A modular framework for the backward error analysis of GMRES

Alfredo Buttari, Nicholas J Higham, Théo Mary, Bastien Vieublé

A MODULAR FRAMEWORK FOR THE BACKWARD ERROR ANALYSIS OF GMRES

ALFREDO BUTTARI [1], NICHOLAS J. HIGHAM [2], THEO MARY [3], AND BASTIEN VIEUBLÉ [4]

Abstract. The Generalized Minimal Residual methods (GMRES) for the solution of general square linear systems is a class of Krylov-based iterative solvers for which there exist backward error analyses that guarantee the computed solution in inexact arithmetic to reach certain attainable accuracies. Unfortunately, these existing backward error analyses cover a relatively small subset of the possible GMRES variants and cannot be used straightforwardly in general to derive new backward error analyses for variants that do not yet have one. We propose a backward error analysis framework for GMRES that substantially simplifies the process of determining error bounds of most existing and future variants of GMRES. This framework describes modular bounds for the attainable normwise backward and forward errors of the computed solution that can be specialized for a given GMRES variant under minimal assumptions. To assess the relevance of our framework we first show that it is compatible with the previous rounding error analyses of GMRES in the sense that it delivers (almost) the same error bounds under (almost) the same conditions. Second, we explain how to use this framework to determine new error bounds for GMRES algorithms that do not have yet a backward error analysis, such as simpler GMRES, CGS2-GMRES, mixed precision GMRES, and more.

1. Introduction

The Generalized Minimal Residual method (GMRES), introduced by Saad and Schultz [41], aims to solve a nonsingular general square linear system

$$Ax = b, \quad A \in \mathbb{R}^{n\times n}, \quad 0 \neq b \in \mathbb{R}^n, \tag{1.1}$$

by iteratively building optimal approximate solutions x_k from a nested sequence of Krylov subspaces $\mathcal{K}_k(A, r_0) = \text{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}$. The process chooses the kth iterate $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ to minimize the 2-norm of the linear system residual $r_k = Ax_k - b$ and delivers the exact solution in at most $k = n$ iterations in exact arithmetic.

In practice, GMRES is implemented in floating-point arithmetic and the true solution cannot be computed exactly (see [10] for an up-to-date survey of floating-point arithmetic). Therefore, to derive bounds on the attainable backward and forward errors of GMRES, that is, bounds on the smallest backward and forward errors that can be obtained, we need a backward error analysis of GMRES.

The first backward error analysis of GMRES is presented by Drkošová et al. [18] and appears in the Ph.D. thesis of Rozložník [39]. In their analysis, the authors of [18] proved that GMRES with Householder orthogonalization (HH-GMRES) is normwise backward stable, meaning that it produces a computed solution whose normwise backward error is of order the unit roundoff of the floating-point arithmetic. To the best of our knowledge, this is the first backward error analysis of GMRES. A subsequent backward stability result on HH-GMRES with relaxed accuracy on the matrix–vector product was given in [19].

The second significant backward error analysis of GMRES concerns GMRES with modified Gram-Schmidt orthogonalization (MGS-GMRES). It is stated in the concluding remarks of the analysis of HH-GMRES [18] that, even though numerical experiments suggested that

1. IRIT, CNRS, F-31071 Toulouse, France
2. Department of Mathematics, The University of Manchester, M13 9PL Manchester, United Kingdom
3. Sorbonne Université, LIP6, CNRS, F-75005 Paris, France
4. Academy of Mathematics and Systems Science, CAS, 100190 Beijing, China

MGS-GMRES was also backward stable, the analysis of HH-GMRES cannot be straightforwardly extended to MGS-GMRES. The underlying reason for this difficulty is the loss of orthogonality in the computed Krylov basis generated by the MGS orthogonalization process. It required ten years and different keystone results, including but not limited to those in [20] and [37], for the MGS-GMRES algorithm to be finally proven backward stable by Paige et al. [36]. Because MGS-GMRES is generally preferred over HH-GMRES in practice, and because of the technicality of the proof and the various sub-results needed that widen our understanding of other algorithms, the work [36] is an important milestone in backward error analysis.

The third and last major backward error analysis of GMRES was proposed by Arioli et al. in [4] and [5]. It covers flexible GMRES (FGMRES), a variant of right-preconditioned GMRES accommodating variable preconditioners [40]. The first study [5] bounds the backward error of FGMRES with arbitrary preconditioners and, subsequently, uses this result to prove the normwise backward stability of FGMRES preconditioned by the LU factors of A computed with an unstable pivoting strategy. The second study [4] completes the previous one by improving the earlier backward error bound with arbitrary preconditioners and by demonstrating that FGMRES preconditioned by the LU factors computed in low precision is normwise backward stable.

Other more recent backward error analyses of GMRES exist in the literature but are strongly based on one of the abovementioned analyses. For example, the authors of [3] derived a backward error bound for MGS-GMRES using an arbitrary matrix–vector product by relying on the analysis of Paige et al. [36]. The authors of [13] proposed a backward error analysis of a split-preconditioned FGMRES in mixed precision by extending the work of Arioli et al. [4] and [5].

