ISSN 1346-5597

National Institute of Informatics

NII Technical Report

A Stabilized GMRES Method for Solving Underdetermined Least Squares Problems

Zeyu LIAO, Ken HAYAMI, Keiichi MORIKUNI, and

Jun-Feng YIN

NII-2020-001E
Mar. 2020

URL: http://www.nii.ac.jp/ Tokyo 101-8430, JAPAN +81-3-4212-2000 FAX: +81-3-3556-1916

A Stabilized GMRES Method for Solving Underdetermined Least Squares Problems

Zeyu LIAO*[1] | Ken HAYAMI[1,2] | Keiichi MORIKUNI[3] | Jun-Feng YIN[4]

[1]Department of Informatics, School of Multidisciplinary Sciences, The Graduate University for Advanced Studies (SOKENDAI), Tokyo, Japan
[2]National Institute of Informatics, Tokyo, Japan
[3]Faculty of Engineering, Information and Systems, University of Tsukuba, Ibaraki, Japan
[4]School of Mathematical Science, Tongji University, Shanghai, China

Correspondence
*Zeyu LIAO, Department of Informatics, School of Multidisciplinary Sciences, The Graduate University for Advanced Studies (SOKENDAI), 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan, Email: zeyu@nii.ac.jp

Summary

Consider using the right-preconditioned generalized minimal residual (AB-GMRES) method, which is an efficient method for solving underdetermined least squares problems. Morikuni (Ph.D. thesis, 2013) showed that for some inconsistent and ill-conditioned problems, the iterates of the AB-GMRES method may diverge. This is mainly because the Hessenberg matrix in the GMRES method becomes very ill-conditioned so that the backward substitution of the resulting triangular system becomes numerically unstable. We propose a stabilized GMRES based on solving the normal equations corresponding to the above triangular system using the standard Cholesky decomposition. This has the effect of shifting upwards the tiny singular values of the Hessenberg matrix which lead to an inaccurate solution. Thus, the process becomes numerically stable and the system becomes consistent, rendering better convergence and a more accurate solution. Numerical experiments show that the proposed method is robust and efficient for solving inconsistent and ill-conditioned underdetermined least squares problems. The method can be considered as a way of making the GMRES stable for highly ill-conditioned inconsistent problems.

KEYWORDS:
least squares problems, Krylov subspace methods, GMRES, underdetermined systems, inconsistent systems, regularization

1 | INTRODUCTION

Consider solving the inconsistent underdetermined least squares problem

$$\min_{x\in\mathbb{R}^n} \|b - Ax\|_2, \qquad A \in \mathbb{R}^{m\times n}, \qquad b \in \mathbb{R}^m, \qquad b \notin \mathcal{R}(A), \qquad m < n, \tag{1}$$

where A is ill-conditioned and may be rank-deficient. Here, $\mathcal{R}(A)$ denotes the range space of A. Such problems may occur in ill-posed problems where b is given by an observation which contains noise. The least squares problem (1) is equivalent to the normal equations

$$A^\mathsf{T} Ax = A^\mathsf{T} b. \tag{2}$$

The standard direct method for solving the least squares problem (1) is to use the QR decomposition. However, when A is large and sparse, iterative methods become necessary. The CGLS[1] and LSQR[2] are mathmetically equivalent to applying the conjugate gradient (CG) method to (2). The convergence of these methods deteriorates for ill-conditioned problems and they require reorthogonalization[3] to improve the convergence. Here, we say (1) is ill-conditioned if the condition number $\kappa_2(A) = \|A\|_2\|A^\dagger\|_2 \gg 1$, where $A^\dagger$ is the pseudoinverse of A. The LSMR[4] applies MINRES[5] to (2).

Hayami et al.[3] proposed preconditioning the $m \times n$ rectangular matrix A of the least squares problem by an $n \times m$ rectangular matrix B from the right and the left, and using the generalized minimal residual (GMRES) method[6] for solving the preconditioned least squares problems (AB-GMRES and BA-GMRES methods, respectively). For ill-conditioned problems, the AB-GMRES and BA-GMRES were shown to be more robust compared to the preconditioned CGNE and CGLS, respectively. Note here that the BA-GMRES works with Krylov subspaces in n-dimensional space, whereas the AB-GMRES works with Krylov subspaces in m-dimensional space. Since $m < n$ in the underdetermined case, the AB-GMRES works in a smaller dimensional space than the BA-GMRES and should be more computationally efficient compared to the BA-GMRES for each iteration. Moreover, the AB-GMRES has the advantage that the weight of the norm in (1) does not change for arbitrary B. Thus, we mainly focus on using the AB-GMRES to solve the underdetermined least squares problem (1). Morikuni[7] showed that the AB-GMRES may fail to converge to a least squares solution in finite-precision arithmetic for inconsistent problems. We will review this phenomenon. The GMRES applied to inconsistent problems was also studied in other papers[8–12].

In this paper, we first analyze the deterioration of convergence of the AB-GMRES. To overcome the deterioration, we use the normal equations of the upper triangular matrix arising in the AB-GMRES to change the inconsistent subproblem to a consistent one. In finite precision arithmetic, forming the normal equations for the subproblem will not square its condition number as would be predicted by theory. In the ill-conditioned case, the tiny singular values are shifted upwards due to rounding errors. In finite precision arithmetic, applying the standard Cholesky decomposition to the normal equations will result in a well-conditioned lower triangular matrix, which will ensure that the forward and backward substitutions work stably, and overcome the problem. Numerical experiments on a series of ill-conditioned Maragal matrices[13] show that the proposed method converges to a more accurate approximation than the original AB-GMRES. The method can also be used to solve general inconsistent singular systems.

The rest of the paper is organized as follows. In Section 2, we briefly review the AB-GMRES and a related theorem. In Section 3, we demonstrate and analyze the deterioration of the convergence. In Section 4, we propose and present a stabilized GMRES method and explain a regularization effect of the method based on the normal equations for ill-conditioned problems. In Section 5, numerical results for the underdetermined case and the square case are presented. In Section 6, we conclude the paper.

All the experiments in this paper were done using MATLAB R2017b in double precision, unless specified otherwise (where we extended the arithmetic precision by using the Multiprecision Computing Toolbox for MATLAB[14]), and the computer uesd was Alienware 15 CAAAW15404JP with CPU Inter(R) Core(TM) i7-7820HK (2.90GHz).

2 | DETERIORATION OF CONVERGENCE OF AB-GMRES FOR INCONSISTENT PROBLEMS

In this section, we review previous results. First, we introduce the right-preconditioned GMRES (AB-GMRES), which is the basic algorithm in this paper. Then, we show the phenomenon that the convergence of the AB-GMRES deteriorates for inconsistent problems. Finally, we cite a related theorem to analyze the deterioration.

2.1 | AB-GMRES method

AB-GMRES for least squares problems applies GMRES to $\min_{u \in \mathbb{R}^m} \|b - ABu\|_2$ with $x = Bu$, where $B \in \mathbb{R}^{n \times m}$. Let x_0 be the initial solution (in all our numerical experiments, we set $x_0 = 0$), and $r_0 = b - Ax_0$. Then, AB-GMRES searches for u in the Krylov subspace $\mathcal{K}_i(AB, r_0) = \operatorname{span}\{r_0, ABr_0, \dots, (AB)^{i-1} r_0\}$. The algorithm is given in Algorithm 1[3]. Here, $H_{i+1,i} = (h_{pq}) \in \mathbb{R}^{(i+1) \times i}$ and $e_1 = (1, 0, \dots, 0)^\mathsf{T} \in \mathbb{R}^{i+1}$.

To find $y_i \in \mathbb{R}^i$ that minimizes $\|r_i\|_2 = \| \|r_0\|_2 e_i - H_{i+1,i} y_i \|_2$ in Algorithm 1, the standard approach computes the QR decomposition of $H_{i+1,i}$

$$H_{i+1,i} = Q_{i+1} R_{i+1,i}, \qquad Q_{i+1} \in \mathbb{R}^{(i+1)\times(i+1)}, \qquad R_{i+1,i} = \begin{pmatrix} R_i \\ 0^\mathsf{T} \end{pmatrix} \in \mathbb{R}^{(i+1)\times i}, \qquad R_i \in \mathbb{R}^{i \times i}, \tag{3}$$

Algorithm 1 AB-GMRES

```
Choose x_0 ∈ ℝ^n,   r_0 = b − A x_0,   v_1 = r_0/‖r_0‖_2
for i = 1, 2, …, k do
    w_i = A B v_i
    for j = 1, 2, …, i do
        h_{i,j} = w_i^T v_j,   w_i = w_i − h_{j,i} v_j
    end for
    h_{i+1,i} = ‖w_i‖_2,   v_{i+1} = w_i / h_{i+1,i}
    Compute y_i ∈ ℝ^i which minimizes ‖r_i‖_2 = ‖ ‖r_0‖_2 e_1 − H_{i+1,i} y_i ‖_2
    x_i = x_0 + B[v_1, v_2, …, v_i] y_i,   r_i = b − A x_i
    if ‖A^T r_i‖_2 < ε‖A^T r_0‖_2 then
        stop
    end if
end for
```

where Q_{i+1} is an orthogonal matrix and R_i is an upper triangular matrix. Then, backward substitution is used to solve a system with the coefficient matrix R_i as follows

$$\|r_i\|_2 = \min_{y_i \in \mathbb{R}^i} \|Q_{i+1}^{\mathsf{T}} \beta e_1 - R_{i+1,i} y_i\|_2, \tag{4}$$

where

$$\beta = \|r_0\|_2, \qquad Q_{i+1}^{\mathsf{T}} \beta e_1 = \begin{pmatrix} t_i \\ \rho_{i+1} \end{pmatrix}, \qquad t_i \in \mathbb{R}^i, \qquad \rho_{i+1} \in \mathbb{R}, \qquad y_i = R_i^{-1} t_i, \tag{5}$$

$$x_i = V_i y_i = V_i (R_i^{-1} t_i), \qquad V_i = [v_1, v_2, \dots, v_i] \in \mathbb{R}^{n \times i}, \qquad V_i^{\mathsf{T}} V_i = I, \tag{6}$$

where I is the identity matrix.

Note the following theorem.

Theorem 1. (Corollary 3.8 of Hayami et al.[3]) If $\mathcal{R}(A) = \mathcal{R}(B^{\mathsf{T}})$ and $\mathcal{R}(A^{\mathsf{T}}) = \mathcal{R}(B)$, then AB-GMRES determines a least squares solution of $\min_{x \in \mathbb{R}^n} \|b - Ax\|_2$ for all $b \in \mathbb{R}^m$ and for all $x_0 \in \mathbb{R}^n$ without breakdown.

Here, breakdown means $h_{i+1,i} = 0$ in Algorithm 1. See Appendix B of[11].

