct preconditioners effectively for any nonsingular block pentadiagonal H-matrices is also described. In addition, the availability of these preconditioners is illustrated with some numerical experiments arising from two dimensional biharmonic equation.

Keywords—Parallel algorithm, Pentadiagonal matrix, Polynomial approximate inverse, Preconditioners, Stair matrix.

I. Introduction

PENTADIAGONAL matrix is a general and important type of special matrices, these types of matrices are widely used in areas of science and engineering. For example, using finite difference method or finite element method to discrete partial differential equations (PDEs) in 2D or 3D, leads often to large sparse block pentadiagonal linear systems [1], [2], [3], [4], [5], [6], [7], [8]. As well as, Linear algebraic equations of the form (1) are obtained, In this paper, we consider a special linear system of the form

$$Ax = b, \quad x, b \in \mathcal{R}^n, \tag{1}$$

where $A \in \mathcal{R}^{n\times n}$ is a large sparse block pentadiagonal matrix blocked in the form

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} & & & \\ A_{21} & A_{22} & A_{23} & & & \\ A_{31} & A_{32} & A_{33} & & & \\ & \ddots & \ddots & \ddots & \ddots & \\ & & & & A_{n-2,n-1} & A_{n-2,n} \\ & & & & A_{n-1,n-1} & A_{n-1,n} \\ & & & A_{n,n-2} & A_{n,n-1} & A_{n,n} \end{pmatrix}, \tag{2}$$

It is assumed that the diagonal blocks A_{ii} of A are square matrices with the same order. For the above block pentadiagonal matrix (2), we briefly denote

$$A = pentadiag(A_{i,i-2}, A_{i,i-1}, A_{ii}, A_{i,i+1}, A_{i,i+2}),$$

where $i = 1, \cdots, n,\ n \geq 3$. In fact, a stair matrix defined in [29] can be regarded as a special block pentadiagonal matrix (see Definition 2.1 of Section 2), we use analogous representations of [29] to definite a new type of stair matrix.

In principle, there are two groups of methods for the solution of linear systems (1). One group of methods are the so-called direct methods, or elimination methods, that is, the exact solution is determined through a finite number of arithmetic operations (in real arithmetic without considering the roundoff errors). It is not efficient to obtain the exact solution of (1) by using a direct method such as Gaussian elimination, if the coeffient matrix A is large and sparse, Since A is a large sparse matrix, direct methods become prohibitively expensive because of a lot of fillin elements. Iterative methods therefore competitive with direct methods provided the number of iterations that are required to converge is either independent of n or scales sublinearly with respects to n, As an alternative, we usually consider nonstationary iterative methods-conjugate gradient method and Krylov subspace methods such as BCG [19], GMRES [32], and BiCGSTAB [33], etc. However, in general, the convergence of Krylov subspace methods is not guaranteed or may be extremely slow [11], [31]. Hence, the original problem (1) must be transformed into a more tractable form. To do so, preconditioned Krylov subspace methods are widely used, that is, we consider an easily invertible matrix M called the preconditioning matrix or preconditioner and apply the iterative solvers either to the left preconditioned linear system $MAx = Mb$ or to the right preconditioned linear system $AMy = b$, where $y = M^{-1}x$ (as well as considering the fast multipole methods).

Generally speaking, the preconditioner M should be chosen so that MA or AM is a good approximation to the identity matrix and has to satisfy the following three conditions [24]:
(1) AM (or MA) should have aclusteredspectrum;
(2) M should be efficiently and suitable computable in parallel;
(3) "$M \times$ vector" should be fast and time-saving to compute in parallel.

Recently, various preconditioners have been widely introduced in the literature [11], [31]. Often used preconditioners are block Jacobi preconditioners, polynomial preconditioner or incomplete LU-decompositions of A [13]. But these preconditioners either lead to unsatisfactory convergence or are not easily implemented for parallel computation [13]. However, Noting that H.B. Li *et al.* present a new stair matrix splitting for block tridiagonal matrix by using ideas of [29] and structures some new parallel preconditioners for block tirdiagonal linear systems suitable for vectors and parallel processors [36].

X. M. Gu, T. Z. Huang, and H. B. Li are all with School of Mathematics Science, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, P. R., China. e-mail: guxianming@yahoo.cn (corresponding author); tingzhuhuang@126.com (corresponding author)

A very promising preconditioner is the sparse approximate inverse (SAI) preconditioner [11], [18], [20], [31], which is a sparse matrix M that directly approximates the inverse of the coefficient matrix A, i.e.,

$$M \approx A^{-1}, \tag{3}$$

Thus, in the basic iterative scheme only matrix-vector multiplications with M appear and it is not essential to solve a linear system in M like in the incomplete LU-approach. However, A^{-1} is a full matrix in general, and hence not for every sparse matrix A there will exist a good sparse approximate inverse matrix M.

Another interesting approach is the polynomial preconditioners [31], based on a splitting, $A = P - Q$, of A, where P is nonsingular. If $H = P^{-1}Q$ and $\rho(H) < 1$ (Here, $\rho(H)$ denotes the spectral radius of H), then (see [30], Theorem 3.4.1) one has

$$A^{-1} = (\sum_{i=1}^{\infty} H^i)P^{-1}, \tag{4}$$

The Neumann expansion (4) suggests taking the matrix $M_m = P(I + H + H^2 + \cdots + H^{m-1})^{-1}$ for $m = 1, 2, 3, \ldots$, as an approximation to A. This matrix is called the *m-step polynomial preconditioner* [9]. Thus, depending on the splitting $A = P - Q$, specific preconditioners may be obtained.

To obtain efficient algorithms in parallel systems, a generalization is introduced in [12], [16], [17] using multisplittings:given k splitting $A = P_l - Q_l, l = 1, 2, \ldots, k$, of A, the m-step preconditioner is defined by

$$\begin{aligned} M^{-1} = \tfrac{1}{k}(I + W + W^2 + \cdots + W^{m-1})(P_1^{-1} + \\ \cdots + P_k^{-1}) \quad \text{and} \quad W = \tfrac{1}{k}\textstyle\sum_{l=1}^{k} P_l^{-1}Q_l. \end{aligned} \tag{5}$$

Hence, according to the above SAI and the m-step polynomial preconditioner, the purpose of this paper is to propose some new parallel polynomial approximate inverse preconditioners for the above block pentadiagonal matrix (2) and the corresponding computation can be done in parallel based on sparse block matrix-vector multiplications.

