BULLETIN OF THE INTERNATIONAL MATHEMATICAL VIRTUAL INSTITUTE
ISSN (p) 2303-4874, ISSN (o) 2303-4955
www.imvibl.org /JOURNALS / BULLETIN
Vol. 6(2016), 269-280

Former
BULLETIN OF THE SOCIETY OF MATHEMATICIANS BANJA LUKA
ISSN 0354-5792 (o), ISSN 1986-521X (p)

A NEW FOURTH-ORDER METHOD TO COMPUTE THE WEIGHTED MOORE-PENROSE INVERSE

Hamid Esmaeili, Raziyeh Erfanifar, and Mahdis Rashidi

Abstract. In this paper, we propose a new method to find weighted Moore-Penrose inverse of matrices. Every iteration of the method involves four matrix multiplications. It is proved that this method converge with fourth-order. The discussions cover both theoretical and computational aspects. A wide set of numerical comparisons of our method with other methods shows that the average number of matrix multiplications and the average CPU time of proposed method are considerably less than those of other methods both in computing Moore-Penrose inverse and weighted Moore-Penrose inverse. So, our new method can be considered as a fast method. For each of the sizes $m \times (m + 50)$, $m = 100, 200, 300, 400, 500$, ten dense matrices were chosen randomly to make these comparisons.

1. Introduction

In numerical mathematics, there is an interest in applications of the generalized inverses of matrices or operators. In fact, many computational and theoretical problems require different types of generalized inverses, when a matrix is singular or rectangular [**1**]. One of the most important generalized inverses is the so-called Weighted Moore-Penrose inverse (or, WMP inverse, for short) of a complex matrix. The introduction and importance of weighted Moore-Penrose inverse for an arbitrary matrix has been made in [**1**, **5**, **17**].

Let $A \in \mathbb{C}^{m\times n}$, $M \in \mathbb{C}^{m\times m}$ and $N \in \mathbb{C}^{n\times n}$ be complex matrices, where M and N are invertible. A matrix $X \in \mathbb{C}^{n\times m}$ is said to be the WMP inverse of A with respect to M and N, and denoted by $A^{\dagger}_{MN}$, if the following four weighted Penrose

2010 *Mathematics Subject Classification.* 15A09, 65F30, 65F50.
Key words and phrases. Weighted Moore-Penrose inverse; Iterative method; Order of convergence; Schulz-type method; Matrix multiplication.

Communicated by Kyung Ho Kim.

equations are satisfied

$$(1.1) \qquad AXA = A, \quad XAX = X, \quad (MAX)^* = MAX, \quad (NXA)^* = NXA,$$

in which $*$ denotes the conjugate transpose. It is proved that the WMP inverse $A^{\dagger}_{MN}$ uniquely exists when M and N are Hermitian positive definite matrices [**1**]. Throughout this work, we consider that the matrices M and N to be Hermitian positive define and subsequently the WMP inverse is uniquely defined.

In particular, when $M = I_m$ and $N = I_n$, the matrix X is called the Moore-Penrose inverse or the generalized pseudo-inverse of A and is denoted by $A^{\dagger}$, while (2.4) reduced to the well-known Penrose equations originally attributed to [**11**] in the following form:

$$(1.2) \qquad AXA = A, \quad XAX = X, \quad (AX)^* = AX, \quad (XA)^* = XA.$$

The WMP inverse is viewed as a generalized Moore-Penrose inverse, and is widely used in control system analysis, statistics, singular differential and difference equations, Markov chains, iterative methods, weighted least-squares problems, perturbation theory, neural network problems and many other subjects found in the literatures (see, e.g. [**2**, **6**, **9**, **10**, **21**]).

Algorithms for computing the (weighted) Moore-Penrose inverse of a matrix are a subject of current research (see, e.g., [**8**, **12**, **14**, **16**, **18**, **22**]). Greville's partitioning method for numerical computation of generalized inverses was introduced in [**4**]. Wang in [**20**] generalized Greville's method to the weighted Moore-Penrose inverse. Many numerical algorithms for computing the (weighted) Moore-Penrose inverse lack numerical stability. The Greville's algorithm requires more operations and consequently it accumulates more rounding errors. Furthermore, it is widely known that the Moore-Penrose inverse is not necessarily a continuous function of the elements of the matrix. The existence of this discontinuity provides more burden in its computation [**1**]. It is therefore clear that cumulative round-off errors should be totally eliminated, which is possible only by means of the symbolic implementation. In this case, variables are stored in the "exact" form or can be left "unassigned", resulting in no loss of accuracy during the calculation. Anyway, by increasing the dimension of the input matrix, the computation of its (weighted) MoorePenrose inverse by the symbolic implementation will take too much time. This made some numerical analysts to suggest and rely on numerically stable matrix methods.