In fact, if $\mathcal{R}(A^{\mathsf{T}}) = \mathcal{R}(B)$ and $x_0 \in \mathcal{R}(A^{\mathsf{T}})$, the solution is a minimum-norm solution since $x = Bu \in \mathcal{R}(A^{\mathsf{T}}) = \mathcal{N}(\mathrm{A})^{\perp}$, where $\mathcal{N}(A)$ is the null space of A.

From now on, we use AB-GMRES to solve (1) with $B = A^{\mathsf{T}}$ and $x = Bu$, which means using the Krylov subspace $\mathcal{K}_i(AA^{\mathsf{T}}, r_0) = \langle r_0, AA^{\mathsf{T}} r_0, \dots, (AA^{\mathsf{T}})^{i-1} r_0 \rangle$ to approximate u. Hence, Theorem 1 guarantees the convergence in exact arithmetic even in the inconsistent case. However, in finite precision arithmetic, AB-GMRES may fail to converge to a least squares solution for inconsistent problems, as shown later.

2.2 | AB-GMRES for inconsistent problems

In this section, we perform experiments to show that the convergence of AB-GMRES deteriorates for inconsistent problems. Experiments were done on the transpose of the matrix Maragal_3[13], denoted by Maragal_3T etc. Table 1 gives the information on the Maragal matrices, including the density of nonzero entries, rank and condition number. Here, the rank and condition number were determined by using the MATLAB functions `spnrank`[15] and `svd`, respectively.

Figure 1 shows the relative residual norm $\|A^{\mathsf{T}} r_i\|_2 / \|A^{\mathsf{T}} b\|_2$ and $\kappa_2(R_i)$ versus the number of iterations for AB-GMRES with $B = A^{\mathsf{T}}$ for Maragal_3T, where $r_i = b - Ax_i$, and the vector b was generated by the MATLAB function `rand` which returns a vector whose entries are uniformly distributed in the interval $(0, 1)$. Here $\kappa_2(R_i) = \kappa_2(H_{i+1,i})$ holds from (3). The value of $\kappa_2(R_i)$ was computed by the MATLAB function `cond`. The relative residual norm $\|A^{\mathsf{T}} r_i\|_2 / \|A^{\mathsf{T}} b\|_2$ decreased to 10^{-8} until the 525th iteration, and then increased sharply. The value of `cond`(R_i) started to increase rapidly around iterations 450–550. This observation shows that R_i becomes ill-conditioned before convergence. Thus, AB-GMRES failed to converge to a least squares solution. This phenomnenon was observed by Morikuni[7].

TABLE 1 Information on the Maragal matrices.

matrix	m	n	density[%]	rank	$\kappa_2(A)$
Maragal_3T	858	1682	1.27	613	1.10×10^3
Maragal_4T	1027	1964	1.32	801	9.33×10^6
Maragal_5T	3296	4654	0.61	2147	1.19×10^5
Maragal_6T	10144	21251	0.25	8331	2.91×10^6
Maragal_7T	26525	46845	0.10	20843	8.91×10^6

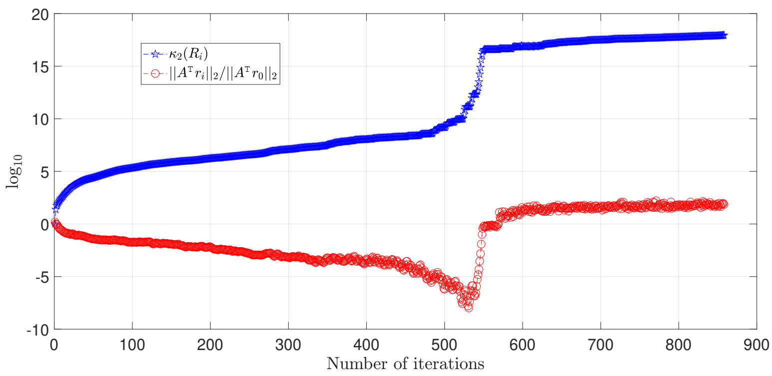

FIGURE 1 $\kappa_2(R_i)$ and relative residual norm versus the number of iterations for Maragal_3T.

The reason why R_i becomes ill-conditioned before convergence in the inconsistent case will be explained by a theorem in the next subsection.

2.3 | GMRES for inconsistent problems

Brown and Walker[8] introduced an effective condition number to explain why GMRES fails to converge for inconsistent least squares problems

$$\min_{x\in\mathbb{R}^m} \|b - \widetilde{A}x\|_2, \tag{7}$$

where $\widetilde{A} \in \mathbb{R}^{m\times m}$ is singular, in the following Theorem 2.

Let $b|_{\mathcal{R}(\widetilde{A})}$ denote the orthogonal projection of b onto $\mathcal{R}(\widetilde{A})$. Assume $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\mathsf{T})$ and grade$(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$. Here, grade$(\widetilde{A}, \widetilde{b})$ for $\widetilde{A} \in \mathbb{R}^{m\times m}$, $\widetilde{b} \in \mathbb{R}^m$ is defined as the minimum k such that $\mathcal{K}_{k+1}(\widetilde{A}, \widetilde{b}) = \mathcal{K}_k(\widetilde{A}, \widetilde{b})$. Then, $\dim(\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})) = \dim(\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})) = \dim(\widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})) = \dim(\widetilde{A}\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})) = k$ (See Appendix A). Since $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\mathsf{T})$, we obtain $\widetilde{A}b|_{\mathcal{R}(\widetilde{A})} = \widetilde{A}b$ and $\dim(\widetilde{A}\mathcal{K}_{k+1}(\widetilde{A}, b)) = \dim(\widetilde{A}\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})) = k$. If $b \notin \mathcal{R}(\widetilde{A})$ and $\dim(\widetilde{A}\mathcal{K}_k(\widetilde{A}, b)) = k$, $\dim(\mathcal{K}_{k+1}(\widetilde{A}, b)) = k + 1$ (See Appendix B).

Let x_0 be the initial solution and $r_0 = b - \widetilde{A}x_0$. In the inconsistent case, a least squares solution is obtained at iteration k, and at iteration $k+1$ breakdown occurs because of $\dim(\widetilde{A}\mathcal{K}_{k+1}(\widetilde{A}, r_0)) < \dim(\mathcal{K}_{k+1}(\widetilde{A}, r_0))$, i.e. rank deficiency of $\min_{z\in\mathcal{K}_{k+1}(\widetilde{A}, r_0)} \|b - \widetilde{A}(x_0 + z)\|_2 = \min_{z\in\mathcal{K}_{k+1}(\widetilde{A}, r_0)} \|r_0 - \widetilde{A}z\|_2$[8]. This case is also called the hard breakdown[10].

However, even if $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\mathsf{T})$, when (7) is inconsistent, the least squares problem $\min_{z\in\mathcal{K}_i(\widetilde{A}, r_0)} \|r_0 - \widetilde{A}z\|_2$ may become ill-conditioned as shown below.

Theorem 2. [8] Assume $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\mathsf{T})$, and denote the least squares residual of *(7)* by r^*, the residual at the $(i-1)$st iteration by r_{i-1}. If $r_{i-1} \neq r^*$, then

$$\kappa_2(A_i) \geq \frac{\|A_i\|_2}{\|\bar{A}_i\|_2} \frac{\|r_{i-1}\|_2}{\sqrt{\|r_{i-1}\|_2^2 - \|r^*\|_2^2}}, \tag{8}$$

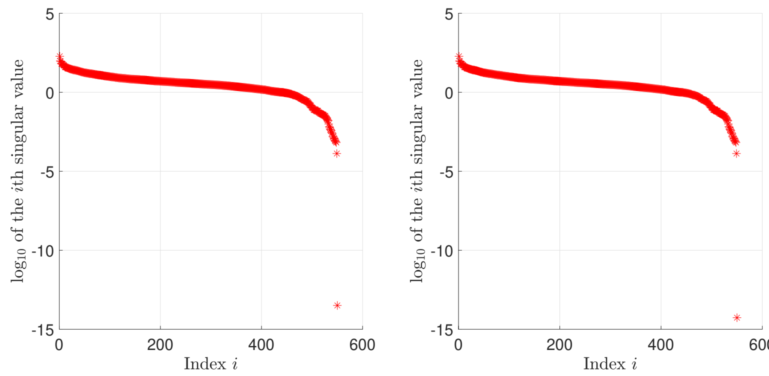

FIGURE 2 Singular value distribution of R_{550} for Maragal_3T in double and quadruple precision arithmetic.

where $A_i \equiv \widetilde{A}|_{\mathcal{K}_i(A,r_0)}$and $\bar{A}_i \equiv \widetilde{A}|_{\mathcal{K}_i(A,r_0)+\mathrm{span}\{r^*\}}$. Here, $\widetilde{A}|_S$ is the restriction of $\widetilde{A}$ to a subspace $S \subseteq \mathbb{R}^m$.

Theorem 2 implies that GMRES suffers ill-conditioning for $b \notin \mathcal{R}(\widetilde{A})$ as $\|r_i\|$ approaches $\|r^*\|$. We can apply Theorem 2 to AB-GMRES for least-squares problems by setting $\widetilde{A} \equiv AA^\mathsf{T}$. Theorem 2 also implies that even if we choose B as A^T, which satisfies the conditions in Theorem 1, AB-GMRES still may not converge numerically because of the ill-conditioning of R_i, losing accuracy in the solution computed in finite-precision arithmetic when r_{i-1} approaches r^*.

3 | ANALYSIS OF THE DETERIORATION OF CONVERGENCE

In this section, we illustrate, the deterioration of convergence of GMRES through numerical experiments. There are two points to note in this section. The first point is that the condition number of R_i tends to become very large as the iteration proceeds for inconsistent problems. Due to $H_{i+1,i} = Q_{i+1}R_{i+1,i}$, the condition number of $H_{i+1,i}$ is the same as that of R_i, and will also become very large. The second point is as follows. Since $y_i = R_i^{-1}t_i$, y_i is obtained by applying backward substitution to the triangular system $R_iy_i = t_i$. When the triangular system becomes ill-conditioned, backward substitution becomes numerically unstable, and fails to give an accurate solution y_i.

Figure 1 shows that at step 550 the relative residual norm suddenly increases. To understand this increase, observe the singular values of R_{550}.

The left of Figure 2 shows the singular values of R_{550} which were computed in double precision arithmetic. The smallest singular value of R_{550} is 3.21×10^{-14}, which means that the triangular matrix R_{550} is very ill-conditioned and nearly singular in double precision arithmetic.

The right of Figure 2 shows the singular values of R_{550} which were computed in quadruple precision arithmetic using the Multiprecision Computing Toolbox for MATLAB [14]. The smallest singular value of R_{550} is 5.39×10^{-15}. Since quadruple precision is more accurate, from now on, we mainly show singular value distributions computed in quadruple precision.