The remainder of this paper is organized as follows. In Section 2, we present some notations, definitions and preliminary results on stair matrices, which we refer to later. In Section 3, by exploiting stair matrices, we describe how to construct the block polynomial preconditioners effectively for the special type of matrix (2) and their theoretical properties are investigated. In Section 4, we present some numerical results of the preconditioned BiCGSTAB method with our polynomial preconditioners, and these results are compared with those polynomial preconditioners using standard block Jacobi splitting. Finally, some conclusions are drawn.

II. Preliminaries and notations

In this section, we will recall some properties of stair matrices defined in [29]. These properties will be useful in the following sections since iterative methods based on these matrices can be easily performed on a parallel computing platform.

From now on, we shall use the following notations and definitions: Let $\mathcal{R}^n$ and $\mathcal{C}^{n\times n}(\mathcal{R}^{n\times n})$ be the n-dimensional real vector space and the set of all $n \times n$ complex (real) matrices, respectively. We denote $A = (a_{ij})$ an $n \times n$ matrix and set offdiag$(A) = A - diag(A)$. For two matrices $A = (a_{ij})$ and $B = (b_{ij})$, $A \le B$ denotes $a_{ij} \le b_{ij}$ for all i and j,and $A \ge B$ denotes $a_{ij} \ge b_{ij}$ for all i and j.

We now recall stair matrices (see Definition 2.1) and their properties (see Theorem 2.1) introduced in the first part of [29]. All notations are similar to those in [29].

Definition 2.1: An $n \times n$ block pentadiagonal matrix

$$A = pentadiag(A_{i,i-2}, A_{i,i-1}, A_{ii}, A_{i,i+1}, A_{i,i+2}),$$

is called a stair matrix if one of the following conditions is satisfied.

$$(I) \left\{ \begin{array}{ll} A_{i,i-2},\ A_{i,i+2} \neq 0, & i = 3, \ldots, 3 + 4\lfloor \frac{n-3}{4} \rfloor; \\ A_{i,i-1} \neq 0, & i = 2, 3, \ldots, 2 + 4\lfloor \frac{n-3}{4} \rfloor, 3 + 4\lfloor \frac{n-3}{4} \rfloor; \\ A_{i,i+1} \neq 0, & i = 3, 4, \ldots, 3 + 4\lfloor \frac{n-3}{4} \rfloor, 4 + 4\lfloor \frac{n-3}{4} \rfloor; \\ A_{ii} \neq 0, & i = 1, \ldots, n. \end{array} \right.$$

$$(II) \left\{ \begin{array}{ll} A_{i,i-2},\ A_{i,i+2} \neq 0, & i = 1, 5, \ldots, 1 + 4\lfloor \frac{n-3}{4} \rfloor, 5 + 4\lfloor \frac{n-3}{4} \rfloor; \\ A_{i,i-1} \neq 0, & i = 4, 5, \ldots, 4 + 4\lfloor \frac{n-3}{4} \rfloor, 5 + 4\lfloor \frac{n-3}{4} \rfloor; \\ A_{i,i+1} \neq 0, & i = 1, 2, \ldots, 1 + 4\lfloor \frac{n-3}{4} \rfloor, 2 + 4\lfloor \frac{n-3}{4} \rfloor; \\ A_{ii} \neq 0, & i = 1, \ldots, n. \end{array} \right.$$

Where $A_{ij} \neq 0$ stand for the block entries of previous block pentadiagnoal matrix constraining invariant, others of the block matrix are zero, and a stair matrix is of type I if condition I is satisfied and is of type II if condition II holds.

According to its form, a stair matrix is denoted by

$$A = stair(A_{i(i-2)}, A_{i(i-1)}, A_{ii}, A_{i(i+1)}, A_{i(i+2)}).$$

In particular,

$$A = stair1(A_{i(i-2)}, A_{i(i-1)}, A_{ii}, A_{i(i+1)}, A_{i(i+2)}),$$

and

$$A = stair2(A_{i(i-2)}, A_{i(i-1)}, A_{ii}, A_{i(i+1)}, A_{i(i+2)}),$$

represent a stair matrix of type I and a stair matrix of type II, respectively.

Theorem 2.1: An $n \times n$ block stair matrix

$$A = stair(A_{i(i-2)}, A_{i(i-1)}, A_{ii}, A_{i(i+1)}, A_{i(i+2)}),$$

is nonsingular if and only if $A_{ii}, i = 1, 2, \cdots, n$ are nonsingular. Furthermore, if A is nonsingular, then

$$A^{-1} = stair(B_{i(i-2)}, B_{i(i-1)}, B_{ii}, B_{i(i+1)}, B_{i(i+2)}), \tag{6}$$

where the block B_{ij} are given by

$$B_{ij} = \left\{ \begin{array}{l} -A_{ii}^{-1}A_{ij}A_{jj}, \quad \text{if } j = i-1, i+1; \\ A_{ii}^{-1}, \quad \text{if } j = i; \\ S_{ij}, \quad \text{if } j = i-2, i+2. \end{array} \right. \tag{7}$$

In fact, where $S_{ij} = A_{ij} - A_{i,j+1}A_{j+1,j+1}^{-1}A_{j+1,j}$ $(j = i - 2, i + 2)$ are well-known Schur complements.

For example, a 5×5 block stair matrix is of the form as follows

$$\begin{pmatrix} A_{11} & & & & \\ A_{21} & A_{22} & & & \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} \\ & & & A_{45} & A_{45} \\ & & & & A_{55} \end{pmatrix},$$

A stair matrix of type I

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} & & \\ & A_{22} & A_{23} & & \\ & & A_{33} & & \\ & & A_{43} & A_{44} & \\ & & A_{53} & A_{54} & A_{55} \end{pmatrix},$$

A stair matrix of type II

If $det(A) \neq 0$, then, by Theorem 2.1, we, respectively, have that

$$\begin{pmatrix} A_{11}^{-1} & & & & \\ U_{21} & A_{22}^{-1} & & & \\ U_{31} & U_{32} & A_{33}^{-1} & U_{34} & U_{35} \\ & & & A_{44}^{-1} & -A_{44}^{-1}A_{45}A_{55}^{-1} \\ & & & & A_{55}^{-1} \end{pmatrix}, \quad (8)$$

