Applied Mathematics and Computation 316 (2018) 89–101

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

An efficient computation of generalized inverse of a matrix☆

V.Y. Pan[a,c], F. Soleymani[b,*], L. Zhao[a,c]

[a] Department of Mathematics and Computer Science, Lehman College of CUNY, Bronx, NY 10468, USA
[b] Department of Mathematics, Institute for Advanced Studies in Basic Sciences, Zanjan, Iran
[c] Department of Mathematics and Computer Science, The Graduate Center of CUNY, New York, NY 10036, USA

ARTICLE INFO

Keywords:
Generalized inverses
Hyperpower method
Moore–Penrose inverse
Convergence analysis
Drazin inverse

ABSTRACT

We propose a hyperpower iteration for numerical computation of the outer generalized inverse of a matrix which achieves 18th order of convergence by using only seven matrix multiplications per iteration loop. This yields a high efficiency index for that computational task. The algorithm has a relatively mild numerical instability, and we stabilize it at the price of adding two extra matrix multiplications per iteration loop. This implies an efficiency index that exceeds the known record for numerically stable iterations for this task, which means substantial acceleration of the long standing algorithms for an important problem of numerical linear algebra. Our numerical tests cover a variety of examples in the category of generalized inverses, such as Drazin case, rectangular case, and preconditioning of linear systems. The test results are in good accordance with our formal study.

1. Our subject, motivation, related works, and our progress

1.1. Generalized inverses: some applications

It has been stated already by Forsythe et al. [[12], p. 31] that in the great majority of practical computational problems, it is unnecessary and inadvisable to actually compute the inverse of a nonsingular matrix. This general rule still remains essentially true for modern matrix computations (see, e.g., [[41], pages 39 and 180]). In contrast the computation or approximation of *generalized inverses* is required in some important matrix computations (cf., e.g., [23]). For example, special cases of the generalized inverses are used in several application such as for preconditioning large scale (diagonal) linear systems of equations [4] and [[6], pages 171–208] and for updating the regression estimates based on the addition or deletion of the data in linear regression analysis [[3], pages 253–294]. Furthermore the computation of the so-called zero initial state system inverses for linear time-invariant state-space systems is essentially equivalent to determining generalized inverses of the matrices of the associated transfer functions.

There exists a generalized inverse of an arbitrary matrix, and it turns into a unique inverse when the matrix is nonsingular, but we must compute generalized inverses in order to deal with rectangular and rank deficient matrices [34,39]. Some generalized inverses can be defined in any mathematical structure that involves associative multiplication, i.e., in a semigroup [[46], chapter 1].

☆ Supported by NSF Grant CCF–1116736 and PSC CUNY Award 68862–00 46.

* Corresponding author.
E-mail addresses: victor.pan@lehman.cuny.edu (V.Y. Pan), fazl_soley_bsb@yahoo.com (F. Soleymani), lzhao1@gc.cuny.edu (L. Zhao).
URL: http://comet.lehman.cuny.edu/vpan/ (V.Y. Pan)

http://dx.doi.org/10.1016/j.amc.2017.08.010
0096-3003/

A system $Ax = b$ of linear equations has a solution if and only if the vector $A\dagger b$ is a solution, and if so, then all solutions are given by the following expression:

$$x = A^{\dagger}b + [I - A^{\dagger}A]w, \tag{1.1}$$

where we can choose an arbitrary vector w and any generalized inverse $A\dagger$.

1.2. Outer generalized inverse

Hereafter $\mathbb{C}^{m\times n}$ denotes the set of all complex $m \times n$ matrices, $\mathbb{C}_r^{m\times n}$ denotes the set of all complex $m \times n$ matrices of rank r, I_m denotes the $m \times m$ identity matrix, and we drop the subscript if the dimension m is not important or is clear from context. Furthermore A^*, $R(A)$, and $N(A)$ denote the conjugate (Hermitian) transpose, the Range, and the Null Space of a matrix $A \in \mathbb{C}^{m\times n}$, respectively.

