A generalization of T. Chan’s preconditioner

Ming-Chao Cai a, Xiao-Qing Jin a,∗,1, Yi-Min Wei b,2

aDepartment of Mathematics, University of Macau, Macau, China
bDepartment of Mathematics, Fudan University, Shanghai 200433, China

Received 18 April 2004; accepted 13 April 2005
Available online 29 June 2005
Submitted by E. Tyrtyshnikov

Abstract

In this paper, we propose a new preconditioner for solving linear systems by the preconditioned conjugate gradient (PCG) method. The preconditioner can be thought of as a generalization of the well-known T. Chan’s preconditioner. For Hermitian positive definite matrix, we observe that our preconditioner is also Hermitian positive definite. The operation cost and convergence of the PCG method are discussed. Numerical experiments have been performed on structured problems to show the competitiveness of this preconditioner.

© 2005 Elsevier Inc. All rights reserved.

AMS classification: 65F10; 65F15

Keywords: T. Chan’s preconditioner; Hadamard product; PCG method

1. Introduction

In 1988, Tony F. Chan proposed a circulant preconditioner which is efficient especially for solving some Toeplitz-like linear systems, see [1,2,7–10]. In this paper,
we will propose a preconditioner which can be thought of as a generalization of T. Chan’s preconditioner.

For a general matrix $A_n = [a_{ij}] \in \mathbb{C}^{n \times n}$, consider

$$P = F^*[(FA_n F^*) \circ \text{diag}(I_{n-m}, I_m)]F$$

(1)

or

$$P = F^*[(FA_n F^*) \circ \text{diag}(J_m, I_{n-m})]F$$

(2)

to be an approximation to $A_n$. Here, $I_{n-m}$ is the identity matrix of order $n-m$, $J_m$ is an $m$-by-$m$ matrix with elements being equal to 1, $\text{diag}(\cdot)$ denotes a block diagonal matrix and $F$ is the $n$-by-$n$ Fourier matrix whose entries are given by:

$$(F)_{j,k} = \frac{1}{\sqrt{n}} e^{2\pi i j k / n}, \quad i \equiv \sqrt{-1}, \quad 0 \leq j, k \leq n-1.$$  

Moreover, in (1), “$\circ$” is the Hadamard product [5]. More precisely, the Hadamard product of $A = [a_{ij}] \in \mathbb{C}^{m \times n}$ and $B = [b_{ij}] \in \mathbb{C}^{m \times n}$ is defined by $A \circ B = [a_{ij} b_{ij}] \in \mathbb{C}^{m \times n}$.

The matrix $P$ can be thought of as a generalization of T. Chan’s preconditioner because when $m = 1$, it is just T. Chan’s preconditioner

$$c(A_n) \equiv F^*\delta(FA_n F^*)F.$$  

(3)

Here $\delta(FA_n F^*)$ is a diagonal matrix whose diagonal is equal to the diagonal of $FA_n F^*$. As shown in [2], T. Chan’s preconditioner $c(A_n)$ is an optimal approximation of $A_n$ in the sense that

$$\|c(A_n) - A_n\|_F = \min_{W_n \in \mathcal{M}_F} \|W_n - A_n\|_F,$$

where $\| \cdot \|_F$ is the Frobenius norm and

$$\mathcal{M}_F \equiv \{ F^* A_n F \mid A_n \text{ is any } n\text{-by-}n \text{ diagonal matrix} \}.$$  

Actually, $\mathcal{M}_F$ is the set of all circulant matrices [3]. It is well-known that T. Chan’s preconditioner is a good preconditioner for solving some structured systems, see [1,2,7,8].

When $m = n$, we have $P \equiv A_n$. We would also remark here that in (1) when $m = n - 1$,

$$\|P - A_n\|_F = \min_{W_n \in \mathcal{M}} \|W_n - A_n\|_F,$$

where $\mathcal{M}$ is a matrix class with matrices having the property that each row sum is equal to each column sum, refer to Theorem 6.8.1 in [3]. Thus, by above discussion,
we know that $P$ in fact is a perturbation of T. Chan’s preconditioner if $m$ is small and we can expect $P$ is a good approximation to some structured matrix $A_n$, therefore a good preconditioner for solving $A_n x = b$.

We now review some important results of T. Chan’s preconditioner. For matrix $E_n \in \mathbb{C}^{n \times n}$, let $\lambda_j(E_n)$ be the $j$th eigenvalue of $E_n$. We have the following theorem, see [6,10].