The inverse matrix of a stair matrix of the type I

where $U_{21} = -A_{22}^{-1}A_{21}A_{11}^{-1}$, $U_{32} = -A_{33}^{-1}A_{32}A_{22}^{-1}$, $U_{34} = -A_{33}^{-1}A_{34}A_{44}^{-1}$, $U_{35} = A_{33}^{-1}(A_{34}A_{44}^{-1}A_{45} - A_{35})A_{55}^{-1}$ and $U_{31} = A_{33}^{-1}(A_{32}A_{22}^{-1}A_{21} - A_{31})A_{11}^{-1}$. or

$$\begin{pmatrix} A_{11}^{-1} & L_{12} & L_{13} & & \\ & A_{22}^{-1} & -A_{22}^{-1}A_{23}A_{33}^{-1} & & \\ & & A_{33}^{-1} & & \\ & & -A_{44}^{-1}A_{43}A_{33}^{-1} & A_{44}^{-1} & \\ & & L_{53} & L_{54} & A_{55}^{-1} \end{pmatrix}. \quad (9)$$

The inverse matrix of a stair matrix of the type II

where $L_{12} = -A_{11}^{-1}A_{12}A_{22}^{-1}$, $L_{54} = -A_{55}^{-1}A_{54}A_{44}^{-1}$, $A_{11}^{-1}(A_{12}A_{22}^{-1}A_{23} - A_{13})A_{33}^{-1}$, $L_{53} = A_{55}^{-1}(A_{54}A_{44}^{-1}A_{43} - A_{53})A_{33}^{-1}$

Applying Theorem 2.1, we immediately obtain the following Algorithm I to compute the inverse matrices of stair matrices.

Algorithm I: Let

$$A = stair(A_{i(i-2)}, A_{i(i-1)}, A_{ii}, A_{i(i+1)}, A_{i(i+2)})$$

be a nonsingular block stair matrix and

$$A^{-1} = stair(B_{i(i-2)}, B_{i(i-1)}, B_{ii}, B_{i(i+1)}, B_{i(i+2)})$$

If (A is of the type I)
for $i = 1:1:n$
$B_{ii} = A_{ii}^{-1}$
endfor i
for $i = 2:4:2+4\lfloor\frac{n-3}{4}\rfloor$ and $i = 3:4:3+4\lfloor\frac{n-3}{4}\rfloor$ and $i = 4:4:4+4\lfloor\frac{n-3}{4}\rfloor$
$B_{ij} = -A_{ii}^{-1}A_{ij}A_{jj}, j = i-1, i+1$
endfor i
for $i = 3:4:3+4\lfloor\frac{n-3}{4}\rfloor$
$B_{ij} = A_{ii}^{-1}(A_{i,j+1}A_{j+1,j+1}^{-1}A_{j+1,j} - A_{ij})A_{jj}^{-1},\ j = i-2, i+2$
endfor i
endif
If (A is of the type II)
for $i = 1:1:n$
$B_{ii} = A_{ii}^{-1}$
endfor i
for $i = 4:4:4+4\lfloor\frac{n-3}{4}\rfloor$ and $i = 5:4:5+4\lfloor\frac{n-3}{4}\rfloor$ and $i = 1:4:1+4\lfloor\frac{n-3}{4}\rfloor$ and $i = 2:4:2+4\lfloor\frac{n-3}{4}\rfloor$
$B_{ij} = -A_{ii}^{-1}A_{ij}A_{jj}, j = i-1, i+1$
endfor i
for $i = 1:4:1+4\lfloor\frac{n-3}{4}\rfloor$ or $i = 5:4:5+4\lfloor\frac{n-3}{4}\rfloor$
$B_{ij} = A_{ii}^{-1}(A_{i,j+1}A_{j+1,j+1}^{-1}A_{j+1,j} - A_{ij})A_{jj}^{-1},\ j = i-2, i+2$
endfor i
endif

where $B_{ii} = 0$, if $i < 1$ or $i > n$. A remarkable feature of the algorithm I is its high parallelism. For example, if A is a stair matrix of the type I, first, for all i, the computations of B_{ii} can be fulfilled by different processors at same time, and then proceed to compute B_{ij} in parallel for $i = 2:4:2+4\lfloor\frac{n-3}{4}\rfloor$, $i = 3:4:3+4\lfloor\frac{n-3}{4}\rfloor$ and $i = 4:4:4+4\lfloor\frac{n-3}{4}\rfloor$ at the same time. Thus, if n is reasonably large, we may achieve fair parallelism.

III. Factorized sparse approximate inverse block polynomial preconditioners for block pentadiagonal matrices

Now, based on the above analysis, we construct an effective block polynomial preconditioner for any nonsingular block pentadiagonal H-matrix.

Let $A = pentadiag(A_{i,i-2}, A_{i,i-1}, A_{ii}, A_{i,i+1}, A_{i,i+2})$ as in (2), and a representation $A = S-P$ is called a stair-splitting of A when $S = stair(A_{i,i-2}, A_{i,i-1}, A_{ii}, A_{i,i+1}, A_{i,i+2})$ is nonsingular. If $\rho(S^{-1}P) < 1$, then the stair-splitting is a convergent splitting of A and one has

$$A^{-1} = (I - S^{-1}P)^{-1}S^{-1}, \quad (10)$$

and

$$\begin{aligned} A^{-1} \approx M_m = (I - S^{-1}P + (S^{-1}P)^2 + \cdots \\ +(S^{-1}P)^{m-1})S^{-1}, \quad m = 1, 2, \ldots. \end{aligned} \quad (11)$$

The matrix M_m is called the m-step polynomial preconditioner for block pentadiagonal matrices. If the terms that have been dropped in the Neumann series (11) are of small norm, the matrix M_m is close to A^{-1} and it can be used an effective preconditioner. Quantifying the "deviation" gives $\|A^{-1} - M_m\|^2 = O(\|S^{-1}P\|_2^m)$.

Remark 3.1: It is worth noting that the condition $\rho(S^{-1}P) < 1$ is not too difficult to be satisfied in general [22], [25], [26], [27], [35], see also the following Theorems 3.1 and 3.2. In addition, the matrix $S^{-1}P$ is also very interesting that if S is a stair matrix of type I, then S^{-1} is the same form as S (see, Theorem 2.1 or (2.3)) and the $1+4\lfloor\frac{n-3}{4}\rfloor$ columns of $S^{-1}P$ are zero vectors, if S is a stair matrix of type II, then S^{-1} is the same form as S and the $3+4\lfloor\frac{n-3}{4}\rfloor$ columns of $S^{-1}P$ are zero vectors.

