Filomat 27:7 (2013), 1269–1276
DOI 10.2298/FIL1307269S

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

An iterative method to compute Moore-Penrose inverse based on gradient maximal convergence rate

Xingping Sheng[a], Tao Wang[b]

[a]*School of Mathematics and Computational Science, Fuyang Normal College, Fuyang, P.R.China*
[b]*School of Mathematical Sciences, Anhui University, Hefei 230039, P.R. China*

Abstract. In this paper, we present an iterative method based on gradient maximal convergence rate to compute Moore-Penrose inverse $A^{\dagger}$ of a given matrix A. By this iterative method, when taken the initial matrix $X_0 = A^*$, the M-P inverse $A^{\dagger}$ can be obtained with maximal convergence rate in absence of roundoff errors. In the end, a numerical example is given to illustrate the effectiveness, accuracy and its computation time, which are all superior than the other methods for the large singular matrix.

1. Introduction

In 1920, Moore generalized the notion of inverse of a matrix including all the matrices, rectangular as well as singular [1].

For an $m \times n$ matrix A over field C of complex number, in 1955, Penrose [2] gave an equivalent definition of Moore-Penrose inverse $A^{\dagger}$, which is the unique solution to the following four equations

$$AXA = A \qquad (1) \qquad XAX = X \qquad (2)$$
$$(AX)^* = AX \qquad (3) \qquad (XA)^* = XA \qquad (4)$$

where $(\cdot)^*$ denotes the conjugate transpose of a complex matrix. In general, let $\phi \neq \eta\{1, 2, 3, 4\}$. If X satisfies all conditions of η, then X is called a η inverse of A, denoted by $A\{\eta\}$.

There have been many approaches available for the determination M-P inverse and weighted M-P inverse, such as singular value decomposition (SVD), full-rank decomposition[3∼ 5], and so on. In 1960, Greville first gave a finite iterative method for computing M-P inverse $A^{\dagger}$ by computing $A_k^{\dagger}$ with n iterations, where A_k is the submatrix of A consisting of the first k columns in [6]. In 1999, Udwadia and Kalaba gave a unified approach for the recursive determination of generalized inverse in [7]. In 2005, Pian improved Udwadia and Kalaba's method in [8]. In the later Wei in [9,10] gave different methods to compute the weighted M-P inverse. In the recent work [24-27], some papers propose iterative methods to compute the M-P inverse for large and spare matrix. In this paper we will design an iterative method based on gradient

2010 *Mathematics Subject Classification*. Primary 15A09; Secondary 65A10,15A24
Keywords. M-P inverse; Iterative method; Gradient maximal convergence rate; Numerical computing
Received: 04 January 2013; Accepted: 27 April 2013
Communicated by Predrag Stanimirović
Research supported by NSF China (No.11071079), Anhui Provincial Natural Science Foundation (No.10040606Q47), and The University Natural Science Research key Project of Anhui Province (No.KJ2013A204)
Email addresses: xingpingsheng@163.com (Xingping Sheng), wanghanbing007@163.com (Tao Wang)

to solve the matrix equation $AXA = A$, when the initial matrix $X_0 = A^*$ is taken, the M-P inverse $A^\dagger$ can be got in maximal convergence rate. We must point that the coefficient matrix of A is not needed full column rank. In the end we give a numerical example to show this iteration is quite efficient.

2. Notations and Preliminaries

In the past decades, research on solutions to linear matrix equations has been very plentiful. For example, by using the well-known Kronecker product, the linear matrix equation can be converted into the standard linear system $Ax = b$ which can readily be solved by some classical methods or the recently developed methods such as SOR-type iteration in [11], AOR-type iterations in [12], preconditioned conjugate gradient method in [13] and multiple search direction conjugate gradient method in [14].