For $A \in \mathbb{C}^{m\times n}$, outer generalized inverses or {2}-inverses are defined [3] by

$$A\{2\} = \{X \in \mathbb{C}^{n\times m} : XAX = X\}. \tag{1.2}$$

For two fixed subspaces $S \subseteq \mathbb{C}^n$ and $T \subseteq \mathbb{C}^m$, define the generalized inverse $A^{(2)}_{T,S} \in A\{2\}$ of a complex matrix $A \in \mathbb{C}^{m\times n}$ as the matrix $X \in \mathbb{C}^{n\times m}$ such that $R(X) = T$ and $N(X) = S$.

Lemma 1.1. *Let a matrix $A \in \mathbb{C}^{m\times n}$ have rank r and let T and S be subspaces of $\mathbb{C}^n$ and $\mathbb{C}^m$, respectively, with $dimT = dimS^{\perp} = t \leq r$. Then A has a {2}−−inverse X such that $R(X) = T$ and $N(X) = S$ if and only if*

$$AT \bigoplus S = \mathbb{C}^m, \tag{1.3}$$

in which case X is unique and is denoted by $A^{(2)}_{T,S}$ (see, e.g., [48]).

The traditional generalized inverses, e.g., the pseudo-inverse $A\dagger$ (a.k.a. Moore–Penrose inverse), the weighted Moore–Penrose inverses $A^{\dagger}_{MN}$ (where M and N are two square Hermitian positive definite matrices), the Drazin-inverse A^D, the group inverse $A^{\#}$, the Bott–Duffin inverse A^{-1}_L [32], the generalized Bott–Duffin inverse $A^{\dagger}_L$, and so on, each of special interest in matrix theory, are special cases of the generalized outer inverse $X = A^{(2)}_{T,S}$.

1.3. The known iterative algorithms for generalized inverses

A number of direct and iterative methods has been proposed and implemented for the computation of generalized inverses (e.g., see [25,31]). Here we consider iterative methods. They approximate generalized inverse preconditioners, can be implemented efficiently in parallel architecture, converge particularly fast in some special cases (see, e.g., [21]), and compute various generalized inverses by using the same procedure for different input matrices, while direct methods usually require much more computer time and space in order to achieve such results.

Perhaps the most general and well-known scheme in this category is the following family of hyperpower matrix iterations [9,38,42],

$$X_{k+1} = X_k(I + R_k + \cdots + R_k^{p-1}) = X_k \sum_{i=0}^{p-1} R_k^i, \quad R_k = I - AX_k, \quad k \geq 0. \tag{1.4}$$

Straightforward implementation of the iteration (1.4) of order p involves p matrix-matrix products. For $p = 2$ it turns into the Newton–Schulz–Hotelling matrix iteration (SM), originated in [17,29]:

$$X_{k+1} = X_k(2I - AX_k), \tag{1.5}$$

and for $p = 3$ into the cubically convergent method of Chebyshev–Sen–Prabhu (CM) [30]:

$$X_{k+1} = X_k(3I - AX_k(3I - AX_k)). \tag{1.6}$$

The paper [35] proposed the following seventh-order factorization (FM) for computing outer generalized inverse with prescribed range and null space assuming an appropriate initial matrix X_0 (see Section 4 for its choices):

$$\begin{cases} \psi_k = I - AX_k, \\ \zeta_k = I + \psi_k + \psi_k^2, \\ \upsilon_k = \psi_k + \psi_k^4, \\ X_{k+1} = X_k(I + \upsilon_k \zeta_k). \end{cases} \tag{1.7}$$

Chen and Tan [7] proposed computing $A^{(2)}_{T,S}$ by iterations based on splitting matrices.

For further background of iterative methods for computing generalized inverses, one may consult [[18], pages 82–84], [[22], chapter 1], [3,6,23,48]. Ben-Israel [2], Pan [24] and Sticrel [42] have presented general introductions into iterative methods for computing $A^{(2)}_{T,S}$. Recently such methods have been studied extensively together with their applications (see, e.g., [8,20,27]).