INSTITUTE OF MATHEMATICS

THE CZECH ACADEMY OF SCIENCES

Iterated Gauss-Seidel GMRES

Stephen Thomas
Erin Carson
Miroslav Rozložník
Arielle Carr
Kasia Świrydowicz

Preprint No. 16-2023

PRAHA 2023

ITERATED GAUSS-SEIDEL GMRES

STEPHEN THOMAS*, ERIN CARSON†, MIROSLAV ROZLOŽNÍK‡, ARIELLE CARR§, AND KASIA ŚWIRYDOWICZ¶

Abstract. The GMRES algorithm of Saad and Schultz (1986) is an iterative method for approximately solving linear systems $A\mathbf{x} = \mathbf{b}$, with initial guess $\mathbf{x}_0$ and residual $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$. The algorithm employs the Arnoldi process to generate the Krylov basis vectors (the columns of V_k). It is well known that this process can be viewed as a QR factorization of the matrix $B_k = [\, \mathbf{r}_0, AV_k \,]$ at each iteration. Despite an $\mathcal{O}(\varepsilon)\kappa(B_k)$ loss of orthogonality, for unit roundoff ε and condition number κ, the modified Gram-Schmidt formulation was shown to be backward stable in the seminal paper by Paige et al. (2006). We present an iterated Gauss-Seidel formulation of the GMRES algorithm (IGS-GMRES) based on the ideas of Ruhe (1983) and Świrydowicz et al. (2020). IGS-GMRES maintains orthogonality to the level $\mathcal{O}(\varepsilon)\kappa(B_k)$ or $\mathcal{O}(\varepsilon)$, depending on the choice of one or two iterations; for two Gauss-Seidel iterations, the computed Krylov basis vectors remain orthogonal to working precision and the smallest singular value of V_k remains close to one. The resulting GMRES method is thus backward stable. We show that IGS-GMRES can be implemented with only a single synchronization point per iteration, making it relevant to large-scale parallel computing environments. We also demonstrate that, unlike MGS-GMRES, in IGS-GMRES the relative Arnoldi residual corresponding to the computed approximate solution no longer stagnates above machine precision even for highly non-normal systems.

1. Introduction. We consider linear systems of the form $A\mathbf{x} = \mathbf{b}$, where A is an $n \times n$ real-valued matrix, solved via the generalized minimal residual method (GMRES) [1]. Let $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$ denote the initial residual for the approximate solution $\mathbf{x}_0$. Inside GMRES, the Arnoldi-QR algorithm is applied to orthogonalize the basis vectors for the Krylov subspace $\mathcal{K}_k(A, \mathbf{r}_0)$ spanned by the columns of the $n \times k$ matrix V_k, where $k \ll n$. After k iterations, this produces the $(k+1) \times k$ upper Hessenberg matrix $H_{k+1,k}$ in the Arnoldi expansion such that

$$v_1 = \mathbf{r}_0/\rho, \quad \rho := \|\mathbf{r}_0\|_2, \quad AV_k = V_{k+1}H_{k+1,k}. \tag{1.1}$$

The expansion (1.1) is equivalent to a QR factorization of $B_k = [\, \mathbf{r}_0,\ AV_k \,] = V_{k+1}\, [\, \rho\, \mathbf{e}_1,\ H_{k+1,k} \,]$ and the columns of V_k form an orthonormal basis for the Krylov subspace $\mathcal{K}_k(\, A,\ \mathbf{r}_0\,)$ [2]. The approximate solution in step k is given by $\mathbf{x}_k = \mathbf{x}_0 + V_k\mathbf{y}_k$, where the vector $\mathbf{y}_k$ minimizes the Arnoldi residual

$$\| \, \rho\, \mathbf{e}_1 - H_{k+1,k}\mathbf{y}_k \, \|_2 = \min_y \, \| \, \rho\, \mathbf{e}_1 - H_{k+1,k}\mathbf{y} \, \|_2. \tag{1.2}$$

When the Krylov vectors are orthogonalized via the modified Gram-Schmidt (MGS) algorithm in finite-precision arithmetic, their loss of orthogonality is related in a straightforward way to the convergence of GMRES. Orthogonality among the Krylov vectors is effectively maintained until the norm-wise relative backward error approaches the machine precision as discussed in the work of Paige and Strakoš [3] and Paige et al. [2]. The growth of the condition number $\kappa(B_k) = \sigma_{\max}(B_k)/\sigma_{\min}(B_k)$, where $\sigma_{\max}(B_k)$ and $\sigma_{\min}(B_k)$ are the maximum and minimum singular values of the matrix B_k, respectively, is related to the norm-wise backward error

$$\beta(\mathbf{x}_k) := \frac{\|\mathbf{b} - A\mathbf{x}_k\|_2}{\|\mathbf{b}\|_2 + \|A\|_2\|\mathbf{x}_k\|_2}, \quad k = 1, 2, \ldots \tag{1.3}$$

and it is observed that, in exact arithmetic, $\beta(\mathbf{x}_k)\, \kappa(B_k) = \mathcal{O}(1)$. Also in exact arithmetic, the orthogonality of the columns implies the linear independence of the Krylov basis vectors.

However, in finite-precision arithmetic, the columns of the computed $\bar{V}_k$ are no longer orthogonal, as measured by $\|I - \bar{V}_k^T\bar{V}_k\|_F$, and may deviate substantially from the level of machine precision, $\mathcal{O}(\varepsilon)$. When linear independence is completely lost, the Arnoldi relative residual, corresponding to the computed approximate solution $\bar{\mathbf{x}}_k$, stagnates at a certain level above $\mathcal{O}(\varepsilon)$. This occurs when $\|S_k\|_2 = 1$, where

$$S_k = (I + L_k^T)^{-1}L_k^T,$$

*National Renewable Energy Laboratory, Golden, CO
†Charles University, Prague, CZ
‡Czech Academy of Sciences, Institute of Mathematics, Prague, CZ. Supported by Czech Academy of Sciences (RVO 67985840) and by the Grant Agency of the Czech Republic, Grant No. 23-06159S.
§Lehigh University, Bethlehem, PA
¶Pacific Northwest National Laboratory, Richland, WA

and L_k is the $k\times k$ strictly lower triangular part of $\bar{V}_k^T\bar{V}_k = I+L_k+L_k^T$. Note also that $I-S_k = (I+L_k^T)^{-1}$. However, Paige et al. [2] demonstrated that despite $\mathcal{O}(\varepsilon)\kappa(B_k)$ loss of orthogonality, MGS-GMRES is backward stable for the solution of linear systems.

Both modified and classical Gram-Schmidt versions with delayed re-orthogonalization (DCGS-2) were derived in Świrydowicz et al. [4], and these have been empirically observed to result in a backward stable GMRES method. The development of low-synchronization Gram-Schmidt and generalized minimal residual algorithms by Świrydowicz et al. [4] and Bielich et al. [5] was largely driven by applications that need stable, yet scalable, solvers. An inverse compact WY MGS algorithm is presented in [4] and is based upon the application of the approximate projector

$$P_k^{(1)} = I - \bar{V}_k\, T_k^{(1)}\, \bar{V}_k^T, \quad T_k^{(1)} \approx (\bar{V}_k^T\bar{V}_k)^{-1},$$

where $\bar{V}_k$ is again $n \times k$, I is the identity matrix of dimension n, and $T_k^{(1)}$ is a $k \times k$ lower triangular correction matrix. To obtain a low-synchronization algorithm requiring only a single global reduction per iteration, the normalization is delayed to the next iteration. The matrix $T_k^{(1)} = (\ I + L_k\ )^{-1}$ is obtained from the strictly lower triangular part of $\bar{V}_k^T\bar{V}_k$, again denoted L_k. Note that because $\bar{V}_k$ has almost orthonormal columns, the norm of L_k is small, and $T_k^{(1)}$ is nearly the identity matrix of dimension k. The last row of L_k is constructed from the matrix-vector product $L_{k,1:k-1} = \bar{\mathbf{v}}_k^T\bar{V}_{k-1}$.

The purpose of the present work is to derive an iterated Gauss-Seidel formulation of the GMRES algorithm based on the approximate solution of the normal equations in the Gram-Schmidt projector, as described by Ruhe [6], and the low-synchronization algorithms introduced by Świrydowicz et al. [4]. $T_k^{(1)}$ represents the inverse compact WY form of the MGS projector $P_k^{(1)}$ and corresponds to one Gauss-Seidel iteration for the approximate solution of the normal equations

$$\bar{V}_{k-1}^T\bar{V}_{k-1}\mathbf{r}_{1:k-1,k} = \bar{V}_{k-1}^T\, A\, \bar{\mathbf{v}}_k. \tag{1.4}$$

Ruhe [6, pg. 597] suggested applying LSQR, whereas Björck [7, pg. 312] recommended conjugate gradients to solve the normal equations. Instead, we apply two Gauss-Seidel iterations and demonstrate that the loss of orthogonality is at the level of $\mathcal{O}(\varepsilon)$ *without any need for explicit reorthogonalization* in the Arnoldi-QR algorithm. Resembling the Householder transformation-based implementation, the GMRES method with two Gauss-Seidel iterations is backward stable, where the stability result of Drkošová et al. [8] applies. Even for extremely ill-conditioned and non-normal matrices, the two iteration Gauss-Seidel GMRES formulation matches the Householder (HH)-GMRES of Walker [9], where the Arnoldi residual of the least square problem (1.2) continues to decrease monotonically without any stagnation.

Contributions. In this paper, we present an iterated Gauss Seidel formulation of the GMRES algorithm [1], and establish the backward stability of this approach. The computed Krylov basis vectors maintain orthogonality to the level of machine precision by projection onto their orthogonal complement and applying the low-synchronization Gram-Schmidt algorithms of Świrydowicz et al. [4]. The triangular matrix $T_k^{(1)}$ is an approximation of the matrix $(Q_k^TQ_k)^{-1}$ and two iterations result in a $T_k^{(2)}$ that is almost symmetric. Note that here, the matrix Q_k refers to the $A = QR$ factorization via Gram-Schmidt, whereas V_k refers to the orthogonal matrix produced in the Arnoldi-QR expansion. Giraud et al. [10] demonstrated how a rank-k correction could be applied in a post-processing step to recover orthogonality by computing the polar decomposition of Q_{k-1}, the matrix exhibited by Björck and Paige [11].

Our paper is organized as follows. Low-synchronization Gram-Schmidt algorithms for the QR factorization are reviewed in Section 2 and multiplicative iterations are applied to solve the normal equations. A rounding error analysis of the iterated Gauss-Seidel Gram-Schmidt algorithm is presented in Section 3, leading to bounds on the error and the orthogonality of the columns of V_{k-1}. In Section 4, we present the iterated Gauss Seidel GMRES algorithm and prove its backward stability and also derive a variant that requires only a single synchronization. Further, the relationship with Henrici's departure from normality is explored. Finally, numerical experiments on challenging problems studied over the past thirty-five years are presented in Section 5.

Notation. Lowercase bold letters denote column vectors and uppercase letters are matrices (e.g., $\mathbf{v}$ and A, respectively). We use A_{ij} to represent the (i,j) scalar entry of a matrix A, and $\mathbf{a}_k$ denotes the k–th column of A. A_k is a block partition up to the k–th column of a matrix. Subscripts indicate

the approximate solution and corresponding residual (e.g., $\mathbf{x}_k$ and $\mathbf{r}_k$) of an iterative method at step k. Throughout this article, the notation U_k (or L_k) and U_s (or L_s) will explicitly refer to *strictly upper/lower triangular matrices.*[1] Vector notation indicates a subset of the rows and/or columns of a matrix; e.g. $V_{1:m+1,1:k}$ denotes the first $m+1$ rows and k columns of the matrix V and the notation $V_{:,1:k}$ represents the first k columns of V. $H_{k+1,k}$ represents an $(k+1)\times k$ upper Hessenberg matrix, and $h_{k+1,k}$ refers to a matrix element. Bars denote computed quantities (e.g., $\bar{Q}_{k-1}$ or $\bar{R}_{k-1}$). In cases where standard notation in the literature is respected that may otherwise conflict with the aforementioned notation, this will be explicitly indicated. Note that the residual vector at iteration k is denoted $\mathbf{r}_k$, whereas $\mathbf{r}_{:,k}$ is the k–th column of the matrix R.

2. Low-synchronization Gram-Schmidt Algorithms. Krylov subspace methods for solving linear systems are often required for extreme-scale applications on parallel machines with many-core accelerators. Their strong-scaling is limited by the number and frequency of global reductions in the form of `MPI_Allreduce` operations and these communication patterns are expensive [12]. Low-synchronization algorithms are based on the ideas in Ruhe [6], and are designed such that they require only one reduction per iteration to normalize each vector and apply projections. The Gram-Schmidt projector applied to $\mathbf{a}_k$, the k-th column of the $n\times m$ matrix A in the factorization $A=QR$, can be written as

$$P_{k-1}\mathbf{a}_k = \mathbf{a}_k - Q_{k-1}\mathbf{r}_{1:k-1,k} = \mathbf{a}_k - Q_{k-1}\left(Q_{k-1}^TQ_{k-1}\right)^{-1}Q_{k-1}^T\mathbf{a}_k,$$

where the vector $\mathbf{r}_{1:k-1,k}$ is the solution of the normal equations

$$Q_{k-1}^TQ_{k-1}\mathbf{r}_{1:k-1,k} = Q_{k-1}^T\mathbf{a}_k. \tag{2.1}$$

Ruhe [6] established that the MGS algorithm employs a *multiplicative* Gauss-Seidel relaxation scheme with matrix splitting $Q_{k-1}^TQ_{k-1} = M_{k-1} - N_{k-1}$, where $M_{k-1} = I + L_{k-1}$ and $N_{k-1} = -L_{k-1}^T$. The iterated CGS-2 is an *additive* Jacobi relaxation.