To show that $\rho(S^{-1}P) < 1$ holds in many of cases, we need to recall the following definitions and lemmas in [15] [34]:

Definition 3.1: A nonsingular matrix $A = (a_{ij}) \in \mathcal{C}^{n\times n}$ is said to be
(a) a nonsingular M-matrix, if $a_{ij} \leq 0$ for any $i \neq j$, and $A^{-1} \geq 0$ (i.e., A is a monotone matrix);
(b) a nonsingular H-matrix, if its comparison matrix $\langle A\rangle$ is an

invertible M-matrix, where $\langle A\rangle = (\langle a_{ij}\rangle) \in \mathcal{R}^{n\times n}$ is defined by

$$\langle a_{ij}\rangle = \begin{cases} |a_{ii}|, \text{ for } i = 1, 2, \dots, n; \\ -|a_{ij}|, \text{ for } i \neq j, i, j = 1, 2, \dots, n. \end{cases} \tag{12}$$

Definition 3.2: A splitting $A = K - N$ is called a regular splitting of A if K is nonsingular, $K^{-1} \geq 0$, and $N \geq 0$.

Lemma 3.1: Let A be a nonsingular M-matrix and $B = (b_{ij})$ be a real matrix. If the elements of B satisfy the relations

$$a_{ii} \leq b_{ii},\ a_{ij} \leq b_{ij} \leq 0,\ i \neq j,\ 1 \leq i,\ j \leq n,$$

then B is also a nonsingular M-matrix. Moreover, $B^{-1} \leq A^{-1}$.

Theorem 3.1. If A is a nonsingular M-matrix as in (2), then the stair-splitting $A = S - P$ is a convergent splitting of A, i.e., $\rho(S^{-1}P) < 1$.

Proof: Since $A = S - P$ is a stair-splitting and A is a nonsingular M-matrix, then $P \geq 0$ and $S \geq A$. By Lemma 3.1, we know that S is also an M-matrix and $S^{-1} \geq 0$, that is, the stair-splitting $A = S - P$ is a regular splitting of A. Therefore, the result easily follows by Theorem 3.13 of [34]. The proof is completed.

In fact, according to Theorem 4.2 of [21], when A is a nonsingular H-matrix, the above convergent theorem also holds:

Theorem 3.2: Let

$$A = pentadiag(A_{i,i-2}, A_{i,i-1}, A_{ii}, A_{i,i+1}, A_{i,i+2}) \in \mathcal{C}^{n\times n},$$

as in (2). If A is a nonsingular H-matrix, then the stair-splitting $A = S - P$ is a convergent splitting of A, i.e., $\rho(S^{-1}P) < 1$.

Proof: Without loss of generality, we can assume that $A = stair2(A_{i(i-2)}, A_{i(i-1)}, A_{ii}, A_{i(i+1)}, A_{i(i+2)})$. Let

$$L = \begin{pmatrix} C_1 & 0 & 0 & & & & \\ 0 & C_2 & 0 & 0 & & & \\ 0 & 0 & C_3 & 0 & 0 & & \\ & 0 & B_4 & C_4 & 0 & 0 & \\ & & A_5 & B_5 & C_5 & 0 & 0 \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & & & & C_m \end{pmatrix},$$

$$U = \begin{pmatrix} C_1 & D_1 & E_1 & & & & \\ 0 & C_2 & D_2 & 0 & & & \\ 0 & 0 & C_3 & 0 & 0 & & \\ & 0 & 0 & C_4 & 0 & 0 & \\ & & 0 & 0 & C_5 & D_5 & E_5 \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & & & & C_m \end{pmatrix},$$

and $D = diag(C_1, C_2, ..., C_m)$. Then we have that

$$LD^{-1}U = \begin{pmatrix} C_1 & D_1 & E_1 & & & & \\ 0 & C_2 & D_2 & 0 & & & \\ 0 & 0 & C_3 & 0 & 0 & & \\ & 0 & B_4 & C_4 & 0 & 0 & \\ & & A_5 & B_5 & C_5 & D_5 & E_5 \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & & & & C_m \end{pmatrix}$$
$$= stair2(A_i, B_i, C_i, D_i, E_i) = S.$$

Thus $S = LD^{-1}U$ is a sparse block factorization of A (see [14]), which is also a particular case of Theorem 4.2 of [21]. Therefore, $\rho(S^{-1}P) < 1$, i.e., the conclusion holds.

The above analysis bases mainly on characteristics of the coefficient matrix A. For the general estimate of the spectral radius $\rho(S^{-1}P)$ and even the condition number, some efforts have been made in [23], [25], [26], [28].

Next, let us observe that the eigenvalue distribution of the transformed coefficient matrices with the m-step polynomial preconditioner (11) to show the convergence rate of the preconditioned linear systems.

Theorem 3.3: Let $A = S - P$ be a convergent stair-splitting of block pentadiagonal matrix A, then the preconditioned matrix $T = M_m A$ with M_m as in (11) has at least $2 + \lfloor\frac{n-3}{4}\rfloor$ eigenvalues at 1, and other eigenvalues satisfy that

$$\sigma(T) = 1 - (\sigma(S^{-1}P))^m, \tag{13}$$

where $\sigma(C)$ denotes the arbitrary eigenvalue of matrix C.

Proof: Since $A = S - P$, obviously, we have that

$$\begin{aligned} M_m A &= (I + S^{-1}P + (S^{-1}P)^2 + \cdots + (S^{-1}P)^{m-1})S^{-1}A \\ &= (I + S^{-1}P + (S^{-1}P)^2 + \cdots + (S^{-1}P)^{m-1})S^{-1}(S - P) \\ &= (I + S^{-1}P + (S^{-1}P)^2 + \cdots + (S^{-1}P)^{m-1})(I - S^{-1}P), \end{aligned} \tag{14}$$

that is, $M_m A$ is a polynomial of matrix $S^{-1}P$, therefore it easily follows that (13) holds for arbitrary eigenvalue of the matrix $S^{-1}P$.

In addition, noting that matrix $S^{1-}P$ has at least $2 + \lfloor\frac{n-3}{4}\rfloor$ zero eigenvalues (If S is a stair matrix of type I, then the $1 + 4\lfloor\frac{n-3}{4}\rfloor$ columns of $S^{-1}P$ are zero vectors, if S is a stair matrix of type II, then the even columns of $S^{-1}P$ are zero vectors), so the preconditioned matrix $T = M_m A$ has at least $3 + 4\lfloor\frac{n-3}{4}\rfloor$ eigenvalues at 1.