**Theorem 1.** Let $A_n = [a_{ij}] \in \mathbb{C}^{n \times n}$ and $c(A_n)$ be T. Chan’s preconditioner given by (3). Then

(i) If $A_n$ is Hermitian, then $c(A_n)$ is also Hermitian. Moreover, we have

$$\min_j \lambda_j(A_n) \leq \min_j \lambda_j(c(A_n)) \leq \max_j \lambda_j(c(A_n)) \leq \max_j \lambda_j(A_n).$$

In particular, if $A_n$ is positive definite, then so is $c(A_n)$.

(ii) We have

$$c(A_n) = \sum_{j=0}^{n-1} \left( \frac{1}{n} \sum_{p-q=j \text{ (mod } n)} a_{pq} \right) Q^j,$$

where $Q$ is an $n$-by-$n$ circulant matrix given by

$$Q = \begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
1 & 0 & \cdots & \cdots & 0 \\
0 & 1 & \cdots & \cdots & \cdots \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & 1 & 0
\end{bmatrix}.$$ 

In this paper, we will use $P$ defined by (1) or (2) as a preconditioner for solving linear systems $A_n x = b$. We will mainly discuss the preconditioner $P$ given by (1) and the discussion of (2) is similar. The paper is organized as follows. In Section 2, we give some spectral properties of $P$. The operation cost and convergence of the preconditioned conjugate gradient (PCG) method are studied in Section 3. In Section 4, we exhibit some numerical examples to illustrate the effectiveness of our preconditioner.

**2. Spectral properties of preconditioner**

In this section, we discuss the eigenvalue distribution of $P$ in (1). Since $F$ is unitary, the eigenvalue distribution of $P$ is the same as to that of
Therefore, we only need to discuss the eigenvalue distribution of $M$. For simplicity, let
\[
F = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix},
\]
where $F_1 \in \mathbb{C}^{(n-m) \times n}$ and $F_2 \in \mathbb{C}^{m \times n}$. Note that
\[
FA_nF^* = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} A_n \begin{bmatrix} F_1^* \\ F_2^* \end{bmatrix} = \begin{bmatrix} F_1 A_n F_1^* \\ F_2 A_n F_2^* \end{bmatrix}.
\]
In the following discussion, we will write
\[
M = \begin{bmatrix} \delta(F_1 A_n F_1^*) & 0 \\ 0 & F_2 A_n F_2^* \end{bmatrix} = \begin{bmatrix} M_{11} & 0 \\ 0 & M_{22} \end{bmatrix}.
\]
Now, we discuss the eigenvalue distribution of $P$, we have the following theorem.

**Theorem 2.** Let $A_n \in \mathbb{C}^{n \times n}$ be Hermitian positive definite, $M$ and $M_{22}$ be defined by (4) and (5) respectively. Then $M$ is also Hermitian positive definite. If we arrange the eigenvalues as follows,
\[
\lambda_1(A_n) \geq \lambda_2(A_n) \geq \cdots \geq \lambda_n(A_n),
\]
and
\[
\lambda_1(M_{22}) \geq \lambda_2(M_{22}) \geq \cdots \geq \lambda_m(M_{22}),
\]
then
\[
\lambda_i(A_n) \geq \lambda_i(M_{22}), \quad \lambda_{n-i+1}(A_n) \leq \lambda_{n-i+1}(M_{22}),
\]
for $i = 1, \ldots, m$. Moreover, we have
\[
0 < \min_j \lambda_j(A_n) \leq \min_j \lambda_j(P) \leq \max_j \lambda_j(M) \leq \max_j \lambda_j(A_n).
\]
Thus
\[
0 < \min_j \lambda_j(A_n) \leq \min_j \lambda_j(P) \leq \max_j \lambda_j(P) \leq \max_j \lambda_j(A_n),
\]
where $P$ is defined by (1).
Proof. It is straightforward to show that $M$ is Hermitian positive definite. By using Cauchy interlace theorem [4,5], one can prove (6) easily. Moreover, for (7), by using Courant-Fisher theorem [4,5], we have

$$0 < \min_j \lambda_j(A_n) \leq \min_j \lambda_j(M_{22}) \leq \max_j \lambda_j(M_{22}) \leq \max_j \lambda_j(A_n).$$

(9)

For $M_{11}$ in $M$ defined by (5), by using Theorem 1 (i), we have

$$0 < \min_j \lambda_j(A_n) \leq \min_j \lambda_j(M_{11}) \leq \max_j \lambda_j(M_{11}) \leq \max_j \lambda_j(A_n).$$

(10)

Therefore, one can obtain (7) and then (8) easily by combining (9) and (10) together. □

Remark. Theorem 2 still holds when the Fourier matrix $F$ in (1) is replaced by any other unitary matrix.