The inverse compact WY form was derived in Świrydowicz et al. [4], with strictly lower triangular matrix L_{k-1}. Specifically, these inverse compact WY algorithms batch the inner-products together and compute the last row of L_{k-1} as

$$L_{k-1,1:k-2} = \mathbf{q}_{k-1}^T\,Q_{k-2}. \tag{2.2}$$

The approximate projector $P_{k-1}^{(1)}$ and correction matrix are given by

$$P_{k-1}^{(1)} = I - Q_{k-1}\,T_{k-1}^{(1)}\,Q_{k-1}^T, \quad T_{k-1}^{(1)} = (I+L_{k-1})^{-1} = M_{k-1}^{-1}, \tag{2.3}$$

respectively, and correspond to one iteration for the normal equations (2.1) with the zero vector as the initial guess. Iteration k of the Gram-Schmidt algorithm with two Gauss-Seidel iterations is given as Algorithm 2.1 . The compact and inverse compact WY forms of the correction matrix appearing in (2.3) can be derived as follows. The projector can be expressed as

$$\begin{aligned} P_{k-1}^{(1)} &= (\,I - \mathbf{q}_{k-1}\mathbf{q}_{k-1}^T\,)\,(\,I - Q_{k-2}\,T_{k-2}^{(1)}\,Q_{k-2}^T\,) \\ &= I - \mathbf{q}_{k-1}\mathbf{q}_{k-1}^T - Q_{k-2}\,T_{k-2}^{(1)}\,Q_{k-2}^T + \mathbf{q}_{k-1}\mathbf{q}_{k-1}^T\,Q_{k-2}\,T_{k-2}^{(1)}\,Q_{k-2}^T \end{aligned}$$

where $T_{k-1}^{(1)}$ is the lower triangular matrix of basis vector inner products. The WY representation is then given by the matrix form

$$P_{k-1}^{(1)} = I - \begin{bmatrix} Q_{k-2} & \mathbf{q}_{k-1} \end{bmatrix} \begin{bmatrix} T_{k-2}^{(1)} & 0 \\ -\mathbf{q}_{k-1}^T\,Q_{k-2}\,T_{k-2}^{(1)} & 1 \end{bmatrix} \begin{bmatrix} Q_{k-2}^T \\ \mathbf{q}_{k-1}^T \end{bmatrix} \tag{2.4}$$

For the inverse compact WY MGS, given the correction matrix $T_{k-1}^{(1)}$ generated by the recursion

$$(T_{k-1}^{(1)})^{-1} = I + L_{k-1} = \begin{bmatrix} (T_{k-2}^{(1)})^{-1} & 0 \\ \mathbf{q}_{k-1}^TQ_{k-2} & 1 \end{bmatrix}, \tag{2.5}$$

[1] We note that the distinction between these two notations is crucial. For U_k, the size of the strictly upper triangular matrix changes with k, whereas the size of U_s remains fixed.

it is possible to prove that

$$\begin{bmatrix} (T^{(1)})^{-1}_{k-2} & 0 \\ \mathbf{q}_{k-1}^T Q_{k-2} & 1 \end{bmatrix} \begin{bmatrix} T^{(1)}_{k-2} & 0 \\ -\mathbf{q}_{k-1}^T Q_{k-2} T^{(1)}_{k-2} & 1 \end{bmatrix} = \begin{bmatrix} I_{k-2} & 0 \\ 0 & 1 \end{bmatrix}. \tag{2.6}$$

Therefore, the lower triangular correction matrix $T^{(1)}_{k-1} = (I + L_{k-1})^{-1}$ from Świrydowicz et al. [4] is equivalent to the compact WY matrix L_1 from Björck [7, page 313].

To prove that the correction matrix $T^{(2)}_{k-1}$ is close to a symmetric matrix, consider the matrix

$$T^{(2)}_{k-1} = (I + L_{k-1})^{-1} - (I + L_{k-1})^{-1} L^T_{k-1} (I + L_{k-1})^{-1} = M^{-1}_{k-1} \left[I + N_{k-1} M^{-1}_{k-1} \right].$$

From this, we have $T^{(2)}_{k-1} = T^{(1)}_{k-1} - T^{(1)}_{k-1} L^T_{k-1} T^{(1)}_{k-1}$. From the block inverse (2.6),

$$T^{(1)}_{k-1} = \begin{bmatrix} T^{(1)}_{k-2} & 0 \\ -\mathbf{q}_{k-1}^T Q_{k-2} T^{(1)}_{k-2} & 1 \end{bmatrix}, \quad L^T_{k-1} = \begin{bmatrix} L^T_{k-2} & Q^T_{k-2} \mathbf{q}_{k-1} \\ 0 & 0 \end{bmatrix}.$$

By subtracting the matrices above, and dropping $\mathcal{O}(\varepsilon^2)$ terms, the block symmetric matrix

$$T^{(2)}_{k-1} \approx \begin{bmatrix} T^{(2)}_{k-2} & -Q^T_{k-2}\, \mathbf{q}_{k-1} \\ -\mathbf{q}^T_{k-1}\, Q_{k-2} & 1 \end{bmatrix} \tag{2.7}$$

is obtained and its singular values and eigenvalues remain close to one. However, as k increases $\sigma_{\max}(T^{(1)}_{k-1}) > 1$ and $\sigma_{\min}(T^{(1)}_{k-1}) < 1$, and additionally $T^{(1)}_{k-1}$ is non-normal. We provide definitions of the departure from normality in Section 4.

Algorithm 2.1 corresponds to MGS-2 (two passes of MGS) and is equivalent to two Gauss-Seidel iterations in exact arithmetic; see Ruhe [6]. Following the MATLAB notation, the algorithm at the k-th step generates QR factorization $A_{:,1:k} = Q_{:,1:k}\, R_{1:k,1:k}$, for $k = 1, \ldots,$ where $A_{:,1:k} = [\, A_{:,k-1},\, \mathbf{a}_k \,]$. The matrices thus have dimensions $A_{:,1:k}$: $n \times k$, $Q_{:,1:k}$: $n \times k$, and $R_{1:k,1:k}$: $k \times k$. Two initial steps prime a depth-two pipeline.

Algorithm 2.1 Inverse Compact WY Two-Reduce Gauss-Seidel Gram-Schmidt

1: $\mathbf{w}_1 := \mathbf{a}_1$, $R_{1,1} = \|\mathbf{w}_1\|_2$, $\mathbf{q}_1 := \mathbf{w}_1 / R_{1,1}$, $Q_1 = \mathbf{q}_1$
2: $R_{1,2} = \mathbf{q}_1^T\, \mathbf{a}_2$, $\mathbf{w}_2 := \mathbf{a}_2 - R_{1,2}\, \mathbf{q}_1$, $Q_2 = [\, Q_1,\, \mathbf{w}_2 \,]$
3: **for** $k = 3, \ldots, m$ **do**

4: $\quad [\, L^T_{1:k-2,k-1},\, \mathbf{r}^{(0)}_{1:k-1,k}] = [\, Q^T_{k-2}\, \mathbf{w}_{k-1} \;,\; Q^T_{k-1}\, \mathbf{a}_k \,]$ ▷ Global Synchronization
5: $\quad \gamma_{k-1} = \|\mathbf{w}_{k-1}\|_2$

6: $\quad \mathbf{q}_{k-1} = \mathbf{w}_{k-1} \,/\, \gamma_{k-1}$ ▷ Lagged Normalization
7: $\quad \mathbf{r}^{(0)}_{1:k-1,k} = \mathbf{r}^{(0)}_{1:k-1,k} \,/\, \gamma_{k-1}$
8: $\quad L_{k-1,1:k-2} = L_{k-1,1:k-2} \,/\, \gamma_{k-1}$

9: $\quad \mathbf{r}^{(1)}_{1:k-1,k} = (\, I + L_{k-1} \,)^{-1}\, \mathbf{r}^{(0)}_{1:k-1,k}$
10: $\quad \mathbf{u}_k = \mathbf{a}_k - Q_{k-1}\, \mathbf{r}^{(1)}_{1:k-1,k}$ ▷ First Gauss-Seidel

11: $\quad \mathbf{r}^{(2)}_{1:k-1,k} = Q^T_{k-1}\, \mathbf{u}_k$ ▷ Global Synchronization
12: $\quad \mathbf{r}^{(3)}_{1:k-1,k} = (\, I + L_{k-1} \,)^{-1}\, \mathbf{r}^{(2)}_{1:k-1,k}$
13: $\quad \mathbf{w}_k = \mathbf{u}_k - Q_{k-1}\, \mathbf{r}^{(3)}_{1:k-1,k}$ ▷ Second Gauss-Seidel

14: $\quad R_{1:k-1,k} = \mathbf{r}^{(1)}_{1:k-1,k} + \mathbf{r}^{(3)}_{1:k-1,k}$
15: $\quad Q_k = [\, Q_{k-1},\, \mathbf{w}_k \,]$

16: **end for**

The projection is computed after one iteration, where $\mathbf{r}^{(1)}_{1:k-1,k}$ is an approximate solution to the normal equations, and can be expressed as

$$\mathbf{u}_k = \mathbf{a}_k - Q_{k-1}\,\mathbf{r}^{(1)}_{1:k-1,k} = \mathbf{a}_k - Q_{k-1}\,M^{-1}_{k-1}\,\mathbf{r}^{(0)}_{1:k-1,k}. \tag{2.8}$$

After two iterations, the projection is given by

$$\mathbf{w}_k = \mathbf{a}_k - Q_{k-1}\mathbf{r}^{(1)}_{1:k-1,k} - Q_{k-1}M^{-1}_{k-1}N_{k-1}\mathbf{r}^{(1)}_{1:k-1,k}, \tag{2.9}$$

where $R_{k,k} = \|\mathbf{w}_k\|_2$ and $\mathbf{q}_k = \mathbf{w}_k/R_{k,k}$. The corresponding projector and correction matrix for two iterations are then found to be

$$P^{(2)}_{k-1} = I - Q_{k-1}T^{(2)}_{k-1}Q^T_{k-1}, \quad T^{(2)}_{k-1} = (I + L_{k-1})^{-1}[I - L^T_{k-1}(I + L_{k-1})^{-1}] = M^{-1}_{k-1}[I + N_{k-1}M^{-1}_{k-1}],$$

and the correction matrix is close to a symmetric matrix. The projection is applied across Steps 9 through 13 in Algorithm 2.1, and employs the following recurrence:

$$\begin{aligned} \mathbf{r}^{(0)}_{1:k-1,k} &= Q^T_{k-1}\mathbf{a}_k, \\ \mathbf{r}^{(1)}_{1:k-1,k} &= M^{-1}_{k-1}\,\mathbf{r}^{(0)}_{1:k-1,k}, \\ \mathbf{r}^{(1)}_{1:k-1,k} + \mathbf{r}^{(3)}_{1:k-1,k} &= \left[I + M^{-1}_{k-1}N_{k-1}\right]\mathbf{r}^{(1)}_{1:k-1,k}. \end{aligned} \tag{2.10}$$

It follows from (2.8) and (2.9) that the vectors $\mathbf{u}_k$ and $\mathbf{w}_k$ are related as

$$\mathbf{w}_k = \mathbf{u}_k - Q_{k-1}M^{-1}_{k-1}\,N_{k-1}\,\mathbf{r}^{(1)}_{1:k-1,k}. \tag{2.11}$$

The orthogonality of the vectors $\mathbf{u}_k$ and $\mathbf{w}_k$ with respect to columns of Q_{k-1} is given as

$$Q^T_{k-1}\mathbf{u}_k = Q^T_{k-1}\,\mathbf{a}_k - Q^T_{k-1}Q_{k-1}M^{-1}_{k-1}Q^T_{k-1}\mathbf{a}_k = N_{k-1}M^{-1}_{k-1}Q^T_{k-1}\mathbf{a}_k, \tag{2.12}$$

$$Q^T_{k-1}\mathbf{w}_k = Q^T_{k-1}\mathbf{u}_k - Q^T_{k-1}Q_{k-1}M^{-1}_{k-1}N_{k-1}M^{-1}_{k-1}Q^T_{k-1}\mathbf{a}_k = (N_{k-1}M^{-1}_{k-1})^2Q^T_{k-1}\mathbf{a}_k. \tag{2.13}$$

3. Rounding error analysis. We now present a rounding error analysis of the low-synchronization Gram-Schmidt algorithm with two Gauss-Seidel iterations. Björck [13] and Björck and Paige [7] analyzed the traditional MGS algorithm. Our approach is more akin to the analysis of the CGS-2 algorithm presented by Giraud et al. [14]. We begin with the derivation of the formulas for the computed quantities in the QR factorization by invoking an induction argument. In particular, the triangular solves with $M_{k-1} = I + L_{k-1}$ implied by Step 9 and Step 12 of Algorithm 2.1 are backward stable as shown by Higham [15], where

$$(M_{k-1} + E_{k-1})\,\mathbf{x} = \mathbf{b}, \quad \|E_{k-1}\|_2 \leq \mathcal{O}(\varepsilon)\,\|M_{k-1}\|_2.$$

The associated error terms and bounds for the recurrence relations in Algorithm 2.1 are then given as

$$\begin{aligned} \bar{\mathbf{r}}^{(0)}_{1:k-1,k} &= \bar{Q}^T_{k-1}\mathbf{a}_k + \mathbf{e}^{(0)}_k, & \|\mathbf{e}^{(0)}_k\|_2 &\leq \mathcal{O}(\varepsilon)\|\bar{Q}_{k-1}\|_2\,\|\mathbf{a}_k\|_2, \\ (M_{k-1} + E^{(1)}_{k-1})\,\bar{\mathbf{r}}^{(1)}_{1:k-1,k} &= \bar{\mathbf{r}}^{(0)}_{1:k-1,k}, & \|E^{(1)}_{k-1}\|_2 &\leq \mathcal{O}(\varepsilon)\|M_{k-1}\|_2, \\ \bar{\mathbf{r}}^{(2)}_{1:k-1,k} &= \bar{Q}^T_{k-1}\bar{\mathbf{u}}_k + \mathbf{e}^{(2)}_k, & \|\mathbf{e}^{(2)}_k\|_2 &\leq \mathcal{O}(\varepsilon)\|\bar{Q}_{k-1}\|_2\,\|\bar{\mathbf{u}}_k\|_2, \\ (M_{k-1} + E^{(2)}_{k-1})\,\bar{\mathbf{r}}^{(3)}_{1:k-1,k} &= \bar{\mathbf{r}}^{(2)}_{1:k-1,k}, & \|E^{(2)}_{k-1}\|_2 &\leq \mathcal{O}(\varepsilon)\|M_{k-1}\|_2. \end{aligned}$$