When the splitting $A = S - P$ is convergent, i.e., $\rho(S^{-1}P) < 1$, then for any of eigenvalues of T, we have that $|\sigma(T)| \in (0, 2)$. Especially when m is enough large, all of eigenvalues of T will have aclusteredspectrum at 1. Next, let us illustrate this phenomenon by the Fig. 1 and Fig. 2, where $m = 3$ and PDE1 matrix (see it in Section 4) is arised from the numerical solution of two dimensional biharmonic equaiton [1]. In practice, the general difficulty of obtaining a converging splitting $A = M - N$ as shared by the task of constructing a preconditioner, lie in finding a simple and computationally efficient M, The idealized case is when $M^{-1}N$ is sparse while $(M^{-1}N)^j$ decays to zero very quickly as $j \to \infty$ so we can take a small number of terms in Neumann's series in accurately approximating A^{-1} and equally we can take

$$S(A^{-1}) \subset S((I + M^{-1}N)^m), \text{ with } m = 3. \tag{15}$$

However, when m is very large, the computation of the polynomial preconditioner M_m has also very high cost. Therefore, how to improve the preconditioner M_m is very key problem especially when m is not very large. Next, some schemes are presented as follows for the polynomial preconditioner M_m when $m = 3$:

Method 1: we consider a special type of polynomial preconditioner (a kind of accelerated relaxation method for preconditioner)

$$M_3^h = (I + S^{-1}P + h(S^{-1}P)^2)S^{-1}, \tag{16}$$

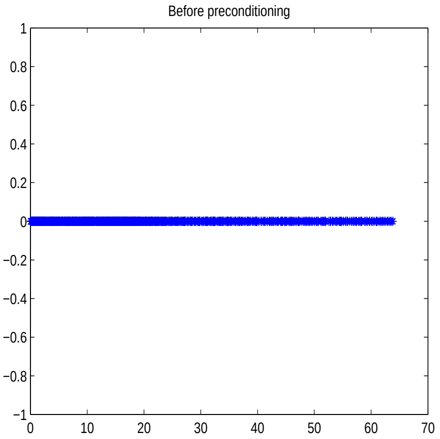

Fig. 1. Spectrum of A for PDE1 matrix.

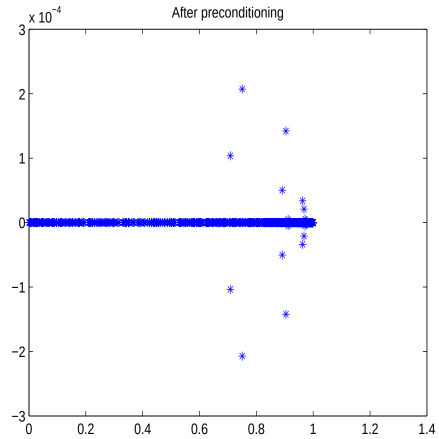

Fig. 2. Spectrum of $M_3 A$ for PDE1 matrix.

where the parameter h is real number, which control the rate of convergence of the transformed coefficient matrices $M_3^h A$. However, the problem arises how to choose h so that the condition number of $M_3^h A$ is as small as possible. Obviously, this problem is computationally expensive, and in general we have to rely on heuristics to obtain an estimate in the following numerical experiments (see Section 4).

Obviously, when $h = 1$, we obtain again the classical m-step polynomial preconditioner (see (3.2)). For the preconditioner M_3^h, the parameter h should be chosen to produce a good preconditioner. Generally speaking, estimating h can be formulated as the problem of finding h_{min} as to minimize $Cond(M_3^h A)$, i.e.,

$$h_{min} = \arg\min_h Cond(M_3^h A). \tag{17}$$

However, this minimization is computationally expensive and we have also to rely on heuristics to obtain an estimate, that is, for those matrices with the same structure of nonzeros, we choose the optimum parameter h of lower order matrix as the optimum parameter of this kind of matrices. Numerical experiments show that this method is very efficient to quickly obtain a better parameter h, see the following Example 4.1.

Method 2: Additive polynomial preconditioner, by considering two different splittings of the matrix A and then averaging the updates of each splitting, that is, for different splittings of the matrix A:

$$A = S_1 - P_1 = S_2 - P_2,$$

where S_1 and S_2 are stair matrices of type I and type II (see Definition 2.1), respectively. Similar to (5), we have the following additive polynomial preconditioner using methods for weighted mean for $m = 1, 2, \ldots,$

$$\begin{array}{l} M_m^\lambda = \frac{1}{1+\lambda}(I + L + \cdots + L^{m-1})(S_1^{-1} + \lambda S_2^{-1}), \\ L = \frac{1}{1+\lambda}(S_1^{-1}P_1 + \lambda S_2^{-1}P_2). \end{array} \tag{18}$$

Obviously, if the matrix A is symmetric, it is easily proved that especial $M_m^1(\lambda = 1)$ is also symmetric (Note that $S_1 = S_2^T$). In fact, if the matrix A is symmetric and positive definite (we say that a matrix P is positive definite if $x^T P x > 0$ for all real nonzero vectors x, see [10]), then M_m^1 is also symmetric and positive definite, that is, it is a valid preconditioner for the system in (1):

Theorem 3.4: Let $A = S_1 - P_1$ and $A = S_2 - P_2$ be two stair-splittings of the symmetric and positive definite matrix A. If S_1 (or S_2) is positive definite, then the matrix M_m^1 is also symmetric and positive definite if one of the following conditions is satisfied
(1) m is odd;
(2) m is even, and $\rho(H) < 1$.

Proof: Since A is symmetric, then $S_1 = S_2^T$. By Theorem 2.2 and Corollary 2.3 of [10], the results immediately follow.

IV. Numerical experiments

In this section, some numerical experiments will be described. The goal of these experiments is to examine the effectiveness of the polynomial preconditioners M_m, M_3^h and M_m^λ for the BiCGSTAB Krylov subspace method [31].

All the numerical experiments were performed in Fortran PowerStation 4.0, to produce our preconditioners, in conjunction with MATLAB R2010a, to implement the iterations. The machine we have used is a *acer* PC-Pentium (R)4, CPU2.20 GHz, 2.00 GB of RAM. In all of our runs we used a zero initial guess, and the right-hand-side vector b is taken as the vector of all ones. The iterative process ends when the residual satisfies

$$\frac{\|r^{(k)}\|_2}{\|r^{(0)}\|_2} < 10^{-6},$$

where $r^{(k)}$ is the residual vector after k-th iterations.

