Journal of Computational and Applied Mathematics 279 (2015) 187–191

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Finding generalized inverses by a fast and efficient numerical method

M. Sharifi, M. Arab *, F. Khaksar Haghani

Department of Mathematics, Shahrekord Branch, Islamic Azad University, Shahrekord, Iran

ARTICLE INFO

Article history:
Received 12 August 2014
Received in revised form 7 November 2014

MSC:
15A09

Keywords:
Generalized inverses
Efficiency index
Hyperpower method
Order

ABSTRACT

In this paper, a method with very high order of convergence is constructed and analyzed. The method is used to compute generalized inverses. The efficiency index has been employed to show its superiority. Numerical experiments re-verify that the proposed iterative expression is more effective than the existing methods of the same type.

1. Introductory notes

High-order matrix iterative methods for finding generalized inverses appeared for the first time in [1] and then they were extensively studied in some recent papers (see e.g. [2–4]). This class of iteration methods has drawn a considerable attention in recent years, which led to the construction of many good methods of this type (see e.g. [5] and the references therein).

The reason for the revived interest in this area is a nice property of high-order methods to overcome theoretical limits of low-order ones concerning the convergence order, informational efficiency and computational efficiency, which are of great practical importance.

High-order matrix methods are primarily introduced with the aim to achieve as high as possible order of convergence using a fixed number of matrix–matrix multiplications (mmm). Since the most impressive cost in implementing such methods (also known as Schulz-type methods) is the cost of mmm. This is closely connected to the definition of important indices such as informational efficiency and computational efficiency indices.

In this work, we consider the latter, i.e. the efficiency index defined by

$$EI = p^{\frac{1}{\theta}}, \tag{1}$$

where p and θ are the convergence order and the number of mmm per cycle of a matrix iterative method.

Considering (1), we are aimed at designing a new iterative method of very high order of convergence which is also economic. That is to say, its computational efficiency index must be reasonable for finding generalized inverses.

Assume that $A \in \mathbb{C}_r^{m\times n}$ is a matrix of rank r, T is a subspace of $\mathbb{C}^n$ of dimension $t \le r$ and S is a subspace of $\mathbb{C}^m$ of dimension $m - t$, then A has a generalized inverse X with range $\mathcal{R}(X) = T$ and null space $\mathcal{N}(X) = S$ if and

* Corresponding author.
E-mail addresses: iau.sharifi@yahoo.com (M. Sharifi), monaarab_f@yahoo.com (M. Arab), haghani1351@yahoo.com (F. Khaksar Haghani).

http://dx.doi.org/10.1016/j.cam.2014.11.009
0377-0427/

only if

$$AT \oplus S = \mathbb{C}^m. \tag{2}$$

In the case when the existence of this outer inverse is ensured, X is unique and it is denoted by $A^{(2)}_{T,S}$, [6].

It is well known that many of the famous inverses such as Moore–Penrose, weighted Moore–Penrose, Drazin, weighted Drazin, etc. are some special cases of $A^{(2)}_{T,S}$, [7].

The Moore–Penrose inverse is a generalization of the inverse of a nonsingular matrix which plays an important role in various fields, such as eigenvalue problems and the linear least square problems. The Drazin inverse has been very useful in Markov chains, multi-body system dynamics, singular difference and differential equations, differential–algebraic equations and numerical analysis [8].

Here we remark that there are many current and related references, some of which are interesting and beneficial to our paper in what follows: computing Moore–Penrose inverses of matrices by high-order Schulz-type iterations [5], computing the group inverses of singular Toeplitz matrices by hyperpower algorithm [9], approximating outer generalized inverse [10], Drazin inverse [11].

The explicit expression for $A^{(2)}_{T,S}$ cannot be directly used in some practical problems. Hence, various numerical solution methods are developed, such as the one in [9]. Note that the validity of these iterative methods is guaranteed under milder conditions.

The hyperpower iteration method of order p [12] can be defined by

$$X_{k+1} = X_k(I + R_k + \cdots + R_k^{p-1}), \quad R_k = I - AX_k. \tag{3}$$

This iteration requires p mmm to achieve pth order of convergence. Choosing $p = 2$ yields to the Schulz matrix iteration [13]

$$X_{k+1} = X_k(2I - AX_k), \tag{4}$$

with quadratic convergence, while choosing $p = 3$ results in the following cubically convergent method [14]

$$X_{k+1} = X_k(3I - AX_k(3I - AX_k)). \tag{5}$$

In such fixed-point-type methods, the initial matrix must be chosen in the form $X_0 = \alpha G$, where α is an appropriate real parameter and G is a matrix of dimensions $n \times m$ and of rank s, $0 < s \le r = \text{rank}(A)$ [8].

The following sections uncover the material in what follows. In Section 2, we consider the high-order scheme of hyperpower for the case $p = 30$. We remark that several results for lower values of p have been published in the literature (see e.g. [15,16] and the references therein). Then, we improve its efficiency index by proper factorization. The derived method is very robust with higher computational efficiency index. Section 3 presents the numerical behavior of the proposed formulation on a matrix problem. A conclusion of this paper will be drawn in Section 4.

2. An efficient iteration

Let us first consider the following scheme extracted from (3) for $p = 30$

$$X_{k+1} = X_k(I + R_k + R_k^2 + \cdots + R_k^{29}). \tag{6}$$

The iteration (6) possesses

$$EI = 30^{\frac{1}{30}} \approx 1.1200, \tag{7}$$

as its computational efficiency index which is terrible. This shows that (6) may not be a challenging iterative method in practice. Here our main goal is to propose a variant of (6) which is much more reliable in case of computational efficiency.

To this end, we should start simplifying (6) by proper matrix factorizing. Doing such a procedure lead us to the following variant

$$X_{k+1} = X_k(I + R_k)(I + R_k^2 + R_k^4)(I + (R_k^2 + R_k^8)(R_k^4 + R_k^{16})), \tag{8}$$

where only nine mmm are required, so that the efficiency index is equal to

$$EI = 30^{\frac{1}{9}} \approx 1.4592. \tag{9}$$

Now we are able to establish the rate of convergence for (8) in what follows.

Theorem 2.1. *Let $A \in \mathbb{C}_r^{m\times n}$ be a given matrix of rank r and $G \in \mathbb{C}_s^{n\times m}$ be a given matrix of rank $0 < s \le r$, which satisfy $\text{rank}(GA) = \text{rank}(G)$. Then, the sequence $\{X_k\}_{k=0}^{k=\infty}$ defined by the iterative method (8) converges to $A^{(2)}_{R(G),N(G)}$ with 30th-order if the initial approximation $X_0 = \alpha G$ satisfies*

$$\|F_0\| = \|AA^{(2)}_{T,S} - AX_0\| < 1. \tag{10}$$