Throughout the entire paper, we employ the $\mathcal{O}(\varepsilon)$ notation, as in Paige and Strakoš [3], which denotes a small multiple of machine precision where the proportionality constant is a low degree polynomial in the matrix dimension n, and the iteration $k \ll n$. Our analysis is based on standard assumptions on the computer arithmetic and these parameters as used in Higham [16]. The computed coefficient vector $\bar{\mathbf{r}}^{(1)}_{1:k-1,k}$ satisfies

$$\bar{\mathbf{r}}^{(1)}_{1:k-1,k} = (M_{k-1} + E^{(1)}_{k-1})^{-1}\,\bar{\mathbf{r}}^{(0)}_{1:k-1,k}$$

and its 2-norm satisfies the inequality

$$\begin{aligned}\|\bar{\mathbf{r}}^{(1)}_{1:k-1,k}\|_2 &\leq \|(M_{k-1}+E^{(1)}_{k-1})^{-1}\|\,\|\bar{Q}^T_{k-1}\mathbf{a}_k+\mathbf{e}^{(0)}_k\|_2\\ &\leq \frac{1+\mathcal{O}(\varepsilon)}{\sigma_{k-1}(M_{k-1})-\|E^{(1)}_{k-1}\|_2}\|\bar{Q}_{k-1}\|_2\,\|\mathbf{a}_k\|_2\\ &\leq \frac{(\,1+\mathcal{O}(\varepsilon)\,)\,\|M^{-1}_{k-1}\|_2}{1-\mathcal{O}(\varepsilon)\kappa(M_{k-1})}\|\bar{Q}_{k-1}\|_2\,\|\mathbf{a}_k\|_2\end{aligned}$$

assuming that $\mathcal{O}(\varepsilon)\kappa(M_{k-1}) < 1$. The computed intermediate coefficient vector $\bar{\mathbf{r}}^{(3)}_{1:k-1,k}$ satisfies the formula

$$\bar{\mathbf{r}}^{(3)}_{1:k-1,k} = (M_{k-1}+E^{(2)}_{k-1})^{-1}\,\bar{\mathbf{r}}^{(2)}_{1:k-1,k},$$

and an upper bound is given by

$$\begin{aligned}\|\bar{\mathbf{r}}^{(3)}_{1:k-1,k}\|_2 &\leq \|(M_{k-1}+E^{(2)}_{k-1})^{-1}\|_2\,\|\bar{Q}^T_{k-1}\bar{\mathbf{u}}_k+\mathbf{e}^{(2)}_k\|_2\|_2\\ &\leq \frac{(1+\mathcal{O}(\varepsilon))\,\|M^{-1}_{k-1}\|_2}{1-\mathcal{O}(\varepsilon)\kappa(M_{k-1})}\|\bar{Q}_{k-1}\|_2\,\|\bar{\mathbf{u}}_k\|_2.\end{aligned}$$

The computed vector $\bar{\mathbf{u}}_k$ in Step 10 of Algorithm 2.1 satisfies

$$\bar{\mathbf{u}}_k = \mathbf{a}_k - \bar{Q}_{k-1}\,\bar{\mathbf{r}}^{(1)}_{1:k-1,k} + \mathbf{e}^{(1)}_k, \tag{3.1}$$

$$\|\mathbf{e}^{(1)}_k\|_2 \leq \mathcal{O}(\varepsilon)\left[\,\|\mathbf{a}_k\|_2 + \|\bar{Q}_{k-1}\|_2\,\|\bar{\mathbf{r}}^{(1)}_{1:k-1,k}\|_2\,\right]. \tag{3.2}$$

Similarly, the computed form of the projection in Step 13 of Algorithm 2.1 satisfies

$$\bar{\mathbf{w}}_k = \bar{\mathbf{u}}_k - \bar{Q}_{k-1}\,\bar{\mathbf{r}}^{(3)}_{1:k-1,k} + \mathbf{e}^{(3)}_k, \tag{3.3}$$

$$\|\mathbf{e}^{(3)}_k\|_2 \leq \mathcal{O}(\varepsilon)\left[\,\|\bar{\mathbf{u}}_k\|_2 + \|\bar{Q}_{k-1}\|_2\,\|\bar{\mathbf{r}}^{(3)}_{1:k-1,k}\|_2\,\right]. \tag{3.4}$$

Summarizing (3.1) and (3.3), the lower bound for the norm of $\bar{\mathbf{u}}_k$ is now determined from the recurrences for the computed quantities

$$A_{k-1} = \bar{Q}_{k-1}\left(\bar{R}^{(1)}_{k-1}+\bar{R}^{(3)}_{k-1}\right) - F_{k-1},\ F_{k-1} = \left[\,\mathbf{e}^{(1)}_1+\mathbf{e}^{(3)}_1 \quad \ldots, \quad \mathbf{e}^{(1)}_{k-1}+\mathbf{e}^{(3)}_{k-1}\,\right], \tag{3.5}$$

$$\mathbf{a}_k = \bar{Q}_{k-1}\,\bar{\mathbf{r}}^{(1)}_{1:k-1,k} + \bar{\mathbf{u}}_k - \mathbf{e}^{(1)}_k, \tag{3.6}$$

where A_{k-1} is given as $A_{k-1} = [\mathbf{a}_1, \ldots, \mathbf{a}_{k-1}]$. Combining these into an augmented matrix form, we obtain

$$A_k = \left[\,A_{k-1}, \quad \mathbf{a}_k\,\right] = \bar{Q}_{k-1}\left[\,\bar{R}^{(1)}_{k-1}+\bar{R}^{(3)}_{k-1}, \quad \bar{\mathbf{r}}^{(1)}_{1:k-1,k}\,\right] - \left[\,F_{k-1}, \quad \mathbf{e}^{(1)}_k-\bar{\mathbf{u}}_k\,\right], \tag{3.7}$$

which results in the bound

$$\left\|\left[\,F_{k-1}, \quad \mathbf{e}^{(1)}_k\,\right]\right\|_2 + \|\bar{\mathbf{u}}_k\|_2 \geq \left\|\left[\,F_{k-1}, \quad \mathbf{e}^{(1)}_k-\bar{\mathbf{u}}_k\,\right]\right\|_2 \geq \sigma_{\min}(A_k). \tag{3.8}$$

Thus, the 2-norm of the projected vector $\bar{\mathbf{u}}_k$ appearing in (3.3) is bounded below as

$$\|\bar{\mathbf{u}}_k\|_2 \geq \sigma_{\min}(A_k) - \left\|\left[\,F_{k-1}, \quad \mathbf{e}^{(1)}_k\,\right]\right\|_2. \tag{3.9}$$

Invoking the inequalities (3.1), (3.3) and the upper bounds on $\|\bar{\mathbf{r}}^{(1)}_{1:k-1,k}\|_2$ and $\|\bar{\mathbf{r}}^{(3)}_{1:k-1,k}\|_2$, together with the bounds $\|\bar{\mathbf{u}}_k\|_2 \approx \|\mathbf{a}_k\|_2 \leq \|A_k\|_2$, it follows that

$$\left\|\left[\,F_{k-1}, \quad \mathbf{e}^{(1)}_k\,\right]\right\|_2 \leq \mathcal{O}(\varepsilon)\frac{(\,1+\mathcal{O}(\varepsilon)\,)\,\|M^{-1}_{k-1}\|_2}{(1-\mathcal{O}(\varepsilon)\kappa(M_{k-1}))^2}\|\bar{Q}_{k-1}\|_2^2\,\|A_k\|_2. \tag{3.10}$$

In order to derive an upper bound for $\|\bar{Q}_{k-1}^T \bar{\mathbf{w}}_k\|_2 / \|\bar{\mathbf{w}}_k\|_2$, assume inductively that $\|N_{k-1}\|_2 \leq \mathcal{O}(\varepsilon)$. Note that the term $\mathcal{O}(\varepsilon)$ depends on the iteration number k. Then it is evident that $\|\bar{Q}_{k-1}\|^2 \leq 1+\mathcal{O}(\varepsilon)$ and $\|M_{k-1}\|_2 \leq 1 + \mathcal{O}(\varepsilon)$, with $\sigma_{\min}(M_{k-1}) \geq 1 - \mathcal{O}(\varepsilon)$. Consequently, it follows that

$$\kappa(M_{k-1}) \leq \frac{1+\mathcal{O}(\varepsilon)}{1-\mathcal{O}(\varepsilon)}.$$

The term $\bar{Q}_{k-1}^T\, \bar{\mathbf{u}}_k$ can be written in the form

$$\begin{aligned}
\bar{Q}_{k-1}^T \bar{\mathbf{u}}_k &= \bar{Q}_{k-1}^T \mathbf{a}_k - \bar{Q}_{k-1}^T \bar{Q}_{k-1} \bar{\mathbf{r}}_{1:k-1,k}^{(1)} - \bar{Q}_{k-1}^T\, \mathbf{e}_k^{(1)} \\
&= \bar{Q}_{k-1}^T \mathbf{a}_k - (M_{k-1} - N_{k-1})(M_{k-1} + E_{k-1}^{(1)})^{-1}\, \bar{\mathbf{r}}_{1:k-1,k}^{(0)} - \bar{Q}_{k-1}^T\, \mathbf{e}_k^{(1)} \\
&= (N_{k-1} + E_{k-1}^{(1)})(M_{k-1} + E_{k-1}^{(1)})^{-1}\, \bar{\mathbf{r}}_{1:k-1,k}^{(0)} - \bar{Q}_{k-1}^T\, \mathbf{e}_k^{(1)} - \mathbf{e}_k^{(0)}.
\end{aligned}$$

Thus, the quotient $\|\, \bar{Q}_{k-1}^T\, \bar{\mathbf{u}}_k \,\|_2 \,/\, \|\, \bar{\mathbf{u}}_k \,\|_2$ can be bounded as

$$\frac{\|\bar{Q}_{k-1}^T \bar{\mathbf{u}}_k \,\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} \leq \|N_{k-1} + E_{k-1}^{(1)}\|_2 \|(M_{k-1} + E_{k-1}^{(1)})^{-1}\|_2 \frac{\|\bar{\mathbf{r}}_{1:k-1,k}^{(0)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} + \frac{\|\bar{Q}_{k-1}^T \mathbf{e}_k^{(1)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} + \frac{\|\mathbf{e}_k^{(0)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2}, \tag{3.11}$$

and therefore

$$\frac{\|\bar{Q}_{k-1}^T \bar{\mathbf{u}}_k \,\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} \leq \left[\frac{(1+\mathcal{O}(\varepsilon))\|M_{k-1}^{-1}\|_2}{1-\mathcal{O}(\varepsilon)\kappa(M_{k-1})} \right] \|N_{k-1}\|_2 \frac{\|\bar{\mathbf{r}}_{1:k-1,k}^{(0)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} + \frac{\|\bar{Q}_{k-1}^T \mathbf{e}_k^{(1)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} + \frac{\|\mathbf{e}_k^{(0)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2}. \tag{3.12}$$

The vector 2-norm $\|\bar{\mathbf{r}}_{1:k-1,k}^{(0)}\|_2$ can be bounded according to

$$\|\bar{\mathbf{r}}_{1:k-1,k}^{(0)}\|_2 \leq (1+\mathcal{O}(\varepsilon))\ \|\bar{Q}_{k-1}\|_2\, \|\mathbf{a}_k\|_2, \tag{3.13}$$

and together with (3.9) and (3.10)

$$\frac{\|\bar{\mathbf{r}}_{1:k-1,k}^{(0)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} \lesssim \frac{\kappa(A_k)}{1-\mathcal{O}(\varepsilon)\kappa(A_k)}$$

under the assumption that $\mathcal{O}(\varepsilon)\kappa(A_k) < 1$. After one Gauss-Seidel iteration, the bound for the loss of orthogonality is proportional to

$$\frac{\|\bar{Q}_{k-1}^T\, \bar{\mathbf{u}}_k \,\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} \lesssim \frac{\mathcal{O}(\varepsilon)\kappa(A_k)}{1-\mathcal{O}(\varepsilon)\kappa(A_k)}. \tag{3.14}$$

This corresponds to the upper bound on the loss of orthogonality originally derived by Björck [13] and further refined by Björck and Paige [11] for the MGS algorithm.