We denote the weighted conjugate transpose matrix of A by

$$A^{\#} = N^{-1}A^*M.$$

Also, I_k denotes the identity matrix of the order k. A basic method to find the WMP inverse is based on the weighted singular value decomposition [**19**] as follows. Suppose that $rank(A) = r$. It is shown there exist $U \in \mathbb{C}^{m\times m}$ and $V \in \mathbb{C}^{n\times n}$, satisfying $U^*MU = I_m$ and $V^*N^{-1}V = I_n$, such that

$$(1.3) \qquad A = U \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} V^*,$$

in which

$$D = diag(\sigma_1, \ldots, \sigma_r), \quad \sigma_1 \geqslant \cdots \geqslant \sigma_r > 0,$$

and $\sigma_1^2, \ldots, \sigma_r^2$ are the nonzero eigenvalues of $A^{\#}A$. Then, the WMP inverse $A_{MN}^{\dagger}$ could be defined by

$$A_{MN}^{\dagger} = N^{-1} V \begin{bmatrix} D^{-1} & 0 \\ 0 & 0 \end{bmatrix} U^* M. \tag{1.4}$$

Furthermore,

$$\|A\|_{MN} = \sigma_1, \quad \|A_{MN}^{\dagger}\|_{NM} = \frac{1}{\sigma_r}, \tag{1.5}$$

in which $\|A\|_{MN} = \|M^{\frac{1}{2}} A N^{-\frac{1}{2}}\|_2$.

The restrictions of computing the WMP inverse using weighted singular value decomposition encouraged some to develop iteration methods for this purpose. In 2006, Huang and Zhang [**8**] applied the Schulz iterative method [**13**]

$$X_{k+1} = X_k(2I - AX_k), \quad k = 0, 1, 2, \ldots, \tag{1.6}$$

to find the WMP inverse using the initial approximation

$$X_0 = \alpha A^{\#}, \quad 0 < \alpha < \frac{2}{\sigma_1^2}\,. \tag{1.7}$$

The Schulz iteration involves only matrix multiplications. The rate of convergence is quadratic, since for the residual matrix $E_k = I - AX_k$ we have $E_{k+1} = E_k^2$. This iteration is numerically stable but for the approximations X_k that are remote from the solution, convergence can be slow. This made the construction of higher order iterative methods. In [**15**], authors stated that the eighth-order convergence, based on Householder's method [**7**], can be obtained with only six matrix multiplications by the formula

$$\begin{aligned} \psi_k &= AX_k \\ \zeta_k &= \psi_k(-2I + \psi_k) \\ \upsilon_k &= 2I + \zeta_k \\ X_{k+1} &= -X_k(-2I + \psi_k)\upsilon_k(2I + \zeta_k \upsilon_k), \quad k = 0, 1, \ldots, \end{aligned} \tag{1.8}$$

in which X_0 is defined as (1.7).

In this paper, we propose a new method to find WMP inverse. Every iteration of the method involves four matrix multiplications. It is proved that this method always converge with fourth-order. Numerical experiments show the efficiency of our method with respect to the methods (1.6) and (1.8). Indeed, instead of constructing a higher order method, we focus on the reduction of the total number of matrix multiplications, and hence, on reduction of the CPU time required to convergence. This leads to a fast iterative method. Toward this goal, a theoretical discussion will be also given to show the behavior of the proposed scheme.

The rest of this paper is organized as follows. In Section 2, we propose our new method to find the WMP inverse and we prove that it is converge with fourth-order. In Section 3, some numerical examples are given to show the performance of the presented method compared with other methods both in computing Moore-Penrose inverse and weighted Moore-Penrose inverse. For each of sizes $m \times (m+50)$,

$m = 100, 200, 300, 400, 500$, ten dense matrices were chosen randomly to make these comparisons. Finally, some conclusions are outlined in Section 4.

