Applied Mathematics and Computation 323 (2018) 64–74

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Computation of weighted Moore–Penrose inverse through Gauss–Jordan elimination on bordered matrices[☆]

Xingping Sheng

School of Mathematics and Statistics, Fuyang Normal College, Anhui 236037, PR China

ARTICLE INFO

MSC:
15A09
65F05

Keywords:
Partitioned matrix
Gauss–Jordan elimination
Weighted Moore–Penrose inverse
Computational complexity

ABSTRACT

In this paper, two new algorithms for computing the Weighted Moore–Penrose inverse $A^{\dagger}_{M,N}$ of a general matrix A for weights M and N which are based on elementary row and column operations on two appropriate block partitioned matrices are introduced and investigated. The computational complexity of the introduced two algorithms is analyzed in detail. These two algorithms proposed in this paper are always faster than those in Sheng and Chen (2013) and Ji (2014), respectively, by comparing their computational complexities. In the end, an example is presented to demonstrate the two new algorithms.

1. Introduction

Throughout the paper we shall use the standard notations of [1–3]. The symbol $C_r^{m\times n}$ denotes the set of all $m\times n$ complex matrices with rank r, C^n stands for the n dimensional complex space. I_n represents an identity matrix of order n. For $A\in C^{m\times n}$, the symbols $\mathcal{R}(A), \mathcal{N}(A), ||A||_F, A^*, A^{-1}$ and $r(A)$ denote its range, null space, the Frobenious norm, the conjugate transpose, regular inverse and rank, respectively. $\mathcal{R}(A)^{\perp}$ and $\mathcal{N}(A)^{\perp}$ are orthogonal complement space of $\mathcal{R}(A)$ and $\mathcal{N}(A)$, respectively.

For any $A\in C^{m\times n}$, we recall that the weighted Moore–Penrose inverse of A, denoted by $A^{\dagger}_{M,N}$, is the unique solutions $X\in C^{n\times m}$ satisfying the following four matrix equations

$$\begin{aligned} AXA &= A \qquad (1)\\ XAX &= X \qquad (2)\\ (MAX)^* &= MAX \qquad (3M)\\ (NXA)^* &= NXA \qquad (4N)\end{aligned}$$

where M and N are Hermitian positive definite matrices of orders m and n respectively. If $M=I_m$ and $N=I_n$, then the weighted Moore–Penrose inverse $A^{\dagger}_{M,N}$ reduces to the Moore–Penrose (abbreviated M–P) inverse $A^{\dagger}$. The matrix $A^{\#}=N^{-1}A^*M$ is called the weighted conjugate transpose matrix of A, it is easy to check $\mathcal{R}(A^{\#})=\mathcal{R}(N^{-1}A^*M)=N^{-1}\mathcal{R}(A^*)$ and $\mathcal{N}(A^{\#})=\mathcal{N}(N^{-1}A^*M)=M^{-1}\mathcal{N}(A^*)$.

☆ This project was supported by NSF China (no. 11471122), Anhui Provincial Natural Science Foundation (no. 1508085MA12) and Key projects of excellent talent fund in Anhui Provincial University support program (no. gxyqZD2016188).

E-mail address: xingpingsheng@163.com

https://doi.org/10.1016/j.amc.2017.11.041
0096-3003/

Table 1
Error and execution time results for computing $A^{\dagger}_{M,N}$ with $\eta = 10^{-2}$.

Method	Time (s)	r_1	r_2	r_3	r_4
Algorithm 2.2	0.668300	1.3129e−11	3.4777e−12	7.1661e−07	4.6343e−07
Algorithm 3.1	0.609692	1.4660e−11	3.4204e−12	5.1026e−07	3.6697e−11
Algorithm 2.3	0.578097	2.4588e−10	1.1145e−09	1.1145e−08	8.7446e−07
Algorithm 3.2	0.401415	1.0820e−10	1.5932e−12	6.0671e−09	5.8628e−08

Table 2
Error and execution time results for computing $A^{\dagger}_{M,N}$ with $\eta = 10^{4}$.

Method	Time (s)	r_1	r_2	r_3	r_4
Algorithm 2.2	0.755466	1.1092e−11	3.0973e−14	1.4398e−08	2.5990e−06
Algorithm 3.1	0.558564	6.8732e−12	3.0930e−14	1.1076e−08	4.3657e−09
Algorithm 2.3	0.750131	1.7855e−08	3.1442e−09	1.8343e−07	2.9696e−06
Algorithm 3.2	0.444912	3.1664e−09	1.4137e−11	1.0243e−07	7.4231e−07

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 $m-s$ such that $AT \oplus S = C^m$. Then there exists a unique matrix X such that $XAX = X$ with $\mathcal{R}(X) = T$ and $\mathcal{N}(X) = S$. This X is called the outer inverse or {2} inverse of A with prescribed range T and null space S and denoted by $A^{(2)}_{T,S}$.