Figure 3 shows $\kappa_2(R_i)$, $\|y_i\|_2$, and the relative residual norm $\|t_i - R_iy_i\|_2/\|t_i\|_2$ versus the number of iterations for AB-GMRES. The relative residual norm increases only gradually when the condition number of R_i is less than 10^8. When the condition number of R_i becomes larger than 10^{10}, the relative residual norm starts to increase sharply. This observation shows that when the condition number of R_i becomes very large, the backward substitution will fail to give an accurate y_i. As a result, we would not get an accurate x_i, and the convergence of AB-GMRES would deteriorate.

4 | STABILIZED GMRES METHOD

In this section, we first propose and present a stabilized GMRES method. Then, we explain its regularization effect comparing it with other regularization techniques.

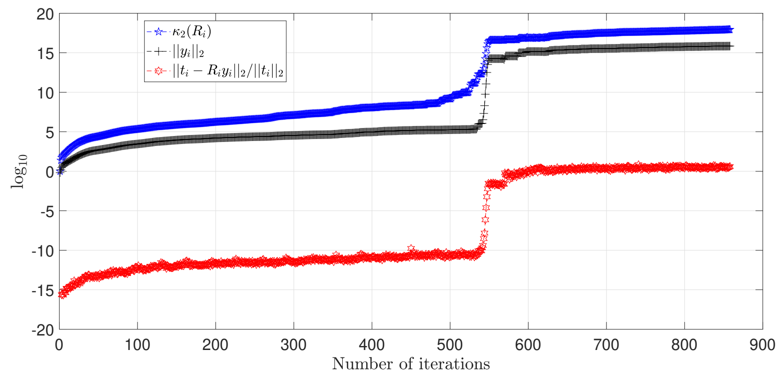

FIGURE 3 $\kappa_2(R_i)$, $\|y_i\|_2$, and $\|t_i - R_i y_i\|_2/\|t_i\|_2$ versus the number of iterations for Maragal_3T.

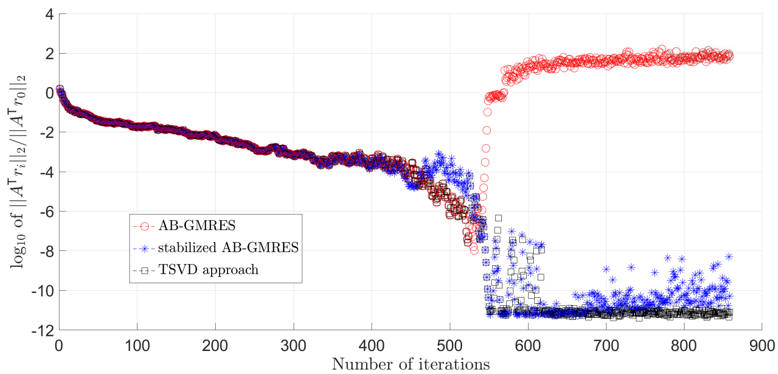

FIGURE 4 Comparison of the standard AB-GMRES with stabilized and TSVD stabilized AB-GMRES with $\mu = 10^{-8}$ AB-GMRES for Maragal_3T.

4.1 | The stabilized GMRES

In order to overcome the deterioration of convergence of GMRES for inconsistent systems, we propose solving the normal equations

$$R_i^\top R_i y_i = R_i^\top t_i \tag{9}$$

instead of $R_i y_i = t_i$, which we will call the stabilized GMRES. This makes the system consistent, and stabilizes the process, as will be shown in the following.

One may also consider using the normal equations of $H_{i+1,i}$. However, before breakdown, we use AB-GMRES, which means we do not have to store $H_{i+1,i}$. We only store R_i and update it in each iteration, which is cheaper.

Figure 4 shows the relative residual norm $\|A^\top r\|_2/\|A^\top r_0\|_2$ versus the number of iterations for the standard AB-GMRES and stabilized AB-GMRES with $B = A^\top$ for Maragal_3T. The stabilized method reaches the relative residual norm level of 10^{-11} which improves a lot compared to the standard method. The method which we used for solving the normal equations (9) is the standard Cholesky decomposition. We replace line 8 of Algorithm 1 by Algorithm 2.

We first checked that the method works for the standard Cholesky decomposition coded by ourselves. Later we applied the backslash function of Matlab to (9) to speed up. We checked that in the backslash, the Cholesky decomposition method `chol` is used until the GMRES residual norm stagnates at a small level as seen in Figure 4. In order to continue with further GMRES iterations, the `chol` is automatically switched to the `ldl`, which works even for singular systems.

In spite of the above mentioned merits of stabilization, solving the normal equations in AB-GMRES is expensive. Actually, we only need the stabilized AB-GMRES when R_i becomes ill-conditioned. Thus, we can speed up the process by switching AB-GMRES to stabilized AB-GMRES only when R_i becomes ill-conditioned. The condition number of an incrementaly enlarging

Algorithm 2 Normal equations stabilization approach

1: Compute the QR decomposition of $H_{i+1,i} = Q_{k+1} R_{i+1,i}$.
2: $R_{i+1,i} = \begin{pmatrix} R_i \\ 0^\top \end{pmatrix}, \quad Q_{i+1}^\top \beta e_1 = \begin{pmatrix} t_i \\ \rho_{i+1} \end{pmatrix}, \quad \widetilde{R_i} = R_i^\top R_i, \quad \widetilde{t_i} = R_i^\top t_i.$
3: Compute the Cholesky decomposition of $\widetilde{R_i} = LL^\top$.
4: Solve $Lz_i = \widetilde{t_i}$ by forward substitution.
5: Solve $L^\top y_i = z_i$ by backward substitution.

TABLE 2 Comparison regarding the smallest attainable relative residual norm $\|A^\top r_i\|_2 / \|A^\top r_0\|_2$.

matrix	Maragal_3T	Maragal_4T	Maragal_5T	Maragal_6T	Maragal_7T
iter.	531	465	1110	2440	1864
standard AB-GMRES	1.05×10^{-8}	2.09×10^{-7}	5.35×10^{-6}	8.26×10^{-6}	4.53×10^{-6}
iter.	552	598	1226	3002	2459
stabilized AB-GMRES	5.99×10^{-12}	5.59×10^{-8}	4.22×10^{-6}	3.88×10^{-6}	2.80×10^{-7}

triangular matrix can be estimated by techniques in [16]. In this paper, we adopt the switching strategy by monitoring the relative residual norm $\|A^\top r_i\|_2/\|A^\top r_0\|_2$. Let ATR($i$)=$\|A^\top r_i\|_2/\|A^\top r_0\|_2$ for the ith iteration. When ATR(υ)/ $\min_{i=1,2,\dots,\upsilon-1}$ATR($i$)>10, we judge that a jump in relative residual norm has occured, and we switch AB-GMRES to stabilized AB-GMRES at the υth iteration.

Motivated by the stabilized AB-GMRES, we also applied the truncated singular value decomposition (TSVD) stabilization method and compared it with the stabilized AB-GMRES. The method modifies R_i by truncating singular values smaller than μ. More specifically, let $R_i = U\Sigma V^\top$ be the SVD of R_i, where the columns of $U = [u_1, u_2, \dots, u_i]$ and $V = [v_1, v_2, \dots, v_i]$ are the left and right singular vectors, respectively, and the diagonal entries of $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \dots, \sigma_i)$ are the singular values of R_i in discending order $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_i$. Then, the TSVD approximates $R_i \simeq \sum_{j=1}^{k} \sigma_j u_j v_j^\top$ with k such that $\sigma_{k+1} \leq \mu\sigma_1 \leq \sigma_k$ and $y_i = R_i^{-1} t_i \simeq \sum_{j=1}^{k} \frac{1}{\sigma_j} v_j u_j^\top t_i$.

When $\mu = 10^{-13}, 10^{-12}, \dots, 10^{-4}$, the method converges but when μ is smaller than 10^{-13} or larger than 10^{-4}, it diverges and is similar to the original AB-GMRES. Numerical experiments showed that $\mu = \sqrt{\epsilon} \simeq 10^{-8}$, where ϵ is the machine epsilon (about 10^{-16} in double presion arithmetic), gave the best result among $\mu = 10^{-1}, 10^{-2}, \dots, 10^{-16}$ in terms of the relavtive residual as shown in Figure 4 for the problem Maragal_3T. The convergence behaviour of the TSVD stabilization method is similar to the stabilized AB-GMRES method, which suggests that eliminating tiny singular values which are less than 10^{-8} is effctive for sovling problem (1). However, the TSVD method requires computing the truncated singular value decomposition of R_i, and requires choosing the value of the threshold parameter μ, whereas the stabilized AB-GMRES does not require either of them.

Table 2 gives more results for the Maragal matrices. The table shows that the stabilized AB-GMRES is more accurate than the standard AB-GMRES. This seems paradoxical, since forming the normal equations whose coefficient matrix $R_i^\top R_i$ would square the condition number compared to R_i, which would make the ill-conditioned problem even worse. Why can the stabilized AB-GMRES give a more accurate solution? We will explain why the stabilized AB-GMRES works in the next subsection.

4.2 | Why the stabilized GMRES method works

Consider solving $R_i y_i = t_i, R_i \in \mathbb{R}^{i\times i}, t_i \in \mathbb{R}^i$ by solving the normal equations (9), which, in theory, squares the condition number and makes the problem become harder to solve numerically. However, in finite precision arithmetic, the condition number of the normal equations is not neccessarily squared. We will continue to illustrate the phenomenon by using the example in Section 3.

We used the MATLAB function `svd` in quadruple precision arithmetic [14] to calculate the singular values. The smallest singular value of R_{550} is 5.39×10^{-15}, so its square is 2.91×10^{-29}.

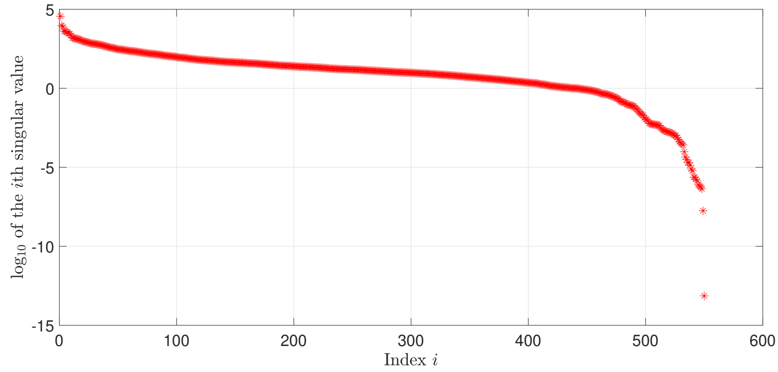

FIGURE 5 Singular values $\sigma_i(\mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550}))$, $i = 1, 2, \ldots, 550$ in quadruple precision arithmetic.