In addition, it should be mentioned that the preconditioned matrix $M_3^h A$ ($M_m A$ or $M_m^\lambda A$) does not need to be formed explicitly since $M_3^h v$ ($M_m v$ or $M_m^\lambda v$) can be computed for any vector v from a sequence of matrix-by-vector products. In experiments, to further reduce computational cost, we may solve them based on the vector and parallel processors. For example, to compute $w = M_3^h v$, we write

$$w = M_3^h(I + (I + (I + hS^{-1}P)S^{-1}P)S^{-1}v,$$

and then apply the following Algorithm II to obtain the vector w in a nested manner.

Algorithm II:
Input S, P, h and v.
Output w.
Step 1: Let $S^{-1}v = w_1$ and solve the linear systems $Sw_1 = v$ to obtain w_1 by using the following parallel Algorithm III.
Step 2: Compute $w_2 = Pw_1$ by the matrix-by-vector product.
Step 3: Let $S^{-1}w_2 = w_3$ and solve the linear systems $Sw_3 = w_2$ to obtain w_3 by using Algorithm III.
Step 4: Compute $w_4 = Pw_3$ by the matrix-by-vector product.
Step 5: Let $S^{-1}w_4 = w_5$ and solve the linear systems $Sw_5 = w_4$ to obtain w_5 by using Algorithm III.
Step 6: Output $w = w_1 + w_3 + hw_5$.

Algorithm III ([29]): This algorithm solves the block stair linear system $Ax = b$, where A is a stair matrix. The solution overwrites b.

If (A is of the type I)
for $i = 1 : 4 : 1 + 4\lfloor \frac{n-3}{4} \rfloor$ and $i = 5 : 4 : 5 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1} b_i$
endfor i
for $i = 2 : 4 : 2 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - A_{i,i-1} b_{i-1})$
endfor i
for $i = 4 : 4 : 4 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - A_{i,i+1} b_{i+1})$
endfor i
for $i = 3 : 4 : 3 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - (A_{i,i-1}A_{i-1,i-1}^{-1}A_{i-1,i-2} - A_{i,i-2})b_{i-2} - A_{i,i-1}A_{i-1,i-1}^{-1}b_{i-1} - A_{i,i+1}A_{i+1,i+1}^{-1}b_{i+1} + (A_{i,i+1}A_{i+1,i+1}^{-1}A_{i+1,i+2} - A_{i,i+2})b_{i+2})$
endfor i
endif
If (A is of the type II)
for $i = 3 : 4 : 3 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1} b_i$
endfor i
for $i = 2 : 4 : 2 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - A_{i,i+1} b_{i+1})$
endfor i
for $i = 4 : 4 : 4 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - A_{i,i-1} b_{i-1})$
endfor i
for $i = 1 : 4 : 1 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - A_{i,i+1}A_{i+1,i+1}^{-1}b_{i+1} + (A_{i,i+1}A_{i+1,i+1}^{-1}A_{i+1,i+2} - A_{i,i+2})b_{i+2})$
endfor i
for $i = 5 : 4 : 5 + 4\lfloor \frac{n-3}{4} \rfloor$
  $b_i = A_{ii}^{-1}(b_i - A_{i,i-1}A_{i-1,i-1}^{-1}b_{i-1} + (A_{i,i-1}A_{i-1,i-1}^{-1}A_{i-1,i-2} - A_{i,i-2})b_{i-2})$
endfor i
endif

where $b_i = 0$, if $i < 1$ or $i > n$. It is readily seen that in block case, Algorithm III needs n matrix-vector products of the form $A_{ij}b_j$, $j = i-2, i-1, i+1, i+2$, n vector additions and solving n small linear systems of the form $A_{ii}^{-1}d$. Obviously, this algorithm has relatively high parallelism. For example, if A is a stair matrix of the type I, first, for all $i = 1 : 4 : 5 + 4\lfloor \frac{n-3}{4} \rfloor$ the computations of $A_{ii}^{-1}b_i$ can be fulfilled by different processors at same time. Then $b_i = A_{ii}^{-1}(b_i - A_{i(i-1)}b_{i-1})$ are easily computed in parallel for even $i = 2 : 4 : 2 + 4\lfloor \frac{n-3}{4} \rfloor$, and $b_i = A_{ii}^{-1}(b_i - A_{i(i+1)}b_{i+1})$ are easily computed in parallel for $i = 4 : 4 : 4 + 4\lfloor \frac{n-3}{4} \rfloor$. also $b_i = A_{ii}^{-1}(b_i - (A_{i,i-1}A_{i-1,i-1}^{-1}A_{i-1,i-2} - A_{i,i-2})b_{i-2} - A_{i,i-1}A_{i-1,i-1}^{-1}b_{i-1} - A_{i,i+1}A_{i+1,i+1}^{-1}b_{i+1} + (A_{i,i+1}A_{i+1,i+1}^{-1}A_{i+1,i+2} - A_{i,i+2})b_{i+2})$ are not hardly computed for $i = 3 : 4 : 3 + 4\lfloor \frac{n-3}{4} \rfloor$. Thus, the high parallelism of Algorithm II is achieved if all A_{ij} are small blocks (see [29]).

Let us consider the linear system of the form

$$Ax = b, \quad , x, b \in \mathcal{R}^n, \tag{19}$$

where the matrix A is a block pentadiagnoal matrix, which arises from the numerical solution of two dimensional biharmonic equation as follows (see [1]):

$$\triangle^2 u = \frac{\partial^4 u}{\partial x^4} + 2\frac{\partial^4 u}{\partial x^2 \partial^2 y} + \frac{\partial^4 u}{\partial y^4} = f(x, y), \ 0 \le x, y \le 1. \tag{20}$$

Next, we use three uniform meshes of $n_1 = 1/51, n_2 = 1/61$ and $n_3 = 1/71$ refer to the mesh sizes in the x-direction and y-direction, which lead to three matrices of order $n = 50 \times 50$ and $n = 60 \times 60$ and $n = 70 \times 70$, respectively, the corresponding matrices are called PDE1, PDE2, PDE3. Their characteristics are given in Table I, where n denotes the order of matrix, the function $n_z(A)$ denotes the number of nonzero elements of A and $Cond(A)$ represents the condition number of matrix A and diagonal dominance of matrices can abbreviated to "DD".

TABLE I
CHARACTERISTICS OF TEST MATRICES PDE1-3.