Recently, many papers discuss the iterative algorithms based on gradient to solve linear system. For instance, Ding and Zhou made a great deal of research on this problem in papers [15-22]: papers [15,16] presented gradient-based iterative algorithms for some matrix equations, in [17-19], least squares solutions were obtained, and by using the hierarchical identification principle, Ding et al. [21] introduced a gradient-based iterative algorithms of the generalized Sylvester-matrix equation. Zhou et al. [22] obtained the maximal convergence rate iterative method based on gradient to linear matrix equations with a little knowledge of discrete-time linear system theory. All the algorithms given by Ding and Zhou can work well only on the condition that the matrix equations considered should have the unique solution, which needs that the coefficient matrix is full column-rank.

In space $C^{m\times n}$, we define inner product as $(A, B) = traceB^*A$ for all $A, B \in C^{m\times n}$. Then the norm of a matrix A generated by this inner product is, obviously, Frobenius norm and denoted by $\| A \|$. Throughout this paper the following notations are used: the set of all $m \times n$ complex matrices of rank r is denoted by $C_r^{m\times n}$. For $A \in C^{m\times n}$, $R(A)$ and $N(A)$ denote the range and null space of A, A^* and $A^\dagger$ denote conjugate transpose and M-P inverse of A. $\sigma_{max}(A) =\| A \|_2$, $\sigma_{min}(A)$ are the maximal singular value, the minimal nonzero singular value of A. P_L denotes the orthogonal projection on L. The symbol of I_n represents an identity matrix of order n.

In this paper the following Lemmas are needed in what follows:

Lemma 2.1[4] Let $A \in C^{m\times n}$ be of rank r, T be a subspace of C^n of dimension $s \le r$ and S be a subspace of C^m of dimension $m - s$. Then A has a {2} inverse X such that $R(X) = T$ and $N(X) = S$, if and only if

$$AT \oplus S = C^m$$

In which case X is unique and this X is denoted by $A^{(2)}_{T,S}$.

Lemma 2.2[4] Let $A \in C^{m\times n}$ be of rank r, any two of the following three statements imply the third:

$$\begin{aligned} X &\in A\{1\}, \\ X &\in A\{2\}, \\ rankA &= rankX. \end{aligned}$$

Lemma 2.3[4] Let $A \in C^{m\times n}$, then for the M-P inverse $A^\dagger$, one has $A^\dagger = A^{(2)}_{R(A^*),N(A^*)}$.

Lemma 2.4[4] Let $C^n = L \oplus M$ and $A \in C^{m\times n}$ be of rank r, then

(1) $P_{L,M}A = A$ if and only if $R(A) \subset L$.

(2) $AP_{L,M} = A$ if and only if $M \subset N(A)$.

Lemma 2.5 Let $A \in C^{m\times n}$, then the set of solution to $AXA = A$ is equivalent to that of $A^*AXAA^* = A^*AA^*$.

Proof Let $A\{1\}$ be the set of the solution to $AXA = A$, then for any $X_0 \in A\{1\}$, we have $AX_0A = A$. This implies $A^*AX_0AA^* = A^*AA^*$.

Conversely, if X is solution to $A^*AXAA^* = A^*AA^*$, then postmultiplying and premultiplying it by $(A^\dagger)^*$, which gives

$$\begin{aligned}(A^\dagger)^*A^*AXAA^*(A^\dagger)^* &= (A^\dagger)^*A^*AA^*(A^\dagger)^*\\ (AA^\dagger)^*AXA(A^\dagger A)^* &= (AA^\dagger)^*A(A^\dagger A)^*\\ AA^\dagger AXAA^\dagger A &= AA^\dagger AA^\dagger A\\ AXA &= A\end{aligned}$$

Lemma 2.6 Let $A \in C^{m\times n}$ and $X \in C^{n\times m}$ satisfy $AXA = 0$, if $R(X) \subset R(A^*)$ and $N(X) \supset N(A^*)$ then $X = 0$.

Proof Postmultiplying and premultiplying $AXA = 0$ by $A^\dagger$, which gives $A^\dagger AXAA^\dagger = 0$. This shows $P_{R(A^*)}XP_{R(A)} = 0$, then from the conditions and the Lemma 2.4, we have $X = 0$.

In order to study the properties of iterative method for computing $A^\dagger$, we need the following two Lemmas.