Given (3.3), the projected vector is expanded as

$$\begin{aligned}
\bar{Q}_{k-1}^T \bar{\mathbf{w}}_k &= \bar{Q}_{k-1}^T \bar{\mathbf{u}}_k - \bar{Q}_{k-1}^T \bar{Q}_{k-1}\, \bar{\mathbf{r}}_{1:k-1,k}^{(3)} - \bar{Q}_{k-1}^T\, \mathbf{e}_k^{(3)} \\
&= \bar{Q}_{k-1}^T\, \bar{\mathbf{u}}_k - (M_{k-1} - N_{k-1})(M_{k-1} + E_{k-1}^{(2)})^{-1}\, \bar{\mathbf{r}}_{1:k-1,k}^{(2)} - \bar{Q}_{k-1}^T\, \mathbf{e}_k^{(3)} \\
&= (N_{k-1} + E_{k-1}^{(2)})(M_{k-1} + E_{k-1}^{(2)})^{-1}\, \bar{\mathbf{r}}_{1:k-1,k}^{(2)} - \bar{Q}_{k-1}^T\, \mathbf{e}_k^{(3)} - \mathbf{e}_k^{(2)}.
\end{aligned}$$

Therefore, the quotient $\|\bar{Q}_{k-1}^T\, \bar{\mathbf{w}}_k \,\|_2 \,/\, \|\bar{\mathbf{w}}_k \,\|_2$ is bounded as

$$\begin{aligned}
\frac{\|\bar{Q}_{k-1}^T \bar{\mathbf{w}}_k \,\|_2}{\|\bar{\mathbf{w}}_k \,\|_2} &\leq \|N_{k-1} + E_{k-1}^{(2)}\|_2 \|(M_{k-1} + E_{k-1}^{(2)})^{-1}\|_2 \frac{\|\bar{\mathbf{r}}_{1:k-1,k}^{(2)}\|_2}{\|\bar{\mathbf{w}}_k \,\|_2} \\
&\quad + \frac{\|\bar{Q}_{k-1}^T\, \mathbf{e}_k^{(3)}\|_2}{\|\bar{\mathbf{w}}_k \,\|_2} + \frac{\|\mathbf{e}_k^{(2)}\|_2}{\|\bar{\mathbf{w}}_k \,\|_2} \\
&\leq \left[\|N_{k-1} + E_{k-1}^{(2)}\|_2 \|(M_{k-1} + E_{k-1}^{(2)})^{-1}\|_2 \frac{\|\bar{\mathbf{r}}_{1:k-1,k}^{(2)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} \right. \\
&\quad \left. + \frac{\|\bar{Q}_{k-1}^T\, \mathbf{e}_k^{(3)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} + \frac{\|\mathbf{e}_k^{(2)}\|_2}{\|\bar{\mathbf{u}}_k \,\|_2} \right] \times \frac{\|\bar{\mathbf{u}}_k\|_2}{\|\bar{\mathbf{w}}_k\|_2}.
\end{aligned}$$

The term $\|\bar{\mathbf{r}}^{(2)}_{1:k-1,k}\|_2$ satisfies the bound

$$\|\mathbf{e}^{(2)}_k\| = \|\bar{\mathbf{r}}^{(2)}_{1:k-1,k} - \bar{Q}^T_{k-1}\,\bar{\mathbf{u}}_k\|_2 \leq \mathcal{O}(\varepsilon)\|\bar{Q}_{k-1}\|_2\|\bar{\mathbf{u}}_k\|_2. \tag{3.15}$$

After two Gauss-Seidel iterations, the essential orthogonality relation is obtained as

$$\begin{aligned}\frac{\|\bar{Q}^T_{k-1}\bar{\mathbf{w}}_k\,\|_2}{\|\bar{\mathbf{w}}_k\,\|_2} \leq \Bigg[&\frac{(1+\mathcal{O}(\varepsilon))\|M^{-1}_{k-1}\|_2}{1-\mathcal{O}(\varepsilon)\kappa(M_{k-1})}\,\|N_{k-1}\|_2\frac{\|\bar{Q}^T_{k-1}\,\bar{\mathbf{u}}_k\|_2 + \|\mathbf{e}^{(2)}_k\|_2}{\|\bar{\mathbf{u}}_k\,\|_2} \\ &+ \frac{\|\bar{Q}^T_{k-1}\mathbf{e}^{(3)}_k\|_2}{\|\bar{\mathbf{u}}_k\,\|_2} + \frac{\|\mathbf{e}^{(2)}_k\|_2}{\|\bar{\mathbf{u}}_k\,\|_2}\Bigg] \times \frac{\|\bar{\mathbf{u}}_k\|_2}{\|\bar{\mathbf{w}}_k\|_2}.\end{aligned} \tag{3.16}$$

By considering (3.3), an upper bound for the ratio $\|\bar{\mathbf{u}}_k\|_2/\|\bar{\mathbf{w}}_k\|_2$ is obtained as

$$\frac{\|\bar{\mathbf{w}}_k\|_2}{\|\bar{\mathbf{u}}_k\|_2} \geq \frac{\|\bar{\mathbf{u}}_k\|_2}{\|\bar{\mathbf{u}}_k\|_2} - \|\bar{Q}_{k-1}\|_2\frac{\|\bar{\mathbf{r}}^{(3)}_{1:k-1,k}\|_2}{\|\bar{\mathbf{u}}_k\|_2} - \frac{\|\mathbf{e}^{(3)}_k\|_2}{\|\bar{\mathbf{u}}_k\|_2}, \tag{3.17}$$

and substituting the upper bound for $\|\bar{\mathbf{r}}^{(3)}_{1:k-1,k}\|_2$ this can be rewritten as

$$\frac{\|\bar{\mathbf{u}}_k\|_2}{\|\bar{\mathbf{w}}_k\|_2} \leq \frac{1}{1-\mathcal{O}(\varepsilon)\kappa(A_k)}. \tag{3.18}$$

An analogous bound was derived in Giraud et al. [14, eq. 3.33] for the classical Gram-Schmidt algorithm (CGS-2) with reorthogonalization. Thus, assuming $\mathcal{O}(\varepsilon)\kappa(A_k) < 1$ and given Björck's result (3.14), it follows that the loss of orthogonality is maintained at the level of machine precision:

$$\|\bar{Q}^T_{k-1}\,\bar{\mathbf{q}}_k\|_2 \lesssim \mathcal{O}(\varepsilon).$$

4. Iterated Gauss-Seidel GMRES. The GMRES algorithm of Saad and Schultz [1] has a thirty-five year history and various alternative formulations of the basic algorithm have been proposed over that time frame. A comprehensive review of these is presented by Zou [17]. In particular, pipelined, s-step, and block algorithms are better able to hide latency in parallel implementations; see, e.g., Yamazaki et al. [18]. The low-synchronization MGS-GMRES algorithm described in Świrydowicz et al. [4] improves parallel strong-scaling by employing one global reduction for each GMRES iteration; see Lockhart et al. [12]. A review of compact WY Gram Schmidt algorithms and their computational costs is given in [5]. Block generalizations of the DGCS-2 and CGS-2 algorithm are presented in Carson et al. [19, 20]. In [20] the authors generalize the Pythagorean theorem to block form and derive `BCGS-PIO` and `BCGS-PIP` algorithms with the more favorable communication patterns described herein. An analysis of the backward stability of the these block Gram-Schmidt algorithms is also presented. Low-synch iterated Gauss-Seidel GMRES algorithms are now presented with one and two global reductions.

The low-synchronization DCGS-2 algorithm introduced by Świrydowicz [4] was employed to compute the Arnoldi-QR expansion in the Krylov-Schur eigenvalue algorithm by Bielich et al. [5]. The algorithm exhibits desirable numerical characteristics including the computation of invariant subspaces of maximum size for the Krylov-Schur algorithm of Stewart [21]. In the case of iterated Gauss-Seidel GMRES, the backward error analysis derived for the two-reduce Gauss-Seidel Gram-Schmidt Algorithm 2.1 can be applied to the IGS-GMRES algorithms. In the case of the DCGS-2 algorithm, the symmetric correction matrix T_{k-1} was derived in Appendix 1 of [4] and is given by $T_{k-1} = I - L_{k-1} - L^T_{k-1}$. This correction matrix was employed in s-step and pipelined MGS-GMRES. When the matrix T_{k-1} is split into $I - L_{k-1}$ and L^T_{k-1} and applied across two iterations of the DCGS-2 algorithm, the resulting loss of orthogonality is empirically observed to be $\mathcal{O}(\varepsilon)$. Indeed, it was noted in Bielich et al. [5] that two iterations of classical Gram-Schmidt (CGS) are needed to achieve vectors orthogonal to the level $\mathcal{O}(\varepsilon)$.

Two-Reduce IGS-GMRES. The iterated Gauss-Seidel GMRES algorithm (IGS-GMRES) presented in Algorithm 4.1 requires computing $\mathbf{v}_k$ in Step 4 and a norm in Step 6, followed by vector scaling in Step 7. Two Gauss-Seidel iterations are applied in Step 13 and 16. The normalization for the Krylov

vector $\mathbf{v}_k$ at iteration k represents the delayed scaling of the vector $\mathbf{w}_{k-1}$ in the matrix-vector product $\mathbf{v}_k = A\mathbf{w}_{k-1}$. Therefore, an additional Step 8 is required, together with Step 9. The subdiagonal element γ_{k-1} in the Arnoldi-QR expansion (1.1) is computed in Step 6 and the remaining entries $H_{1:k-1,k-1}$ are computed after the second Gauss-Seidel iteration in Step 16 of Algorithm 4.1. In order to lag the normalization to the next iteration, the norm $\|\mathbf{w}_{k-1}\|_2$ is included in the global reduction in Steps 5 and 6.

Using MATLAB notation, the algorithm denoted as IGS-GMRES at the k-th iteration step computes the Arnoldi expansion $A\,V_{:,1:k-1} = V_{:,1:k}\,H_{1:k,1:k-1}$. A lagged normalization leads to a pipeline of depth two. Thus, two initial iteration steps prime the pipeline.

Algorithm 4.1 Low-Synchronization Two-Reduce Iterated Gauss-Seidel GMRES

1: $\mathbf{r}_0 = b - A\,\mathbf{x}_0$, $\rho = \|\mathbf{r}_0\|_2$, $\mathbf{w}_1 = \mathbf{r}_0$, $\mathbf{v}_1 = \mathbf{w}_1/\rho$
2: $\mathbf{w}_2 = A\mathbf{v}_1$, $H_{1,1} = \mathbf{v}_1^T\,\mathbf{w}_2$, $\mathbf{w}_2 = \mathbf{w}_2 - H_{1,1}\,\mathbf{v}_1$, $V_2 = [\,V_1,\ \mathbf{w}_2\,]$
3: **for** $k = 3, \ldots n$ **do**

4: $\quad\mathbf{v}_k = A\mathbf{w}_{k-1}$
5: $\quad[\,L_{1:k-2,k-1}^T,\ \mathbf{r}_{1:k-1,k}^{(0)}\,] = [\,V_{k-2}^T\,\mathbf{w}_{k-1},\ V_{k-1}^T\,\mathbf{v}_k\,]$ ▷ Global Synchronization
6: $\quad\gamma_{k-1} = \|\mathbf{w}_{k-1}\|_2$
7: $\quad\mathbf{v}_{k-1} = \mathbf{w}_{k-1}\,/\,\gamma_{k-1}$

8: $\quad\mathbf{r}_{1:k-1,k}^{(0)} = \mathbf{r}_{1:k-1,k}^{(0)}\,/\,\gamma_{k-1}$ ▷ Scale for Arnoldi
9: $\quad\mathbf{r}_{k-1,k}^{(0)} = \mathbf{r}_{k-1,k}^{(0)}\,/\,\gamma_{k-1}$
10: $\quad L_{k-1,1:k-2} = L_{k-1,1:k-2}\,/\,\gamma_{k-1}$
11: $\quad\mathbf{v}_k = \mathbf{v}_k\,/\,\gamma_{k-1}$

12: $\quad\mathbf{r}_{1:k-1,k}^{(1)} = (I + L_{k-1})^{-1}\,\mathbf{r}_{1:k-1,k}^{(0)}$
13: $\quad\mathbf{u}_k = \mathbf{v}_k - V_{k-1}\,\mathbf{r}_{1:k-1,k}^{(1)}$ ▷ First Gauss-Seidel

14: $\quad\mathbf{r}_{1:k-1,k}^{(2)} = V_{k-1}^T\,\mathbf{u}_k$ ▷ Global Synchronization
15: $\quad\mathbf{r}_{1:k-1,k}^{(3)} = (I + L_{k-1})^{-1}\,\mathbf{r}_{1:k-1,k}^{(2)}$
16: $\quad\mathbf{w}_k = \mathbf{u}_k - V_{k-1}\,\mathbf{r}_{1:k-1,k}^{(3)}$ ▷ Second Gauss-Seidel

17: $\quad H_{1:k-1,k-1} = \mathbf{r}_{1:k-1,k}^{(1)} + \mathbf{r}_{1:k-1,k}^{(3)}$
18: $\quad V_k = [\,V_{k-1},\ \mathbf{w}_k\,]$

19: **end for**

For the least squares solution, we solve $\mathbf{y}_m = \text{argmin} \parallel H_m\mathbf{y}_m - \rho\,\mathbf{e}_1 \parallel_2$ and then compute $\mathbf{x}_m = \mathbf{x}_0 + V_m\mathbf{y}_m$. The subdiagonal element γ_{k-1} is computed as $\|\mathbf{w}_{k-1}\|_2$ in Algorithm 4.1. To reduce the number of global synchronizations the recurrence

$$\|\mathbf{w}_{k-1}\|_2^2 = \parallel \mathbf{u}_{k-1} \parallel_2^2 - \parallel \mathbf{r}_{1:k-2,k-1}^{(2)} \parallel_2^2 \tag{4.1}$$

would have to be employed. Therefore, we have

$$\|\mathbf{w}_{k-1}\|_2 \geq \|\mathbf{u}_{k-1}\|_2 - \|\mathbf{r}_{1:k-2,k-1}^{(2)}\|_2.$$

This means that the magnitude of the vector $\mathbf{w}_{k-1}$ is at least as large as the difference between the magnitudes of the vectors $\mathbf{u}_{k-1}$ and $\mathbf{r}_{1:k-2,k-1}^{(3)}$. Because $\|\mathbf{r}_{1:k-2,k-1}^{(3)}\|_2$ is small, it follows that $\|\mathbf{w}_{k-1}\|_2$ is approximately equal to $\|\mathbf{u}_{k-1}\|_2$. Thus, if $\|\mathbf{r}_{1:k-2,k-1}^{(3)}\|_2$ is small, then the inequality is a tight bound, and $\|\mathbf{w}_{k-1}\|_2$ is approximately equal to $\|\mathbf{u}_{k-1}\|_2$. Therefore, we can say that $\|Q_{k-1}^T\mathbf{q}_k\|_2 = \|\,|Q_{k-1}^T\mathbf{w}_k\|_2/\|\mathbf{w}_k\|_2$ is $\mathcal{O}(\varepsilon)$.