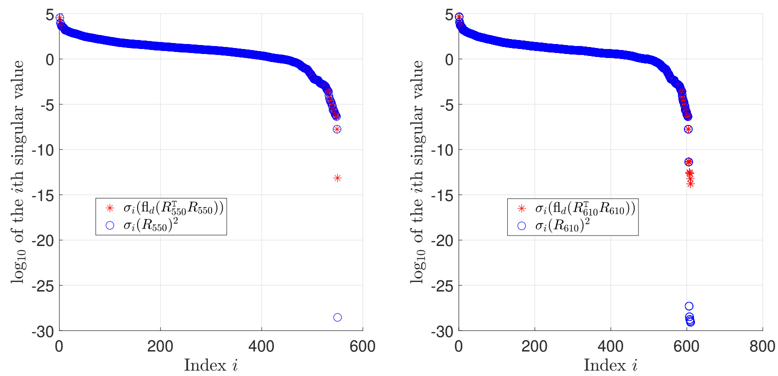

FIGURE 6 Singular values $\sigma_i(\mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550}))$, $\sigma_i(R_{550})^2$, $\sigma_i(\mathrm{fl}_d(R_{610}^{\mathsf{T}} R_{610}))$, and $\sigma_i(R_{610})^2$ in quadruple precision arithmetic.

Let $\mathrm{fl}(\cdot)$ denote the evaluation of an expression in floating point arithmetic and $\mathrm{fl}_d(\cdot)$ and $\mathrm{fl}_q(\cdot)$ denote the result in double precision arithmetic and quadruple precision arithimetic, respectively. Figure 5 shows that, numerically, the smallest singular value of $\mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550})$ is 7.21×10^{-14}, which is much larger than 2.91×10^{-29}. Further, the Cholesky factor L of $\mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550}) = LL^{\mathsf{T}}$ computed in double precision precision arithmetic has the smallest singular value 3.50×10^{-7}, which is also larger than $\sqrt{2.91 \times 10^{-29}} = 5.39\times10^{-15}$. Thus, the triangular systems $Lz_i = \widetilde{t_i}$ and $L^{\mathsf{T}} y_i = z_i$ are better-conditioned than $R_i y_i = t_i$, which will ensure the stability of the forward and backward substitutions and succeeds in obtaining a much more accurate solution than the standard approach.

The left of Figure 6 compares the singular values $\sigma_i(\mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550}))$ and $\sigma_i(R_{550})^2, i = 1, 2, \ldots, 550$. The first to the 549th singular values of $\mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550})$ and the corresponding $\sigma(R_{550})^2$ are almost the same, while the last one is different. What will happen when R_i contains a cluster of small singular values?

The upper triangular matrix R_{610} contains a cluster of small singular values. The right of Figure 6 compares the singular values $\sigma_i(\mathrm{fl}_q(R_{610}^{\mathsf{T}} R_{610}))$ and $\sigma_i(R_{610})^2$. The larger singular values are the same as the 'exact' values, while the smaller singular values become larger than the 'exact' ones.

Experiment results show that finite precision arithmetic has the effect of shifting the tiny singular value upwards. That is the reason why the normal equations (9) help to reduce the condition number and makes the problem become better-conditioned.

Next, we computed the multiplication $R_{550}^{\mathsf{T}} R_{550}$ in quadruple precision arithmetic and observed that the smallest singular values of $R_{550}^{\mathsf{T}} R_{550}$ conincided with the squared singular values $\sigma_i(R_{550})^2$ (blue circle symbol) in the left of Figure 6, unlike in double precision computation. Since the maximum of the elements of $\mid \mathrm{fl}_q(R_{550}^{\mathsf{T}} R_{550}) - \mathrm{fl}_d(R_{550}^{\mathsf{T}} R_{550}) \mid$ is approximately 8.16×10^{-12} , double precision arithmetic contains error of the order of 10^{-12}. Thus, double precision arithmetic has an effect of regularizing the matrix $R_{550}^{\mathsf{T}} R_{550}$, since double precision matrix multiplication is not accurate enough to keep all the information.

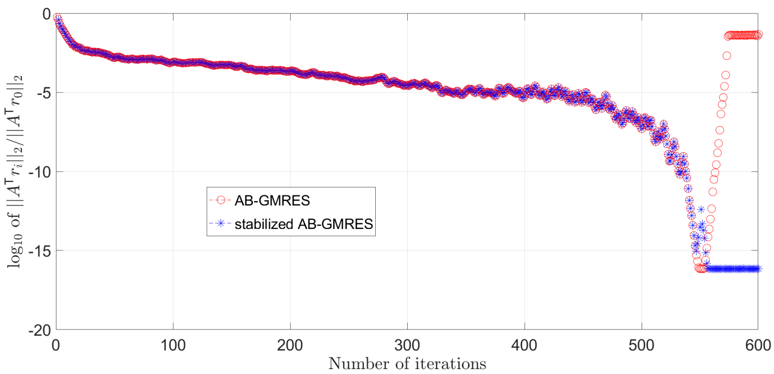

FIGURE 7 Effect of the stabilized method in quadruple precision arithmetic for Maragal_3T.

4.3 | Quadruple precision

In order to see the effect of the machine precision ϵ on the convergence of the AB-GMRES, we compared the stabilized AB-GMRES with the AB-GMRES in quadruple precision arithmetic for the problem Maragal_3T in Figure 7. For both methods, the relative residual norm reached a smaller level of 10^{-16} compared to 10^{-12} and 10^{-8}, respectively, for double precision arithmetic in Figure 4. The curve of the relative residual norm became smoother compared to double precision. As seen in Figure 7, the relative residual norm of the AB-GMRES method jumped to 10^{-1} after reaching 10^{-16}, whereas the relative residual norm of the stabilized GMRES stayed around 10^{-16}.

4.4 | When the stabilized GMRES method works

Motivated by the Läuchli matrix [17], we consider solving the following EP (equal projection) problem $A_3 x = (1,0,0)^\mathsf{T}$, where A_3 is null space symmetric, that is $\mathcal{N}(A_3) = \mathcal{N}(A_3^\mathsf{T})$ with null space $\mathcal{N}(A_3) = \mathrm{span}\{(1,-1,1)^\mathsf{T}\}$.

$$A_3 x = \begin{pmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} - \frac{\sqrt{6\epsilon}}{6} & -\frac{\sqrt{6\epsilon}}{6} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} + \frac{\sqrt{6\epsilon}}{6} & \frac{\sqrt{6\epsilon}}{6} \\ 0 & \frac{\sqrt{6\epsilon}}{3} & \frac{\sqrt{6\epsilon}}{3} \end{pmatrix} x = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \tag{10}$$

where ϵ is the machine epsilon.

Apply GMRES with $x_0 = 0$ to (10). Let $R_s \in \mathbb{R}^{s \times s}$ be the upper triangular matrix obtained at the sth iteration of GMRES. In the second iteration, after applying the Givens rotation to $H_{3,2}$, we obtain the following:

$$R_2 = \begin{pmatrix} 1 & 1 \\ 0 & \sqrt{\epsilon} \end{pmatrix}, \qquad R_2^\mathsf{T} R_2 = \begin{pmatrix} 1 & 1 \\ 1 & 1+\epsilon \end{pmatrix} \simeq \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}. \tag{11}$$

Thus, there is a risk that the stabilized GMRES will give a numerically singular matrix $R_2^\mathsf{T} R_2$ in finite precision arithmetic for nonsingular R_2. We will analyze this phenominon.

We define the following.

$O(\epsilon)$ denotes that there exists a constant c independent of ϵ, such that $-c\epsilon < O(\epsilon) < c\epsilon$. Also, let

$$\mathbb{O}(\epsilon) = \begin{pmatrix} O(\epsilon) \\ O(\epsilon) \\ \vdots \\ O(\epsilon) \end{pmatrix} \in \mathbb{R}^n, \quad \mathcal{O}(\epsilon) = [\mathbb{O}(\epsilon), \mathbb{O}(\epsilon), \cdots, \mathbb{O}(\epsilon)] \in \mathbb{R}^{n \times n}. \tag{12}$$

We assume that the basic arithmetic operations op $= +, -, *, /$ satisfy $\mathrm{fl}(x \text{ op } y) = (x \text{ op } y)(1 + O(\epsilon))$ as in [18].

Note also that the following hold from[18]. Let $x, y \in \mathbb{R}^n$, $A, B \in \mathbb{R}^{n\times n}$, and

$$|x| = \begin{pmatrix} |x_1| \\ |x_2| \\ \vdots \\ |x_n| \end{pmatrix} \quad \text{for} \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \tag{13}$$

$$|A| = \begin{pmatrix} |a_{11}| & |a_{12}| & \cdots & |a_{1n}| \\ |a_{21}| & |a_{22}| & \cdots & |a_{2n}| \\ \vdots & \vdots & \ddots & \vdots \\ |a_{n1}| & |a_{n2}| & \cdots & |a_{nn}| \end{pmatrix} \tag{14}$$

for $A = (a_{pq})$. Then

$$\begin{aligned} \mathrm{fl}(x^\top y) &= x^\top y + O(n\epsilon)|x|^\top |y| = x^\top y + O(n\epsilon), \\ \mathrm{fl}(Ax) &= Ax + \mathbb{O}(n\epsilon)|A||x| = Ax + \mathbb{O}(n\epsilon), \\ \mathrm{fl}(AB) &= AB + O(n\epsilon)|A||B| = AB + \mathcal{O}(n\epsilon). \end{aligned}$$

Note also that the following theorem holds from Theorem 8.10 of[18].

Theorem 3. Let $T = (t_{pq}) \in \mathbb{R}^{n\times n}$ be a triangular matrix and $b \in \mathbb{R}^n$. Then, the computed solution $\hat{x}$ obtained from substitution applied to $Tx = b$ satisfies

$$\hat{x} = x + O(n^2\epsilon)M(T)^{-1}|b|. \tag{15}$$

Here, $M(T) = (m_{ij})$ is the comparison matrix such that

$$m_{ij} = \begin{cases} |t_{ij}|, & i = j, \\ -|t_{ij}|, & i \neq j. \end{cases} \tag{16}$$

Further, we define the following.

Assume $\|A\|_2 = O(1)$. We say $A \in \mathbb{R}^{n\times n}$ is numerically nonsingular if and only if

$$\mathrm{fl}(Ax) = \mathbb{O}(\epsilon) \quad \Rightarrow \quad x = \mathbb{O}(\epsilon). \tag{17}$$

Note that this definition of numerical nonsingularity agrees with that of numerical rank due to the following.