Lemma 2.7[22] For any iterative process

$$x_k = Ax_{k-1}, \tag{2.1}$$

where $A \in C^{n\times n}$ and $x_k \in C^n$. Then the iteration (2.1) converges for arbitrary initial condition x_0 if and only if

$$\beta = \rho(A) := \max_{1\le i\le n} |\ \lambda_i(A)\ | < 1, \tag{2.2}$$

Moreover, if A is a real symmetric matrix, then the 2-convergence rate of iteration (2.1) is $1/\beta$, and

$$\|\ x_k\ \| \le \beta^k \ \|\ x_0\ \|\ .$$

Lemma 2.8[22] Assume that m_i $(i = 1, 2, \dots, n)$ are some given positive scalars. Denote $m_{max} = \max\limits_{1\le i\le n}\{m_i\}$ and $m_{min} = \min\limits_{1\le i\le n}\{m_i\}$. Then

$$\min_{0<\mu<2/m_{max}} \max_{1\le i\le n} |1 - \mu m_i| = \frac{m_{max} - m_{min}}{m_{max} + m_{min}}. \tag{2.3}$$

Moreover, the unique μ_{opt} can be obtained by the following equality

$$\mu_{opt} = \frac{2}{m_{max} + m_{min}}.$$

3. Iterative method for computing $A^\dagger$

In this section we study gradient-based numerical solution to the M-P inverse $A^\dagger$. A gradient-based algorithm for solving the unique minimum Frobenius norm solution of $AXA = A$ is constructed, if the initial matrix X_0 is taken by A^*, then $A^\dagger$ is obtained by using the algorithm. The basic idea of gradient-based algorithm is to search minimum Frobenius norm X such that the following objective function

$$J(X) = \frac{1}{2}\ \|\ A - AXA\ \|^2$$

is minimized. It is easy to compute that

$$\frac{\partial J(X)}{\partial X} = -A^*(A - AXA)A^*.$$

Then the gradient-based algorithm can be constructed as follows:

$$X_k = X_{k-1} - \mu\frac{\partial J(X)}{\partial X} = X_{k-1} + \mu A^*(A - AX_{k-1}A)A^*, \tag{3.1}$$

where μ is called the convergence factor that will be specified later.

About formula (3.1), we have the following basic properties:

Theorem 3.1 Let $A \in C^{m\times n}$, taking the initial matrix $X_0 = A^*$, then the sequences $\{X_i\}$, generated by formula (3.1), satisfies

$$R(X_i) \subset R(A^*) \quad \text{and} \quad N(X_i) \supset N(A^*) \quad (i = 0, 1, 2, \cdots). \tag{3.2}$$

Further, if the sequences $\{X_i\}$ converges, then it converges to $A^\dagger$.

Proof We prove the conclusion by induction.

When $i = 1$, we have

$$\begin{aligned} X_1 &= X_0 + \mu A^*(A - AX_0A)A^* \\ &= A^*[I_m + \mu(A - AX_0A)A^*] \\ &= [I_n + \mu A^*(A - AX_0A)]A^*. \end{aligned}$$

This shows when $i = 1$, the conclusion is right.

Assume that conclusion holds for all $0 \le i \le s(0 < s < k)$. Then there exists matrices U and V such that

$$X_s = A^*U = VA^*.$$

Further, we have

$$\begin{aligned} X_{s+1} &= X_s + \mu A^*(A - AX_kA)A^* \\ &= A^*[U + \mu(A - AX_kA)A^*] \\ &= [V + \mu A^*(A - AX_kA)]A^*. \end{aligned}$$

This means when $i = s + 1$, the conclusion is also right.

By the principle of induction, the conclusion $R(X_i) \subset R(A^*)$ and $N(X_i) \supset N(A^*)$ hold for all $i = 0, 1, 2, \cdots$.