2. A new method to find the WMP inverse

In this section, we investigate our iterative method to compute the WMP inverse. To this goal, consider the following polynomial

$$g(x) = 12x - 38x^2 + 52x^3 - 33x^4 + 8x^5. \tag{2.1}$$

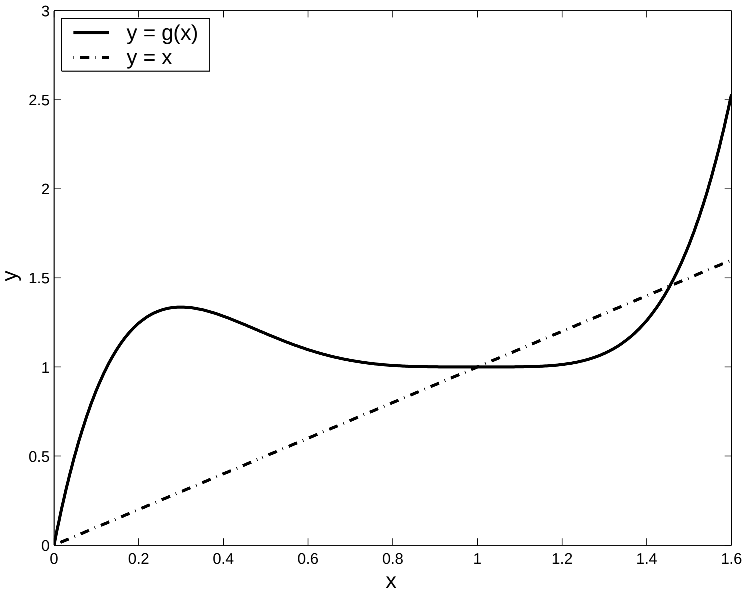

FIGURE 1. Graphs of the line $y = x$ and the function $y = g(x)$.

We can find that the real fixed points and the critical points of $g(x)$ as follows:

$$g(x) = x \quad \Longrightarrow \quad x = 0,\ 1,\ 1+\gamma,$$
$$g'(x) = 0 \quad \Longrightarrow \quad x = 0.3\,,\ 1,\ 1,\ 1,$$

in which

$$\gamma = \frac{1}{24} - \frac{23}{24(829 + 12\sqrt{4857})^{1/3}} + \frac{1}{24}(829 + 12\sqrt{4857})^{1/3} \approx 0.45$$

Noting $g''(0.3) = -13.72 < 0$ and $g^{(4)}(1) = 168 > 0$, we can deduce that $x = 0.3$ is a local maximizer and $x = 1$ is a local minimizer of $g(x)$. On the other hand, $g(0) = 0 < 1 = g(1)$ and $g(0.3) \approx 1.33 < 1 + \gamma = g(1 + \gamma)$. Therefore, $\underline{x} = 0, 1$ and $\overline{x} = 0.3,\ 1 + \gamma$ are minimizer and maximizer of $g(x)$ in the interval $[0, 1 + \gamma]$, respectively. Moreover, the interval $[0, 1 + \gamma]$ maps into itself by the function $g(x)$.

In the following theorem, it is proved that the sequence $x_{k+1} = g(x_k)$ is fourth-order convergent to $x = 1$ for any $x_0 \in (0, 1 + \gamma)$.

Theorem 2.1. *For any initial point $x_0 \in (0, 1+\gamma)$, the sequence $x_{k+1} = g(x_k)$ is fourth-order convergent to $x = 1$, in which the function $g(x)$ is defined by (2.1).*

Proof. We know that the function $g(x)$ maps the interval $[0, 1+\gamma]$ into itself. Considering an arbitrary initial point $x_0 \in (0, 1+\gamma)$, one can easily obtain the following considerations (For clarification, see Figure 1):

- The unique solution of the equation $g(x) = 1$ in the interval $[0, 1)$ is $\frac{1}{8}$.
- $g(x)$ is increasing in the interval $(0, \frac{1}{8})$. Therefore, if $x_k \in (0, \frac{1}{8})$, for some k, then there exists an index $k_0 \geqslant k$ such that either $x_{k_0} = \frac{1}{8}$, and so $x_{k_0+1} = 1$, or $x_{k_0+1} \in (\frac{1}{8}, 1)$.
- If $x_k \in (\frac{1}{8}, 1)$, for some k, then $x_{k+1} \in (1, 1+\gamma)$.
- If $x_k \in (1, 1+\gamma)$, for some k, then the sequence $\{x_{k+s}\}_{s \geqslant 1} \subseteq [1, 1+\gamma)$ is a strictly decreasing sequence converging to $x = 1$.