Let the SVD of $A = U\Sigma V^\top$ where U, V are orthogonal matrices and $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$. Here, $\|A\|_2 = \sigma_1 = O(1)$. If the numerical rank of A is $r < n$, there is a $\sigma_i = O(\epsilon)$, $r + 1 \leq i \leq n$. Then, $Ax = U\Sigma V^\top x = O(\epsilon)$ admits $x' = V^\top x = (x'_1, x'_2, \dots, x'_n)^\top$ such that $x'_i = O(1)$, and hence $x = \mathbb{O}(1)$. Thus, A is numerical singular. Then, the following theorem holds.

Theorem 4. Let $R_s = (r_{pq}) \in \mathbb{R}^{s\times s}$ be an upper-triangular matrix and

$$R_{s+1} = \begin{pmatrix} R_s & d \\ 0^\top & r_{s+1,s+1} \end{pmatrix} \in \mathbb{R}^{(s+1)\times(s+1)}. \tag{18}$$

Assume R_s is numerically nonsingular, and $R_s = \mathcal{O}(1)$, $R_s^{-1} = \mathcal{O}(1)$, $M(R_s)^{-1} = \mathcal{O}(1)$, $d = \mathbb{O}(1)$ and $O(s) = O(s^2) = O(1)$. Then, the following holds:

$$\mathrm{fl}(R_{s+1}^\top R_{s+1}) \text{ is numerically nonsingular} \quad \Longleftrightarrow \quad \mathrm{fl}(r_{s+1,s+1}^2) > \mathrm{fl}(d^\top d)O(\epsilon).$$

Proof. See Appendix C. □

Theorem 4 gives the necessary and sufficient condition so that the stabilized GMRES works at the $(s + 1)$st iteration, i.e. $R_{s+1}^\top R_{s+1}$ is numerically nonsingular.

The difficulty in solving $R_i y_i = t_i$ by backward substitution is not because the diagonals of R_i are tiny. The reason is that R_i has tiny singular values. However, the exceptional example (11) exists where the stabilized AB-GMRES does not work. The condition $\mathrm{fl}(r_{s+1,s+1}^2) > \mathrm{fl}(d^\top d)O(\epsilon)$ in Theorem 4 excludes such exceptions.

Figure 8 shows $r_{s+1,s+1}^2$ and $d^\top d$ together with the convergence of the AB-GMRES and that of the stabilized AB-GMRE for Maragal_3T. The figure shows that upto 613 iterations, the conditions in Theorem 4 are satisfied, and $R_{s+1}^\top R_{s+1}$ is numerically nonsingular, so that the stabilized AB-GMRES works.

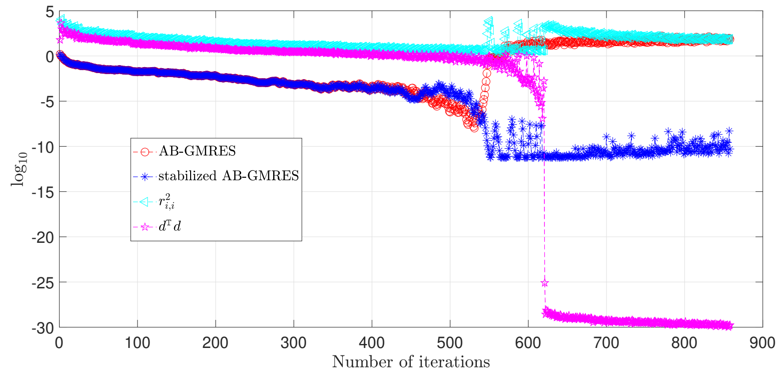

FIGURE 8 $r_{i,i}^2$ $(i = s + 1)$ and $d^\top d$ in stabilized AB-GMRES for Maragal_3T.

4.5 | Comparison with Tikhonov regularization method

Another approach to stabilize the AB-GMRES would be to apply Tikhonov regularization. There are two methods to implement it. The first method is to solve the following square system:

$$(R_i^\top R_i + \lambda I)y_i = R_i^\top t_i, \qquad \lambda \geq 0 \tag{19}$$

using the Cholesky decomposition.

The second method is to solve the regularized least suqares problem

$$\min_{y_i \in \mathbb{R}^i} \left\| \begin{pmatrix} t_i \\ 0 \end{pmatrix} - \begin{pmatrix} R_i \\ \sqrt{\lambda} I \end{pmatrix} y_i \right\|_2 \tag{20}$$

using the QR decomposition.

These two methods are equivalent mathematically. However, they are not equivalent numerically. The behavior of the first method is similar to the stabilized AB-GMRES. Table 3 shows that AB-GMRES combined with the first method converges better when $\lambda = 10^{-16}$ than when $\lambda = 10^{-14}$. This method can be used to shift upwards the small singular values, but is less acurrate compared to the stabilized AB-GMRES (cf. Table 2).

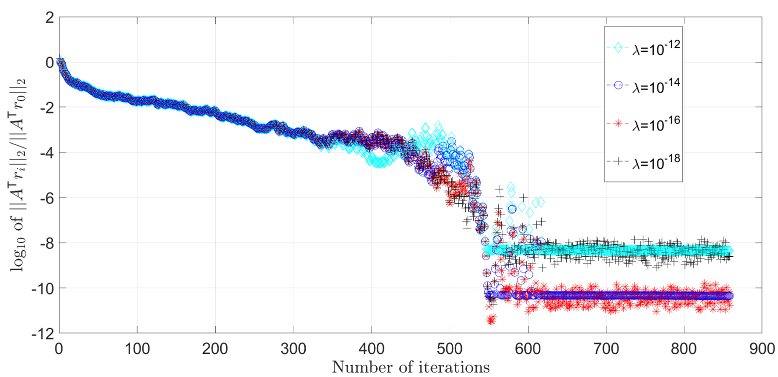

FIGURE 9 Relative residual norm for the regularized AB-GMRES using (20) versus number of iterations for different λ for Maragal_3T.

Table 3 also shows that the second method is even more accurate compared with the stabilized AB-GMRES method. There is no need to form the normal equations, so that less information is lost due to rounding error. However, one needs to choose

TABLE 3 Attainable smallest relative residual norm $\|A^\mathsf{T} r_i\|_2 / \|A^\mathsf{T} r_0\|_2$ for AB-GMRES with Tikhonov regularization using (19) and (20).

matrix	Maragal_3T	Maragal_4T	Maragal_5T	Maragal_6T	Maragal_7T
iter.	552	597	1304	2440	1864
method (19) $\lambda = 10^{-14}$	5.08×10^{-11}	5.57×10^{-8}	1.05×10^{-5}	8.26×10^{-6}	4.53×10^{-6}
iter.	570	598	1226	2440	1864
method (19) $\lambda = 10^{-16}$	5.80×10^{-12}	5.59×10^{-8}	4.22×10^{-6}	8.26×10^{-6}	4.53×10^{-6}
iter.	553	547	1261	2937	2475
method (20) $\lambda = 1.6 \times 10^{-14}$	7.54×10^{-11}	5.59×10^{-8}	1.15×10^{-5}	9.12×10^{-6}	2.78×10^{-7}
iter.	551	547	1262	3037	2475
method (20) $\lambda = 10^{-16}$	3.37×10^{-12}	5.59×10^{-8}	5.64×10^{-7}	1.91×10^{-6}	2.78×10^{-7}

an appropriate value for the regularization parameter λ. Figure 9 shows the relative residual norm $\|A^\mathsf{T} r_i\|_2 / \|A^\mathsf{T} r_0\|_2$ for the regularized AB-GMRES using (20) versus the number of iterations for different values of λ for Maragal_3T. According to Figure 9, $\lambda = 10^{-16}$ was optimal among $10^{-12}, 10^{-14}, 10^{-16}, 10^{-18}$, so we recommend this value in practice.

We here note the following.

Theorem 5. Let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_i$ be the the singular values of R_i. Then, the singular values of

$$R_i' = \begin{pmatrix} R_i \\ \sqrt{\lambda} I \end{pmatrix} \tag{21}$$

are given by $\sqrt{\sigma_1^2 + \lambda} \geq \sqrt{\sigma_2^2 + \lambda} \geq \cdots \geq \sqrt{\sigma_i^2 + \lambda}$.

Proof. See Appendix D. □

Then, let

$$\kappa \equiv \kappa_2(R_i) = \frac{\sigma_1}{\sigma_i}, \quad \kappa'^2 \equiv \kappa_2(R_i')^2 = \frac{\sigma_1^2 + \lambda}{\sigma_1^2/\kappa^2 + \lambda} = 1 + \frac{\sigma_1^2(1 - 1/\kappa^2)}{\sigma_1^2/\kappa^2 + \lambda}. \tag{22}$$

Since $\kappa \geq 1, \mathrm{d}\kappa'/\mathrm{d}\lambda \leq 0$ for $\lambda \geq 0$ and $\kappa'(\lambda = 0) = \kappa, \kappa'(\lambda = +\infty) = 1$. Note also that

$$\lambda = \frac{\sigma_1^2[1 + (\kappa'/\kappa)^2]}{\kappa'^2 - 1} \tag{23}$$

Therefore, for instance, if $\kappa \gg 1$ and we want $\kappa' = \sqrt{\kappa}$,

$$\lambda = \frac{\sigma_1^2(1 + 1/\kappa)}{\kappa - 1} \simeq \frac{\sigma_1^2}{\kappa}. \tag{24}$$

For example, if $\kappa = 10^{16}$ and we want $\kappa' = 10^8$, we should choose $\lambda \simeq \sigma_1^2 \times 10^{-16}$. For Maragal_3T, the largest singular value σ_1 is about 12.64, so that we can estimate a reasonable value of $\lambda \simeq 1.60 \times 10^{-14}$. However, this estimation assumes $\kappa' = \sqrt{\kappa}$, and needs an extra cost for computing σ_1. See [19] for other estimation techniques for the regularization parameter.