It is well known that $A^{\dagger}_{M,N}$ is a special {2} inverse X of A with $\mathcal{R}(X) = \mathcal{R}(A^{\#}) = N^{-1}\mathcal{R}(A^*)$ and $\mathcal{N}(X) = \mathcal{N}(A^{\#}) = M^{-1}\mathcal{N}(A^*)$, this means that $A^{\dagger}_{M,N} = A^{(2)}_{\mathcal{R}(A^{\#}),\mathcal{N}(A^{\#})} = A^{(2)}_{N^{-1}\mathcal{R}(A^*),M^{-1}\mathcal{N}(A^*)}$.

Weighted M–P inverse arises in matrix computation, image reconstruction, large-scale systems and statistics. In the latest fifty years, there have been many famous specialists and scholars, who investigated the weighted M–P inverse $A^{\dagger}_{M,N}$. Its representation and perturbation theories were introduced in [4–14].

One handy method of computing the inverse of a nonsingular matrix A is the Gauss–Jordan elimination procedure by executing elementary row operations on the pair $(A \quad I)$ to transform it into $(I \quad A^{-1})$. Moreover Gauss–Jordan elimination can be used to determine whether or not a matrix is nonsingular. However, one can not directly use this method to compute weighted M–P inverse $A^{\dagger}_{M,N}$ on a square singular matrix A.

In 1987, Anstreicher and Rothblum [15] used Gauss–Jordan elimination to compute the index, generalized null spaces, and Drazin inverse. Recently, the authors [13,16–18] used two different Gauss–Jordan elimination methods to compute the $A\dagger$ and $A^{(2)}_{T,S}$, respectively. More recently, these algorithms were further improved by Ji [14,19–21], Stanimirovic and Petkovic [22].

In [13,16], the author, Chen and Gong proposed an algorithm for computing the outer inverse $A^{(2)}_{T,S}$ and M–P inverse $A\dagger$ starts from elementary row operations on the pair $(GA \quad I)$. Then, Ji [19], Stanimirovic and Petkovic [22] proposed an alternative explicit expressions for $A\dagger$ and $A^{(2)}_{T,S}$, respectively. These methods begin with the elementary row operations on the pair $(G \quad I)$ and do not need to compute A^*A or GA. Following the line [19], Ji [14] develop an algorithm for $A^{\dagger}_{M,N}$ free of computing N^{-1}. More recently the author and Chen [17] start with the elementary row and column operations on the partitioned matrix $\begin{pmatrix} GAG & G \\ G & 0 \end{pmatrix}$ for computing $A^{(2)}_{T,S}$, then in [18] the author improved the algorithm [17] to compute M–P inverse $A\dagger$. In [20,21] Ji proposed a new method for computing the outer inverse $A^{(2)}_{T,S}$ and M–P inverse $A\dagger$ by applying elementary row operations on a blocked matrices.

As a special {2} inverse, the algorithms of [13,17,21,22] can be used to compute $A^{\dagger}_{M,N}$ with $G = A^{\#}$. If we use these methods to compute $A^{\dagger}_{M,N}$, it is not only increase the computational cost to compute the $A^{\#} = N^{-1}A^*M$, but also it worsens the condition number. The goal of this paper is to develop algorithms for $A^{\dagger}_{M,N}$ free of computing $A^{\#} = N^{-1}A^*M$.

In this paper, inspired by the ideas of [14,17,21], we will propose two alternative methods of elementary row and column operations for weighted M–P inverse $A^{\dagger}_{M,N}$ by applying row and column operations on the matrices $\begin{pmatrix} A^* & N \\ M^{-1} & 0 \end{pmatrix}$ and $\begin{pmatrix} 0 & A^*M \\ N^{-1}A^* & 0 \end{pmatrix}$, respectively. Our approach is like the one in [17,21] by working a bordered matrix and the $A^{\dagger}_{M,N}$ is easily read off from the computed result. But the complexities of my two approaches are all less than that in [17,21].

The paper is organized as follows. The ideas of computational $A^{(2)}_{T,S}$ in [17,21] are repeated in the next section. In Section 3, we derive two novel explicit expressions for $A^{\dagger}_{M,N}$ and propose two Gauss–Jordan-like elimination procedure for $A^{\dagger}_{M,N}$ based on the formula. In Section 4, their computational complexities are studied. In Section 5, an illustrative example is presented to explain the corresponding improvements of the algorithm.