Noting the above considerations, we can conclude that the sequence $x_{k+1} = g(x_k)$ is convergent to $x = 1$. On the other hand,

$$g'(1) = g''(1) = g'''(1) = 0$$

implies that the convergence is fourth-order (See [**3**]). $\square$

Using iteration function (2.1), we obtain the following iterative method to find $A^{\dagger}_{MN}$:

$$X_{k+1} = X_k \left[12I - 38(AX_k) + 52(AX_k)^2 - 33(AX_k)^3 + 8(AX_k)^4\right],$$

that can be rewrite as follows:

$$\begin{aligned} \psi_k &= AX_k \\ \zeta_k &= \psi_k^2 \\ X_{k+1} &= X_k \left[12I - 38\psi_k + \zeta_k(52I - 33\psi_k + 8\zeta_k)\right], \quad k = 0, 1, \ldots. \end{aligned} \tag{2.2}$$

The iterative method (2.2) falls within the domain of methods for matrix inversion. It requires an initial matrix to start the process and can rapidly converge, which is an advantage over the existing methods. An important challenge when applying iterative methods for finding the WMP is related to the initial matrix. Here, the initial matrix plays a very crucial significance to provide convergence. Accordingly, we must apply the following initial matrix

$$X_0 = \beta A^{\#}, \tag{2.3}$$

where β is a suitable constant.

In the sequel, we first give a mathematical analysis to observe that under which conditions, (2.2) converges.

Lemma 2.1. *For the sequence $\{X_k\}$ generated by (2.2) with the initial matrix (2.3), it holds that*

$$\begin{aligned} (MAX_k)^* = MAX_k, \quad (NX_kA)^* = NX_kA, \\ X_kAA^{\dagger}_{MN} = X_k, \quad A^{\dagger}_{MN}AX_k = X_k. \end{aligned} \tag{2.4}$$

Proof. We will prove the conclusion by induction on k. For $k = 0$, and using (2.3), the first two equations can be verified easily. So, we only give a verification to the last two equations. Using the facts that $(AA^{\dagger}_{MN})^{\#} = AA^{\dagger}_{MN}$ and $(A^{\dagger}_{MN}A)^{\#} = A^{\dagger}_{MN}A$, we have

$$X_0 AA^{\dagger}_{MN} = \beta A^{\#} AA^{\dagger}_{MN} = \beta A^{\#}(AA^{\dagger}_{MN})^{\#} = \beta(AA^{\dagger}_{MN}A)^{\#} = \beta A^{\#} = X_0,$$
$$A^{\dagger}_{MN} AX_0 = \beta A^{\dagger}_{MN} AA^{\#} = \beta(A^{\dagger}_{MN}A)^{\#} A^{\#} = \beta(AA^{\dagger}_{MN}A)^{\#} = \beta A^{\#} = X_0.$$

Assume now that the conclusion holds for some $k > 0$. We show that it continues to hold for $k + 1$. Using the iterative method (2.2), one has

$$\begin{aligned}(MAX_{k+1})^* &= \left(MAX_k(12I - 38\psi_k + 52\psi_k^2 - 33\psi_k^3 + 8\psi_k^4)\right)^* \\ &= 12(M\psi_k)^* - 38(M\psi_k^2)^* + 52(M\psi_k^3)^* - 33(M\psi_k^4)^* + 8(M\psi_k^5)^* \\ &= 12M\psi_k - 38M\psi_k^2 + 52M\psi_k^3 - 33M\psi_k^4 + 8M\psi_k^5 \\ &= MAX_k(12I - 38\psi_k + 52\psi_k^2 - 33\psi_k^3 + 8\psi_k^4) \\ &= MAX_{k+1},\end{aligned}$$

which uses the fact that $(M\psi_k)^* = M\psi_k$, M is Hermitian positive definite ($M^* = M$), and also e.g.,

$$(M\psi_k^2)^* = \psi_k^*(M\psi_k)^* = \psi_k^*(M\psi_k) = \psi_k^* M^* \psi_k = (M\psi_k)^*\psi_k = M\psi_k\psi_k = M\psi_k^2.$$