5 | COMPARISONS WITH OTHER METHODS

5.1 | Underdetermined inconsistent least squares problems

First, we compared the stabilized AB-GMRES with the range restricted AB-GMRES (RR-AB-GMRES)[20], where the Krylov subspace for the RR-AB-GMRES with $B = A^\mathsf{T}$ is $K_i(AA^\mathsf{T}, AA^\mathsf{T} r_0)$, AB-GMRES with $B = A^\mathsf{T}$, BA-GMRES with $B = A^\mathsf{T}$, LSQR[2] and LSMR[4]. All programs for iterative methods were coded according to the algorithms in[2–4, 20]. Each method was terminated at the iteration step which gives the minimum relative residual norm within m iterations, where m is the number of

TABLE 4 Comparison of the attainable smallest relative residual norm $\|A^\mathsf{T} r_i\|_2 / \|A^\mathsf{T} r_0\|_2$.

matrix		Maragal_3T	Maragal_4T	Maragal_5T	Maragal_6T	Maragal_7T
standard AB-GMRES	iter.	531	465	1110	2440	1864
		1.05×10^{-8}	2.09×10^{-7}	5.35×10^{-6}	8.26×10^{-6}	4.53×10^{-6}
stabilized AB-GMRES	iter.	552	598	1226	3002	2459
		5.99×10^{-12}	5.59×10^{-8}	4.22×10^{-6}	3.88×10^{-6}	2.80×10^{-7}
RR-AB-GMRES	iter.	553	565	1223	2374	2474
		2.57×10^{-11}	5.59×10^{-8}	3.62×10^{-6}	1.63×10^{-5}	2.78×10^{-7}
BA-GMRES	iter.	562	626	1263	4373	5658
		2.88×10^{-14}	7.92×10^{-11}	2.29×10^{-12}	5.12×10^{-11}	2.03×10^{-10}
LSQR	iter.	1682	2375	4576	151013	97348
		5.64×10^{-14}	2.77×10^{-10}	1.11×10^{-11}	5.87×10^{-10}	1.33×10^{-9}
LSMR	iter.	1654	2308	4273	127450	70242
		5.51×10^{-14}	3.00×10^{-10}	3.25×10^{-11}	4.16×10^{-10}	9.95×10^{-10}

TABLE 5 Comparison of the CPU time (seconds) to obtain relative residual norm $\|A^\mathsf{T} r_i\|_2 / \|A^\mathsf{T} r_0\|_2 < 10^{-8}$.

matrix		Maragal_3T	Maragal_4T	Maragal_5T	Maragal_6T	Maragal_7T
standard AB-GMRES	iter.	-	-	-	-	-
		-	-	-	-	-
stabilized AB-GMRES	iter.	546 (526)	-	-	-	-
		2.01	-	-	-	-
RR-AB-GMRES	iter.	545	-	-	-	-
		1.84	-	-	-	-
BA-GMRES	iter.	530	608	1232	3623	5001
		2.10	3.19	4.25×10^{1}	1.81×10^{3}	9.20×10^{3}
LSQR	iter.	1465	2120	4032	101893	54444
		1.27×10^{-1}	2.56×10^{-1}	1.49	2.93×10^{2}	4.33×10^{2}
LSMR	iter.	1456	1989	4013	54017	31206
		1.25×10^{-1}	2.37×10^{-1}	1.49	1.50×10^{2}	2.23×10^{2}

the rows of the matrix. No restarts were used for GMRES. Experiments were done for rank-deficient matrices whose information is given in Table 1. Here, we have deleted the zero rows and columns of the test matrices beforehand. The elements of b were randomly generated using the MATLAB function `rand`. Each experiment was done 10 times for the same right hand side b and the average of the CPU times are shown. Symbol - denotes that $\|A^\mathsf{T} r_i\|_2 / \|A^\mathsf{T} r_0\|_2$ did not reach 10^{-8} within $20n$ iterations.

Table 4 shows that the stabilized AB-GMRES is generally more accurate than the RR-AB-GMRES. The stabilized AB-GMRES took more iterations to attain the same order of the smallest residual norm than the RR-AB-GMRES. Table 4 also shows that for the same underdetermined least squares problems, the BA-GMRES was the best in terms of the attainable smallest relative residual norm and that the LSQR and LSMR are comparable to the BA-GMRES, but require less CPU time according to Tabel 5.

TABLE 6 Information of the singular square matrices.

matrix	size	density[%]	rank	$\kappa_2(A)$	application
Harvard500	500	1.05	170	1.30×10^{2}	web connectivity
netz4504	1961	0.13	1342	3.41×10^{1}	2D/3D finite element problem
TS	2142	0.99	2140	3.52×10^{3}	counter example problem
grid2_dual	3136	0.12	3134	8.58×10^{3}	2D/3D finite element problem
uk	4828	0.06	4814	6.62×10^{3}	undirected graph
bw42	10000	0.05	9999	2.03×10^{3}	partial differential equation [8]

TABLE 7 Comparison of the attainable smallest relative residual norm $\|A^{\mathsf{T}}r_i\|_2/\|A^{\mathsf{T}}r_0\|_2$ for inconsistent square linear systems.

matrix	Harvard500	netz4504	TS	grid2_dual	uk	bw42
iter.	104	144	1487	3134	4620	715
standard AB-GMRES	9.38×10^{-9}	4.51×10^{-10}	1.56×10^{-9}	5.98×10^{-10}	1.35×10^{-9}	8.06×10^{-8}
iter.	175	201	1617	3135	4779	788
stabilized AB-GMRES	4.53×10^{-14}	1.51×10^{-14}	1.54×10^{-9}	1.14×10^{-9}	6.81×10^{-10}	1.66×10^{-7}
iter.	135	200	1652	3134	4706	1163
RR-AB-GMRES	7.78×10^{-14}	3.36×10^{-14}	4.56×10^{-9}	6.52×10^{-8}	8.33×10^{-8}	1.56×10^{-5}
iter.	139	194	1628	3134	4724	1520
BA-GMRES	1.91×10^{-15}	7.27×10^{-16}	8.43×10^{-13}	1.23×10^{-13}	6.94×10^{-14}	1.97×10^{-11}
iter.	391	198	6047	12549	6249	1256
LSQR	3.59×10^{-15}	5.86×10^{-16}	1.96×10^{-12}	2.51×10^{-13}	6.56×10^{-14}	1.59×10^{-11}
iter.	338	195	6219	12497	6199	1212
LSMR	2.01×10^{-15}	5.97×10^{-16}	1.25×10^{-12}	2.34×10^{-13}	7.35×10^{-14}	1.60×10^{-11}

5.2 | Inconsistent systems with highly ill-conditioned square coefficient matrices

The stabilized AB-GMRES is not restricted to solving underdetermined problems but can also be applied to solving the least squares problem $\min_{x\in\mathbb{R}^n}\|b - Ax\|_2$, where $A \in \mathbb{R}^{n\times n}$ is a highly ill-conditioned square matrix. Thus, we also test on square matrices of different kinds. Table 6 gives the information of the matrices.

These matrices are all numerically singular. We generated the right-hand side b by the MATLAB function `rand`, so that the systems are generically inconsistent. We compared the stabilized AB-GMRES with the standard AB-GMRES, RR-AB-GMRES, BA-GMRES with $B = A^{\mathsf{T}}$, LSMR [4], and LSQR [2]. Table 7 gives the smallest relative residual norm and the number of iterations. Table 9 gives the CPU times in seconds required to obtain relative residual norm $\|A^{\mathsf{T}}r_i\|_2/\|A^{\mathsf{T}}r_0\|_2 < 10^{-8}$. The switching strategy which was introduced in Section 4.1 was used for the stabilized AB-GMRES when measuing CPU times. The number of iterations when switching occurred is in brackets.

Table 7 shows that for most problems the BA-GMRES was the best in terms of accuracy of relative residual norm. The LSQR and LSMR are similar and are comparable to the BA-GMRES, beacuse they all change the inconsistent problem into a consistent problem. The LSQR and LSMR are more suitable for large and sparse problems compared to the BA-GMRES because they require less CPU time and memory.

For Harvard500 and bw42, the AB-GMRES could only converge to the level of 10^{-9} regarding the relative residual norm, while the stabilized AB-GMRES converged to the level of 10^{-14}. The stabilized AB-GMRES was robust in the sense that it could continue to compute even when the upper triangular matrix R_i became seriously ill-conditioned, and the relative residual norm did not increase sharply towards the end, but just stagnated at a low level, just like for consistent problems. Comparing the CPU time in Tabel 9, LSMR was the fastest. The stabilized AB-GMRES was usually faster than BA-GMRES.

TABLE 8 Attainable smallest relative residual norm $\|A^{\mathsf{T}} r_i\|_2/\|A^{\mathsf{T}} r_0\|_2$ for bw42.

method	iter.	$\min_i \|A^{\mathsf{T}} r_i\|_2/\|A^{\mathsf{T}} r_0\|_2$
standard GMRES	147	8.08×10^{-9}
stabilized GMRES	219	1.94×10^{-11}
RR-GMRES	220	3.13×10^{-11}

TABLE 9 Comparison of the CPU time (seconds) to obtain relative residual norm $\|A^{\mathsf{T}} r_i\|_2/\|A^{\mathsf{T}} r_0\|_2 < 10^{-8}$ for inconsistent square linear systems.

matrix	Harvard500	netz4504	TS	grid2_dual	uk	bw42
iter.	104	134	1411	3134	4583	-
standard AB-GMRES	4.72×10^{-2}	1.87×10^{-1}	2.14×10	2.16×10^{2}	6.93×10^{2}	-
iter.	104	134	1531 (182)	3134	4679 (4199)	-
stabilized AB-GMRES	4.78×10^{-2}	1.89×10^{-1}	8.19×10	2.21×10^{2}	1.93×10^{3}	-
iter.	114	153	1530	-	-	-
RR-AB-GMRES	6.42×10^{-2}	2.62×10^{-1}	2.68×10	-	-	-
iter.	103	131	1379	3134	4562	738
BA-GMRES	5.48×10^{-2}	1.72×10^{-1}	2.06×10	2.44×10^{2}	7.55×10^{2}	2.33×10
iter.	222	134	4239	11802	5948	913
LSQR	5.63×10^{-3}	6.61×10^{-3}	7.86×10^{-1}	1.15	8.65×10^{-1}	3.12×10^{-1}
iter.	215	132	3913	11746	5898	655
LSMR	5.34×10^{-3}	6.42×10^{-3}	7.04×10^{-1}	1.15	8.42×10^{-1}	2.32×10^{-1}

Thus, our stabilization method also makes AB-GMRES stable for highly ill-conditioned inconsistent systems with square coefficient matrices.

The coefficient matrix A of bw42 is singular and satisfies $\mathcal{N}(A) = \mathcal{N}(A^{\mathsf{T}})$. The problem comes from a finite-difference discritization of a PDE with periodic boundary condition (Experiment 4.2 in Brown and Walker[8] with the original b). Since the matrix is range symmetric, the GMRES, RR-GMRES, and stabilized GMRES can be directly applied to $Ax = b$ (See[8] Theorem 2.4,[21] Theorem 2.7, and[22] Theorem 3.2.) as shown in Table 8. The stabilized GMRES gave the relative residual norm 1.94×10^{-11} for bw42 at the 219th iteration, similar to the BA-GMRES.

6 | CONCLUDING REMARKS

We proposed a stabilized AB-GMRES method for ill-conditioned underdetermined and inconsistent least squares problems. It shifts upwards the tiny singular values of the upper triangular matrix appearing in AB-GMRES, making the process more stable, giving better convergence, and more accurate solutions compared to AB-GMRES. The method is also effective for making AB-GMRES stable for inconsistent least squares problems with highly ill-conditioned square coefficient matrices.

