Journal of Mathematical Modeling
Vol. 5, No. 1, 2017, pp. 1-14

JMM

GGMRES: A GMRES–type algorithm for solving singular linear equations with index one

Alireza Ataei[a*] **and Faezeh Toutounian**[b]

[a] *Mathematics Department, Faculty of Science, Persian Gulf University, Iran*
[b] *Department of Applied Mathematics, School of Mathematical Sciences, Ferdowsi University of Mashhad, Iran*
Emails: ataei@pgu.ac.ir, toutouni@math.um.ac.ir

Abstract. In this paper, an algorithm based on the Drazin generalized conjugate residual (DGMRES) algorithm is proposed for computing the group-inverse solution of singular linear equations with index one. Numerical experiments show that the resulting group-inverse solution is reasonably accurate and its computation time is significantly less than that of group-inverse solution obtained by the DGMRES algorithm.

Keywords: singular linear systems, DGMRES method, group-inverse solution, Drazin-inverse solution, Krylov subspace methods.
AMS Subject Classification: 15A06, 15A09, 65F10, 65F50.

1 Introduction

Consider the linear system

$$Ax = b, \tag{1}$$

where $A \in \mathbb{C}^{N\times N}$ is a singular matrix and $\mathrm{ind}(A)$ is arbitrary. Here $\mathrm{ind}(A)$, the index of A is the size of the largest Jordan block corresponding to the zero eigenvalue of A. We recall that the Drazin-inverse solution of (1) is the vector $A^D b$, where A^D is the Drazin-inverse of the singular matrix A. For

*Corresponding author.
Received: 27 October 2016 / Revised: 6 November 2016 / Accepted: 6 November 2016.

the Drazin-inverse and its properties, we refer to [2] or [5]. In the important special case $k = 1$, this matrix is called the group inverse of A and denoted by $A^{\sharp}$.

The Drazin-inverse has various applications in the theory of finite Markov chains [5], the study of singular differential and difference equations [5], the investigation of Cesaro-Neumann iterations [9], cryptography [8], iterative methods in numerical analysis [6, 7], multibody system dynamics [13] and others.

It is well known that the representations of the Drazin (group) inverse of matrices are very important not only in matrix theory, but also in singular differential and difference equations, probability statistical, numerical analysis, game theory, econometrics, control theory and so on [2, 5], and also singular systems with index one arise naturally in Markov chain modelling [3, 11].

The problem of finding the solution of the form $A^D b$ for (1) is very common in the literature and many different techniques have been developed in order to solve it. In [14], Sidi proposed a general approach to Krylov subspace methods for computing Drazin-inverse solution. And then, he gave several Krylov subspace methods of Arnoldi, the Drazin generalized conjugate residual (DGCR) and Lancoze types. Moreover in [15, 16], Sidi has continued to drive two Krylov subspace methods for computing $A^D b$. One is DGMRES method, which is implementation of the DGCR method for singular systems that is analogues to GMRES for non-singular systems. The other one is the Drazin biconjugate gradient algorithm (DBI-CG) which is Lanczos type algorithm. DGMRES, just like, GMRES method, is a stable numerically and economical computationally and storage wise. DBI-CG method, also just like the biconjugate gradient (BI-CG) for non–singular systems, is a fast algorithm, but when we need a high accuracy, the algorithm is invalid. In the present paper, we develop the group generalized minimal residual (GGMRES) algorithm which is another implementation of DGMRES, for solving the singular linear system (1) with $\mathrm{ind}(A) = 1$. By numerical examples, we show that the computation time of GGMRES algorithm is substantially less than that of DGMRES algorithm.

The paper is organized as follows. In Section 2, we will give a review of DGMRES. In Section 3, we will derive the GGMRES algorithm. In Section 4 the results of some numerical examples are given. Section 5 is devoted to concluding remarks.