Our motivation for designing a new backward error analysis for GMRES lies in one striking aspect of this algorithm: its large number of possible variants. It is not relevant to enumerate them all in this article and, in the following, we only provide an idea of the scale of this number. We redirect the reader to the recent survey [51] or the book of Saad [42] for more details on the various implementations of GMRES. The reason for the extensive number of variants of GMRES lies in the number of options available in each part of this algorithm:

- At the preconditioner level we have a wide range of choices of preconditioners to pick from. These include partial or approximate factorizations of A, approximate inverses, polynomials, or iterative solvers, to quote a few. In addition, we need to consider the four main ways to apply them: right-, left-, split-, or flexible-preconditioning. More information about preconditioning can be found in [50].
- At the orthogonalization level we can pick from a range of algorithms that offer different tradeoffs between numerical stability and performance. The most common choices are Householder and classical or modified Gram-Schmidt with or without reorthogonalization. Numerical comparison of these algorithms can be found in [25, chap. 19], [21], or [32].
- At the restart level we need to choose whether or not to stop and restart GMRES periodically and under which criteria. In doing so, we limit the size of the Krylov subspaces and we can reduce resource consumption, but it comes at the risk of harming the convergence.
- Finally, we need to consider all the remaining techniques that change one or multiple of these parts. For instance, employing mixed precision arithmetic [3] or randomization [6], approximating the matrix–vector product [22], compressing the basis [2], or using block orthogonalization and communication avoiding approaches [16, 28].

Unfortunately, and importantly, among the many possible variants of GMRES only a small number are covered by one of the previous backward error analyses. In addition, because these previous analyses are long, sophisticated, and were not made to be modular, extending

one of them to derive a new backward error analysis for a given variant of GMRES is generally far from straightforward.

Motivated by this issue, the core objective of this article is to present a backward error analysis framework which simplifies the process of deriving bounds for the attainable normwise backward and forward errors of the computed solutions by many GMRES algorithms, in particular, those which do not yet have backward error analyses. To do so, in section 2, we list the set of notations and mathematical tools we will use throughout the article. In section 3, we develop our backward error analysis framework consisting of: an abstract modular algorithm that can be specialized to most of the possible and popular GMRES algorithms; parametric error bounds and minimal assumptions on the operations of this abstract algorithm; modular bounds for the attainable normwise backward and forward errors resulting from the error analysis of this abstract algorithm. In section 4, we consider an extension of this framework for taking into account restarted variants of GMRES. In section 5, we explain how to use our framework by applying it to HH-GMRES, MGS-GMRES, and FGMRES, showing in addition that it provides (almost) the same results under (almost) the same conditions as the previously mentioned existing backward error analyses of GMRES. Finally, in section 6, we use this framework to derive error bounds for simpler GMRES, GMRES using classical Gram-Schmidt with reorthogonalization, and mixed precision restarted GMRES for which no bounds existed previously in the literature. We further discuss how this framework could be applied to deflated GMRES, randomized Gram-Schmidt GMRES, and block GMRES.

2. Notations

In this section we introduce our notation and briefly recall the essential mathematical concepts and tools that we will use throughout the article.

We use the standard model of floating-point arithmetic [25, sect. 2.2], we use the notation $\mathrm{fl}(\cdot)$ to denote the computed value of a given expression, and we put a hat on variables to denote that they represent computed quantities. For any integer k, we define

$$\gamma_k = \frac{ku}{1-ku}.$$

A superscript on γ denotes that u carries that superscript as a subscript; thus $\gamma_k^f = ku_f/(1-ku_f)$, for example. We also use the notation $\widetilde{\gamma}_k = \gamma_{\eta k}$ to hide modest constants η.

Our analysis is a traditional worst case analysis and the error bounds obtained depend on some constants related to the problem dimension n and the size of the basis k. We gather these constants into generic functions $c(n,k)$. For the sake of readability, as these constants are known to be pessimistic (see [17, 26, 27]), we do not always keep track of the precise values of the functions $c(n,k)$. Instead, we guarantee that those functions $c(n,k)$ are polynomials in n and k of low degree.

We use the notation $\lesssim$ and $\approx$ when dropping negligible second order terms in the error bounds, and the notation $\Theta_1 \gg \Theta_2$ to indicate that Θ_1 is much greater than Θ_2. In particular, we consider that if $\Theta_1 \gg \Theta_2$ we can safely assess that $\Theta_1 \gg c(n,k)\Theta_2$, where $c(n,k)$ is a polynomial in n and k of low degree. We also use the notation $\equiv$, which means that we can take the quantity on the left, which is in our control and is not fixed, to be equal to the quantity on the right.

We define the normwise condition number of a square nonsingular $n \times n$ matrix M by $\kappa(M) = \|M^{-1}\|\|M\|$ for a given norm, and M^{-1} is replaced by the pseudoinverse if M is not square. We represent the set of singular values of M by $\sigma_1(M) \geq \sigma_2(M) \geq \cdots \geq \sigma_n(M)$, where we define $\sigma_{\min}(M) = \sigma_n(M)$ and $\sigma_{\max}(M) = \sigma_1(M)$.

Our error analysis uses both the 2-norm and the Frobenius norm, denoted by $\|\cdot\|_2$ and $\|\cdot\|_F$, respectively. The 2-norm of M refers to the induced norm

$$\|M\|_2 = \max_x \frac{\|Mx\|_2}{\|x\|_2} = \sigma_{\max}(M), \quad \|M^{-1}\|_2^{-1} = \min_x \frac{\|Mx\|_2}{\|x\|_2} = \sigma_{\min}(M).$$

We write $\kappa_2(M)$ and $\kappa_F(M)$ for the corresponding condition numbers of M.