ACKNOWLEDGMENTS

Ken Hayami was supported by JSPS KAKENHI Grant Number 15K04768.
Keiichi Morikuni was supported by JSPS KAKENHI Grant Number 16K17639 and Hattori Hokokai Foundation.
Jun-Feng Yin was supported by the National Natural Science Foundation of China (No. 11971354).

APPENDIX

A PROOF OF STATEMENT IN SECTION 2.3

Lemma 1. Assume $\mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \{0\}$, and $\mathrm{grade}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$. Then, $\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})$ holds.

Proof. Note that

$$\widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \mathrm{span}\{\widetilde{A}b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})}, \cdots, \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})}\} \subseteqq \mathrm{span}\{b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A}b|_{\mathcal{R}(\widetilde{A})}, \cdots, \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})}\} = \mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}).$$

$\mathrm{grade}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$ implies that

$$\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \mathrm{span}\{b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A}b|_{\mathcal{R}(\widetilde{A})}, \cdots, \widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})}\}.$$

Hence,

$$\widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})} = c_0 b|_{\mathcal{R}(\widetilde{A})} + c_1 \widetilde{A}b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_{k-1}\widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})}, \quad c_i \in \mathbb{R}, i = 0, 1, 2, \cdots, k-1.$$

If $c_0 = 0$,

$$\widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})} = c_1 \widetilde{A}b|_{\mathcal{R}(\widetilde{A})} + c_2 \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_{k-1}\widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})}.$$

Hence,

$$c_1 \widetilde{A}b|_{\mathcal{R}(\widetilde{A})} + c_2 \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_{k-1}\widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})} - \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})} = \widetilde{A}(c_1 b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_{k-1}\widetilde{A}^{k-2} b|_{\mathcal{R}(\widetilde{A})} - \widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})}) = 0.$$

Hence,

$$c_1 b|_{\mathcal{R}(\widetilde{A})} + c_2 \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_{k-1}\widetilde{A}^{k-2} b|_{\mathcal{R}(\widetilde{A})} - \widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})} \in \mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \{0\}.$$

which implies

$$\widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})} = c_1 b|_{\mathcal{R}(\widetilde{A})} + c_2 \widetilde{A}b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_{k-1}\widetilde{A}^{k-2} b|_{\mathcal{R}(\widetilde{A})}.$$

Thus,

$$\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \mathcal{K}_{k-1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}),$$

which contradicts with $\mathrm{grade}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$. Hence, $c_0 \neq 0$, and

$$b|_{\mathcal{R}(\widetilde{A})} = d_1 \widetilde{A}b|_{\mathcal{R}(\widetilde{A})} + d_2 \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})} + \cdots + d_{k-1}\widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})} + d_k \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})}.$$

Hence,

$$\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \mathrm{span}\{b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A}b|_{\mathcal{R}(\widetilde{A})}, \cdots, \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})}\} \subseteqq \mathrm{span}\{\widetilde{A}b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})}, \cdots, \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})}\} = \widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}).$$

Thus,

$$\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}).$$

□

Corollary 1. Assume $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\top)$, and $\mathrm{grade}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$. Then, $\mathcal{K}_{k+1}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = \widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})$ holds.

Proof. $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\top)$ implies that

$$\mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\top) \cap \mathcal{R}(\widetilde{A}) = \mathcal{R}(\widetilde{A})^\perp \cap \mathcal{R}(\widetilde{A}) = \{0\}.$$

Hence, from Lemma 1, Corollary 1 holds. □

B PROOF OF STATEMENT IN SECTION 2.3

Lemma 2. Assume $\mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \{0\}$, $\mathrm{grade}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$, and $b \notin \mathcal{R}(\widetilde{A})$. Then, $\dim(\mathcal{K}_{k+1}(\widetilde{A}, b)) = k+1$ holds.

Proof. Let $c_0, c_1, \dots, c_k \in \mathbb{R}$ satisfy

$$c_0 b + c_1 \widetilde{A} b + \cdots + c_k \widetilde{A}^k b = 0.$$

Since $\mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \{0\}$,

$$b = b|_{\mathcal{R}(\widetilde{A})} \oplus b|_{\mathcal{N}(\widetilde{A})},$$

where $b|_{\mathcal{N}(\widetilde{A})}$ denotes the orthogonal projection of b onto $\mathcal{N}(\widetilde{A})$. Hence,

$$c_0 b|_{\mathcal{N}(\widetilde{A})} + c_0 b|_{\mathcal{R}(\widetilde{A})} + c_1 \widetilde{A} b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_k \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})} = 0.$$

If $c_0 \neq 0$

$$b|_{\mathcal{N}(\widetilde{A})} = -b|_{\mathcal{R}(\widetilde{A})} - \frac{c_1}{c_0} \widetilde{A} b|_{\mathcal{R}(\widetilde{A})} - \cdots - \frac{c_k}{c_0} \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})} \in \mathcal{R}(\widetilde{A}).$$

Hence,

$$b|_{\mathcal{N}(\widetilde{A})} \in \mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \{0\}.$$

Thus, $b|_{\mathcal{N}(\widetilde{A})} = 0$, which contradicts $b \notin \mathcal{R}(\widetilde{A})$. Hence, we have $c_0 = 0$, and

$$c_1 \widetilde{A} b + c_2 \widetilde{A}^2 b + \cdots + c_k \widetilde{A}^k b = c_1 \widetilde{A} b|_{\mathcal{R}(\widetilde{A})} + c_2 \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})} + \cdots + c_k \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})} = 0.$$

But, since

$$\dim(\mathrm{span}\{\widetilde{A} b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A}^2 b|_{\mathcal{R}(\widetilde{A})}, \cdots, \widetilde{A}^k b|_{\mathcal{R}(\widetilde{A})}\}) = \dim(\widetilde{A}\mathrm{span}\{b|_{\mathcal{R}(\widetilde{A})}, \widetilde{A} b|_{\mathcal{R}(\widetilde{A})} \cdots, \widetilde{A}^{k-1} b|_{\mathcal{R}(\widetilde{A})}\}) = \dim(\widetilde{A}\mathcal{K}_k(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})})) = k$$

holds from Lemma 1, we have $c_1 = c_2 = \cdots = c_k = 0$, which implies $\dim(\mathcal{K}_{k+1}(\widetilde{A}, b)) = k+1$. □

Corollary 2. Assume $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\mathsf{T})$, $\mathrm{grade}(\widetilde{A}, b|_{\mathcal{R}(\widetilde{A})}) = k$, and $b \notin \mathcal{R}(\widetilde{A})$. Then, $\dim(\mathcal{K}_{k+1}(\widetilde{A}, b)) = k+1$ holds.

Proof. $\mathcal{N}(\widetilde{A}) = \mathcal{N}(\widetilde{A}^\mathsf{T})$ implies $\mathcal{N}(\widetilde{A}) \cap \mathcal{R}(\widetilde{A}) = \{0\}$. Hence, the corollary follows from Lemma 2. □

C PROOF OF THEOREM 4

Note that

$$R_{s+1}^\mathsf{T} R_{s+1} = \begin{pmatrix} R_s & 0 \\ d^\mathsf{T} & r_{s+1,s+1} \end{pmatrix} \begin{pmatrix} R_s & d \\ 0^\mathsf{T} & r_{s+1,s+1} \end{pmatrix} = \begin{pmatrix} R_s^\mathsf{T} R_s & R_s^\mathsf{T} d \\ d^\mathsf{T} R_s & d^\mathsf{T} d + r_{s+1,s+1}^2 \end{pmatrix}.$$

Proof of (⇒). Assume $\mathrm{fl}(r_{s+1,s+1}^2) \leq \mathrm{fl}(d^\mathsf{T} d) O(\epsilon)$. Then, since

$$\mathrm{fl}(d^\mathsf{T} d) = d^\mathsf{T} d + O(s\epsilon) d^\mathsf{T} d = (1 + O(s\epsilon)) d^\mathsf{T} d,$$
$$\mathrm{fl}(d^\mathsf{T} d + r_{s+1,s+1}^2) = (d^\mathsf{T} d + r_{s+1,s+1}^2)(1 + O(s\epsilon)) = d^\mathsf{T} d (1 + O(s\epsilon)),$$

we have

$$\mathrm{fl}(R_{s+1}^\mathsf{T} R_{s+1}) = \begin{pmatrix} R_s^\mathsf{T} R_s + O(s\epsilon)|R_s|^\mathsf{T}|R_s| & R_s^\mathsf{T} d + O(s\epsilon)|R_s|^\mathsf{T}|d| \\ d^\mathsf{T} R_s + O(s\epsilon)|d|^\mathsf{T}|R_s| & d^\mathsf{T} d + O(s\epsilon) d^\mathsf{T} d \end{pmatrix} = \begin{pmatrix} R_s^\mathsf{T} \\ d^\mathsf{T} \end{pmatrix} \begin{pmatrix} R_s & d \end{pmatrix} + \mathcal{O}(s\epsilon), \tag{C1}$$

since $R_s = \mathcal{O}(1)$ and $d = \mathbb{O}(1)$. Note

$$\begin{pmatrix} R_s & d \end{pmatrix} \begin{pmatrix} -R_s^{-1} d \\ 1 \end{pmatrix} = -R_s R_s^{-1} d + d = 0,$$

since R_s is nonsingular.

Hence,

$$\mathrm{fl}(\begin{pmatrix} R_s & d \end{pmatrix} \begin{pmatrix} -R_s^{-1} d \\ 1 \end{pmatrix}) = \mathrm{fl}\{R_s \mathrm{fl}(-R_s^{-1} d) + d\} = [\mathrm{fl}\{R_s \mathrm{fl}(-R_s^{-1} d)\} + d]\{1 + O(\epsilon)\}.$$

Note here that

$$\mathrm{fl}\{R_s \mathrm{fl}(-R_s^{-1}d)\} = R_s \mathrm{fl}(-R_s^{-1}d) + O(s\epsilon)|R_s||R_s^{-1}d|,$$

and

$$\mathrm{fl}(-R_s^{-1}d) = -R_s^{-1}d + O(s^2\epsilon)M(R_s)^{-1}|d| \tag{C2}$$

from Theorem 3. Hence,

$$\mathrm{fl}(\begin{pmatrix} R_s & d \end{pmatrix}\begin{pmatrix} -R_s^{-1}d \\ 1 \end{pmatrix}) = O(s^2\epsilon)R_s M(R_s)^{-1}|d| + O(s\epsilon)|R_s||R_s^{-1}d| = \mathbb{O}(s^2\epsilon),$$

since $R_s^{-1} = \mathcal{O}(1)$ and $M(R_s)^{-1} = \mathcal{O}(1)$.