2 DGMRES algorithm

DGMRES method is a Krylov subspace method for computing the Drazin–inverse solution of consistent or inconsistent linear systems (1) [14, 16]. In this method, there is no restriction on the matrix A. Thus, in general, A is non-Hermitian, $a := \mathrm{ind}(A)$ is arbitrary, and the spectrum of A can have any shape. DGMRES starts with an initial vectors x_0 and generates a sequence of vectors $x_0, x_1, \ldots$, as

$$x_m = x_0 + \sum_{i=1}^{m-a} c_i A^{a+i-1} r_0, \qquad r_0 = b - Ax_0.$$

Then

$$r_m = b - Ax_m = b - \sum_{i=1}^{m-a} c_i A^{a+i} r_0.$$

The Krylov subspace we will use is

$$\mathcal{K}_{m-a}\{A; A^a r_0\} = \mathrm{span}\{A^a r_0, A^{a+1} r_0, \ldots, A^{m-1} r_0\}.$$

The vector x_m produced by DGMRES satisfies

$$\|A^a r_m\| = \min_{x \in x_0 + \mathcal{K}_{m-a}\{A; A^a r_0\}} \|A^a (b - Ax)\|_2. \tag{2}$$

As $x_m = x_0 + \sum_{i=1}^{m-a} c_i A^{a+i-1} r_0$, we start by orthogonalizing the krylov vectors $A^a r_0, A^{a+1} r_0, \ldots$, using the Arnoldi–Gram–Schmidt process, see [1, 12], carried out numerically like the modified Gram–Schmidt process:

- For $i = 1, 2, \ldots,$ do
- Compute $h_{ji} = (v_j, Av_i), \quad j = 1, 2, \ldots, i.$
- Compute $\hat{v}_i = Av_i - \sum_{j=1}^{i} v_j h_{ji}$.
- Let $h_{i+1,i} = \|\hat{v}_i\|_2$ and set $v_{i+1} = \hat{v}_i / h_{i+1,i}$.
- EndDo

Consequently, we have a set of orthonormal vectors $v_1, v_2, \ldots$, that satisfies

$$Av_i = \sum_{j=1}^{i+1} v_j h_{ji}, \quad i = 1, 2, \ldots, \tag{3}$$

as long as $i \leq q - 1$, where q is the degree of the minimal polynomial of A with respect to $A^a r_0$, hence with respect to v_1. Furthermore, for each k,

$$\mathrm{span}\{\mathrm{v}_1, \mathrm{v}_2, \ldots, \mathrm{v}_\mathrm{k}\} \quad = \mathrm{span}\{\mathrm{A}^a \mathrm{r}_0, \mathrm{A}^{a+1} \mathrm{r}_0, \ldots, \mathrm{A}^{k+a-1} \mathrm{r}_0\} = \mathcal{K}_\mathrm{k}(\mathrm{A}; \mathrm{A}^a \mathrm{r}_0). \tag{4}$$

If we now define the $N \times k$ matrix $\hat{V}_k$ by

$$\hat{V}_k = [v_1|v_2|\dots|v_k], \quad k = 1, 2, \dots, \tag{5}$$

then, for $m \leq m_0$ (for definition of m_0 see [14] and [16]), we can write

$$x_m = x_0 + \hat{V}_{m-a}\xi_m, \quad \text{for some } \xi_{\mathrm{m}} \in \mathbb{C}^{\mathrm{m-a}} \tag{6}$$

where we need to determine ξ_m. Since $r_m = r_0 + A\hat{V}_{m-a}\xi_m$, we have

$$A^a r_m = A^a r_0 + A^{a+1}\hat{V}_{m-a}\xi_m = \beta v_1 - A^{a+1}\hat{V}_{m-a}\xi_m. \tag{7}$$

Next, provided $k \leq q-1$, from (3) we can write

$$A\hat{V}_k = \hat{V}_{k+1}\bar{H}_k, \bar{H}_k \quad = \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & h_{1k} \\ h_{21} & h_{22} & \cdots & \cdots & h_{2k} \\ 0 & h_{32} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & h_{kk} \\ 0 & \cdots & \cdots & 0 & h_{k+1,k} \end{bmatrix}. \tag{8}$$