The forward error of a computed solution $\widehat{x}$ of the linear system (1.1) is defined as

$$\frac{\|x-\widehat{x}\|_2}{\|x\|_2},$$

while the normwise backward error of $\widehat{x}$ we are using is defined as [25, sect. 7.1]

$$\min\left\{\,\varepsilon : (A+\Delta A)\widehat{x} = b+\Delta b,\ \|\Delta A\|_F \le \varepsilon\|A\|_F,\ \|\Delta b\|_2 \le \varepsilon\|b\|_2\,\right\} = \frac{\|b-A\widehat{x}\|_2}{\|A\|_F\,\|\widehat{x}\|_2+\|b\|_2}.$$

In the remainder of this article, "backward error" will refer implicitly to the "normwise backward error".

3. Backward error analysis framework

In this section, we develop our backward error analysis framework for GMRES. This framework is built upon an abstract algorithm which we call modular GMRES and that is composed of four elemental operations on which we require minimal assumptions. By specializing these operations and meeting these assumptions, modular GMRES can describe most of the GMRES implementations and variants. We perform a backward error analysis of this algorithm resulting in modular bounds on its attainable backward and forward errors. These modular bounds can be used to derive error bounds for any specializations of modular GMRES. This abstract algorithm, its associated assumptions, and its modular error bounds constitute the backward error analysis framework for GMRES.

3.1. The modular GMRES algorithm and its error model.

We define modular GMRES (MOD-GMRES) by Algorithm 1 which delivers an approximation to the solution of the linear system (1.1). To do so, the algorithm minimizes the residual of the left-preconditioned linear system $\widetilde{A}x = \widetilde{b}$, where $\widetilde{A} = M_L^{-1}A$ and $\widetilde{b} = M_L^{-1}b$, over a subspace $\mathcal{Z}$ spanned by the given full-rank basis $Z_k = [z_1, \ldots, z_k]$. In other words, MOD-GMRES can be viewed as a general subspace projection method computing an approximation to the solution of $Ax = b$ in the space $\mathcal{Z}$ under the orthogonality constraint $\widetilde{b} - \widetilde{A}x_k \perp \widetilde{A}\mathcal{Z}$, where $\perp$ is the orthogonality relation induced by the ℓ_2-inner-product.

A few comments are in order. First, note that MOD-GMRES initializes implicitly the first guess of the solution to zero, namely $x_0 = 0$. Doing so lightens our notation without losing generality; our conclusions still straightforwardly hold with $x_0 \neq 0$. Note also that in Algorithm 1 the matrix M_L stands for the potential use of a left-preconditioner. Naturally, setting $M_L = I$ amounts to no left-preconditioner used in the algorithm. The potential use of a right-preconditioner is carried by the basis Z_k which, in exact arithmetic, can take the form $Z_k \equiv M_R^{-1}V_k$ where $M_R \in \mathbb{R}^{n\times n}$ and $V_k = [v_1, \ldots, v_k]$ are the Krylov basis vectors obtained from an Arnoldi process. If we define $Z_k \equiv [M_{R,1}^{-1}v_1, \ldots, M_{R,k}^{-1}v_k]$ with possibly $M_{R,i} \neq M_{R,j}$ for all $i \neq j \le k$ we can even account for non-constant right-preconditioners and flexible variants of GMRES [40]. Finally, note that since MOD-GMRES describes a more general subspace projection method which is, for instance, not necessarily implemented with an Arnoldi procedure (more is said about this in the coming paragraphs), referring to it as "GMRES" might sound surprising. This choice is motivated by the fact that we solely focus on applying our framework to GMRES algorithms in the context of this article. For this reason, we call Algorithm 1 a GMRES method which is consistent with the rest of the content developed in this document.

MOD-GMRES is an abstract modular algorithm in the sense that we are making very few assumptions on the basis Z_k, the left-preconditioner M_L, and the operations at lines 1 to 4 of Algorithm 1. By specializing these different elements (or modules) and meeting their assumptions, MOD-GMRES can describe most of the GMRES algorithms in the literature and implemented in software. The following is about listing those minimal assumptions that are mostly parametric error bounds associated with each operation and that are necessary to derive our backward error analysis. We will later showcase in section 5 how to specialize

Algorithm 1 Modular GMRES (MOD-GMRES)

Input: a matrix $A \in \mathbb{R}^{n\times n}$, a right-hand side $b \in \mathbb{R}^n$, a basis $Z_k \in \mathbb{R}^{n\times k}$, and a left-preconditioner $M_L \in \mathbb{R}^{n\times n}$.
Output: a computed solution x_k to $Ax = b$.
1: Compute $C_k = \widetilde{A}Z_k \in \mathbb{R}^{n\times k}$ where $\widetilde{A} = M_L^{-1}A \in \mathbb{R}^{n\times n}$.
2: Compute $\widetilde{b} = M_L^{-1}b \in \mathbb{R}^n$.
3: Solve $y_k = \arg\min_y \|\widetilde{b} - C_k y\|_2$.
4: Compute the solution approximation $x_k = Z_k y_k$.

MOD-GMRES and how to meet its assumptions for the three previously mentioned GMRES algorithms on which we already have backward error analyses: HH-GMRES, MGS-GMRES, and FGMRES.