3. Operation cost and convergence

We know that in each iteration of the PCG method, it is required to compute two matrix-vector products: $A_ny$ and $P^{-1}y$, see [4,7,8]. If $A_n$ has no good structure, it requires $O(n^2)$ operations in each iteration to compute $A_ny$. To compute $P^{-1}y$, we have by (5),

$$P^{-1} = F^* \begin{bmatrix} \delta(F_1 A_n F_1^*)^{-1} & 0 \\ 0 & (F_2 A_n F_2^*)^{-1} \end{bmatrix} F.$$

Therefore, we need to form $FA_n F^*$ in the initial step. We remark here it can be formed by using Fast Fourier Transform (FFT) in $O(n \log n)$ operations by employing $n$ parallel processes. After forming $FA_n F^*$, we need to compute $\delta(F_1 A_n F_1^*)^{-1}$ and $(F_2 A_n F_2^*)^{-1}$. It is easy to obtain $\delta(F_1 A_n F_1^*)^{-1}$ because it is a diagonal matrix. However, for $(F_2 A_n F_2^*)^{-1}$, we need to choose small $m$ in practice so that the operation cost to compute $(F_2 A_n F_2^*)^{-1}$ could be small.

Actually, we do not need to form the whole matrix of $FA_n F^*$. We are only required to form $\delta(F_1 A_n F_1^*)$ and $F_2 A_n F_2^*$. For $\delta(F_1 A_n F_1^*)$, we can compute it as follows. By Theorem 1 (ii), we need $O(n^2)$ to obtain the first column $c$ of T. Chan’s preconditioner. Note the following relation between the first column and the eigenvalues of $c(A_n)$,

$$\sqrt{n}Fc = A_n 1_n,$$

where $1_n = (1, 1, \ldots, 1)^T$ and $A_n = \delta(FA_n F^*)$ is a diagonal matrix holding the eigenvalues of $c(A_n)$. Therefore, we can obtain $A_n = \delta(FA_n F^*)$ and then $\delta(F_1 A_n F_1^*)$ by using FFT in $O(n \log n)$ operations. For $F_2 A_n F_2^*$, we need $O(mn^2)$
operations in general. Thus, totally, we need $O(mn^2)$ operations to obtain $P$. We emphasize that for Toeplitz matrix, one only needs $O(mn \log n)$ operations to form $P$.

For the convergence of our method, we have the following observations:

1. When $m$ in (1) is small, our preconditioner is in fact T. Chan’s circulant preconditioner with low-rank perturbation. We remark that all well-known results about the spectral clustering of circulant preconditioners in [11] can be adopted to our case. Therefore, it could be a good preconditioner for some structured systems like Toeplitz systems (see also Chapter 1 in [7]). At least in the Hermitian positive definite case, our preconditioner could save about $m$ iterations hopefully compared to T. Chan’s preconditioner.

2. When $m$ becomes larger, our preconditioner would be a better approximation to the original matrix. Therefore, it should be a good preconditioner for any general matrix. However, although the convergence rate of the iterative method may be fast, the operation cost will increase with $m$ and it will be the same as that of the original one with $m$ sufficiently large.

An open problem is that for a given matrix $A_n$, can we find an optimal $m$ to construct the preconditioner $P$?

4. Numerical experiments

In this section, all the experiments were performed in MATLAB 6.5. We used the MATLAB-provided M-file "pcg" with our preconditioner $P$ given by (2) to solve the systems. In all tests, the zero vector is the initial guess and the stopping criterion is

$$
\frac{||r_q||_2}{||r_0||_2} < 10^{-6},
$$

where $r_q$ is the residual after the $q$th iteration. We illustrate the efficiency of our preconditioner by solving the following problems.

**Example 1.** Consider $A_nx = b$ where the matrix $A_n$ is a symmetric positive definite Toeplitz matrix with the first row given by

$$
\left(2, -\frac{1}{2}, -\frac{1}{2^2}, \ldots, -\frac{1}{2^{n-1}}\right)
$$

and $b = (1, 2, \ldots, n)^T$.