Matrices	Size(n)	$n_z(A)$	DD	Symmetric	Cond(A)
PDE1	2500	31504	No	Yes	5.9e+005
PDE2	3600	45604	No	Yes	1.2e+006
PDE3	4900	45178	No	Yes	2.2e+006

Now, we compare the above three polynomial preconditioners using block stair-splitting with polynomial preconditioners using block Jacobi splitting $A = D - C$, where $D = diag(A_{11}, A_{22}, \cdots, A_{mm})$. For convenience, we denote the corresponding block Jacobi polynomial preconditioners by

$$JM_3^h = (I + D^{-1}C + h(D^{-1}C)^2)D^{-1},$$

Especially, let

$$JM_3 = (I + D^{-1}C + (D^{-1}C)^2)D^{-1}.$$

Comparisons are made in terms of the similar ILU(0) method among the diagonal blocks A_{ii} of A to yield the corresponding matrices A_{ii}^{-1}. The results are presented in Tables II and III for various matrices, respectively. The symbol "No precond" means that no preconditioner is used.

As it can be seen, the application of these block stair-splitting preconditioners greatly improves the convergence rate corresponding to classical block Jacobi splitting ones and so reduces the number of iterations in almost all of cases.

TABLE II
NUMBER OF ITERATIONS OBTAINED WITH BICGSTAB USING THE PRECONDITIONERS M_3, M_3^h AND FOR MATRIXES PDE1-3 AND DIFFERENT PARAMETERS h.

			M_3^h					
Matrices	No precond	M_3	$h=0.5$	$h=1.5$	$h=3$	$h=6$	$h=10$	$h=20$
PDE1	377	81	81	74	62	55	52	50
PDE2	534	108	133	118	89	76	72	70
PDE3	717	152	163	137	131	101	98	92

Next, by using the heuristic method and (17), we obtain an estimate for the optimum parameter h for matrices PDE1-3 with the same pattern of nonzero elements. First of all, we simulate, by computer, the function $Cond(M_3^h A)$for the lower order matrix PDE1, see Fig. 3.

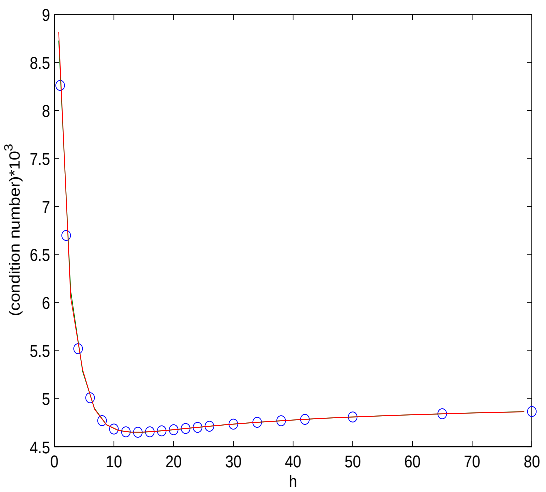

Fig. 3. The function Cond($M_3^h A$) changes as the parameter h increase, where A = PDE1.

We observe (or compute) that the condition number of $M_3^h A$ is minimum when $h = 12$. Thus we choose $h = 12$ as anoptimumparameter for this kind of matrices. Now we test this conjecture for different parameters h (see Table IV). Obviously, the results of Table IV conform to our hypothesis, which confirms our method.

V. CONCLUSIONS

In this paper, exploiting the stair-splitting technique and polynomial preconditioners, we develop the ideas of Axelsson [11], Saad [11] and H. B. Li [36] *et al.* introduce some new parallel polynomial approximate inverse preconditioners for the block pentadiagonal matrix in the form (1.2), whose computation can be done in parallel based on sparse blocks matrix-vector multiplications. If we view block tridiagonal matrix as a special type of block pentadiagonal matrix, then we obtain a new stair matrix splitting about block tridiagonal matrix, we also can structure some new parallel preconditioners for block tridiagonal linear systems. Moreover, theoretical analysis shows that our schemes are effective for any nonsingular block pentadiagonal H-matrices or symmetric positive definite block pentadiagonal matrices, see Theorems 3.2 and 3.4. Finally, The robustness of these preconditioners is also analyzed by some numerical experiments.

As it can be seen, the efficiency of these new preconditioners is confirmed. However, we have to rely on heuristics to obtain an estimate for the optimum parameters h and λ in M_3^h and M_3^λ since their computations are expensive(especially for M_3^λ), see Section 4. In addition, because there has no large and reliable parallel processor in our laboratory, and therefore only theoretic analysis is presented, computational time for the preconditioners and for the solution of the systems is not narrated in this article. These problems are of important and interest, which will be further investigated and solved in a later work.

ACKNOWLEDGMENT

This research is supported by NSFC (61170311, 11026085), Chinese Universities Specialized Research Fund for the Doctoral Program (20110185110020) and the Fundamental Research Funds for the Central Universities (ZYGX2009J103).

REFERENCES

[1] G. Rodrigue, R. S. Varga, "Convergence rate estimate for iterative solutions of the biharmonic equation," *J. Comput. Appl. Math.*, vol. 24, no. 1-2, pp. 129-146, 1988.
[2] M. Vajtersic, "A Direct Block-Five-Diagonal System Solver for the VLSI Parallel Model." *In Proceedings of the 10th International Parallel Processing Symposium (IPPS '96).* IEEE Computer Society, Washington, DC, USA, pp. 886-890, 1996.
[3] M. H. Koulaei, F. Toutounian, "Factored sparse approximate inverse of block tridiagonal and block pentadiagonal matricies," *Appl. Math. Comput.*, vol. 184, no.2, pp. 223-234, 2007.
[4] V. Casulli, "Numerical simulation of three-dimensional free surface flow in isopycnal co-ordinates," *Int. J. Numer. Meth. Fluids,* vol. 25, no. 6, pp. 645-658, 1997.
[5] K. Benkert, R. Fischer, "An Efficient Implementation of the Thomas-Algorithm for Block Penta-diagonal Systems on Vector Computers," *International Conference on Computational Science*, Beijing, China, pp. 144-151, 2007.
[6] R. Samtaney, D. I. Pullin, "On initial-value and self-similar solutions of the compressible Euler equations," *Phys. Fluids,* vol. 8, no. 10, pp. 2650-2655, 1996.
[7] A. Kaveh, H. Rahami, "Tri-diagonal and penta-diagonal block matrices for efficient eigensolutions of problems in structural mechanics," *Acta Mechanica,* vol. 192, no. 1-4, pp. 77-87, 2007.
[8] V. B. L. Boppana, J. S. B. Gajjar, "Global flow instability in a lid-driven cavity," *Int. J. Numer. Meth. Fluids,* vol. 62, no. 8, pp. 827-853, 2010.
[9] L. Adams, "M-step preconditioned conjugate gradient methods", *SIAM J. Sci. Stat. Comput.,* vol. 6, no. 2, pp. 452-462, 1985.
[10] L. Adams, E. Ong, "Additive polynomial preconditioners for parallel computers," *Parallel Comput.,* vol. 9, no.3, pp. 333-345, 1988/89.
[11] O. Axelsson, *Iterative Solution Methods,* Cambridge University Press, New York, USA., 1994.
[12] Z. Z. Bai, "On the convergence of the multisplitting methods for the linear complementarity problem," *SIAM J. Matrix Anal. Appl.,* vol. 21, no.1, pp. 67-78, 2000.

