

ITERATIVE METHOD FOR SHARPENING MATRIX INVERSES

AWNI M. ABU-SAMAN

ABSTRACT. This paper describes a numerical iterative method for sharpening the inverse of a matrix as a mean for solving linear systems of equations. The iterative algorithm, based on iterative procedure, only require scalable matrix. The iterative algorithm used has the benefit of being able to modify the inverse of a nearly singular systems by considering the Gaussian or LU inverse as initial matrix inverse. The algorithm will be applied for solving the system of linear equations.

2000 Mathematics Subject Classification: 15A09.

Keywords: Scaled matrix, Iterative algorithm, Mathematical software.

1. INTRODUCTION

The problem of matrix inversion is considered to be one of the basic problems widely encountered in science and engineering fields. It is usually an essential part of many solutions, e.g., as preliminary steps for optimization, single processing, electromagnetic systems, robotic control, statistics and physics.

Singular and nearly singular matrices as well as matrices with large condition numbers are omnipresent in computations for physics, engineering, medicine and signal processing. Representations of such a matrix enable its fast multiplication by a vector and expression of its inverse via the solutions of a few linear systems of equations. The later problems (of inversion and linear system solving) are highly important for the theory and practice of computing, the real-time inversion of matrices is usually desired. Since the mid 1980s, efforts have been directed toward computational aspects of fast matrix inversion, and many algorithms have thus been proposed.

Some effective direct solution algorithms exploiting displacement representation can be found in [6, 8]. Alternative iterative methods were proposed in [1]. The later methods nontrivially extend some preceding work for general input matrices [8] and can be most effective for well conditioned inputs.

There are many algorithms, such as LU -decomposition algorithm, Gaussian elimination method, recurrent neural networks, successive matrix squaring algorithm and singular value decomposition to compute the matrix inverse [7, 9], including Newtons iteration method [3, 6]. Newtons iteration algorithm for matrix inversion was first proposed by Schults in 1933 and was well studied in [7]. Newtons iteration is simple to describe and to analyze and is numerically stable.

Householder in [2] defined successive improvements of a matrix B to solve the matrix equation $AB = M$, for nonsingular matrix A , using the iterative relation

$$B_{i+1} = B_i + C(M - AB_i), i = 0, 1, \dots \quad (1)$$

A particular case of the above general iterative scheme is defined by the choice $M = I$ and $C_i = B_i$, which turns into the Newtons iterative algorithm [6].

The outline of this paper is as follows. After recalling the LU -decomposition algorithm for the inverse of a matrix, In section 2, we review Newtons iteration algorithm for matrix inverse. In section 3 we introduce the modified Newtons iteration algorithm. Finally, numerical results are reported to illustrate the convergence of the modified algorithm

2. ITERATIVE MATRIX INVERSION

To solve for a matrix inverse, the neural-network design methods are usually based on the definition equation $I - AB = I - BA = 0$, where $A \in R^{n \times n}$ matrix is assumed nonsingular and $I \in R^{n \times n}$ denotes an identity matrix.

Consider a matrix $A \in R^{n \times n}$ with approximate initial inverse B , Newtons iteration algorithm for computing matrix inverse is:

$$B_{i+1} = B_i(I + Eps_i), i = 0, 1, \dots \quad (2)$$

Where the matrix $Eps_i = (I + AB_i)$, and B_0, B_1, B_2, \dots defines a sequence of approximations for the inverse of the matrix A . The residuals $(I + AB_i)$ and $(I + B_iA)$ should be squared in each step. Thus the sequence of the matrices rapidly converge to the inverse of A if initially the norms $\|I - AB_i\|$ and/or $\|I - B_iA\|$ are substantially less than 1.

The above algorithm (2.1) can be modified by multiplying the right hand side by a scaling factor to accelerate the convergence [7]

$$B_{i+1} = a_i B_i(I + Eps_i), i = 0, 1, \dots \quad (3)$$

Let B be the estimate inverse of the matrix A then

$$\begin{aligned} AB &= I + Eps \\ A^{-1}AB &= A^{-1}[I + Eps] \\ A^{-1} &= B[I + Eps]^{-1} \end{aligned}$$

$$A^{-1} = B[I + Eps + Eps^2 + Eps^3 + \dots + Eps^n] \quad (4)$$

The iterative form of (2.3) has the form:

$$B_{i+1} = B_i[I + Eps_i + Eps_i^2 + Eps_i^3 + \dots + Eps_i^n] \quad (5)$$

Where $Eps^n \rightarrow 0$ as $n \rightarrow \infty$

Algorithm 1.1. Iterative method for computing the matrix inverse

- Given A as the $n \times n$ original matrix,
- Factorize A into a product of LU matrices,
- set $L(V) = I, UB = V$,
- Use the forward substitution to Find columns of V , then use backward substitution to find columns of B
- Make sure that B is such that $\|I - AB\|$ and/or $\|I - BA\|$ are less than 1.
- Compute the initial value for the matrix $Eps = (I + AB)$,
- B is the estimate of the inverse such that $AB_i = I + Eps_i$,
- Compute the sequence

$$B_{i+1} = B_i[I + Eps_i + Eps_i^2 + Eps_i^3 + \dots + Eps_i^n] \quad (6)$$

- Repeat until $\|I - AB_i\|$ and/or $\|I - B_iA\|$ are less than tolerance value.