Matrix–matrix product (line 1). For nonsingular $M_L \in \mathbb{R}^{n\times n}$, $A \in \mathbb{R}^{n\times n}$, and $Z_k \in \mathbb{R}^{n\times k}$ we assume that the computed left-preconditioned matrix–matrix product $\widehat{C}_k$ at line 1 of Algorithm 1 satisfies

$$\widehat{C}_k = \mathrm{fl}(\widetilde{A}Z_k) = \widetilde{A}Z_k + \Delta_c, \quad \|\Delta_c\|_F \le \varepsilon_c \|\widetilde{A}Z_k\|_F, \tag{3.1}$$

where $\Delta_c \in \mathbb{R}^{n\times k}$ is the computing error generated during the computation of the matrix–matrix product and ε_c is a parameter bounding the magnitude of this error. The product itself can take many forms and be implemented in many ways as long as assumption (3.1) is satisfied for a given ε_c. For instance, if Algorithm 1 is implemented using the Arnoldi process, the matrix–matrix product is performed iteratively through a succession of matrix–vector products involving the computed Arnoldi Krylov basis vectors $\widehat{V}_k = [\widehat{v}_1, \ldots, \widehat{v}_k]$. If, in addition, a left-preconditioner is used we have $M_L \neq I$, $\widetilde{A} \neq A$, and $Z_k \equiv \widehat{V}_k$. In this situation, the preconditioned matrix $\widetilde{A}$ is rarely fully formed in practice, and its application to a vector is made by the successive applications of A and M_L^{-1}. The linear action of the preconditioner to a vector can be performed by explicitly forming M_L^{-1} and computing a standard matrix–vector product or by decomposing M_L into triangular factors subsequently used in substitution algorithms. In some cases, M_L^{-1} might not be available as a matrix or as a matrix decomposition, and we might only be able to compute its linear action to a vector by another means. If a right-preconditioner is used instead, we have $M_L = I$, $\widetilde{A} = A$, and $Z_k \equiv \mathrm{fl}(M_R^{-1}\widehat{V}_k)$, where $M_R \in \mathbb{R}^{n\times n}$ is nonsingular; it is worth noticing that the basis Z_k is defined as the computed product $\mathrm{fl}(M_R^{-1}\widehat{V}_k)$ rather than the exact one $M_R^{-1}\widehat{V}_k$. Naturally, the same previous comments we made on the left-preconditioned product apply to the computation of the right-preconditioned product $AM_R^{-1}\widehat{V}_k$. Overall, MOD-GMRES allows for different possibilities of product implementations for line 1 that potentially deliver different backward error results leading to different values for ε_c.

Preconditioned right-hand-side (line 2). For nonsingular $M_L \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ we assume that the computed preconditioned right-hand side $\widehat{b}$ at line 2 of Algorithm 1 satisfies

$$\widehat{b} = \mathrm{fl}(M_L^{-1}b) = \widetilde{b} + \Delta_b, \quad \|\Delta_b\|_2 \le \varepsilon_b \|\widetilde{b}\|_2, \tag{3.2}$$

where $\Delta_b \in \mathbb{R}^n$ is the computing error introduced by the application of M_L^{-1} on b and ε_b is a parameter bounding the magnitude of this error. Just as for the computation of the matrix–matrix product at line 1, the preconditioner application can be implemented in various ways as long as assumption (3.2) is met.

Least squares solver (line 3). For $\widehat{C}_k \in \mathbb{R}^{n\times k}$ and $0 \neq \widehat{b} \in \mathbb{R}^n$ we assume that the computed solution $\widehat{y}_k$ of the least squares problem at line 3 of Algorithm 1 satisfies

$$\begin{aligned} \widehat{y}_k &= \arg\min_y \|\widehat{b} + \Delta_{\mathrm{ls}}^b - (\widehat{C}_k + \Delta_{\mathrm{ls}}^c)y\|_2, \\ \|\left[\Delta_{\mathrm{ls}}^b, \Delta_{\mathrm{ls}}^c\right] e_j\|_2 &\le \varepsilon_{\mathrm{ls}} \|\left[\widehat{b}, \widehat{C}_k\right] e_j\|_2 \quad \forall j \le k+1, \end{aligned} \tag{3.3}$$

where $\Delta_{\text{ls}}^b \in \mathbb{R}^n$ and $\Delta_{\text{ls}}^c \in \mathbb{R}^{n\times k}$ are the computing errors generated by the least squares solver and ε_{ls} is a parameter bounding the magnitude of these errors. Assumption (3.3) does not enforce specific methods for solving the least squares problem. In particular, we do not require using the Arnoldi algorithm as classically done in GMRES. In that sense, the MOD-GMRES process is not necessarily iterative. Nevertheless, all the examples of application of our framework throughout this article employ an Arnoldi algorithm. For instance, we will show in section 5 that the solutions of the least squares problem via MGS [41] and Householder [48] Arnoldi meet assumption (3.3) for specific ε_{ls}. Moreover, in section 6 we discuss various other variants of the Arnoldi algorithm for the solution of the least squares problem at line 3 and how they meet this assumption.

Computation of the solution approximation (line 4). For $Z_k \in \mathbb{R}^{n\times k}$ and $\widehat{y}_k \in \mathbb{R}^n$ we assume that the computed approximate solution $\widehat{x}_k$ at line 4 of Algorithm 1 verifies

$$\widehat{x}_k = \text{fl}(Z_k\widehat{y}_k) = Z_k\widehat{y}_k + \Delta_x, \quad \|\Delta_x\|_2 \leq \varepsilon_x\|Z_k\|_F\|\widehat{y}_k\|_2, \tag{3.4}$$

where $\Delta_x \in \mathbb{R}^n$ is the error introduced while computing the matrix–vector product and ε_x is a parameter bounding the magnitude of this error. The implementation of this operation can take different forms, which yield potentially different values for ε_x. In particular, with right-preconditioned GMRES where $Z_k \equiv \text{fl}(M_R^{-1}\widehat{V}_k)$, the application of Z_k might not be a standard matrix–vector product. In that case, Z_k might not be stored explicitly and line 4 consists of a matrix–vector product with $\widehat{V}_k$ and the application of M_R^{-1}.