If the sequences $\{X_i\}$ is convergent, then taking limit on both sides of (3.1), we have

$$X_\infty = X_\infty + \mu A^*(A - AX_\infty A)A^*.$$

From the above equality, we can easy get

$$A^*AA^* = A^*AX_\infty AA^*.$$

By Lemma 2.5, we have $AX_\infty A = A$. Further by (3.2), Lemma 2.3 and Lemma 2.4, we know $X_\infty = A^\dagger$. □

In the next Theorem, the sufficient and necessary condition of the iterative convergence is studied and the maximal convergence rate is obtained when taken $\mu_{opt} = \dfrac{2}{\sigma^4_{max}(A) + \sigma^4_{min}(A)}$.

Theorem 3.2 Assume the sequences $\{X_k\}$ $(k = 1, 2, \ldots)$ are generalized by formula (3.1). Then the formula (3.1) yields $\lim_{k\to\infty} X_k = A^\dagger$ for the special initial matrix $X_0 = A^*$ if and only if

$$0 < \mu < \frac{2}{\sigma^4_{max}(A)}. \tag{3.3}$$

Moreover, the F-convergence rate of the method is maximized when

$$\mu = \mu_{opt} = \frac{2}{\sigma^4_{max}(A) + \sigma^4_{min}(A)}. \tag{3.4}$$

In this case, the F-convergence rate is $\gamma_{max} = 1/\beta(\mu_{opt})$ with

$$\beta(\mu_{opt}) = \frac{cond^4(A) - 1}{cond^4(A) + 1}. \tag{3.5}$$

i.e. the iteration error matrix $\tilde{X}_k = X_k - A^{\dagger}$ satisfies

$$\| \tilde{X}_k \| \le \beta^k(\mu_{opt}) \| \tilde{X}_0 \|, \ \ k \ge 0. \tag{3.6}$$

Proof Minusing $A^{\dagger}$ on both sides of (3.1), then we rewrite (3.1) as

$$\tilde{X}_k = \tilde{X}_{k-1} - \mu A^* A \tilde{X}_{k-1} A A^*. \tag{3.7}$$

Using Kronecker product on formula (3.7), we have

$$vec(\tilde{X}_k) = vec(\tilde{X}_{k-1}) - \mu \mathcal{A}\mathcal{A}^* vec(\tilde{X}_{k-1}) = (I - \mu \mathcal{A}\mathcal{A}^*) vec(\tilde{X}_{k-1}), \tag{3.8}$$

where $\mathcal{A} = A \otimes A^*$.
From the initial matrix $X_0 = A^*$ and Theorem 3.1, we know

$$R(\tilde{X}_k) = R(X_k - A^{\dagger}) \subset R(A^*) \quad \text{and} \quad N(\tilde{X}_k) = N(X_k - A^{\dagger}) \supset N(A^*). \tag{3.9}$$

This implies that there exists a matrix W such that

$$\tilde{X}_k = A^* W A^*.$$

So all $\tilde{X}_k$ satisfy

$$vec(\tilde{X}_k) = \mathcal{A} vec(W) \subset R(\mathcal{A}) = R(\mathcal{A}\mathcal{A}^*). \tag{3.10}$$

There exists some matrices $\tilde{Y}_k$ such that

$$vec(\tilde{X}_k) = \mathcal{A}\mathcal{A}^T \tilde{Y}_k. \tag{3.11}$$

Since $\mathcal{A}\mathcal{A}^*$ is Hermite, we have

$$\mathcal{A}\mathcal{A}^* = P^{-1} diag(\lambda_1, \lambda_2, \ldots, \lambda_s, 0, \ldots, 0) P, \tag{3.12}$$

where P is an unitary matrix, $diag(a_1, a_2, \ldots, a_t)$ represents a diagonal matrix with diagonal elements $a_1, a_2, \ldots, a_t$, $\lambda_i (i = 1, 2, \ldots, s)$ are the nonzero eigenvalues of $\mathcal{A}\mathcal{A}^*$ with $\lambda_1 \ge \lambda_2 \ge \ldots \ge \lambda_s > 0$, it is obvious that $\lambda_i = \sigma_i^4(A)$.
From (3.10) and (3.12), we have