Then,

$$\mathrm{fl}(R_{s+1}^\top R_{s+1}\begin{pmatrix} -R_s^{-1}d \\ 1 \end{pmatrix}) = \mathrm{fl}(\{\begin{pmatrix} R_s^\top \\ d^\top \end{pmatrix}\begin{pmatrix} R_s & d \end{pmatrix} + \mathcal{O}(s\epsilon)\}\begin{pmatrix} -R_s^{-1}d + \mathcal{O}(s^2\epsilon)M(R_s)^{-1}|d| \\ 1 \end{pmatrix}) = \mathbb{O}(s^2\epsilon) = \mathbb{O}(\epsilon),$$

since (C1), (C2), and $O(s^2) = O(1)$. Since $\begin{pmatrix} -R_s^{-1}d \\ 1 \end{pmatrix} = \mathbb{O}(1)$, $R_{s+1}^\top R_{s+1}$ is numerically singular. By contraposition, ($\Leftarrow$) holds.

□

Proof of ($\Leftarrow$). Assume $R_{s+1}^\top R_{s+1}$ is not numerically singular. Then, there exists a vector $\begin{pmatrix} z \\ w \end{pmatrix} \in \mathbb{R}^{s+1}$ such that $\left|\begin{pmatrix} z \\ w \end{pmatrix}\right| > \mathbb{O}(\epsilon)$, and

$$\mathrm{fl}\{R_{s+1}^\top R_{s+1}\begin{pmatrix} z \\ w \end{pmatrix}\} = R_{s+1}^\top\left(R_{s+1}\begin{pmatrix} z \\ w \end{pmatrix} + |R_{s+1}|\left|\begin{pmatrix} z \\ w \end{pmatrix}\right|O((s+1)\epsilon)\right) +$$

$$\left|R_{s+1}^\top\right|\left|R_{s+1}\begin{pmatrix} z \\ w \end{pmatrix} + |R_{s+1}|\left|\begin{pmatrix} z \\ w \end{pmatrix}\right|O((s+1)\epsilon)\right|O((s+1)\epsilon) = \mathbb{O}(\epsilon)$$

assuming $O(s+1) = O(1)$.

Hence,

$$\mathrm{fl}\{R_{s+1}^\top R_{s+1}\begin{pmatrix} z \\ w \end{pmatrix}\} = \begin{pmatrix} R_s^\top R_s & R_s^\top d \\ d^\top R_s & d^\top d + r_{s+1,s+1}^2 \end{pmatrix}\begin{pmatrix} z \\ w \end{pmatrix} + \mathbb{O}(\epsilon) = \mathbb{O}(\epsilon).$$

Thus,

$$R_s^\top R_s z + w R_s^\top d = \mathbb{O}(\epsilon), \tag{C3}$$

$$d^\top R_s z + (d^\top d + r_{s+1,s+1}^2)w = \mathbb{O}(\epsilon). \tag{C4}$$

(C3) can be expressed as $R_s^\top(R_s z + wd) = \mathbb{O}(\epsilon)$. From Lemma 3, $R_s^\top$ is numerically nonsingular, so that

$$R_s z + wd = \mathbb{O}(\epsilon). \tag{C5}$$

Hence, from (C4), $d^\top R_s z + w(d^\top d + r_{s+1,s+1}^2) = d^\top(R_s z + wd) + w r_{s+1,s+1}^2 = O(\epsilon)$. Thus, $w r_{s+1,s+1}^2 = O(\epsilon)$. If $w = O(\epsilon)$, $R_s z = \mathbb{O}(\epsilon)$ from (C5). Since R_s is numerically nonsingular, $z = \mathbb{O}(\epsilon)$, which contradicts with the assumption.

Hence, $|w| > O(\epsilon)$, so that $r_{s+1,s+1}^2 = O(\epsilon)$, which gives

$$\mathrm{fl}(r_{s+1,s+1}^2) = O(\epsilon) \le \mathrm{fl}(d^\top d)O(\epsilon).$$

□

Lemma 3. Let $n = O(1)$. If $A \in \mathbb{R}^{n\times n}$ is numerically nonsingular, and $A^{-1} = \mathcal{O}(1)$, then $A^\top$ is numerically nonsingular.

Proof. If

$$\mathrm{fl}(A^\top x) = A^\top x + \mathbb{O}(n\epsilon)|A^\top||x| = \mathbb{O}(n\epsilon),$$

then

$$\mathrm{fl}(x^\top A) = x^\top A + \mathbb{O}^\top(n\epsilon) = \mathbb{O}^\top(n\epsilon).$$

Thus,

$$\mathrm{fl}(x^\top A y) = \mathrm{fl}(x^\top A)y + O(n\epsilon)|\mathrm{fl}(x^\top A)||y| = O(n\epsilon)$$

holds for all $y = \mathbb{O}(1)$.

For arbitrary $z = \mathbb{O}(1) \in \mathbb{R}^n$, let

$$y = A^{-1}z = \mathbb{O}(1).$$

Then,

$$\mathrm{fl}(Ay) = Ay + O(n\epsilon)|A||y| = z + O(n\epsilon)|A||y|.$$

Hence,

$$z = \mathrm{fl}(Ay) + O(n\epsilon)|A||y| = \mathrm{fl}(Ay) + \mathbb{O}(n\epsilon).$$

Thus, we have

$$\mathrm{fl}(x^\mathsf{T} z) = x^\mathsf{T} z + O(n\epsilon)|x|^\mathsf{T}|z| = \mathrm{fl}(x^\mathsf{T} Ay) + O(n\epsilon) = O(n\epsilon)$$

for arbitrary $z = \mathbb{O}(1) \in \mathbb{R}^n$. Hence, $x = \mathbb{O}(\epsilon)$, so that A^T is numerically nonsingular. □

D PROOF OF THEOREM 5 IN SECTION 4.5

Proof. Let the singular value decomposition of R_i be given by $R_i = U\Sigma V^\mathsf{T} \in \mathbb{R}^{i\times i}$, where U, V are orthogonal matrices and $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \dots, \sigma_i)$. Let $I_i \in \mathbb{R}^{i\times i}$ be the identity matrix. Then, we have $R_i' = \begin{pmatrix} R_i \\ \sqrt{\lambda} I_i \end{pmatrix} = U'\Sigma' V^\mathsf{T}$, where $U' = \begin{pmatrix} U & 0 \\ 0 & V \end{pmatrix}$ and $\Sigma' = \begin{pmatrix} \Sigma \\ \sqrt{\lambda} I_i \end{pmatrix}$. Since $\Sigma'^\mathsf{T}\Sigma' = \Sigma^2 + \lambda I_i = \mathrm{diag}(\sigma_1^2 + \lambda, \sigma_2^2 + \lambda, \dots, \sigma_i^2 + \lambda)$, the singular values of $\begin{pmatrix} R_i \\ \sqrt{\lambda} I_i \end{pmatrix}$ are $\sqrt{\sigma_1^2 + \lambda} \geq \sqrt{\sigma_2^2 + \lambda} \geq \cdots \geq \sqrt{\sigma_i^2 + \lambda}$. □

References

1. Hestenes MR, and Stiefel E. Methods of conjugate gradients for solving linear systems. J Research Nat Bur Standards. 1952;**49**(6):409–436.
2. Paige CC, and Saunders MA. LSQR: An algorithm for sparse linear equations and sparse least squares. ACM Trans Math Software. 1982;**8**(1):43–71.
3. Hayami K, Yin JF, and Ito T. GMRES methods for least squares problems. SIAM J Matrix Anal Appl. 2010;**31**(5):2400–2430.
4. Fong DCL, and Saunders M. LSMR: An iterative algorithm for sparse least-squares problems. SIAM J Sci Comput. 2011;**33**(5):2950–2971.
5. Paige CC, and Saunders MA. Solution of sparse indefinite systems of linear equations. SIAM J Numer Anal. 1975;**12**(4):617–629.
6. Saad Y, and Schultz MH. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J Sci Comput. 1986;**7**(3):856–869.
7. Morikuni K. Inner-iteration Preconditioning for Least Squares Problems. Doctoral Thesis, Department of Informatics, School of Multidisciplinary Sciences, The Graduate University for Advanced Studies; 2013.
8. Brown P, and Walker H. GMRES on (nearly) singular systems. SIAM J Matrix Anal Appl. 1997;**18**(1):37–51.
9. Calvetti D, Lewis B, and Reichel L. GMRES-type methods for inconsistent systems. Linear Algebra Appl. 2000;**316**(1):157–169.
10. Reichel L, and Ye Q. Breakdown-free GMRES for singular systems. SIAM J Matrix Anal Appl. 2005;**26**(4):1001–1021.
11. Morikuni K, and Hayami K. Convergence of inner-iteration GMRES methods for rank-deficient least squares problems. SIAM J Matrix Anal Appl. 2015;**36**(1):225–250.

12. Morikuni K, and Rozložník M. On GMRES for Singular EP and GP Systems. SIAM J Matrix Anal Appl. 2018;**39**(2):1033–1048.

13. Davis T, and Hu Y. The University of Florida sparse matrix collection. ACM Trans Math software. 2011;**38**(1):1–25. Available from: https://sparse.tamu.edu/.

14. Advanpix LLC. Multiprecision Computing Toolbox for MATLAB. Version 4.4.5.12711;. Available from: https://www.advanpix.com/.

15. Foster L., San Jose State University Singular Matrix Database;. Available from: http://www.math.sjsu.edu/singular/matrices/.

16. Tebbens JD, and Tůma M. On incremental condition estimators in the 2-norm. SIAM J Anal Appl. 2014;**35**(1):174–197.

17. Higham NJ. The Test Matrix Toolbox for Matlab (version 3.0). University of Manchester, Manchester; 1995.

18. Higham NJ. *Accuracy and Stability of Numerical Algorithms,*Second Ed. SIAM. Philadelphia; 2002.

19. Brezinski C, Rodriguez G, and Seatzu S. Error estimates for the regularization of least squares problems. Numer Algorithms. 2009;**51**(1):61–76.

20. Neuman A, Reichel L, and Sadok H. Algorithms for range restricted iterative methods for linear discrete ill-posed problems. Numer Algorithms. 2012;**59**(2):325–331.

21. Hayami K, and Sugihara M. A geometric view of Krylov subspace methods on singular systems. Numer Linear Algebra Appl. 2011;**18**(3):449–469.

22. Calvetti D, Lewis B, and Reichel L. GMRES-type methods for inconsistent systems. Linear Algebra Appl. 2000;**316**(1-3):157–169.

23. Hansen PC. *Discrete Inverse Problems: Insight and Algorithms*. SIAM. Philadelphia, P.A.; 2010.

24. Neuman A, Reichel L, and Sadok H. Implementations of range restricted iterative methods for linear discrete ill-posed problems. Linear Algebra Appl. 2012;**436**(10):3974–3990.

25. Paige CC, Rozložník M, and Strakoš Z. Modified Gram-Schmidt (MGS), least squares, and backward stability of MGS-GMRES. SIAM J Matrix Anal Appl. 2006;**28**(1):264–284.