Additional assumptions. Finally, in addition to the previous assumptions on lines 1 to 4, we require the basis Z_k not to be numerically singular to the accuracy ε_x, that is,

$$\sigma_{\min}(Z_k) \gg \varepsilon_x\|Z_k\|_F. \tag{3.5}$$

In particular, this assumption means that if Z_k is ill-conditioned we need correspondingly high accuracy in computing line 4. We also require all the accuracy parameters to be substantially less than 1; that is,

$$0 \leq \varepsilon_c, \varepsilon_b, \varepsilon_{\text{ls}}, \varepsilon_x \ll 1. \tag{3.6}$$

The role of assumption (3.6) is mostly to ensure that second order terms can be dropped safely in our inequalities. We emphasize that the accuracy parameters do not need to be smaller than 1 by many orders of magnitude. Guaranteeing the parameters to be equal or lower than 0.01 for instance is likely enough for the results of this article to be valid.

3.2. **The key dimension.** A significant challenge in determining attainable forward and backward errors for MOD-GMRES lies in the fact that the quality of the computed solution depends strongly on the chosen basis Z_k. As for the operations used within MOD-GMRES, we require few assumptions on the basis Z_k. In particular, the basis does not have to be a Krylov basis or to be constructed from an Arnoldi process. That being said, it is trivial that good errors on the computed solution are not achieved in general for a very small basis Z_k that would span a subspace $\mathcal{Z}$ not descriptive enough. Therefore, we shall answer the question: what are minimal conditions on the basis Z_k such that the computed solution is guaranteed to have reached small backward and forward errors? Thinking of the dimension k of the basis as increasing, we can reformulate the question as: at which dimension $k \leq n$ is the computed solution guaranteed to have reached small errors? We define such a key dimension as the first $k \leq n$ for which Z_k satisfies

$$\sigma_{\min}\big(\big[\widetilde{b}\phi, \widetilde{A}Z_k\big]\big) \leq c(n,k)(\varepsilon_c + \varepsilon_b + \varepsilon_{\text{ls}})\|\big[\widetilde{b}\phi, \widetilde{A}Z_k\big]\|_F \tag{3.7}$$

and

$$\sigma_{\min}\big(\widetilde{A}Z_k\big) \gg (\varepsilon_c + \varepsilon_b + \varepsilon_{\text{ls}})\|\widetilde{A}Z_k\|_F \tag{3.8}$$

for all scalar $\phi > 0$ and where $c(n,k)$ is a polynomial of low degree in n and k. The role of conditions (3.7) and (3.8) is to capture the exact moment where $\widetilde{b}$ lies in the range of $\widetilde{A}Z_k$, that is, the moment where our basis Z_k contains the solution. Indeed, condition (3.7)

requires $[\widetilde{b}\phi, \widetilde{A}Z_k]$ to be nearby rank deficient while condition (3.8) enforces $\widetilde{A}Z_k$ to be full-rank. The combination of the two conditions imposes $\widetilde{b}$ to be in the range of $\widetilde{A}Z_k$.

Even though condition (3.8) imposes the smallest singular value of $\widetilde{A}Z_k$ to be sufficiently higher than the accuracy parameters associated with the matrix–matrix product (line 1), the preconditioned right-hand side (line 2), and the least squares problem (line 3), it is important to note that it does not require the quantities A, M_L, or Z_k to have, individually, a high enough smallest singular value relative to these accuracies. That would be a substantially stronger assumption since $\widetilde{A}Z_k$ generally tends to be better conditioned than A if the preconditioners are well-chosen.

This key dimension definition expressed by conditions (3.7) and (3.8) has been heavily inspired by the analysis of Paige , Rozloznik, and Strakos [36]. In that work, the conditions (3.7) and (3.8) do not appear as "conditions" or in this exact form but can be found indirectly in [36, eq. (8.6)]. Overall, the approach is the same but it is shaped differently. Note also that the introduction of a scalar ϕ in condition (3.7), also present in [36], becomes necessary for the proof of Theorem 3.1 where it is used alongside [36, Thm. 2.4] to bound the residual of the left-preconditioned linear systems.

A difficulty in applying our framework is showing that a dimension k exists such that conditions (3.7) and (3.8) hold. Fortunately, for the most stable orthogonalization algorithms, such as the Householder orthogonalization, the existence of such an iteration is relatively direct as we will show in section 5.2. When the orthogonalization method faces loss of orthogonality, the proof of the existence of the key dimension is less direct. We explain how the MGS orthogonalization, which faces such loss of orthogonality, still meets conditions (3.7) and (3.8) in section 5.3.

3.3. **Backward error analysis of MOD-GMRES.** This section is dedicated to the central theorem of this article which establishes bounds on the errors of the computed solution to (1.1) by MOD-GMRES. In more detail, Theorem 3.1 shows that under the assumptions made in section 3.1 and the existence of a key dimension k defined in section 3.2, the forward and backward errors of the computed solution $\widehat{x}_k$ at this dimension k are bounded by functions of the accuracy parameters ε_c, ε_b, ε_{ls}, and ε_x.