Note that $\bar{H}_k \in \mathbb{C}^{(k+1)\times k}$ and $\bar{H}_k$ has full rank when $k \leq q-1$ [16]. Now, by using (7), (8), and $\hat{V}_{m+1}^* \hat{V}_{m+1} = I_{(m+1)\times(m+1)}$ we can reduce the $n \times (m-a)$ least squares problem of (2) to the $(m+1) \times (m-a)$ least squares problem

$$\begin{aligned} \|A^a r_m\| &= \|\hat{V}_{m+1}(\beta e_1 - \hat{H}_m \xi_m)\| \\ &= \min_{\xi \in \mathbb{C}^{m-a}} \|\hat{V}_{m+1}(\beta e_1 - \hat{H}_m \xi)\| \\ &= \min_{\xi \in \mathbb{C}^{m-a}} \|\beta e_1 - \hat{H}_m \xi\|, \end{aligned} \tag{9}$$

where

$$\hat{H}_m = \bar{H}_m \bar{H}_{m-1} \dots \bar{H}_{m-a}, \tag{10}$$

and $\hat{H}_m \in \mathbb{C}^{(m+1)\times(m-a)}$. In general, the value of n is very large and $m \ll n$, which implies that the problem in (9) is very small. The minimization problem (9) is accomplished by using the QR decomposition of $\hat{H}_m$. For more details we refer the reader to [14] and [16].

We now summarize the steps of DGMRES for the solution (1) where $a = \mathrm{ind(A)}$ is known.

Algorithm 1 DGMRES algorithm

1. Pick x_0 and compute $r_0 = b - Ax_0$ and $A^a r_0$.
2. Compute $\beta = \|A^a r_0\|$ and set $v_1 = \beta^{-1}(A^a r_0)$.
3. Orthogonalize the Krylov vectors $A^a r_0, A^{a+1} r_0, \ldots,$ via the Arnoldi-Gram-Schmidt process carried out like the modified Gram-Schmidt process:
   For $i = 1, 2, \ldots,$ do
   Compute $h_{ij} = (v_j, Av_i), j = 1, 2, \ldots, i.$
   Compute $\hat{v}_i = Av_i - \sum_{j=1}^{i} v_j h_{ij}.$
   Let $h_{i+1,i} = \|\hat{v}_i\|$ and set $v_{i+1} = \hat{v}_i / h_{i+1,i}.$
4. EndDo
5. For $k = 1, 2, \ldots,$ form the matrices $\hat{V}_k \in \mathbb{C}^{N \times k}$ and $\bar{H}_k \in \mathbb{C}^{(k+1) \times k}$ as defined in (5) and (8), respectively.
6. For $m = a + 1, ...,$ form the matrix $\hat{H}_m = \bar{H}_m \bar{H}_{m-1} \ldots \bar{H}_{m-a}$.
7. Compute the QR factorization of $\hat{H}_m$: $\hat{H}_m = Q_m R_m; Q_m \in \mathbb{C}^{(m+1) \times (m-a)}$ and $R_m \in \mathbb{C}^{(m-a) \times (m-a)}$.
8. Solve the (upper triangular) system $R_m \xi_m = \beta(Q_m^* e_1)$, where $e_1 = [1, 0, \ldots, 0]$.
9. Compute $x_m = x_0 + \hat{V}_{m-a} \xi_m$ (then $\|A^a r_m\| = \beta \sqrt{1 - \|Q_m^* e_1\|^2}$).
10. Compute $\|A^a r_m\| = \beta \sqrt{1 - \|Q_m^* e_1\|^2}$.

3 GGMRES algorithm

In this section, we develop a new implementation of the DGMRES algorithm for the case $\mathrm{ind}(A) = 1$.

Let $\bar{H}_{i\bullet}^{(m)}$ and $\bar{H}_{\bullet j}^{(m)}$ represent the row i and the column j of $\bar{H}_m$, respectively. By partitioning $\bar{H}_m$ and $\hat{H}_m = \bar{H}_m \bar{H}_{m-1}$ as

$$\bar{H}_m = \begin{bmatrix} \bar{H}_{1\bullet}^{(m)} \\ \bar{R}_m \end{bmatrix} \qquad \text{and} \qquad \widehat{H}_m = \begin{bmatrix} d_m^T \\ F_m \end{bmatrix}, \tag{11}$$