$$\begin{aligned} \tilde{X}_k &= (I - \mu \mathcal{A}\mathcal{A}^*) vec(\tilde{X}_{k-1}) \\ &= (I - \mu \mathcal{A}\mathcal{A}^*) \mathcal{A}\mathcal{A}^* \tilde{Y}_{k-1} \\ &= P^{-1} diag(1 - \mu\lambda_1, 1 - \mu\lambda_2, \ldots, 1 - \mu\lambda_s, 1, \ldots, 1) P \\ &\quad\ P^{-1} diag(\lambda_1, \lambda_2, \ldots, \lambda_s, 0, \ldots, 0) P \tilde{Y}_{k-1} \\ &= P^{-1} diag((1 - \mu\lambda_1)\lambda_1, (1 - \mu\lambda_2)\lambda_2, \ldots, (1 - \mu\lambda_s)\lambda_s, 0, \ldots, 0) P \tilde{Y}_{k-1} \\ &= P^{-1} diag(1 - \mu\lambda_1, 1 - \mu\lambda_2, \ldots, 1 - \mu\lambda_s, 0, \ldots, 0) P \tilde{X}_{k-1}. \end{aligned} \tag{3.13}$$

The equation (3.13) is a linear matrix equation with coefficient matrix $\Phi = P^{-1} diag(1 - \mu\lambda_1, 1 - \mu\lambda_2, \ldots, 1 - \mu\lambda_s, 0, \ldots, 0)P$. Therefore, the gradient-based algorithm converges for the special arbitrary initial condition $X_0 = A^*$ if and only if

$$\rho(\Phi) = \max_{1 \le i \le s} \left\{ |\lambda_i(\Phi)| \right\} = \max_{1 \le i \le s} \left\{ |1 - \mu\lambda_i| \right\} < 1,$$

which is equivalent to inequality (3.3).
Since $\Phi^* = \Phi$, 2-convergence rate of the iterative process (3.8) is $\gamma(\mu) = 1/\beta(\mu)$ from Lemma 2.7, where

$$\beta(\mu) = \max_{1 \le i \le s} \left\{ |1 - \mu\lambda_i(\mathcal{A}\mathcal{A}^*)| \right\} = \max_{1 \le i \le s} \left\{ |1 - \mu\sigma_i^4(A)| \right\}. \tag{3.14}$$

According to Lemma 2.8, we get that $\beta(\mu)$ is minimized, i.e. $\gamma(\mu)$ is maximized, when

$$\mu = \mu_{opt} = \frac{2}{\sigma_{max}^4(A) + \sigma_{min}^4(A)}.$$

From equation (3.14) and Lemma 2.8, we get equation (3.5) and

$$\| vec(\tilde{X}_k) \| \leq \beta(\mu_{opt}) \| vec(\tilde{X}_{k-1}) \| . \tag{3.15}$$

By using the property that $\|X\| = \|vec(X)\|_2$, (3.6) can be obtained from (3.15).

When $\mu = \mu_{opt}$, the formula (3.1) gets maximal convergence rate, this iteration is called gradient-based maximal convergence rate, abbreviated GBMC.

In the next theorem, we will use another way to prove the necessary condition of the convergence for the formula (3.1) and clearly to see the decreasing for the residual norm of $\widetilde{X}_{k-1} = X_{k-1} - A^{\dagger}$.

Theorem 3.3 Let $A \in C^{m\times n}$ and $X_0 = A^*$. If μ satisfied the inequality (3.3), then $\{X_k\}$ generated by the gradient-based formula (3.1) converges to $A^{\dagger}$ and the residual norm of the error matrix $\widetilde{X}_k = \widetilde{X}_{k-1} - A^{\dagger}$ Frobenius norm is decreasing.