**Example 2.** Consider $A_nx = b$ where $b = (1, 2, \ldots, n)^T$. The matrix $A_n$ is

$$
A_n = T_n + \frac{1}{n} H_n,
$$
where $T_n$ is a symmetric positive definite Toeplitz matrix with the first row given by
\[
\begin{pmatrix}
1, & -\frac{1}{3}, & -\frac{1}{3^2}, & \cdots, & -\frac{1}{3^{n-1}}
\end{pmatrix},
\]
and $H_n = [h_{jk}]_{j,k=1}^n$ is a Hermitian matrix with the upper diagonal part defined by
\[
h_{jk} = \begin{cases}
\frac{4}{j^2+k^2}, & j < k, \\
0, & j = k.
\end{cases}
\]

In Tables 1 and 2, the number of iterations are given for different preconditioners. In the tables, $n$ is the matrix size, “4000+” means that the number of iterations is larger than 4000, $I$ means no preconditioner is used, $T$ is T. Chan’s preconditioner and $P(m)$ is given by (2) for $m = 5, 10, 20$. We note that although $m$ is small, when $P(m)$ applied, the number of iterations of $P(m)$ is less than that of $T$. In Tables 3 and 4, we give the computing time for different preconditioners. One can see that the CPU time is also reduced by using our preconditioner.

**Table 1**
Number of iterations for Example 1

<table>
<thead>
<tr>
<th>$n$</th>
<th>$I$</th>
<th>$T$</th>
<th>$P(5)$</th>
<th>$P(10)$</th>
<th>$P(20)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6400</td>
<td>2154</td>
<td>49</td>
<td>36</td>
<td>32</td>
<td>31</td>
</tr>
<tr>
<td>12800</td>
<td>4000+</td>
<td>65</td>
<td>51</td>
<td>43</td>
<td>42</td>
</tr>
<tr>
<td>25600</td>
<td>4000+</td>
<td>91</td>
<td>71</td>
<td>60</td>
<td>55</td>
</tr>
<tr>
<td>51200</td>
<td>4000+</td>
<td>126</td>
<td>101</td>
<td>84</td>
<td>75</td>
</tr>
</tbody>
</table>

**Table 2**
Number of iterations for Example 2

<table>
<thead>
<tr>
<th>$n$</th>
<th>$I$</th>
<th>$T$</th>
<th>$P(5)$</th>
<th>$P(10)$</th>
<th>$P(20)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>514</td>
<td>28</td>
<td>21</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>2000</td>
<td>1016</td>
<td>36</td>
<td>27</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>3000</td>
<td>1518</td>
<td>42</td>
<td>32</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>4000</td>
<td>2019</td>
<td>48</td>
<td>36</td>
<td>32</td>
<td>31</td>
</tr>
</tbody>
</table>

**Table 3**
Computing time in seconds for Example 1

<table>
<thead>
<tr>
<th>$n$</th>
<th>$T$</th>
<th>$P(5)$</th>
<th>$P(10)$</th>
<th>$P(20)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6400</td>
<td>3.43</td>
<td>3.32</td>
<td>3.03</td>
<td>3.14</td>
</tr>
<tr>
<td>12800</td>
<td>9.58</td>
<td>9.51</td>
<td>8.32</td>
<td>8.47</td>
</tr>
<tr>
<td>25600</td>
<td>28.53</td>
<td>27.73</td>
<td>25.20</td>
<td>23.98</td>
</tr>
<tr>
<td>51200</td>
<td>91.76</td>
<td>88.54</td>
<td>85.32</td>
<td>78.75</td>
</tr>
</tbody>
</table>
Table 4
Computing time in seconds for Example 2

<table>
<thead>
<tr>
<th>n</th>
<th>T</th>
<th>P(5)</th>
<th>P(10)</th>
<th>P(20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>3.19</td>
<td>2.44</td>
<td>2.33</td>
<td>3.45</td>
</tr>
<tr>
<td>2000</td>
<td>14.98</td>
<td>11.54</td>
<td>9.96</td>
<td>9.97</td>
</tr>
<tr>
<td>3000</td>
<td>31.55</td>
<td>24.12</td>
<td>21.78</td>
<td>21.82</td>
</tr>
<tr>
<td>4000</td>
<td>81.25</td>
<td>63.17</td>
<td>59.93</td>
<td>58.92</td>
</tr>
</tbody>
</table>

Acknowledgments

The comments of the referee and the editor Prof. Eugene Tyrtyshnikov are essential for the improvement of our paper. We would like to express our sincere acknowledgment to both of them.

References