Hybrid MGS-CGS GMRES. We now derive a new variant of Algorithm 4.1 that requires only a single synchronization point. The vector $\mathbf{w}_{k-1}$ is available at Step 4 of Algorithm 4.1. Alternatively, the vector $\mathbf{v}_k = A\mathbf{u}_{k-1}$ can be computed during the Gauss-Seidel iteration in Step 13, by replacing Step 11 in Algorithm 4.1, where $\mathbf{v}_k = A\mathbf{w}_{k-1} \,/\, \gamma_{k-1}$, and $\mathbf{w}_{k-1} = \mathbf{u}_{k-1} - V_{k-2}\, V_{k-2}^T\, \mathbf{u}_{k-1}$, i.e.,

$$\mathbf{v}_k = \frac{1}{\gamma_{k-1}} \left[A\mathbf{u}_{k-1} - V_{k-1} H_{1:k-1,1:k-2}\, \mathbf{r}^{(2)}_{1:k-2,k-1} \right]. \tag{4.2}$$

The recurrence for Step 11 is instead written as $\mathbf{u}_k = A\,\mathbf{v}_{k-1} - V_{k-1} V_{k-1}^T A\,\mathbf{v}_{k-1}$. It follows that

$$\begin{aligned}\mathbf{u}_k = &\frac{1}{\gamma_{k-1}} \left[A\,\mathbf{u}_{k-1} - V_{k-2}\, V_{k-2}^T\, A\,\mathbf{u}_{k-1} \right] - \frac{1}{\gamma_{k-1}} \mathbf{v}_{k-1}\, \mathbf{v}_{k-1}^T\, A\,\mathbf{u}_{k-1} \\ &+ \frac{1}{\gamma_{k-1}}\, V_{k-1} \left(I - V_{k-1}^T\, V_{k-1} \right)\, H_{1:k-1,1:k-2}\, \mathbf{r}^{(3)}_{1:k-2,k-1}.\end{aligned}$$

Noting that $\mathbf{r}^{(0)}_{1:k-2,k} = V_{k-2}^T\, A\,\mathbf{u}_{k-1}$, it is possible to compute $\mathbf{r}^{(1)}_{k-1,k}$ as follows:

$$\begin{aligned}\mathbf{v}_{k-1}^T\, A\,\mathbf{u}_{k-1} &= \frac{1}{\gamma_{k-1}} \left[\mathbf{u}_{k-1} - V_{k-2}\, \mathbf{r}^{(2)}_{1:k-2,k-1} \right]^T A\,\mathbf{u}_{k-1} \\ &= \frac{1}{\gamma_{k-1}} \left[\mathbf{r}^{(0)}_{k-1,k} - L_{k-1,1:k-2}\, \mathbf{r}^{(0)}_{1:k-2,k} \right],\end{aligned}$$

where $L_{k-1,1:k-2} = \mathbf{u}_{k-1}^T\, V_{k-2}$. This is "Stephen's trick" from Świrydowicz et al. [4, eq. 4] and Bielich et al. [5] applied to the Arnoldi-QR algorithm. The first projection step is then applied in Steps 9 and 11 using the compact WY form given in equation (2.4), i.e.,

$$\mathbf{u}_k = A\,\mathbf{u}_{k-1} \,/\, \gamma_{k-1} - \begin{bmatrix} V_{k-2} & \mathbf{v}_{k-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\gamma_{k-1}^{-1}\, L_{k-1,1:k-2} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{(0)}_{1:k-2,k} \\ \mathbf{r}^{(0)}_{k-1,k} \end{bmatrix}, \tag{4.3}$$

where the implied triangular inverse simplifies to $(\, I + L_{k-1} \,)^{-1} = I - L_{k-1}$, only when the last row contains non-zero off-diagonal elements, as in Step 12 of Algorithm 4.2. The correction matrix then takes the simplified form

$$T^{(1)}_{k-1} = \begin{bmatrix} I & 0 \\ -\mathbf{v}_{k-1}^T\, V_{k-2} & 1 \end{bmatrix},$$

where the spectral radius ρ_{k-1} of the matrix $M_{k-1}^{-1}\, N_{k-1}$ is identical to that of the two-reduce algorithm. The projection matrix is given by

$$\begin{aligned}P^{(1)}_{k-1} &= (\, I - \mathbf{v}_{k-1}\mathbf{v}_{k-1}^T \,)\, (\, I - V_{k-2}\, T^{(1)}_{k-1}\, V_{k-2}^T \,) \\ &= I - \mathbf{v}_{k-1}\mathbf{v}_{k-1}^T - V_{k-2}\, V_{k-2}^T + \mathbf{v}_{k-1}\mathbf{v}_{k-1}^T\, V_{k-2}\, V_{k-2}^T.\end{aligned}$$

After substitution of this expression, it follows that

$$\mathbf{u}_k = \frac{1}{\gamma_{k-1}} \left[A\mathbf{u}_{k-1} - V_{k-1}\, \mathbf{r}^{(1)}_{1:k-1,k} \right].$$

The one-synch hybrid MGS-CGS GMRES algorithm is presented in Algorithm 4.2. The algorithm can be characterized by an MGS iteration at Step 13, combined with a lagged CGS iteration at Step 10. The update of the Hessenberg matrix in Step 14 is the same as earlier. We note that our backward stability analysis in the following section only applied to IGS-GMRES, Algorithm 4.1, with two global synchronizations. A complete stability analysis of Algorithm 4.2 remains future work.

The triangular solve and matrix-vector multiply for the multiplicative iterations require $(k-1)^2$ flops at iteration k and thus lead to a slightly higher operation count versus the traditional MGS algorithm, which is $2m^2n$ for an $n \times m$ matrix. The matrix-vector multiplies in Step 4 of Algorithm 4.2 have complexity $4nk$ at iteration k and the norm in Step 6 requires $2n$ flops, for a total of $4mn^2 + 3/3n^3 + \mathcal{O}(mn)$

Algorithm 4.2 Low-Synch One-Reduce hybrid MGS-CGS GMRES

1: $\mathbf{r}_0 = b - A\,\mathbf{x}_0$, $\rho = \|\mathbf{r}_0\|_2$, $\mathbf{w}_1 = \mathbf{r}_0$, $\mathbf{v}_1 = \mathbf{w}_1/\rho$
2: $\mathbf{w}_2 = A\mathbf{v}_1$, $H_{1,1} = \mathbf{v}_1^T\,\mathbf{w}_2$, $\mathbf{u}_2 = (\mathbf{u}_2 - \mathbf{v}_1\,H_{1,1}\,)\,/\,H_{1,1}$, $V_2 = [\,V_1,\,\mathbf{w}_2\,]$
3: **for** $k = 3, \ldots n$ **do**

4: $\begin{bmatrix} \mathbf{r}^{(2)}_{1:k-2,k-1} & \mathbf{r}^{(0)}_{1:k-1,k} \\ \|\mathbf{u}_{k-1}\|_2^2 & \mathbf{r}^{(0)}_{k-1,k} \end{bmatrix} = [\,V_{k-2},\,\mathbf{u}_{k-1}\,]^T\,[\,\mathbf{u}_{k-1},\,A\mathbf{u}_{k-1}\,]$ ▷ Global Synchronization

5: $\mathbf{r}^{(3)}_{1:k-2,k-1} = (\,I + L_{k-2}\,)^{-1}\,\mathbf{r}^{(2)}_{1:k-2,k-1}$ ▷ Normal equations

6: $\gamma_{k-1} = \left\{(\,\|\mathbf{u}_{k-1}\|_2^2 - \|\,\mathbf{r}^{(2)}_{1:k-2,k-1}\|_2^2\,)\right\}^{1/2}$ ▷ Pythagorean thm

7: $\mathbf{r}^{(0)}_{1:k-1,k} = \mathbf{r}^{(0)}_{1:k-1,k}\,/\,\gamma_{k-1}$ ▷ Scale for Arnoldi
8: $\mathbf{r}^{(0)}_{k-1,k} = \mathbf{r}^{(0)}_{k-1,k}\,/\,\gamma_{k-1}$
9: $L_{k-1,1:k-2} = \mathbf{r}^{(2)}_{1:k-2,k-1}/\,\gamma_{k-1}$

10: $\mathbf{w}_{k-1} = \mathbf{u}_{k-1} - V_{k-2}\,\mathbf{r}^{(2)}_{1:k-2,k-1}$ ▷ Jacobi iteration
11: $\mathbf{v}_{k-1} = \mathbf{w}_{k-1}\,/\,\gamma_{k-1}$

12: $\mathbf{r}^{(1)}_{1:k-1,k} = \begin{bmatrix} I & 0 \\ -L_{k-1,1:k-2} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{(0)}_{1:k-2,k} \\ \mathbf{r}^{(0)}_{k-1,k} \end{bmatrix}$
13: $\mathbf{u}_k = A\mathbf{u}_{k-1}\,/\,\gamma_{k-1} - V_{k-1}\,\mathbf{r}^{(1)}_{1:k-1,k}$ ▷ Gauss-Seidel iteration

14: $H_{1:k-2,k-1} = \mathbf{r}^{(4)}_{1:k-2,k-1} + \mathbf{r}^{(3)}_{1:k-2,k-1}$ ▷ Arnoldi expansion H_{k-1}
15: $\mathbf{r}^{(4)}_{1:k-1,k} = \mathbf{r}^{(0)}_{1:k-1,k} - H_{1:k-1,1:k-2}\,\mathbf{r}^{(3)}_{1:k-2,k-1}$

16: **end for**

flops. The number of global reductions is decreased from $k-1$ at iteration k in MGS-GMRES to only one when combined with the lagged normalization of the Krylov basis vectors. These costs are comparable to the DCGS-2 algorithm requiring $4mn^2$ flops. Indeed, the MGS-CGS GMRES can be viewed as a hybrid algorithm with Gauss-Seidel (MGS) and Jacobi (CGS) iterations.

The traditional MGS-GMRES algorithm computes an increasing number of inner products at each Arnoldi iteration. These take the form of global reductions implemented as `MPI_Allreduce`. A global reduction requires $\mathcal{O}(\log P)$ time to complete, where P is the number of MPI ranks running on a parallel machine. This can be further complicated by heterogeneous computer architectures based on graphical processor units (GPUs). However, the single GPU performance of DCGS-2 is well over 200 GigaFlops/sec, and merges the matrix-vector products in Steps 10 and 13 above into one GPU kernel call for increased execution speed. Recent improvements to MPI collective global communication operations, that reduce latencies, include node-aware optimizations as described by Bienz et al. [22]. Among the different parallel variants of the algorithms studied by Yamazaki et al. [18], the low-synch implementation of MGS-GMRES exhibited the best strong scaling performance on the ORNL Summit supercomputer.

Backward stability. The MGS-GMRES algorithm was proven to be backward stable for the solution of linear systems $A\mathbf{x} = \mathbf{b}$ in [2] and orthogonality is maintained to $\mathcal{O}(\varepsilon)\kappa(B_k)$, where $B_k = [\mathbf{r}_0,\,A\bar{V}_k]$ and $\bar{V}_k$ is the matrix generated in finite precision arithmetic at iteration k, with computed vectors as columns. Our backward error analysis of the iterated Gauss-Seidel Gram-Schmidt Algorithm 2.1 is also based on the QR factorization of the matrix $\bar{B}_k$. The error matrix, F_k, for the computed Arnoldi expansion after k iterations is expressed as

$$A\bar{V}_k - \bar{V}_{k+1}\bar{H}_{k+1,k} = F_k,$$

and is a matrix that grows in size by one column at each iteration. Recall that the strictly lower triangular matrix L_k is incrementally computed one row per iteration as in (2.2) and is obtained from the relation

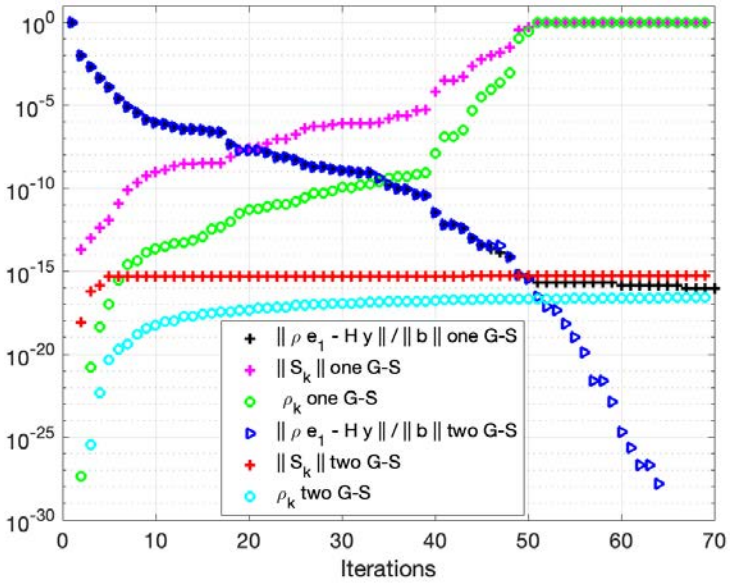

FIG. 1. *fs1836 matrix. Arnoldi relative residual, spectral radius of $M_k^{-1}N_k$ and Paige's $\|S_k\|$ for one and two Gauss-Seidel iterations.*

$\bar{V}_k^T\bar{V}_k = I + L_k + L_k^T$. Thus it follows from the analysis in Section 3 that the loss of orthogonality is

$$\| I - \bar{V}_k^T\bar{V}_k \|_2 \lesssim \mathcal{O}(\varepsilon). \tag{4.4}$$

The IGS-GMRES algorithm (Algorithm 4.1) thus maintains orthogonality to the level $\mathcal{O}(\varepsilon)$. Therefore, it follows from Drkošová et al. [8], that under the reasonable assumption of the numerical non-singularity of the coefficient matrix, the algorithm is also backward stable.