Thus, the first equality in (2.4) holds for $k + 1$, and the second equality can be proved in a similar way. For the third equality in (2.4), using the assumption that $X_k AA^{\dagger}_{MN} = X_k$ or equivalently $\psi_k AA^{\dagger}_{MN} = \psi_k$ and the iterative method (2.2), we could write down

$$\begin{aligned}X_{k+1}AA^{\dagger}_{MN} &= X_k(12I - 38\psi_k + 52\psi_k^2 - 33\psi_k^3 + 8\psi_k^4)AA^{\dagger}_{MN} \\ &= 12X_kAA^{\dagger}_{MN} - 38X_k\psi_kAA^{\dagger}_{MN} + 52X_k\psi_k\psi_kAA^{\dagger}_{MN} \\ &\quad -33X_k\psi_k^2\psi_kAA^{\dagger}_{MN} + 8X_k\psi_k^3\psi_kAA^{\dagger}_{MN} \\ &= 12X_k - 38X_k\psi_k + 52X_k\psi_k\psi_k - 33X_k\psi_k^2\psi_k + 8X_k\psi_k^3\psi_k \\ &= X_k(12I - 38\psi_k + 52\psi_k^2 - 33\psi_k^3 + 8\psi_k^4) \\ &= X_{k+1}.\end{aligned}$$

Consequently, the third equality in (2.4) holds for $k + 1$. The fourth equality can similarly be proved, and the desired result follows. □

Lemma 2.2. *Let A is the matrix (1.3) and take X_0 as (2.3). Considering the conditions of Lemma 2.1, for each approximate inverse produced by (2.2), it holds that*

$$(N^{-1}V)^{-1}X_k(U^*M)^{-1} = \begin{bmatrix} D_k & 0 \\ 0 & 0 \end{bmatrix},$$

in which D_k is a diagonal matrix.

PROOF. Using relation (1.3) and definition of $A^{\#}$, we have

$$X_0 = \beta A^{\#} = (N^{-1}V) \begin{bmatrix} D_0 & 0 \\ 0 & 0 \end{bmatrix} (U^* M),$$

in which

$$D_0 = \beta D$$

is a diagonal matrix. Therefore,

$$(N^{-1}V)^{-1} X_0 (U^* M)^{-1} = \begin{bmatrix} D_0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Now, the principle of mathematical induction and Lemma 2.1 lead to

$$(N^{-1}V)^{-1} X_k (U^* M)^{-1} = \begin{bmatrix} D_k & 0 \\ 0 & 0 \end{bmatrix},$$

in which D_k is the following diagonal matrix:

(2.5)

$$D_{k+1} := \varphi(D_k) = D_k \left[12I - 38(DD_k) + 52(DD_k)^2 - 33(DD_k)^3 + 8(DD_k)^4\right].$$

This complete the proof. □

In the following theorem, we prove that the method (2.2) is fourth-order convergent.

THEOREM 2.2. *Assume that A is a $m \times n$ matrix whose weighted singular value decomposition is given by (1.4). Let furthermore that the initial matrix is available by (2.3) in which*

$$0 < \beta < \frac{7}{5\,\sigma_1^2}. \tag{2.6}$$

Then, the sequence of iterates produced by (2.2) converges to the WMP inverse $A^{\dagger}_{MN}$ with fourth-order.

PROOF. By considering (1.4) and in order to establish this result, we must show that

$$\lim_{k\to\infty} (N^{-1}V)^{-1} X_k (U^* M)^{-1} = \begin{bmatrix} D^{-1} & 0 \\ 0 & 0 \end{bmatrix}. \tag{2.7}$$

Let

$$D_k = diag(d_1^{(k)}, \ldots, d_r^{(k)}),$$

in which

$$d_i^{(0)} = \beta\sigma_i, \quad i = 1, \ldots, r.$$

It follows from Lemma 2.2 and relation (2.5) that

(2.8)

$$d_i^{(k+1)} = \varphi(d_i^{(k)}) = d_i^{(k)} \left[12 - 38\sigma_i d_i^{(k)} + 52(\sigma_i d_i^{(k)})^2 - 33(\sigma_i d_i^{(k)})^3 + 8(\sigma_i d_i^{(k)})^4\right].$$

Take $t_i^{(k)} = \sigma_i d_i^{(k)}$. Then, $t_i^{(k+1)} = g(t_i^{(k)})$ and, regarding (2.6), $t_i^{(0)} = \beta\sigma_i^2 \in (0, \frac{7}{5})$. According to Theorem 2.1, we have $\lim_{k\to\infty} t_i^{(k)} = 1$ with fourth-order. Thus, $d_i^{(k)} \to$

σ_i^{-1} with fourth-order, too. This shows the fourth-order of convergence for the presented method (2.2). The proof is complete. $\square$

3. Numerical experiments

In this section, we will make some numerical comparisons of our proposed method (2.2) with the method (1.8) and Schulz method (1.6). To do so, we focus on the total number of matrix multiplications and CPU times required for convergence. We compare the behavior of different methods for some randomly generated dense matrices.