Theorem 3.1. *Suppose Algorithm 1 is applied with a basis $Z_k \in \mathbb{R}^{n\times k}$ and a left-preconditioner $M_L \in \mathbb{R}^{n\times n}$ to solve (1.1), where conditions (3.1) to (3.8) are satisfied for given parameters ε_c, ε_b, ε_{ls}, and ε_x. Then the computed solution $\widehat{x}_k$ of $Ax = b$ has backward and forward errors satisfying respectively*

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)\xi\kappa_F(M_L), \tag{3.9}$$

and

$$\frac{\|\widehat{x}_k - x\|_2}{\|x\|_2} \lesssim c(n,k)\xi\kappa_F(\widetilde{A}), \tag{3.10}$$

where

$$\xi = \alpha\varepsilon_c + \beta\varepsilon_b + \beta\varepsilon_{ls} + \lambda\varepsilon_x \tag{3.11}$$

with

$$\alpha = \sigma_{\min}^{-1}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}, \qquad \beta = \max\left(1, \sigma_{\min}^{-1}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}\right), \qquad \lambda = \sigma_{\min}^{-1}(Z_k)\|Z_k\|_F, \tag{3.12}$$

and where $c(n,k)$ is a polynomial in n and k of low degree.

Proof. The proof is split into four parts. We will first show that the computed solution $\widehat{y}_k$ of the least squares problem obtained at line 3 of Algorithm 1 is an accurate solution for this least squares problem. We then show that the residual of the left-preconditioned linear system, associated with the computed solution $\widehat{y}_k$ of the least squares problem, is small. We subsequently use the bound on the residual to determine the backward error of

the left-preconditioned linear system from which we finally deduce bounds on the forward and backward errors of the original linear system (1.1).

1. Backward error of the least squares problem computed solution. Given a basis Z_k of rank $k \leq n$ for which conditions (3.1) to (3.8) are satisfied, we begin the proof by bounding the backward error of the computed solution $\widehat{y}_k$ for the least squares problem $\min_y \|\widetilde{b} - \widetilde{A}Z_k y\|$ at line 3 of Algorithm 1. The least squares solver, satisfying condition (3.3), is applied on the least squares problem $\min_y \|\widehat{b} - \widehat{C}_k y\|_F$ which accounts for the errors generated during the computations of the product $\widetilde{A}Z_k$ and the preconditioned right-hand-side at line 1 and 2. The computed solution $\widehat{y}_k$ at line 3 therefore satisfies

$$\begin{aligned} \widehat{y}_k &= \arg\min_y \|\widetilde{b} + \Delta\widetilde{b}^{(1)} - (\widetilde{A}Z_k + \Delta C_k)y\|_2, \\ \|\Delta C_k\|_F &= \|\Delta_c + \Delta_{\text{ls}}^c\|_F \lesssim (\varepsilon_c + \varepsilon_{\text{ls}})\|\widetilde{A}Z_k\|_F, \\ \|\Delta\widetilde{b}^{(1)}\|_2 &= \|\Delta_b + \Delta_{\text{ls}}^b\|_2 \lesssim (\varepsilon_b + \varepsilon_{\text{ls}})\|\widetilde{b}\|_2. \end{aligned} \tag{3.13}$$

The bound on $\|\Delta C_k\|_F$ comes from the combination of the errors in computing the matrix–matrix product (3.1) and the solution of the least squares problem $\min_y \|\widehat{b} - \widehat{C}_k y\|_F$; these errors are bounded by, respectively,

$$\|\Delta_c\|_F \leq \varepsilon_c \|\widetilde{A}Z_k\|_F \quad \text{and} \quad \|\Delta_{\text{ls}}^c\|_F \leq \varepsilon_{\text{ls}} \|\,\text{fl}(\widetilde{A}Z_k)\|_F \approx \varepsilon_{\text{ls}}\|\widetilde{A}Z_k\|_F.$$

Equivalently, the bound on $\|\Delta\widetilde{b}^{(1)}\|_2$ comes from conditions (3.2) and (3.3).

2. Bound on the left-preconditioned linear system residual. The second part of the proof consists of demonstrating that the computed solution $\widehat{y}_k$ of the least squares problem (3.13) achieves a small enough residual

$$r_k = \widetilde{b} + \Delta\widetilde{b}^{(1)} - (\widetilde{A}Z_k + \Delta C_k)\widehat{y}_k \tag{3.14}$$

for the left-preconditioned linear system. This is the most challenging part of the proof. Fortunately, Paige et al. [36] opened a pathway to achieve this and our approach follows in their footsteps. The core idea is to use the key dimension conditions (3.7) and (3.8) which we developed in section 3.2. The proof then relies on the very useful [36, Thm. 2.4] which gives an upper bound on the residual (3.14). This theorem applied to the inexact least squares problem (3.13) gives for all $\phi > 0$ such that

$$\delta_k = \frac{\sigma_{\min}\big([(\widetilde{b} + \Delta\widetilde{b}^{(1)})\phi, \widetilde{A}Z_k + \Delta C_k]\big)}{\sigma_{\min}\big([\widetilde{A}Z_k + \Delta C_k]\big)} < 1, \tag{3.15}$$

$$\|r_k\|_2^2 \leq \sigma_{\min}^2\big([(\widetilde{b} + \Delta\widetilde{b}^{(1)})\phi, \widetilde{A}Z_k + \Delta C_k]\big)\big(\phi^{-2} + \|\widehat{y}_k\|_2^2/[1 - \delta_k^2]\big). \tag{3.16}$$