respectively, where $\bar{R}_m$ is an $m \times m$ upper triangular matrix, F_m is an $m \times (m-1)$ upper Hessenberg matrix, and $d_m^T \in \mathbb{R}^{1 \times (m-1)}$, we see that

$$d_2^T = \bar{H}_{1\bullet}^{(2)} \bar{H}_{\bullet 1}^{(1)}, \quad d_{m+1}^T = (d_m^T | \bar{H}_{1\bullet}^{(m+1)} \bar{H}_{\bullet m}^{(m)}), \quad m = 2, 3, \ldots \tag{12}$$

and

$$F_2 = \bar{R}_2 \bar{H}_{\bullet 1}^{(1)}, \quad F_{m+1} = (\tilde{F}_m | \bar{R}_{m+1} \bar{H}_{\bullet m}^{(m)}), \quad m = 2, 3, \ldots \tag{13}$$

where

$$\tilde{F}_m = \begin{bmatrix} F_m \\ 0 \end{bmatrix}.$$

If m steps of the Arnoldi process have been taken and $\hat{V}_{m+1}$, the Arnoldi basis associated with DGMRES is of full rank, then F_m is of full rank. This result follows from the fact that for the elements $\hat{h}_{i+2,i}$, $i = 1, 3, \ldots, m-1$, of $\hat{H}$, we have

$$\hat{h}_{i+2,i} = \bar{h}_{i+2,i+1}\bar{h}_{i+1,i} \neq 0.$$

In what follows we suppose that F_m is of full rank.

In order to get the solution ξ_m of the least squares problem (9), we can consider the normal equation

$$\hat{H}_m^T \hat{H}_m \xi_m = \beta \hat{H}_m^T e_1. \tag{14}$$

The use of (11) implies that

$$\left(d_m d_m^T + F_m^T F_m\right) \xi_m = \beta \widehat{H}_m^T e_1 = \beta d_m.$$

Let

$$\lambda_m = \beta - d_m^T \xi_m, \tag{15}$$

then, we have

$$F_m^T F_m \xi_m = \lambda_m d_m. \tag{16}$$

By assumption, F_m is of full rank and $\lambda_m = 0$ implies that $\xi_m = 0$, which is not the desired solution. Therefore, λ_m must be nonzero. By defining

$$u_m = \frac{\xi_m}{\lambda_m}, \tag{17}$$

the equation (16) can be written as

$$F_m^T F_m u_m = d_m. \tag{18}$$

For solving this positive definite system, we form the QR factorization

$$Q_m F_m = \begin{bmatrix} R_m \\ o \end{bmatrix},$$

where R_m is an $(m-1) \times (m-1)$ nonsingular upper triangular matrix and $Q \in \mathbb{R}^{m \times m}$ is an unitary matrix. This gives the following equation:

$$R_m^T R_m u_m = d_m. \tag{19}$$

With setting $z_m = R_m u_m$, the vector u_m can be computed by solving the lower and upper triangular systems $R_m^T z_m = d_m$ and $R_m u_m = z_m$, respectively.

From (19), we have $u_m^T d_m = \|R_m u_m\|^2 \geq 0$. So, the relations (15) and (17) imply that

$$\lambda_m = \frac{\beta}{1 + d_m^T u} > 0 \qquad \text{and} \qquad \xi_m = \lambda_m u_m, \tag{20}$$

which can be used for computing λ_m and ξ_m.

We note that F_{m+1} can be obtained as a simple update of F_m by first appending a row of zeros at the bottom of F_m and following that by appending the $(m+1)$-vector $\bar{R}_{m+1}\bar{H}_{:m}^{(m)}$ as the mth column. After forming F_m, for obtaining R_m, we can factorize F_m by appling a series of Givens rotations to the columns of F_m.