TABLE III
NUMBER OF ITERATIONS OBTAINED WITH BICGSTAB USING THE PRECONDITIONERS M_3^1, M_3^λ AND FOR MATRIXES PDE1-3 AND DIFFERENT PARAMETERS λ.

			M_3^λ					
Matrices	No precond	M_3^1	$\lambda = 0.1$	$\lambda = 0.25$	$\lambda = 0.5$	$\lambda = 1.5$	$\lambda = 3$	$\lambda = 6$
PDE1	377	85	83	79	86	87	81	79
PDE2	534	124	113	123	116	116	116	137
PDE3	717	190	171	165	180	194	229	156

TABLE IV
NUMBER OF ITERATIONS OBTAINED WITH BICGSTAB USING THE BLOCK JACOBI SPLITTING POLYNOMIAL PRECONDITIONERS JM_3, JM_3^h AND FOR MATRIXES PDE1-3 AND DIFFERENT PARAMETERS h.

			JM_3^h					
Matrices	No precond	JM_3	$h = 0.5$	$h = 1.5$	$h = 3$	$h = 6$	$h = 10$	$h = 20$
PDE1	377	260	158	300	365	451	479	473
PDE2	534	371	232	436	599	600	654	719
PDE3	717	495	313	617	742	910	957	1037

[13] R. Barret, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods,* SIAM press, Philadelphia, PA, USA., 1994.

[14] R. Beauwens, M. Ben Bouzid, "On sparse block factorization iterative methods," *SIAM J. Numer. Anal.,* vol. 24, no. 5, pp. 1066-1076, 1987.

[15] A. Berman, R.J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences,* SIAM Press, Philadelphia, USA., 1994.

[16] R. Bru, C. Corral, J. Mas, "A preconditioned conjugate gradient method on a distributed memory multiprocessor," *Appl. Math. Lett.,* vol. 8, no. 3, pp. 49-53, 1995.

[17] R. Bru, C. Corral, A. Martínez, J. Mas, "Multisplitting preconditioners based on incomplete Choleski factorizations," *SIAM J. Matrix Anal. Appl.,* vol. 16, no. 4, pp. 121-1222, 1995.

[18] E. Chow, Y. Saad, "Approximate inverse techniques for block-partitioned matrices," *SIAM J. Sci. Comput.,* vol. 18, no. 6, pp. 1657-1675, 1997.

[19] R. Fletcher, *Conjugate gradient methods for indefinite systems, in: Lecture Notes in Mathematics, vol. 506,* Springer, Berlin, 1976, pp. 73-89.

[20] M. J. Grote, T. Huckle, "Parallel preconditioning with sparse approximate inverses," *SIAM J. Sci. Comput.,* vol. 18, no. 3, pp. 838-853, 1997.

[21] C. H. Guo, "Some results on sparse block factorization iterative methods," *Linear Algebra Appl.,* vol. 145, pp. 187-199, 1991.

[22] C. H. Guo, "Incomplete block factorization preconditioning for linear systems arising in the numerical solution of the Helmholtz equation," *Appl. Numer. Math.,* vol. 19, no. 4, pp. 495-508, 1996.

[23] J. G. Hu, "The upper and lower bounds of the eigenvalues of $M^{1-}N$," *Math. Numer. Sinica,* vol. 8, no. 1, pp. 41-46, 1986. (in Chinese)

[24] T. Huckle, "Approximate sparsity patterns for the inverse of a matrix and preconditioning," *Appl. Numer. Math.,* vol. 30, no. 2-3, pp. 291-303, 1999.

[25] H. B. Li, T. Z. Huang, H. Li, "An improvement on a new upper bound for moduli of eigenvalues of iterative matrices," *Appl. Math. Comput.,* vol. 137, no. 2, pp. 977-984, 2006.

[26] H. B. Li, T. Z. Huang, H. Li, "On some subclasses of P-matrices," *Numer. Lin. Alg. Appl.,* vol. 14, no. 5, pp. 391-405, 2007.

[27] H. B. Li, T. Z. Huang, S. Q. Shen, H. Li, "Lower bounds for the minimum eigenvalue of Hadamard product of an M-matrix and its inverse," *Lin. Alg. Appl.,* vol. 421, no. 1, pp. 235-247, 2007.

[28] H. B. Li, T. Z. Huang, H. Li, "Inclusion sets for singular values," *Lin. Alg. Appl.,* vol. 428 no. 8-9, pp. 2220-2235, 2008.

[29] H. Lu, "Stair matrices and their generalizations with applications to iterative methods I: a generalization of the successive overrelaxation method," *SIAM J. Numer. Anal.,* vol. 37, no. 1, pp. 1-17, 2001.

[30] J. M. Ortega, *Introduction to Parallel and Vector Solution of Linear Systems,* Plenum Press, New York, USA., 1988.

[31] Y. Saad, *Iterative Methods for Sparse Linear Systems,* PWS Publishing Company, Boston, MA, USA., 1996.

[32] Y. Saad, M. H. Schultz, "GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM J. Sci. Statist. Comput.,* vol. 7, no.3, pp. 856-869, 1986.

[33] H. A. Van der Vorst, "Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems," *SIAM J. Sci. Statist. Comput.,* vol. 13, no.2, pp. 631-644, 1992.

[34] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, NJ, USA., 1962.

[35] J. H. Yun, "Block ILU preconditioners for a nonsymmetric block-tridiagonal M-matrix", *BIT*, vol. 40, no. 3, pp. 583-605, 2000.

[36] H. B. Li, T. Z. Huang, Y. Zhang, X. P. Liu, H. Li, "On some new approximate factorization methods for block tridiagonal matrices suitable for vector and parallel processors," *Mathematics and Computers in Simulation*, vol. 79, no. 7, pp. 2135-2147, 2009.