3. NUMERICAL RESULTS

Example 3.1.

$$A = \begin{pmatrix} 4 & 6 & 4 & 1 \\ 10 & 20 & 15 & 4 \\ 20 & 45 & 36 & 10 \\ 35 & 84 & 70 & 20 \end{pmatrix}$$

The above matrix has condition number $K(A) = 1.82 * 10^4$

For $b = (1 \ 3 \ 2 \ 4)^T$, the solution is $x = (-10 \ 36 \ -83 \ 175)^T$. After making small change in the coefficient matrix A

$$A = \begin{pmatrix} 4 & 6 & 4 & 1 \\ 10 & 20 & 15 & 4.03 \\ 20 & 45 & 36 & 10 \\ 34.98 & 84 & 70 & 20 \end{pmatrix}$$

We obtain a significant change in the solution

$$x = (-1.9725 \ 9.2153 \ -226950 \ 44.3781)^T$$

The initial matrix inverse is taken as:

$$B = \begin{pmatrix} 3.978 & -6.015 & 4.000 & -1.000 \\ -10.006 & 20.000 & -15.000 & 4.000 \\ 20.000 & -45.000 & 36.000 & -10.000 \\ 34.962 & 84.000 & 70.010 & 20.000 \end{pmatrix}$$

$$C = I - A * B = \begin{pmatrix} 0.0860 & 0.0600 & 0.100 & 0 \\ 0.1880 & 0.1500 & 0.0400 & 0 \\ 0.3300 & 0.3000 & 0.1000 & 0 \\ 0.5140 & 0.5250 & 0.2000 & 0 \end{pmatrix}$$

Norm of the matrix $C = 0.9257$, after 13 iterations with double precision MATLAB [4] calculations, the initial inverse is sharpened to

$$A^{-1} = \begin{pmatrix} 4.000 & -6.001 & 4.000 & -1.000 \\ -10.000 & 20.000 & -15.000 & 4.000 \\ 20.000 & -45.000 & 36.000 & -10.000 \\ -35.000 & 84.000 & -70.000 & 20.000 \end{pmatrix}$$

4. CONCLUSIONS

In this paper, we first recalled Newtons iteration for the inversion of nearly singular matrices, iterative relations derived from Newtons iterative algorithm, and then presented an alternative strategy based on criterion based on modification of Newtons algorithm, The initial inverse matrix is taken as the LU -factorization inverse. Thus the sequence of the matrices proved to be more rapidly convergent to the inverse of the matrix A .

REFERENCES

- [1] D. Bini, V. Pan, *Polynomial and Matrix Computations*, vol. 1: Fundamental Algorithms, Birkaser, Boston, 1994.
- [2] A. S. Householder, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964, Chapter 7.
- [3] T. Kailath, *Fast Reliable Algorithms for Matrices with Structure*, SIAM Publications, Philadelphia, 1999.
- [4] MATLAB Users Guide, Version 7.3, *The Math Works Inc.*, 2006.
- [5] V. Y. Pan, A. Zheng, *Superfast algorithms for Cauchy-like matrix computations and extensions*, Linear Algebra and Its Applications, Vol. 310 (2000), pp. 83-108.
- [6] V. Y. Pan, A. Zheng, X. Huang, O. Dias, *Newtons iteration for inversion of Cauchy-like and other Structure matrices*, Journal of Complexity, Vol. 13 (1997), pp. 108-124.
- [7] V. Y. Pan, R. Schreiber, *An improved Newton iteration for the generalized inverse of a matrix, with applications*, SIAM Journal on Scientific and Statistical Computing, Vol. 12 (5), (991), pp. 1109-1131.
- [8] T. Sderstrm, G. Stewart, *On the numerical properties of an iterative method for computing the Moore-Penrose generalized inverse*, SIAM Journal of Numerical Analysis, Vol. 11 (1974), pp. 61-74.
- [9] Y. Wei, *Recurrent neural networks for computing Weighted Moore Penrose inverse*, Applied Math. Comp. Vol. 116 (2000), pp. 279-287.

Awni M. Abu-Saman
Mathematics Department
Alazhar University-Gaza
P.O.Box 1277, Gaza-Gaza Strip
Palestinian Authority