Now, we show that the norm $\|Ar_m\|$ can be obtained without actually having to form x_m and r_m. From (9) and (11) there holds

$$\begin{aligned}
\|Ar_m\|_2^2 &= \|\beta e_1 - \hat{H}_m \xi_m\|_2^2 \\
&= \|\beta e_1 - \begin{bmatrix} d_m^T \\ F_m \end{bmatrix} \xi_m\|_2^2 \\
&= \left\| \begin{bmatrix} \beta - d_m^T \xi_m \\ -F_m \xi_m \end{bmatrix} \right\|_2^2 \\
&= \left\| \begin{bmatrix} \lambda_m \\ -\lambda_m F_m u_m \end{bmatrix} \right\|_2^2 \\
&= \lambda_m^2 (1 + \|F_m u_m\|_2^2) \\
&= \lambda_m^2 \left[1 + \left\| Q_m \begin{bmatrix} R_m \\ o \end{bmatrix} u_m \right\|_2^2 \right] \\
&= \lambda_m^2 (1 + \|R_m u_m\|_2^2) \\
&= \lambda_m^2 (1 + \|z_m\|_2^2).
\end{aligned}$$

So, we have

$$\|Ar_m\|_2 = \lambda_m \sqrt{1 + \|z_m\|_2^2}.$$

This relation enables us to obtain the norms $\|Ar_m\|_2$, $m \geq 1$, without actually having to form x_m and r_m. Now, we summarize the steps of the new method, called GGMRES method, for the solution group of a linear system (1) when A is singular and $\text{ind}(A) = 1$, as follows.

Algorithm 2 GGMRES algorithm

1. Pick x_0 and compute $r_0 = b - Ax_0$ and Ar_0.
2. Compute $\beta = \|A^a r_0\|$ and set $v_1 = \beta^{-1}(A^a r_0)$.
3. Orthogonalize the Krylov vectors $A^a r_0, A^{a+1} r_0, \ldots,$ via the Arnoldi-Gram-Schmidtprocess carried out like the modified Gram-Schmidt process:
   For $i = 1, 2, \ldots,$ do
   Compute $h_{ij} = (v_j, Av_i), j = 1, 2, \ldots, i.$
   Compute $\hat{v}_i = Av_i - \sum_{j=1}^{i} v_j h_{ij}.$
   Let $h_{i+1,i} = \|\hat{v}_i\|$ and set $v_{i+1} = \hat{v}_i / h_{i+1,i}.$
4. EndDo
5. For $k = 1, 2, \ldots,$ form the matrices $\hat{V}_k \in \mathbb{C}^{N \times k}$ and $\bar{H}_k \in \mathbb{C}^{(k+1)\times k}$as defined in (5)and (8), respectively.
6. Form the vector d_m and the matrix F_m by using the recursive formula

$$d_2^T = \bar{H}_{1.}^{(2)} \bar{H}_{.1}^{(1)}, \quad d_{k+1}^T = (d_k^T | \bar{H}_{1.}^{(k+1)} \bar{H}_{.k}^{(k)}), \quad k = 2, 3, \ldots, m-1$$

   and

$$F_2 = \bar{R}_2 \bar{H}_{.1}^{(1)}, \quad F_{k+1} = (\tilde{F}_k | \bar{R}_{k+1} \bar{H}_{.k}^{(k)}), \quad k = 2, 3, \ldots, m-1,$$

   where $\tilde{F}_k = \begin{bmatrix} F_k \\ 0 \end{bmatrix}$ and $\bar{R}_k$ is defined in (11).
7. Compute the QR factorization of F_m : $F_m = Q_m R_m$; $Q_m \in \mathbb{R}^{m \times m-1}$ and $R_m \in \mathbb{R}^{(m-1)\times(m-1)}$.
8. Solve $R_m^T z_m = d_m$ and $R_m u_m = z_m$.
9. Compute $\lambda_m = \frac{\beta}{1 + d_m^T u_m}$ and $\xi_m = \lambda_m u_m$.
10. Compute $x_m = x_0 + \hat{V}_{m-1} \xi_m$.
11. Compute $\|Ar_m\|_2 = \lambda_m \sqrt{1 + \|z_m\|_2^2}$.

It is possible to implement the GGMRES algorithm in a progressive manner. The columns of $\bar{H}_m$, F_m, and R_m can be computed step by step and $\|Ar_m\|_2$ can be computed with no additional operations. In addition, as DGMRES algorithm, the computation of the vector x_m requires that the m vectors $v_1, v_2, \ldots, v_{m-1}$ and x_0 (all of dimension n) to be stored.