Proof From (3.7), we have

$$\begin{aligned}
\| \tilde{X}_k \|^2 &= tr\Big[\tilde{X}_k^*\tilde{X}_k\Big] \\
&= \| \tilde{X}_{k-1} \|^2 + \mu^2 \| A^*A\tilde{X}_{k-1}AA^* \|^2 - \mu tr\Big[\tilde{X}_{k-1}^*A^*A\tilde{X}_{k-1}AA^* + AA^*\tilde{X}_{k-1}^*AA^*\tilde{X}_{k-1}\Big] \\
&= \| \tilde{X}_{k-1} \|^2 + \mu^2 \| A^*A\tilde{X}_{k-1}AA^* \|^2 - 2\mu \| A\tilde{X}_{k-1}A \|^2 \\
&\leq \| \tilde{X}_{k-1} \|^2 - \mu\Big(2 - \mu \| A \|_2^4 \Big)\|A\tilde{X}_{k-1}A\|^2 \\
&\leq \| \tilde{X}_0 \|^2 - \mu\Big(2 - \mu\sigma_{max}^4(A)\Big)\sum_{i=1}^{k} \| A\tilde{X}_{i-1}A \|^2 \\
&= \| A^* - A^{\dagger} \|^2 - \mu\Big(2 - \mu\sigma_{max}^4(A)\Big)\sum_{i=1}^{k} \| A\tilde{X}_{i-1}A \|^2
\end{aligned}$$

If the condition (3.3) is satisfied then the Frobenius norm of $\widetilde{X}_k$ is decreasing and

$$\sum_{k=1}^{\infty} \| A\tilde{X}_kA \|^2 \leq \frac{\| A^* - A^{\dagger} \|^2}{\mu\Big(2 - \mu\sigma_{max}^4(A)\Big)} < \infty,$$

It follows that

$$\lim_{k\to\infty} A\tilde{X}_kA = 0.$$

By Theorem 3.1 and Lemma 2.6, we can obtain

$$\lim_{k\to\infty} \tilde{X}_k = 0,$$

which yields

$$\lim_{k\to\infty} X_k = A^{\dagger}.$$

4. Numerical examples

In this section, we will give a numerical example to illustrate our results. All the tests are performed by Matlab 6.5.1 Service pack 1 version of software, which is used on a Pentium(R) Dual-Core processor system running at 2.6GHz with 1G of RAM memory using the windows XP professional 32 bit Operating System. The singular test matrix A with size 200×200 is obtained from the Matrix Computation Toolbox (mctoolbox) [23] (which includes test matrices from Matlab itself, i.e. A=matrix(8,200)) and the initial iterative matrices are chosen as $X_0 = A^*$. Because of the influence of the error of roundoff, we regard the matrix A as zero matrix if $\| A \| < 1.0e - 10$.

In the following Table, we will perform numerical experiments to compare Petković and Stanimirović's method [26] (PSI), Toutounian and Ataei's method [27] (CGSI) with the proposed method GBMC.

Table 1: error and computational time Results

Method	Time	$\| AA^\dagger A - A \|_2$	$\| A^\dagger AA^\dagger - A^\dagger \|_2$	$\| AA^\dagger - (AA^\dagger)^* \|_2$	$\| A^\dagger A - (A^\dagger A)^* \|_2$
PSI	76.672	9.9884e-11	7.4645e-13	2.3170e-13	3.0985e-15
CGSI	no result	no result	no result	no result	no result
GBMC	4.484	9.4827e-11	1.1711e-13	4.6409e-13	4.6091e-16

Note: The CGSI method was not able to produce numerical results for matrix(8,200), even after one day running. From Table 1, the proposed method (GBMC) is superior to PSI and CGSI on convergence time and accuracy.

5. Conclusion

In this paper, we presente an iterative formula to compute M-P inverse $A^\dagger$ based on gradient maximal convergence rate, where the matrix A is not full-rank. In the last section, a numerical example is given to illustrate the effectiveness, accuracy and its computation time, which are all superior than the other methods for the large singular matrix.

6. Acknowledgement

The authors wish to express their sincere thanks to the anonymous referees for their helpful comments and suggestions.