Our first experiment illustrates that the bounds derived in the previous sections properly capture the behavior of the IGS-GMRES algorithm in finite precision. In the next section, we demonstrate that our bounds continue to hold for a broad class of matrices. In particular, we examine the `fs1836` matrix studied by Paige and Strakoš [3]. Our Figure 1 should be compared with Figures 7.1 and 7.3 of their (2002) paper. In order to demonstrate empirically that the backward error is reduced by the iteration matrix $M_{k-1}^{-1}N_{k-1}$, the quantity $\|S_k^{(2)}\|_2$ is computed, as defined by Paige et al. [2], which measures the loss of orthogonality for two iterations. The spectral radius ρ_k of the matrix $M_k^{-1}N_k$ is highly correlated with, and follows the metric, $\|S_k^{(2)}\|_2$. The metric $\|S_k^{(2)}\|_2$ is plotted for one and two Gauss-Seidel iterations in Figure 1. The Arnoldi relative residuals are plotted as in Figure 7.1 of Paige and Strakoš [3], which, for one iteration, stagnate near iteration forty-three at 1×10^{-7} before reaching $\mathcal{O}(\varepsilon)$. For two Gauss-Seidel iterations, the (Arnoldi) relative residual continues to decrease monotonically and the norm-wise relative backward error (1.3) reaches the level of machine precision: $\beta(\mathbf{x}_k) = 6.6 \times 10^{-17}$ at iteration fifty. Most notably, the 2-norm of A is large, where $\|A\|_2 = 1.2 \times 10^9$.

A normal matrix $X \in \mathbb{C}^{n\times n}$ satisfies $X^*X = XX^*$. Henrici's definition of the departure from normality

$$\mathrm{dep}(X) = \sqrt{\|X\|_F^2 - \|\Lambda(X)\|_F^2}, \tag{4.5}$$

where $\Lambda(X) \in \mathbb{C}^{n\times n}$ is the diagonal matrix containing the eigenvalues of X [23], serves as a useful metric for the loss of orthogonality. While we find practical use for this metric for measuring the degree of (non)normality of a matrix, there are of course other useful metrics to describe (non)normality. We refer the reader to [23–25] and references therein. In particular, the loss of orthogonality is signaled by the departure from normality as follows:

$$\mathrm{dep}^2(M_k^{-1}) = \mathrm{dep}^2(I + L_k) = \|I + L_k\|_F^2 - \|I\|_F^2 = \|I\|_F^2 + \|L_k\|_F^2 - k = \|L_k\|_F^2. \tag{4.6}$$

Note that for the iteration matrix $M_k^{-1}N_k = (I + L_k)^{-1}L_k^T$ we have, and it is observed in practice, that $\|M_k^{-1}N_k\|_2 \leq \|M_k^{-1}N_k\|_F \approx \|L_k\|_F$ up to the first order in ε.

5. Numerical Results. Numerically challenging test problems for GMRES have been proposed and analyzed over the past 35 years. These include both symmetric and non-symmetric matrices. Simoncini and Szyld [26] introduced a symmetric, diagonal matrix with real eigenvalues, causing MGS-GMRES residuals to stagnate. Highly non-normal matrices from Walker [9] were used to explore the convergence characteristics of HH-GMRES and then the non-normal fs1836 from Paige et al. [2] and west0132 from Paige and Strakoš [3] encountering stagnation. In addition to these, we consider the impcol_e matrix from Greenbaum et al. [27]. Matrices with complex eigenvalues forming a disc inside the unit circle such as the Helmert matrix from Liesen and Tichý [28], are also evaluated. Results from a pressure continuity solver with AMG preconditioner and a circuit simulation with the ADD32 matrix from Rozložník, Strakoš, and Tůma [29] are also presented. Numerical results were obtained with Algorithm 4.1 and we verified that Algorithm 4.2 computations are comparable. In particular, the diagonal elements $R_{k,k}$ computed with the Pythagorean identity agree with those from Algorithm 4.1 to at least 16 significant digits in all of our examples. In all cases, the orthogonality between the computed vectors remains on the level $\mathcal{O}(\varepsilon)$, the smallest singular value of $\bar{V}_k$ remains close to one, and the Krylov basis vectors remain linearly independent. Therefore the norm of the true relative residual reaches the level $\mathcal{O}(\varepsilon)$ at least by the final iteration, while the Arnoldi residual continues to decrease far below this level.

Ill-Conditioned Diagonal Matrix. Simoncini and Szyld [26] consider several difficult and very ill-conditioned problems that can lead to stagnation of the residual before converging to the level of machine precision $\mathcal{O}(\varepsilon)$. In their example 5.5, they construct $A = \text{diag}([1e-4, 2, 3, \ldots, 100])$, a diagonal matrix, and the right-hand side $\mathbf{b} = \text{randn}(100, 1)$ is normalized so that $\|\mathbf{b}\|_2 = 1$. The condition number of this matrix is $\kappa(A) = 1 \times 10^6$ and $\|A\|_2 = 100$. With the MGS-GMRES algorithm, the relative residual stagnates at the level 1×10^{-12} after 75 iterations, when $\|S_k\|_2 = 1$, indicating that the Krylov vectors are not linearly independent. In the case of the iterative Gauss-Seidel formulation of GMRES (with two Gauss-Seidel iterations), the convergence history is plotted in Figure 2, where it can be observed that the relative Arnoldi residual continues to decrease monotonically. Furthermore, the true relative residual is plotted along with $\|L_{k-1}\|_F$. The latter indicates that a significant loss of orthogonality does not occur.

Ill-Conditioned Symmetric and Non-Symmetric Matrices. Figures 1.1 and 1.2 from Greenbaum et al. [27] describe the results for STEAM1 (using the HH and MGS implementations of GMRES, respectively). Similarly, Figures 1.3 and 1.4 from Greenbaum et al. [27] correspond to IMPCOLE. They emphasize that the convergence behavior illustrated in these plots is typical of the MGS-GMRES and HH-GMRES algorithms. The condition number of the system matrix is $\kappa(A) = 2.855 \times 10^7$ and $\|A\|_2 = 2.2 \times 10^7$ for STEAM1, whereas $\kappa(A) = 7.102 \times 10^6$ and $\|A\|_2 = 5.0 \times 10^3$ for IMPCOLE.

Greenbaum et al. [27] observe that although orthogonality of the Krylov vectors is not maintained near machine precision, as is the case for the Householder implementation, the relative residuals of the MGS-GMRES algorithm are almost identical to those of the HH-GMRES until the smallest singular value of the matrix $\bar{V}_k$ begins to depart from one. At that point the MGS-GMRES relative residual norm begins to stagnate close to its final precision level. This observation is demonstrated with the numerical examples for matrices STEAM1 ($N = 240$, symmetric positive definite matrix used in oil recovery simulations) and IMPCOLE ($N = 225$, nonsymmetric matrix from modelling of the hydrocarbon separation problem). In both experiments $\mathbf{x} = (1, ..., 1)^T$, $\mathbf{b} = A\mathbf{x}$ and $\mathbf{x}_0 = \mathbf{0}$. The convergence histories for the iterated Gauss-Seidel GMRES algorithm applied to these matrices are plotted in Figures 3 and 4. A significant loss of orthogonality is not observed until the last iteration. Otherwise the computed metric $\|L_{k-1}\|_F$ and the true relative residual remain near $\mathcal{O}(\varepsilon)$. The Arnoldi residual continues to decrease and the smallest singular value of V_k is 0.99985.

Highly Non-Normal Matrices. Bidiagonal matrices with a δ off-diagonal were studied by Embree [30]. These are non-normal matrices where $0 < \delta \leq 1$ and are completely defective for all $\delta \neq 0$. A defective matrix is a square matrix that does not have a complete basis of eigenvectors, and is therefore not diagonalizable, and the pseudo-spectra [25] of these matrices are discs in the complex plane. Our

IGS-GMRES algorithm leads to convergence after 16 iterations without stagnation and orthogonality is maintained to machine precision as plotted in Figure 5. The matrix 2-norm is $\|A\|_2 = 1.1$ and condition $\kappa(A) = 1.2$. Walker [9] employed the highly non-normal matrix in equation (5.1) to compare the Gram-Schmidt and Householder implementations of GMRES. The element α controls both the condition number $\kappa(A)$ and the departure from normality $\mathrm{dep}(A)$ of the matrix of size $n \times n$. Here $\|A\|_2 = 2.0 \times 10^3$.

$$A = \begin{bmatrix} 1 & 0 & \cdots & 0 & \alpha \\ 0 & 2 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & n \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{5.1}$$

For large values of α, Walker found that the MGS residual would stagnate and that the CGS algorithm led to instability. Furthermore, it was found that even CGS-2 with re-orthogonalization exhibited some instability near convergence. HH-GMRES maintains $\mathcal{O}(\varepsilon)$ orthogonality as measured by $\|I - \bar{V}_k^T \bar{V}_k\|_F$ and reduces the relative residual to machine precision.

In our experiments, the value $\alpha = 2000$ leads to a matrix with $\kappa(A) = 4 \times 10^5$. The departure from normality, based on Henrici's metric, is large: $\mathrm{dep}(A) = 2000$. The convergence history is displayed in Figure 6. The loss of orthogonality remains near $O(\varepsilon)$ and our upper bound is close for this problem. Notably, the oscillations present in the relative residual computed by the CGS-2 variant are not present in the iterated Gauss-Seidel convergence history plots, where the Arnoldi relative residual decreases monotonically.

Paige and Strakoš [3] experimented with MGS-GMRES convergence for the non-normal matrices FS1836 and WEST0132. In all their experiments $\mathbf{b} = (1, \ldots, 1)^T$. The matrix FS1836 has dimension $n = 183$, with $\|A\|_2 \approx 1.2 \times 10^9$, and $\kappa(A) \approx 1.5 \times 10^{11}$. The matrix WEST0132 has $n = 132$, $\|A\|_2 \approx 3.2 \times 10^5$, and $\kappa(A) \approx 6.4 \times 10^{11}$. Their Figure 7.1 indicates that the relative residual for FS1836 stagnates at 1×10^{-7} at iteration 43 when orthogonality is lost. The relative residual for the WEST0132 matrix also stagnates at the 1×10^{-7} level after 130 iterations. These results contrast with our Figures 7 and 8. In both cases the Arnoldi residuals continue to decrease and $\|L_{k-1}\|_F$ grows slowly or remains close to machine precision. The smallest singular value remains at $\sigma_{\min}(\bar{V}_k) = 1$ for both matrices.

Complex Eigenvalues in a Disc. Liesen and Tichý [28] employ the Helmert matrix generated by the MATLAB command `gallery('orthog',18,4)`. Helmert matrices occur in a number of practical problems, for example in applied statistics. The matrix is orthogonal, and the eigenvalues cluster around -1, as in the right panels of their Figure 4.4. The worst-case MGS-GMRES residual norm decreases quickly throughout the iterations and stagnates at the 12–th iteration, where the relative residual remains at 1×10^{-10}. From the convergence history plotted in Figure 9, the loss of orthogonality remains near machine precision and the Arnoldi relative residual does not stagnate. The quantity $\|L_{k-1}\|_F$ is an excellent predictor of the orthogonality.

Nalu-Wind Model. Nalu-Wind solves the incompressible Navier-Stokes equations, with a pressure projection. The governing equations are discretized in time with a BDF-2 integrator, where an outer Picard fixed-point iteration is employed to reduce the nonlinear system residual at each time step. Within each time step, the Nalu-Wind simulation time is often dominated by the time required to setup and solve the linearized governing equations. The pressure systems are solved using GMRES with an AMG preconditioner, where a polynomial Gauss-Seidel smoother is now applied are described in Mullowney et al. [31]. Hence, relaxation is a compute time intensive component, when employed as a smoother.

The McAlister experiment for wind-turbine blades is an unsteady RANS simulation of a fixed-wing, with a NACA0015 cross section, operating in uniform inflow. Resolving the high-Reynolds number boundary layer over the wing surface requires resolutions of $\mathcal{O}(10^{-5})$ normal to the surface resulting in grid cell aspect ratios of $\mathcal{O}(40,000)$. These high aspect ratios present a significant challenge. The simulations were performed for the wing at 12 degree angle of attack, 1 m chord length, denoted c, aspect ratio of 3.3, i.e., $s = 3.3c$, and square wing tip. The inflow velocity is $u_\infty = 46$ m/s, the density is $\rho_\infty = 1.225$ kg/m^3, and dynamic viscosity is $\mu = 3.756 \times 10^{-5}$ kg/(m s), leading to a Reynolds number $Re = 1.5 \times 10^6$. Due to the complexity of mesh generation, only one mesh with approximately 3 million grid points was generated.

The smoother is hybrid block-Jacobi with two sweeps of polynomial Gauss-Seidel relaxation applied locally on a subdomain and then Jacobi smoothing for globally shared degrees of freedom. The coarsening rate for the wing simulation is roughly $4\times$ with eight levels in the V-cycle for *hypre* [32]. Operator complexity C is approximately 1.6 indicating more efficient V-cycles with aggressive coarsening, however, an increased number of solver iterations are required compared to standard coarsening. The convergence history is plotted in Figure 10, where the loss of orthogonality is completely flat and close to machine precision.

Circuit Simulation. Rozložník et al. [29] study a typical linear system arising in circuit simulation (the matrix from a 32-bit adder design). The matrix has $\|A\|_2 = 0.05$ and $\kappa(A) = 213$. In exact arithmetic the Arnoldi vectors are orthogonal. However, in finite precision computation the orthogonality is lost, which may potentially affect approximate solution. In their Figure 3, the authors have plotted the loss of orthogonality of the computed Krylov vectors for different implementations of the GMRES method (MGS, Householder and CGS). The comparable results for IGS-GMRES are plotted in Figure 11. The smallest singular value remains at $\sigma_{\min}(\bar{V}_k) = 1$.

6. Conclusions. The essential contribution of our work was to derive an iterative Gauss-Seidel formulation of the GMRES algorithm due to Saad and Schultz [1] that employs approximate solutions of the normal equations appearing in the Gram-Schmidt projector, based on observations of Ruhe [6] and the low-synchronization algorithms introduced by Świrydowicz et al. [4].