All tests were carried out with a Matlab code, while the computer specifications are Microsoft Windows XP Intel(R), Pentium(R) 4, CPU 3.2 GHz, with 2 GB of RAM.

Consider the initial matrix X_0 according to (2.3), with β from (2.6). Since σ_1^2 is a (the largest) eigenvalue of $A^{\#}A$, we have $\sigma_1^2 \leqslant \|A^{\#}A\|_\infty \leqslant \|A^{\#}\|_\infty \, \|A\|_\infty$. Therefore, the selection

$$\beta = \frac{1}{\|A^{\#}\|_\infty \, \|A\|_\infty}$$

satisfies both in (1.7) and (2.3). The stop criterion is

$$\frac{\|X_{k+1} - X_k\|_\infty}{1 + \|X_k\|_\infty} < 10^{-7}$$

and the maximum number of iterations is set to 100 in our written codes as the maximum number of cycle for the methods considered in comparisons.

We present two different types of tests. Test 1 is devoted to compare the schemes for finding the weighted Moore-Penrose inverse, while test 2 gives some comparison for finding the Moore-Penrose inverse.

Test 1. In this test, we compute the weighted Moore-Penrose inverse of randomly generated dense matrix A of the size $m \times n$, $n = m + 50$, as follows:

$$A = 10\, rand(m, n) - 10\, rand(m, n),$$

where different Hermitian positive definite matrices M and N (which have also been constructed randomly) are in what follows:

$$M = qr(10\, rand(m, m) - 10\, rand(m, m)), \qquad M = M^* M;$$
$$N = qr(10\, rand(n, n) - 10\, rand(n, n)), \qquad N = N^* N.$$

For each $m = 100, 200, 300, 400, 500$, we have performed 10 tests and compared the average values of the matrix multiplications and the elapsed times in seconds. The results of comparisons are reported in Fig. 2, in terms of the number of matrix multiplications, and the computational time (in seconds) in Fig. 3. We observe that method (2.2) is better than others both in number of matrix multiplications and CPU time.

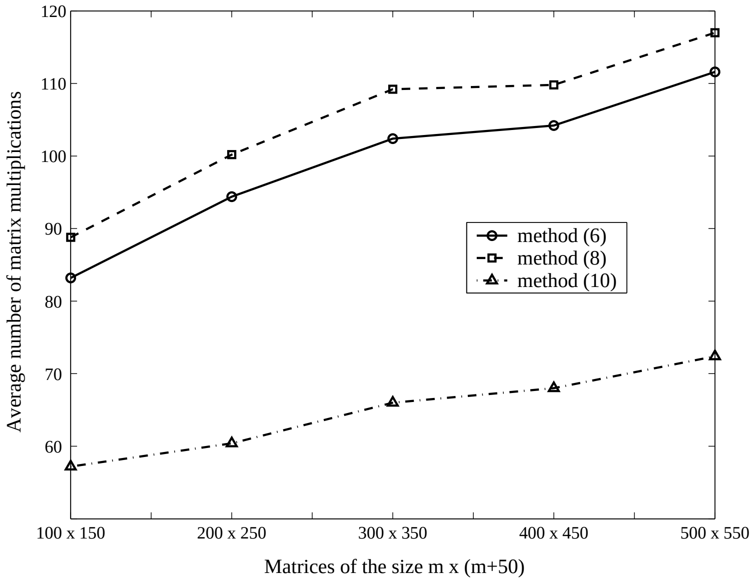

FIGURE 2. The average number of matrix multiplications to compute the weighted Moore-Penrose inverse by different methods