4 Numerical examples

To compare the behavior of the proposed GGMRES method discussed in the previous section with the DGMRES method, in this section, we present

numerical results for two examples. Our examples, which have a singular coefficient matrix, are derived by the finite difference method for elliptic partial differential equations. The numerical computations are performed in MATLAB (R213a) with double precision. All computations were performed running the code on an Intel (R) Core (TM) i7-2600, 3.40 GHz machine with 8 GB of RAM memory using Windows 7 professional 64-bit operating system. The initial vector x_0 is the zero vector. All the tests were stopped as soon as $\|Ar_m\|_2 \leq 10^{-12}$.

Example 1. We form the linear system $Ax = b$ by discretizing Poisson equation with Neumann boundary conditions:

$$\begin{cases} (\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})u(x,y) = f(x,y), & (x,y) \in \Omega = [0,1] \times [0,1], \\ \frac{\partial}{\partial n}u(x,y) = \varphi(x,y), & x, y \in \partial\Omega. \end{cases}$$

This linear system also has been formed by Sidi [16] for testing DGMRES algorithm. The problem has also been considered by Hank and Hochbruck [10] for testing the Chebyshev-type semi-iterative method.

Let M be an odd integer, we discretize the Poisson equation on a uniform grid of mesh size $h = 1/M$ via central differences, and then by taking the unknowns in the red-black order we obtain the system $Ax = b$, where the $(M+1)^2 \times (M+1)^2$ nonsymmetric matrix A is as follows

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_1 \end{bmatrix}, \tag{21}$$

where $A_1 = 4I$,

$$A_2 = \begin{bmatrix} T_2 & -2I & o & \cdots & \cdots & \cdots & \cdots & o \\ -I & T_1 & -I & o & & & & \vdots \\ o & -I & T_2 & -I & o & & & \vdots \\ \vdots & o & -I & T_1 & -I & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & o \\ \vdots & & & & o & -I & T_2 & -I \\ o & \cdots & \cdots & \cdots & \cdots & o & -2I & T_1 \end{bmatrix},$$

and

$$A_3 = \begin{bmatrix} T_1 & -2I & o & \cdots & \cdots & \cdots & \cdots & o \\ -I & T_2 & -I & o & & & & \vdots \\ o & -I & T_1 & -I & o & & & \vdots \\ \vdots & o & -I & T_2 & -I & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & o \\ \vdots & & & & o & -I & T_1 & -I \\ o & \cdots & \cdots & \cdots & \cdots & o & -2I & T_2 \end{bmatrix}.$$

Here, I and o denote, respectively, the $(M+1)/2 \times (M+1)/2$ identity and zero matrices and the $(M+1)/2 \times (M+1)/2$ matrices T_1 and T_2 are given by

$$T_1 = \begin{bmatrix} -2 & o & \cdots & \cdots & o \\ -1 & -1 & \ddots & & \vdots \\ o & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & & -1 & o \\ o & \cdots & o & -1 & -1 \end{bmatrix}, \quad T_2 = \begin{bmatrix} -1 & -1 & o & \cdots & o \\ o & -1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & o \\ \vdots & & & -1 & -1 \\ o & \cdots & \cdots & o & -2 \end{bmatrix}.$$

The numerical experiment is performed for $M = 31, 63, 127$.

Example 2. As shown in [4], applying 5−point central differences to the partial differential equation

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + d\frac{\partial U}{\partial x} = f(x,y), \quad 0 < x, y < 1,$$

over the unit square $\Omega = (0,1)\times(0,1)$ with the periodic boundary condition:

$$u(x,0) = u(x,1), \quad u(0,y) = u(1,y),$$

yields a singular system with a nonsymmetric coefficient matrix. The mesh size is chosen as $h = 1/m$ for Ω, so that the resulting system has the following $n \times n$ coefficient matrix (where $n = m^2$):

$$A := \frac{1}{h^2} \begin{bmatrix} D_m & I_m & \cdots & & & I_m \\ I_m & D_m & I_m & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & I_m & D_m & I_m \\ I_m & & & & I_m & D_m \end{bmatrix}. \tag{22}$$

Here I_m is the $m \times m$ unit matrix and D_m the $m \times m$ matrix is given by

$$D_m := \begin{bmatrix} -4 & \alpha_+ & & & \alpha_- \\ \alpha_- & -4 & \alpha_+ & & \\ & \ddots & \ddots & \ddots & \\ & & \alpha_- & -4 & \alpha_+ \\ \alpha_+ & & & \alpha_- & -4 \end{bmatrix},$$

where $\alpha_\pm = 1 \pm \frac{dh}{2}$. The numerical experiment is done for $d = 0.1$, $d = 0.3$, $d = 0.5$, and $m = 60$.