Essentially, the following is about choosing a proper ϕ for which we can show that the residual is "small enough". We proceed as in [36, sect. 8.2] and we wish to choose this ϕ such that it satisfies

$$\phi^{-2} = \|\widehat{y}_k\|_2^2/[1 - \delta_k^2] \tag{3.17}$$

which allows a direct simplification of the expression $(\phi^{-2} + \|\widehat{y}_k\|_2^2/[1 - \delta_k^2])$ in the residual bound (3.16). The previous definition (3.17) is equivalent to the following

$$\begin{aligned} &\text{LHS}(\phi) = \text{RHS}(\phi), \\ &\text{LHS}(\phi) \equiv \sigma_{\min}^2\big(\widetilde{A}Z_k + \Delta C_k\big) - \sigma_{\min}^2\big([(\widetilde{b} + \Delta\widetilde{b}^{(1)})\phi, \widetilde{A}Z_k + \Delta C_k]\big), \\ &\text{RHS}(\phi) \equiv \sigma_{\min}^2\big(\widetilde{A}Z_k + \Delta C_k\big)\|\widehat{y}_k\phi\|_2^2. \end{aligned}$$

We now need to show that their exist a ϕ verifying the above and at the same time satisfies $\phi > 0$ and $\delta_k < 1$ such that the quantities are well-defined and the theorem is applicable. For $\phi = 0$, $\text{LHS}(\phi) > \text{RHS}(\phi)$, while for $\phi = \|\widehat{y}_k\|_2^{-1}$, $\text{LHS}(\phi) < \text{RHS}(\phi)$, so by continuity, there exists $\phi \in (0, \|\widehat{y}_k\|_2^{-1})$ satisfying both (3.17) and

$$\delta_k < 1, \quad 0 < \phi < \|\widehat{y}_k\|_2^{-1}. \tag{3.18}$$

The remainder will be about to show that for this value of ϕ the scaled right-hand side satisfies $\|\widetilde{b}\phi\|_2 \approx \|\widetilde{A}Z_k\|_F$, and the quantites $\sigma^2_{\min}([(\widetilde{b}+\Delta\widetilde{b}^{(1)})\phi, \widetilde{A}Z_k + \Delta C_k])$ and δ_k are small. From (3.7) and (3.13), we obtain

$$\begin{aligned}
&\sigma_{\min}\big(\big[(\widetilde{b}+\Delta\widetilde{b}^{(1)})\phi, \widetilde{A}Z_k + \Delta C_k\big]\big) \\
&\quad = \min_{\|w\|_2=1} \big\|\big[(\widetilde{b}+\Delta\widetilde{b}^{(1)})\phi, \widetilde{A}Z_k + \Delta C_k\big]w\big\|_2 \\
&\quad \le \min_{\|w\|_2=1} \big\|\big[\widetilde{b}\phi, \widetilde{A}Z_k\big]w\big\|_2 + \max_{\|w\|_2=1} \big\|\big[\Delta\widetilde{b}^{(1)}\phi, \Delta C_k\big]w\big\|_2 \\
&\quad \le \sigma_{\min}\big(\big[\widetilde{b}\phi, \widetilde{A}Z_k\big]\big) + \big\|\big[\Delta\widetilde{b}^{(1)}\phi, \Delta C_k\big]\big\|_F \\
&\quad \lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\big\|\big[\widetilde{b}\phi, \widetilde{A}Z_k\big]\big\|_F + (\varepsilon_b+\varepsilon_{\text{ls}})\|\widetilde{b}\phi\|_2 \\
&\qquad + (\varepsilon_c+\varepsilon_{\text{ls}})\|\widetilde{A}Z_k\|_F \\
&\quad \le c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})(\|\widetilde{b}\phi\|_2 + \|\widetilde{A}Z_k\|_F),
\end{aligned} \tag{3.19}$$

from which, in addition of (3.17), we revisit (3.16) to give the following bound on the residual

$$\|r_k\|_2^2 \lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})^2(\|\widetilde{b}\phi\|_2 + \|\widetilde{A}Z_k\|_F)^2\phi^{-2}. \tag{3.20}$$

Observing that $\widetilde{b} = r_k + (\widetilde{A}Z_k + \Delta C_k)\widehat{y}_k - \Delta\widetilde{b}^{(1)}$ from (3.14) and using (3.13), (3.18), and (3.20) yields

$$\begin{aligned}
\|\widetilde{b}\phi\|_2 &\le \|r_k\phi\|_2 + \|\widetilde{A}Z_k\widehat{y}_k\phi\|_2 + \|\Delta C_k\widehat{y}_k\phi\|_2 + \|\Delta\widetilde{b}^{(1)}\phi\|_2 \\
&\lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})(\|\widetilde{b}\phi\|_2 + \|\widetilde{A}Z_k\|_F) + \|\widetilde{A}Z_k\|_F,
\end{aligned}$$

from which we obtain

$$\|\widetilde{b}\phi\|_2 \lesssim \frac{(1 + c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}}))}{(1 - c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}}))}\|\widetilde{A}Z_k\|_F \approx \|\widetilde{A}Z_k\|_F. \tag{3.21}$$

Using (3.21), bound (3.19), and the nonsingularity condition (3.8) gives

$$\begin{aligned}
\delta_k &\lesssim \frac{c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})(\|\widetilde{b}\phi\|_2 + \|\widetilde{A}Z_k\|_F)}{\sigma_{\min}(\widetilde{A}Z_k) - \|\Delta C_k\|_F} \\
&\lesssim \frac{c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\|\widetilde{A}Z_k\|_F}{\sigma_{\min}(\widetilde{A}Z_k) - (\varepsilon_c+\varepsilon_{\text{ls}})\|\widetilde{A}Z_k\|_F} \ll 1,
\end{aligned} \tag{3.22}$$