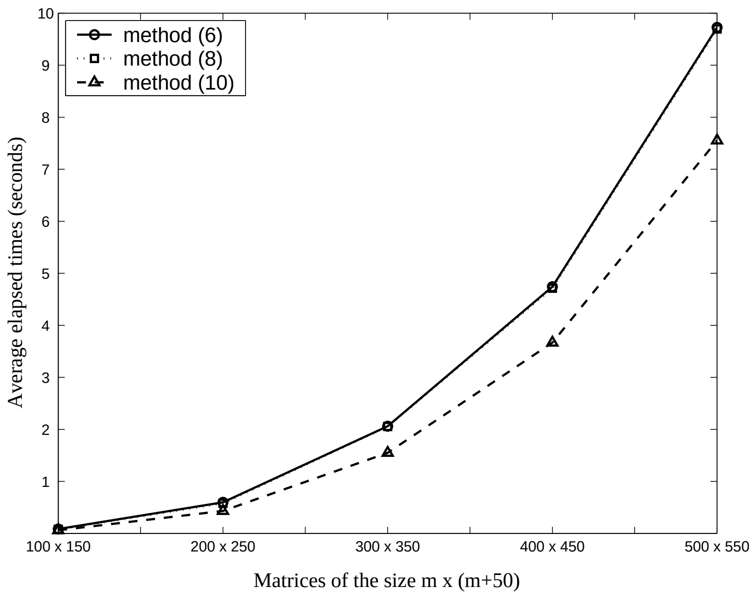

FIGURE 3. The average elapsed times to compute the weighted Moore-Penrose inverse by different methods

Test 2. In this test, we compute the Moore-Penrose inverse of randomly generated dense matrix A of the size $m \times n$, $n = m + 50$, as follows:

$$A = 10\, rand(m, n) - 10\, rand(m, n).$$

Again, for each $m = 100, 200, 300, 400, 500$, we have performed 10 tests and compared the average values of the matrix multiplications and the elapsed times in seconds. The results of comparisons are reported in Fig. 4, in terms of the number of matrix multiplications, and the computational time (in seconds) in Fig. 5. We observe that method (2.2) is better than others both in number of matrix multiplications and CPU time.

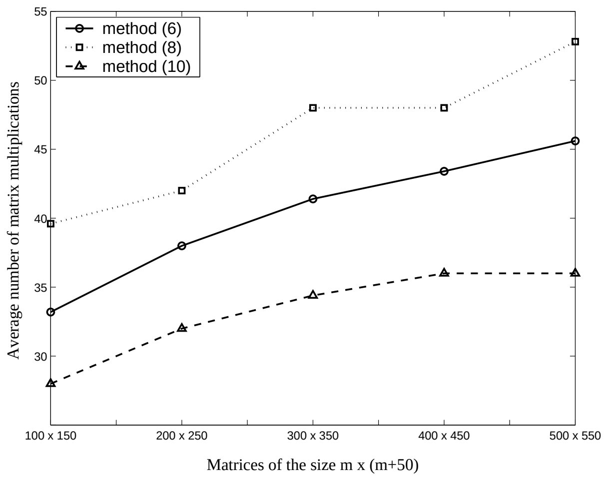

FIGURE 4. The average number of matrix multiplications to compute the Moore-Penrose inverse by different methods

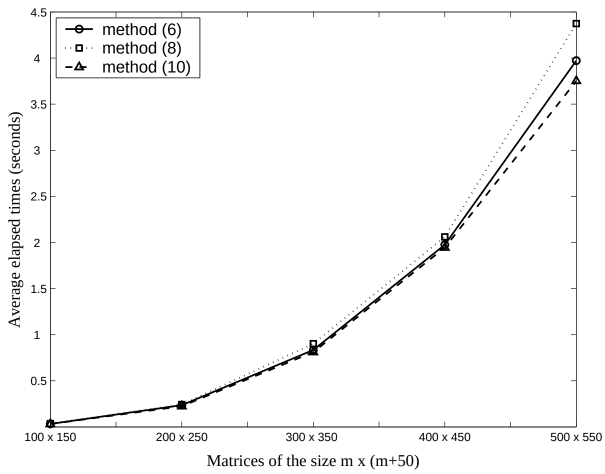

FIGURE 5. The average elapsed times to compute the Moore-Penrose inverse by different methods

4. Conclusions

In this paper, we proposed a new method to find the weighted Moore-Penrose inverse. It is proved that this method converge with fourth-order. Although our method is not a higher order scheme, a wide set of random numerical experiments showed that its number of matrix multiplications and its CPU time are considerably less than those of higher order methods. So, our method could be considered as an efficient method.

Acknowledgement

The authors would like to thank the referees for their useful comments and constructive suggestions which substantially improved the quality of this paper.