For the matrix A of both (21) and (22) the identity $Ae = A^T e = 0$ holds, so that $\text{Null}(A) = \text{Null}(A^T) = \text{Span}\{e\}$, where $e = (1, 1, \ldots, 1)^T$. Furthermore, $\text{ind}(A) = 1$, as mentioned in [10,16]. Even if the continuous problem has a solution, the discretized problem need not to be consistent. Here, we consider only the Group-inverse solution of the system for arbitrary right side b, not necessarily related to f and φ.

As [17], we first construct a consistent system with known solution $\hat{s} \in R(A)$ via $\hat{s} = Ay$, where $y = [0, \ldots, 0, 1]^T$. Then we perturb $A\hat{s}$, the right-hand side of $Ax = A\hat{s} = \hat{b}$, with a constant multiple of the null space vector e and we obtain the right-hand side

$$b = \hat{b} + \delta \frac{e}{\|e\|_2}.$$

Consequently the system $Ax = \hat{b} + \delta \frac{e}{\|e\|_2}$is solved for x. The perturbation parameter δ is selected as 10^{-2} in our experiments.

For these examples, the solution we are looking for is the vector $\hat{s}$, whose components are zeros except

$$\hat{s}_{2\hat{M}^2-\hat{M}} = -1, \ \hat{s}_{2\hat{M}^2-1} = -1, \ \hat{s}_{2\hat{M}^2} = -2, \ \hat{s}_{4\hat{M}^2} = 4,$$

where $\hat{M} = (M+1)/2$ for Example 1 and except

$$\hat{s}_m = 1, \quad \hat{s}_{m^2-m} = 1, \quad \hat{s}_{m^2-m+1} = \alpha_-, \quad \hat{s}_{m^2-1} = \alpha_+, \quad \hat{s}_{m^2} = 4,$$

for Example 2.

In Tables 1-4, we give the number of iterations (Its), the CPU time (Time) required for convergence, and the error (Error) for the DGMRES and GGMRES methods. As shown in Tables 1-4 the GGMRES algorithm is effective and less expensive than the DGMRES algorithm.

Table 1: Application of GGMRES implementation to the consistent singular system for Example 1.

Size of A	1024×1024			4096×4096			16384×16384		
Method	Its	Time	Error	Its	Time	Error	Its	Time	Error
DGMRES	164	0.33	$8.87e-013$	310	4.20	$9.92e-13$	471	29.46	$9.00e-13$
GGMRES	164	0.24	$8.67e-013$	310	3.23	$9.88e-13$	471	24.34	$9.83e-13$

Table 2: Application of GGMRES implementation to the inconsistent singular system for Example 1.

Size of A	1024×1024			4096×4096			16384×16384		
Method	Its	Time	Error	Its	Time	Error	Its	Time	Error
DGMRES	164	0.38	$8.74e-013$	310	4.04	$9.90e-13$	471	29.48	$9.96e-13$
GGMRES	164	0.31	$8.67e-013$	310	3.35	$9.88e-13$	471	22.99	$9.83e-13$

Table 3: Application of GGMRES implementation to the consistent singular system for Example 2 with $m = 60$(n=3600).

A	0.1			0.3			0.5		
Method	Its	Time	Error	Its	Time	Error	Its	Time	Error
DGMRES	217	1.74	$9.97e-13$	240	2.11	$9.70e-13$	246	2.27	$9.35e-13$
GGMRES	217	1.38	$9.93e-13$	240	1.60	$9.68e-13$	246	1.64	$9.30e-13$

Table 4: Application of GGMRES implementation to the inconsistent singular systemfor Example 2 with $m = 60$(n=3600).