The insights gained from the seminal work of Ruhe [6] led us to the conclusion that the modified Gram-Schmidt algorithm is equivalent to one step of a *multiplicative* Gauss-Seidel iteration method applied to the normal equations $Q_{k-1}^T Q_{k-1}\, \mathbf{r}_{1:k-1,k} = Q_{k-1}^T \mathbf{a}_k$. Similarly, the classical Gram-Schmidt algorithm can be viewed as one step of an *additive* Jacobi relaxation. The projector is then given by $P\mathbf{a}_k = \mathbf{a}_k - Q_{k-1}\, T_{k-1}\, Q_{k-1}^T \mathbf{a}_k$, where T_{k-1} is a correction matrix. In the case of DCGS-2, with delayed re-orthogonalization, Bielich et al. [5] split and apply the symmetric (normal) correction matrix across two Arnoldi iterations and then apply Stephen's trick to maintain orthogonality. For MGS, the lower triangular matrix $T_{k-1} \approx (\, Q_{k-1}^T Q_{k-1}\,)^{-1}$ appearing in Świrydowicz et al. [4] was identified as the inverse compact WY form with $T_{k-1}^{(1)} = (I + L_{k-1})^{-1}$, where the strictly lower triangular matrix L_{k-1} was computed from the loss of orthogonality relation

$$Q_{k-1}^T Q_{k-1} = I + L_{k-1} + L_{k-1}^T.$$

The matrix T_{k-1} from the inverse compact WY form of the Gram-Schmidt projector was also present, without having been explicitly defined, in the rounding error analysis of Björck [13], in Lemma 5.1. In effect, the low-synchronization Gram-Schmidt algorithm presented in [4] represents one iteration step, with a zero initial guess, to construct an approximate projector. When two iteration steps are applied, the resulting correction matrix $T_{k-1}^{(2)}$ is close to a symmetric (normal) matrix:

$$T_{k-1}^{(2)} = M_{k-1}^{-1}\, [\, I + N_{k-1}\, M_{k-1}^{-1}\,] = T_{k-1}^{(1)} - T_{k-1}^{(1)}\, L_{k-1}^T\, T_{k-1}^{(1)}.$$

However, the Gauss-Seidel formulation described by Ruhe [6] differs from MGS-2 in floating-point arithmetic and the incremental MGS-2 iterative refinement steps allow us to prove backward stability. When employed to compute the Arnoldi-QR expansion, the GMRES formulation with two Gauss Seidel iterations results in an $\mathcal{O}(\varepsilon)$ backward error, preserving the orthogonality of the Krylov basis vectors to the level $\mathcal{O}(\varepsilon)$, which is measured by $\|L_{k-1}\|_F$. This result is related to recent work on the iterative solution of triangular linear systems using Jacobi iterations that may diverge for highly non-normal triangular matrices [33].

Here, the departure from normality of the inverse of the correction matrix $T_{k-1}^{(1)}$ is a measure of the loss of orthogonality. In this formulation, the matrix is lower triangular and can substantially depart from normality as signaled by $\sigma_{\max}^2(I + L_{k-1}) > 1$. Because the correction matrix $T_{k-1}^{(2)}$ associated with two iterations of Gauss–Seidel is close to a symmetric (normal) matrix, the singular values of the inverse remain close to one. A departure from normality indicates a possible loss of numerical rank for the Krylov basis vectors V_k with the smallest singular value decreasing from one. Our numerical experiments, on challenging problems proposed over the past thirty-five years, demonstrate the robust

convergence and low relative error achievable with our IGS-GMRES. Furthermore, with one iteration step the loss of orthogonality is at most $\mathcal{O}(\varepsilon)\kappa(B_k)$ and for the algorithm with two iterations, it remains near machine precision.

We have demonstrated that the iterated Gauss Seidel GMRES with two iterations is backward stable and does not exhibit stagnation in the Arnoldi residual as defined by Greenbaum et al. [27]. This refers to the approximate residual norm computed as $\|\rho\,\mathbf{e}_1 - H_{k+1,k}\mathbf{y}_k\,\|_2/\rho$, where $\mathbf{x}_k = \mathbf{x}_0 + V_k\mathbf{y}_k$ is the approximate solution and the true residual is given by $\mathbf{b} - A\mathbf{x}_k$. The Arnoldi residual does not stagnate and will continue to decrease monotonically. In particular, the true relative residual will decrease towards machine precision level and as noted by Paige et al. [2], the norm-wise relative backward error $\beta(\mathbf{x}_k)$ will be $\mathcal{O}(\varepsilon)$. In the original algorithm, the Arnoldi residual can stagnate when the Krylov vectors lose linear independence and the smallest singular value of V_k decreases towards zero. For the iterative Gauss-Seidel GMRES algorithm described herein, the singular values remain close to one and $\beta(\mathbf{x}_k)$ remains close to machine precision at convergence.

We have also presented a one-reduce algorithm which can be interpreted as an MGS-CGS hybrid. This algorithm merits further study. Algorithm 4.1 is equivalent to the MGS-CGS GMRES when $L_{k-2} = 0$ at step $k-1$ in exact arithmetic and produces similar results in finite precision (in terms of the loss of orthogonality and Arnoldi residual). Our numerical experiments demonstrated that the computed results are comparable to the two-reduce IGS-GMRES and diagonal elements $R_{k,k}$ agree to machine precision when the Pythagorean identity is employed. In particular, the loss of orthogonality remains at $\mathcal{O}(\varepsilon)$ in all of the examples, implying backward stability according to Drkošová et al. [8].

We anticipate that the iterated Gauss-Seidel algorithms will facilitate the construction of backward stable iterative solvers based on block Gram-Schmidt algorithms including enlarged Krylov subspace methods and s-step methods. Mixed-precision formulations may also benefit. Randomization and sketching may be applied to the normal equations in the projector, with an oblique inner-product $\mathbf{x}^T B^T B\mathbf{x}$ and sketching matrix A, leading to even greater computational efficiency. In addition, these developments could be relevant for Anderson acceleration of nonlinear fixed-point iterations, which is currently being applied to optimization algorithms in deep learning for artificial intelligence. The algorithms may also be useful for computing eigenvalues with the Krylov-Schur algorithm of Stewart [21], where $\mathcal{O}(\varepsilon)$ orthogonality is required to obtain an invariant subspace of maximal dimension. We plan to explore the parallel strong-scaling performance of the IGS-GMRES and hybrid MGS-CGS GMRES for large-scale scientific computation on exascale supercomputers.

Acknowledgement. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. The second author was additionally supported by Charles University PRIMUS project no. PRIMUS/19/SCI/11 and Charles University Research program no. UNCE/SCI/023. The third author was supported by the Academy of Sciences of the Czech Republic (RVO 67985840) and by the Grant Agency of the Czech Republic, Grant No. 20-01074S. Our work was inspired by the contributions of A. Björck, A. Ruhe, C. C. Paige, Z. Strakoš, and the legacy of Y. Saad and M. H. Schultz that is MGS-GMRES. We also wish to thank our friends and colleagues Julien Langou and Luc Giraud for their thoughtful insights over many years. A sincere thank-you to the organizers of the Copper Mountain conferences on Krylov and multigrid methods for their rich 38+ year tradition of innovative numerical linear algebra. We would also like to thank the two anonymous reviewers for their careful reading of the manuscript. Their thoughtful comments resulted in a much improved presentation.