References

[1] E.H. Moore, On the reciprocal of the general algebra matrix (abstract), Bull. Amer. Math. Soc. 26(1920)394-395.
[2] R. Penrose, A generalized inverse for matrices, Pro. Cambridge Philos. Soc. 51(1955)406-413.
[3] A. Ben-Israel and T.N.E. Greville. Generalized inverse: Theory and Applications. 2nd Edition, NewYork: Springer Verlag, 2003.
[4] C.R. Rao and S.K. Mitra, Generalized Inverse of Matrices and Its Application. Wily, NewYork,1971.
[5] S.L.Campbell and C.D.Meyer. Generalized Inverses of Linear Transformations. Fearon-Pitman.Belmont,Calif.1979.
[6] T.N.E. Greville, some applications of the pseudoinverse of a matrix, SIAM J. Rev. 2(1960)1,15-22.
[7] F.E. Udwadia and R.E. Kalaba, A unified approach for the recursive determination of generalized inverse. Comput Math Appl, 37(1999)1, 125-130.
[8] J. Pian, X. Ding and G. Chen. On the recursive computing formulae of generalized inverse and weighted generalized inverse. Journal of Lanzhou Univerity, 41(2005)112-116(in Chinese).
[9] Y. Wei, Recurrent neutral networks for computing weighted Moore-Penrose inverse, Appl. Math. Comput. 116(2000), 279-287.
[10] Y. Wei, The representation and approximation for weighted Moore-Penrose inverse, Appl. Math. Comput. 121(2001), 17-28.
[11] L.Zheng, C.Li and D.J.Evans, Chebyshev acceleration for SOR-like method, Int. J. Comput. Math. 82(2005)5, 583-593
[12] T.Huang, G.Cheng, D.J.Evans and X.Cheng, AOR type iterations for solving preconditioned linear systems, Int. J. Comput. Math. 82(2005)8, 969-976.

[13] I.E.Kaporin, Explicitly preconditioned conjugate gradient method for the solution of unsymmetric linear systems, Int. J. Comput. Math. 44(1992)1-4, 169-187.

[14] T.Gu, X.Liu, Z.Mo and X.Chi, Multiple search direction conjugate gradient method I: methods and their propositions, Int. J. Comput. Math. 81(2004)9, 1133-1143.

[15] F. Ding. Transformations between some special matrices. Comput. Math. Appl. 59 (8)(2010) 2676-2695.

[16] F. Ding, T. Chen. Gradient based iterative algorithms for solving a class of matrix equations, IEEE Trans. Automat. Control 50 (2005) 1216-1221.

[17] F. Ding, T. Chen. Hierarchical gradient-based identification of multivariable discrete-time systems, Automatica 41 (2005) 315-325.

[18] F.Ding, T. Chen. Iterative least squares solutions of coupled Sylvester matrix equations, Systems Control Lett. 54 (2005) 95-107.

[19] F. Ding, T. Chen. Hierarchical least squares identification methods for multivariable systems, IEEE Trans. Automat. Control 50 (2005) 397-402.

[20] F. Ding, T. Chen. On iterative solutions of general coupled matrix equations, SIAM J. Control Optim. 44 (2006) 2269-2284.

[21] F. Ding, P.X. Liu, J. Ding. Iterative solutions of the generalized Sylvester matrix equations by using the hierarchical identification principle, Appl. Math. Comput. 197 (2008) 41-50.

[22] B. Zhou, J. Lam, G.R. Duan. Gradient-based maximal convergence rate iterative method for solving linear matrix equations, Int. J. Comput. Math. 87 (3)(2010) 515-527.

[23] N.J. Higham, The Matrix Computation Toolbox, http://www.ma.man.ac.uk/higham/ mctoolbox.

[24] P.S. Stanimirovic, D.S., Cvetkovic-Ilic, Successive matrix squaring algorithm for computing outer inverses, Appl. Math. Comput. 203(2008)19-29.

[25] H.S. Najan, M.S. Solary, Computational algorithms for computing the inverse of a square matrix, quasi-inverse of a non-square matrix and block matrices, Appl. Math. Comput. 183(2006)539-550.

[26] M.D. Petković and P.S. Stanimirović , Iterative method for computing the Moore-Penrose inverse based on Penrose equations, J. Comput. Appl. Math. 235(2011)1604-1613.

[27] F. Toutounian and A. Ataei, A new method for computing Moore-Penrose inverse matrices, J. Comput. Appl. Math. 228(2009)412-417.