A	0.1			0.3			0.5		
Method	Its	Time	Error	Its	Time	Error	Its	Time	Error
DGMRES	217	1.80	$9.99e-13$	240	2.09	$9.76e-13$	246	2.29	$9.96e-13$
GGMRES	217	1.45	$9.93e-13$	240	1.68	$9.68e-13$	246	1.64	$9.57e-13$

5 Conclusion

In this paper, we have presented a new method, called GGMRES, for computing the group-inverse solution of singular linear equations with index one. This method is based on DGMRES algorithm. Numerical experiments show that the group-inverse solution obtained by this method is reasonably accurate, and its computation time is less than that of solution obtained by the DGMRES method. So, we can conclude that the GGMRES algorithm is a robust and efficient tool for computing the group-inverse solution of singular linear equations with index one.

Acknowledgements

We would like to thank the editor and anonymous referees for their carefully reading and useful comments. We are also grateful to editor in chief of the journal for his comments that improved our results.

References

[1] W.E. Arnoldi, *The principle of minimized iterations in the solution of the matrix eigenvalue problem,* Quart. Appl. Math. **9** (1951) 17–29.

[2] A. Ben-Israel and T.N.E. Grevile, *Generalized Inverses: Theory and Applications*, second ed., Springer-Verlag, New York, 2003.

[3] A. Berman and R.J. Plemmons, *Nonnegative Matrices in the Mathematical Science,* Academic Press, New York, 1979.

[4] P.N. Brown and H.F. Walker, *GMRES on (nearly) singular systems,* SIAM J. Matrix Anal. Appl. **18** (1997) 37–51.

[5] S.L. Campell and C.D. Meyer, *Generalized Inverses of Linear Transformations*, Pitman, London, 1979, Dover, New York, 1991.

[6] M. Eiermann, I. Marek and W. Niethammer, *On the solution of singular linear systems of algebric equations by semiiterative methods*, Numer. Math. **53** (1988) 265–283.

[7] R.W. Freund and M. Hochbruck, *On the use of two QMR algorithms for solving singular systems and applications in Markov chain modeling,* Numer. Linear algebra Appl. **1** (1994) 403–420.

[8] R.E. Hartwig and J. Levine, *Applications of the Drazin inverse to the Hill cryptographic system, Part III*, Cryptologia **5** (1981) 67–77.

[9] R.E. Hartwig and F. Hall, *Applications of the Drazin inverse to Cesaro-Neumann iterations, in: S.L. Campbell (Ed.), Recent Applications of Generalized Inverses*, Pitman, London, Program **66** (1982) 145–195.

[10] M. Hank and M. Hochbruck, *A Chebyshev–like semiiteration for inconsistent linear systems,* Electron. Trans. Numer. Anal. **1** (1993) 315–339.

[11] I. Marrek and D.B. Szyld, *Iterative and semi–iterative methods for computing stationary probablity vectors of Markov operators*, Math. Comput. **61** (1993) 719–731.

[12] Y. Saad and M. Schultz, *A generalized minimal reseidual algorithm for solving non–symetric linear systems*, SIAM J. Sci. Statist. Comput. **7** (1986) 56–869.

[13] B. Simeon, C. Fuhrer and P. Rentrop, *The Drazin inverse in multibody system dynamics*, Numer. Math. **64** (1993) 521–539.

[14] A. Sidi, *A unified approach to krylov subspace methods for the Drazin–inverse solution of singular non-symmetric linear systems*, Linear Algebra Appl. **298** (1999) 99–113.

[15] A. Sidi and V. Kluzner, *A Bi–CG type iterative method for Drazin inverse solution of singular inconsistent non-symmetric linear systems of arbitrary index*, Electron. J. Linear Algebra Appl. **6** (1999) 72–94.

[16] A. Sidi, *DGMRES: A GMRES–type algorithm for Drazin–inverse solution of singular non–symmetric linear systems*, Linear Algebra Appl. **335** (2001) 189–204.

[17] J. Zhou and Y. Wei, *DFOM algorithm and error analysis for projection methods for solving singular linear system*, Appl. Math. Comput. **157** (2004) 313–329.