References

[1] A. Ben-Israel and T.N.E. Greville, *Generalized Inverses, Theory and Applications*, second ed. Springer, NY, 2003.

[2] A. Bojanczyk, N.J. Higham and H. Patel, Solving the indefinite least squares problem by hyperbolic QR factorization, *SIAM J. Matrix Anal. Appl.* **24**(4)(2003), 914–931.

[3] R.L. Burden and J.D. Faires, *Numerical Analysis*, 9th Ed. Brooks/Cole, Cengage Learning, Boston, MA., 2011.

[4] T.N.E. Greville, Some applications of the pseudo-inverse of matrix, *SIAM Rev.* **2**(1)(1960), 15–22.

[5] M.E. Gulliksson, P.A. Wedin and Y. Wei, Perturbation identities for regularized Tikhonov inverse and weighted pseudo inverse, *BIT Num.math.* **40**(3)(2000), 513–523.

[6] M. Gulliksson, X. O. Jin and Y.Wei, Perturbation bounds for constrained and weighted least squares problems, *Linear Algebra Appl.* **349**(1-3)(2002), 221–232.

[7] A.S. Householder, *The Theory of Matrices in Numerical Analysis*, Dover Publications, New York, 1964.

[8] F. Huang and X. Zhang, An improved Newton iteration for the weighted Moore-Penrose inverse, *Appl. Math. Comput.* **174**(2)(2006), 1460–1486.

[9] A. Kılıçman, Z. Al Zhour, The general common exact solutions of coupled linear matrix and matrix differential equations, *J. Anal. Comput.* **1**(2005), 15–30.

[10] A. Kılıçman and Z. Al Zhour, Iterative solutions of coupled matrix convolution equations, *Soochow J. Math* **33**(1)(2007), 167–180.

[11] R. Penrose, A generalized inverses for matrices, *Proc. Cambridge Philo. Soc.* **51**(3)(1955), 406–413.

[12] M.D. Petković, P.S. Stanimirović and M.B. Tasić, Effective partitioning method for computing weighted Moore-Penrose inverse, *Comput. Math. Appl.* **55**(8)(2008), 1720–1734.

[13] G. Schulz, Iterative Berechnung der reziproken matrix, *Z. Angew. Math. Mech.* **13**(1)(1933), 57–59.

[14] S. K. Sen, H. Agarwal and S. Sen, Chemical equation balancing: an integer programming approach, *Math. Comput. Model.* **44**(7-8)(2006), 678–691.

[15] F. Soleimani, F. Soleymani, A. Cordero and J.R. Torregrosa, On the extension of Householder's method for weighted Moore-Penrose inverse, *Appl. Math. Comput.* **231**(2014), 407–413.

[16] F. Soleymani, On a fast iterative method for approximate inverse of matrices, *Commun. Korean Math. Soc.* **28**(2)(2013), 407–418.

[17] W. Sun and Y. Wei, Inverse order rule for weighted generalized inverse, *SIAM J. Matrix Anal. Appl.* **19**(3)(1998), 772–775.

[18] M.Z. Ullah, F. Soleymani and A.S. Al-Fhaid, An efficient matrix iteration for computing weighted Moore-Penrose inverse, *Appl. Math. Comput.* **226**(2014),441–454.

[19] C.F. Van Loan, Generalizing the singular value decomposition, *SIAM J. Numer. Anal.* **13**(1)(1976), 76–83.

[20] G.R. Wang, A new proof of Greville's method for computing the weighted MP inverse, *J. Shanghai Normal University (Nat. Sci.),* **3**(1985), 32–38.

[21] Y. Wei, Perturbation bound of singular linear system, *Appl. Math. Comput.* **105**(2-3)(1999), 211–220.

[22] L. Weiguo, L. Juan, Q. Tiantian, A family of iterative methods for computing Moore-Penrose inverse of a matrix, *Linear Algebra Appl.* **438**(1)(2013), 47–56.

Received by editor 31.07.2016; Available online 05.09.2016.

Department of Mathematics, Bu-Ali Sina University, Hamedan, Iran
E-mail address: esmaeili@basu.ac.ir

Department of Mathematics, Bu-Ali Sina University, Hamedan, Iran
E-mail address: rerfanifar92@basu.ac.ir

Department of Mathematics, Bu-Ali Sina University, Hamedan, Iran
E-mail address: mrashidi94@basu.ac.ir