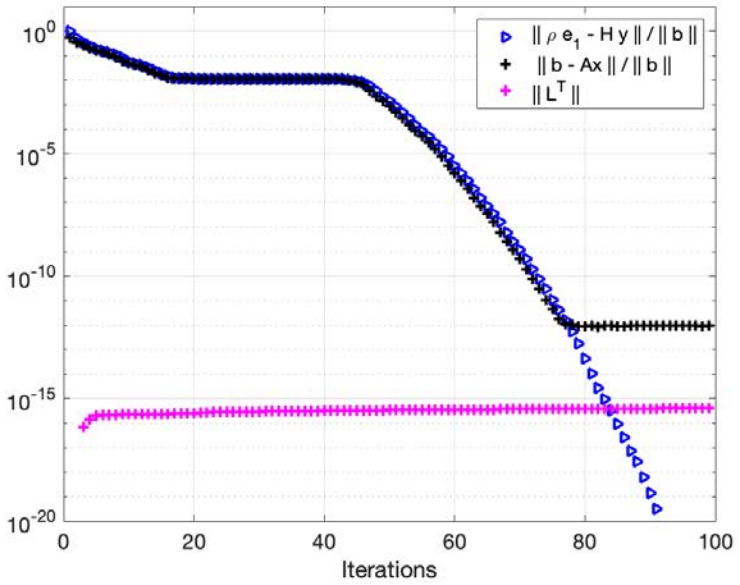

FIG. 2. *Simoncini matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

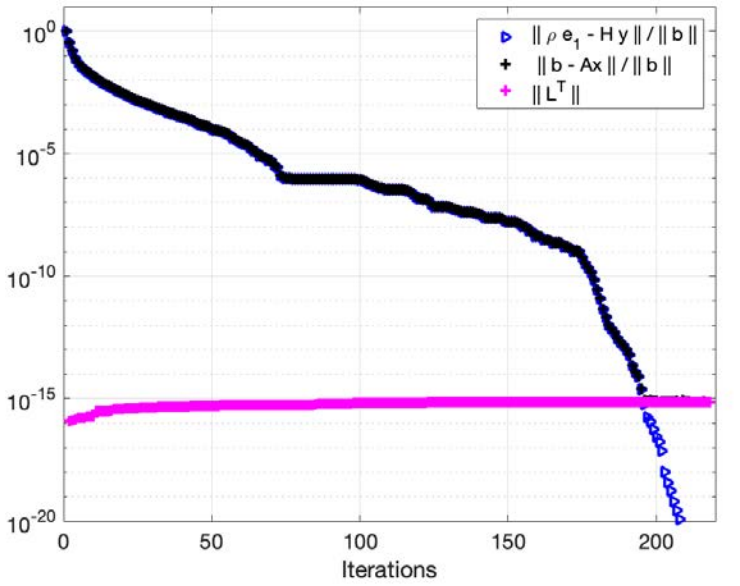

FIG. 3. *steam1 matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

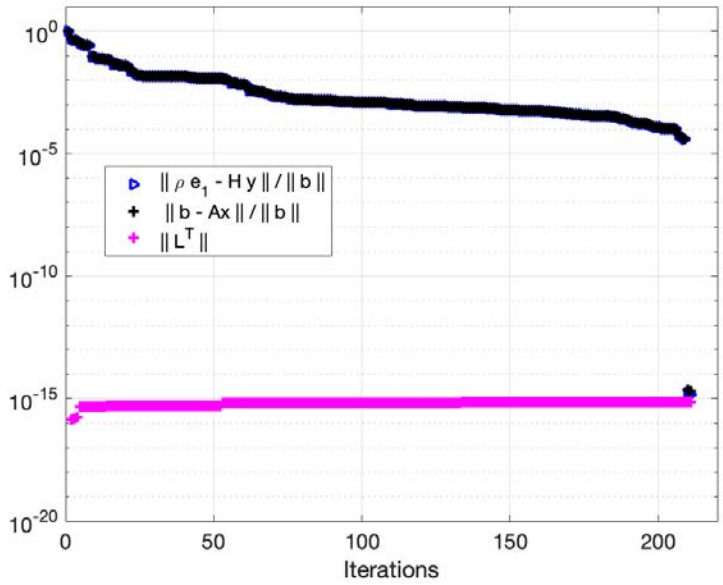

FIG. 4. *impcol_e matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

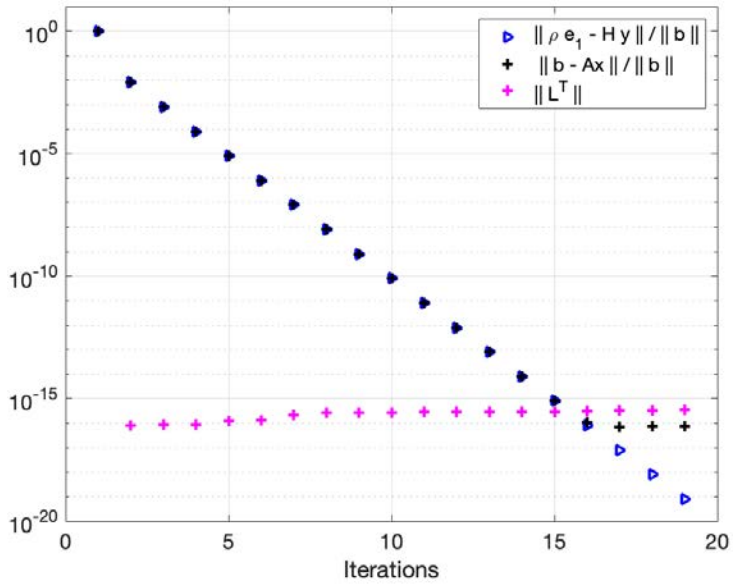

FIG. 5. *Embree δ matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

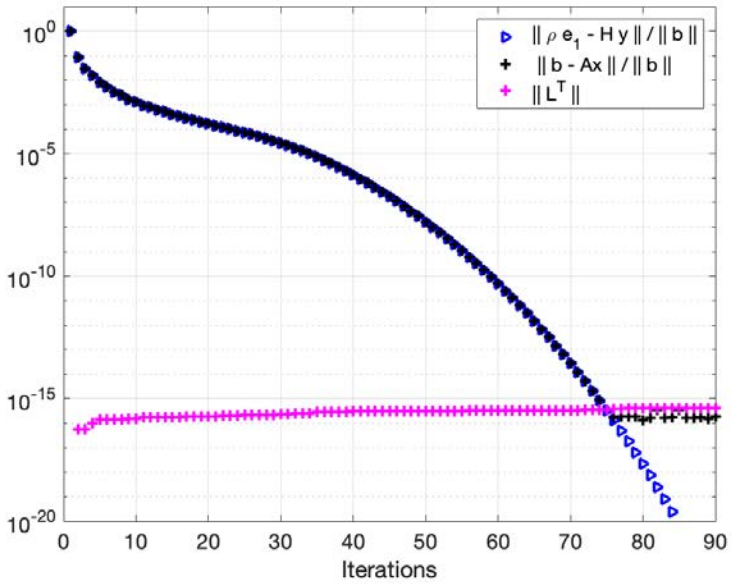

FIG. 6. *Walker matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

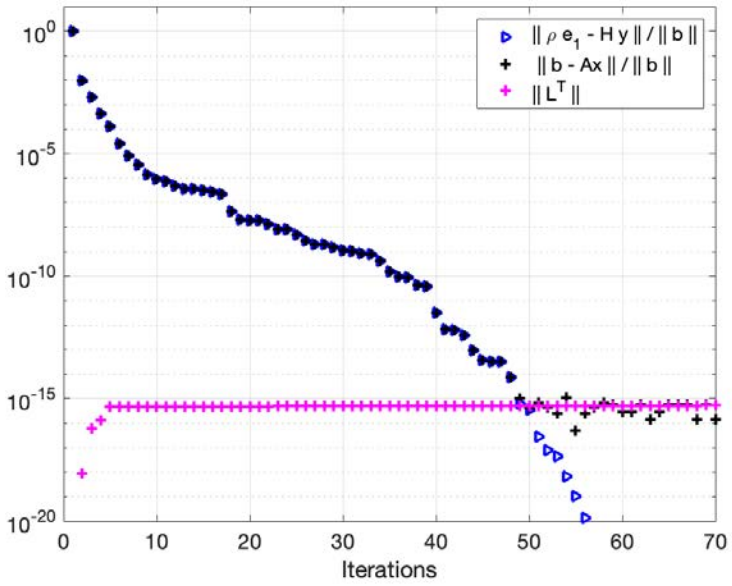

FIG. 7. *fs1863 matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

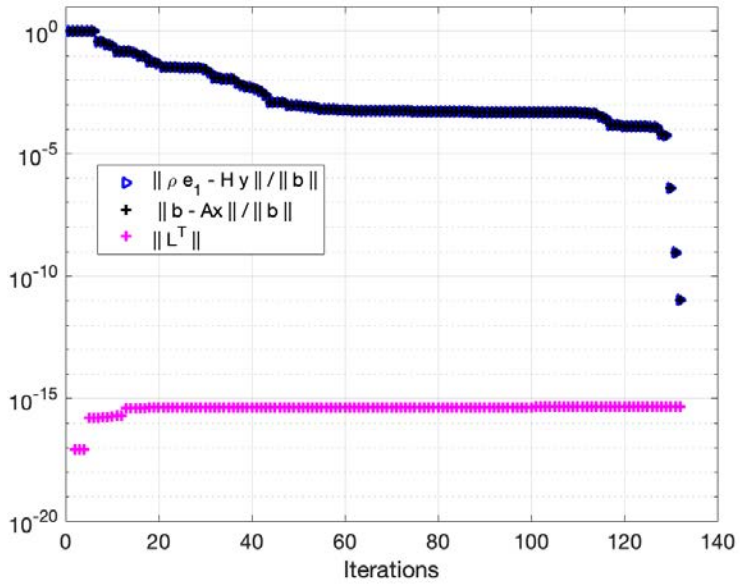

FIG. 8. *west0132 matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

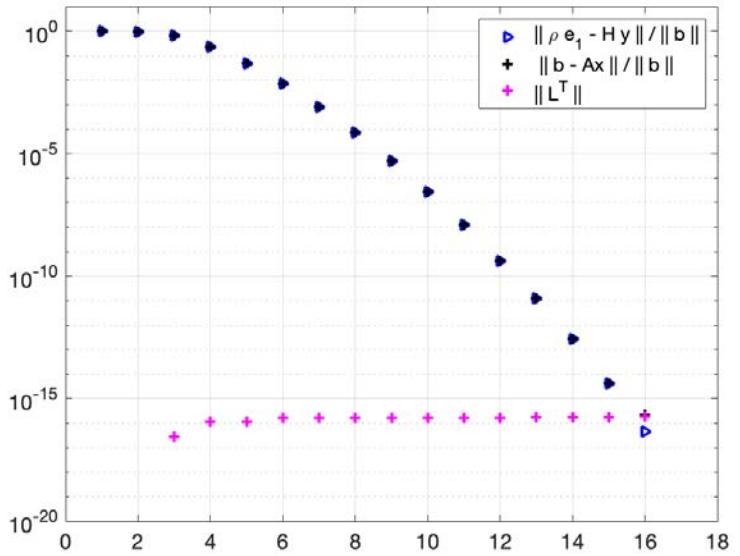

FIG. 9. *Helmert matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

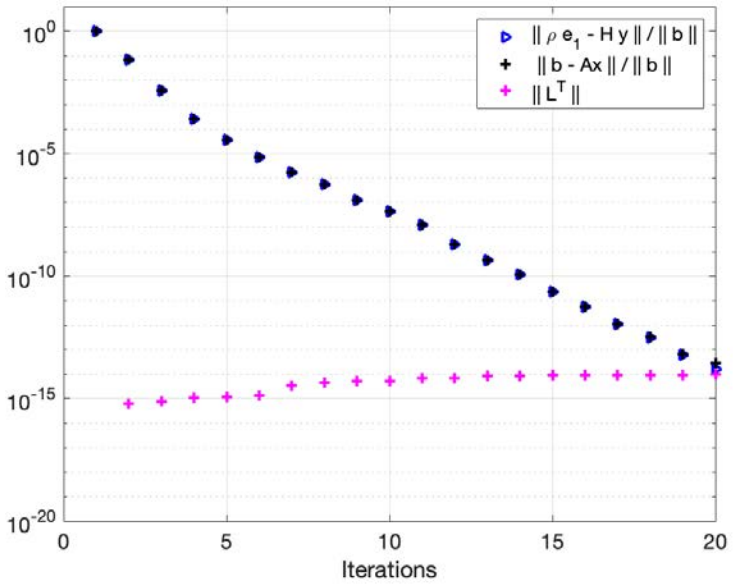

FIG. 10. *Pressure matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

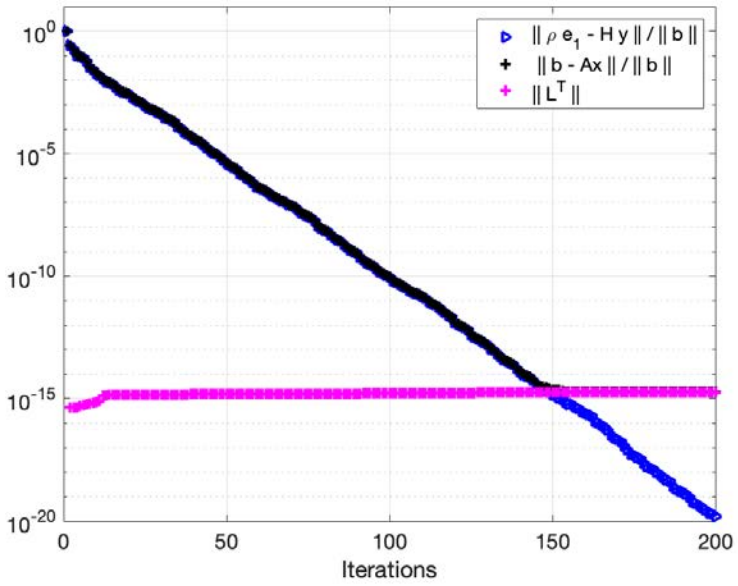

FIG. 11. *Add32 matrix. Arnoldi relative residual. Loss of orthogonality relation (4.6).*

REFERENCES

[1] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on scientific and statistical computing 7 (3) (1986) 856–869.

[2] C. C. Paige, M. Rozložník, Z. Strakoš, Modified Gram-Schmidt (MGS), least squares, and backward stability of MGS-GMRES, SIAM Journal on Matrix Analysis and Applications 28 (1) (2006) 264–284.

[3] C. C. Paige, Z. Strakoš, Residual and backward error bounds in minimum residual Krylov subspace methods, SIAM Journal on Scientific and Statistical Computing 23 (6) (2002) 1899–1924.

[4] K. Świrydowicz, J. Langou, S. Ananthan, U. Yang, S. Thomas, Low synchronization Gram-Schmidt and generalized minimal residual algorithms, Numerical Linear Algebra with Applications 28 (2020) 1–20.

[5] D. Bielich, J. Langou, S. Thomas, K. Świrydowicz, I. Yamazaki, E. Boman, Low-synch Gram-Schmidt with delayed reorthogonalization for Krylov solvers, Parallel Computing (2021).

[6] A. Ruhe, Numerical aspects of Gram-Schmidt orthogonalization of vectors, Linear Algebra and its Applications 52 (1983) 591–601.

[7] Å. Björck, Numerics of Gram-Schmidt orthogonalization, Linear Algebra and Its Applications 197 (1994) 297–316.

[8] J. Drkošová, A. Greenbaum, M. Rozložník, Z. Strakoš, Numerical stability of GMRES, BIT Numerical Mathematics 35 (1995) 309–330.

[9] H. F. Walker, Implementation of the GMRES method using Householder transformations, SIAM Journal on Scientific and Statistical Computing 9 (1) (1988) 152–163.

[10] L. Giraud, S. Gratton, J. Langou, A rank-k update procedure for reorthogonalizing the orthogonal factor from modified Gram-Schmidt, SIAM J. Matrix Analysis and Applications 25 (4) (2004) 1163–1177.

[11] Å. Björck, C. C. Paige, Loss and recapture of orthogonality in the modified Gram-Schmidt algorithm, SIAM Journal on Matrix Analysis and Applications 13 (1992) 176–190.

[12] S. Lockhart, D. J. Gardner, C. S. Woodward, S. Thomas, L. N. Olson, Performance of low synchronization orthogonalization methods in Anderson accelerated fixed point solvers, in: Proceedings of the 2022 SIAM Conference on Parallel Processing for Scientific Computing (PP), 2022, pp. 49–59.

[13] A. Björck, Solving least squares problems by Gram–Schmidt orthogonalization, BIT 7 (1967) 1–21.

[14] L. Giraud, J. Langou, M. Rozložník, J. v. d. Eshof, Rounding error analysis of the classical Gram-Schmidt orthogonalization process, Numerische Mathematik 101 (2005) 87–100.

[15] N. J. Higham, The accuracy of solutions to triangular systems, SIAM J. Numer. Anal. 26 (5) (1989) 1252–1265.

[16] N. J. Higham, Accuracy and Stability of Numerical Algorithms, 2nd Edition, SIAM, 2002.

[17] Q. Zou, GMRES algorithms over 35 years (2021). arXiv:2110.04017.

[18] I. Yamazaki, S. Thomas, M. Hoemmen, E. G. Boman, K. Świrydowicz, J. J. Elliott, Low-synchronization orthogonalization schemes for s-step and pipelined Krylov solvers in Trilinos, in: SIAM 2020 Conference on Parallel Processing for Scientific Computing, SIAM, 2020, pp. 118–128.

[19] E. Carson, K. Lund, M. Rozložník, S. Thomas, Block Gram-Schmidt algorithms and their stability properties, Linear Algebra and its Applications 638 (2022) 150–195.

[20] E. Carson, K. Lund, M. Rozložník, The stability of block variants of classical Gram–Schmidt, SIAM Journal on Matrix Analysis and Applications 42 (3) (2021) 1365–1380.

[21] G. W. Stewart, A Krylov–Schur algorithm for large eigenproblems, SIAM Journal on Matrix Analysis and Applications 23 (3) (2001) 601–614.

[22] A. Bienz, W. Gropp, L. Olson, Node-aware improvements to allreduce, in: Proceedings of the 2019 IEEE/ACM Workshop on Exascale MPI (ExaMPI), Association for Computing Machinery, 2019, pp. 1–10.

[23] P. Henrici, Bounds for iterates, inverses, spectral variation and fields of values of non-normal matrices, Numerische Mathematik 4 (1) (1962) 24–40.

[24] I. C. Ipsen, A note on the field of values of non-normal matrices, Tech. rep., North Carolina State University. Center for Research in Scientific Computation (1998).

[25] L. Trefethen, M. Embree, The behavior of nonnormal matrices and operators, Spectra and Pseudospectra (2005).

[26] V. Simoncini, D. B. Szyld, Theory of inexact krylov subspace methods and applications to scientific computing, SIAM Journal on Scientific Computing 25 (2) (2003) 454–477.

[27] A. Greenbaum, M. Rozložník, Z. Strakoš, Numerical behaviour of the modified Gram-Schmidt GMRES implementation, BIT 37 (3) (1997) 706–719.

[28] J. Liesen, P. Tichý, The worst-case GMRES for normal matrices, BIT Numer Math 44 (2004) 79–98.

[29] M. Rozložník, Z. Strakoš, M. Tůma, On the role of orthogonality in the GMRES method, in: K. G. Jeffery, J. Král, M. Bartosek (Eds.), SOFSEM '96: Theory and Practice of Informatics, Proceedings, Vol. 1175 of Lecture Notes in Computer Science, Springer, 1996, pp. 409–416.

[30] M. Embree, How descriptive are GMRES convergence bounds?, Tech. Rep. Tech. Rep. 99/08, Mathematical Institute, University of Oxford, UK (1999).

[31] P. Mullowney, R. Li, S. Thomas, S. Ananthan, A. Sharma, A. Williams, J. Rood, M. A. Sprague, Preparing an incompressible-flow fluid dynamics code for exascale-class wind energy simulations, in: Proceedings of the ACM/IEEE Supercomputing 2021 Conference, ACM, 2021, pp. 1–11.

[32] R. D. Falgout, J. E. Jones, U. M. Yang, The design and implementation of hypre, a library of parallel high performance preconditioners, in: A. M. Bruaset, A. Tveito (Eds.), Numerical Solution of Partial Differential Equations on Parallel Computers, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 267–294.

[33] E. Chow, H. Anzt, J. Scott, J. Dongarra, Using jacobi iterations and blocking for solving sparse triangular systems in incomplete factorization preconditioning, Journal of Parallel and Distributed Computing 119 (2018) 219–230.