Finally, we can now refine our bound on the residual (3.20); using (3.17), (3.21) and (3.22), we can state

$$\|r_k\|_2 \lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\|\widetilde{A}Z_k\|_F\|\widehat{y}_k\|_2. \tag{3.23}$$

3. Backward error of the left-preconditioned linear system. Now that we have a bound on the residual expressed in terms of $\widetilde{A}$, Z_k, and $\widehat{y}_k$, the third part of the proof will be about retrieving the backward error of the preconditioned system $\widetilde{A}x = \widetilde{b}$. To achieve this, we first want to substitute $\widehat{y}_k$ in (3.23) by $\widehat{x}_k = \text{fl}(Z_k\widehat{y}_k) = Z_k\widehat{y}_k + \Delta_x$. Using assumptions (3.4) and (3.5), we can conclude that

$$\begin{aligned}
\|\widehat{x}_k\|_2 = \|Z_k\widehat{y}_k + \Delta_x\|_2 &\ge \left(\frac{\|Z_k\widehat{y}_k\|_2}{\|\widehat{y}_k\|_2} - \frac{\|\Delta_x\|_2}{\|\widehat{y}_k\|_2}\right)\|\widehat{y}_k\|_2 \\
&\ge \left(\min_y \frac{\|Z_k y\|_2}{\|y\|_2} - \varepsilon_{\text{x}}\|Z_k\|_F\right)\|\widehat{y}_k\|_2 \approx \sigma_{\min}(Z_k)\|\widehat{y}_k\|_2,
\end{aligned} \tag{3.24}$$

which allows us to rework (3.23) as

$$\begin{aligned}
\|r_k\|_2 &\lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\sigma^{-1}_{\min}(Z_k)\|\widetilde{A}Z_k\|_F\|\widehat{x}_k\|_2 \\
&\le c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\sigma^{-1}_{\min}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}(\|\widetilde{A}\|_F\|\widehat{x}_k\|_2 + \|\widetilde{b}\|_2).
\end{aligned} \tag{3.25}$$

In addition, we define

$$\Delta\widetilde{A}^{(1)} = (\Delta C_k\widehat{y}_k - \widetilde{A}\Delta_x)\|\widehat{x}_k\|_2^{-2}\widehat{x}_k^T,$$

which gives, using (3.4),

$$\begin{aligned}(\widetilde{A}+\Delta\widetilde{A}^{(1)})\widehat{x}_k &= \widetilde{A}\widehat{x}_k + \Delta C_k\widehat{y}_k - \widetilde{A}\Delta_x = \widetilde{A}(Z_k\widehat{y}_k+\Delta_x) + \Delta C_k\widehat{y}_k - \widetilde{A}\Delta_x,\\ &= (\widetilde{A}Z_k + \Delta C_k)\widehat{y}_k.\end{aligned}$$

Hence, using (3.14),

$$r_k = \widetilde{b} + \Delta\widetilde{b}^{(1)} - (\widetilde{A}+\Delta\widetilde{A}^{(1)})\widehat{x}_k, \tag{3.26}$$

and, using (3.4), (3.13), and (3.24),

$$\begin{aligned}\|\Delta\widetilde{A}^{(1)}\|_F &\lesssim \left(\varepsilon_c + \varepsilon_{\text{ls}} + \frac{\|\widetilde{A}\|_F\|Z_k\|_F}{\|\widetilde{A}Z_k\|_F}\varepsilon_x\right)\|\widetilde{A}Z_k\|_F\|\widehat{y}_k\|_2/\|\widehat{x}_k\|_2\\ &\lesssim \left(\varepsilon_c + \varepsilon_{\text{ls}} + \frac{\|\widetilde{A}\|_F\|Z_k\|_F}{\|\widetilde{A}Z_k\|_F}\varepsilon_x\right)\sigma_{\min}^{-1}(Z_k)\|\widetilde{A}Z_k\|_F.\end{aligned} \tag{3.27}$$

We now form the quantities

$$\Delta\widetilde{b}^{(2)} = -\frac{\|\widetilde{b}\|_2}{\|\widetilde{A}\|_F\|\widehat{x}_k\|_2 + \|\widetilde{b}\|_2}r_k \quad \text{and} \quad \Delta\widetilde{A}^{(2)} = \frac{\|\widetilde{A}\|_F\|\widehat{x}_k\|_2}{\|\widetilde{A}\|_F\|\widehat{x}_k\|_2 + \|\widetilde{b}\|_2}r_k\frac{\widehat{x}_k^T}{\|\widehat{x}_k\|_2^2}$$

verifying $r_k = \Delta\widetilde{A}^{(2)}\widehat{x}_k - \Delta\widetilde{b}^{(2)}$ and which can be bounded using (3.25) such that

$$\begin{aligned}\|\Delta\widetilde{b}^{(2)}\|_2 &\lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\sigma_{\min}^{-1}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}\|\widetilde{b}\|_2,\\ \|\Delta\widetilde{A}^{(2)}\|_F &\lesssim c(n,k)(\varepsilon_c+\varepsilon_b+\varepsilon_{\text{ls}})\sigma_{\min}^{-1}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}\|\widetilde{A}\|_2.\end{aligned} \tag{3.28}$$

Finally, by replacing r_k by $\Delta\widetilde{A}^{(2)}\widehat{x}_k - \Delta\widetilde{b}^{(2)}$ in (3.26), we can conclude that MOD-GMRES will deliver a computed solution $\widehat{x}_k$ that is the exact solution of the perturbed linear system $(\widetilde{A}+\Delta\widetilde{A})\widehat{x}_k = \widetilde{b}+\Delta\widetilde{b}$ where

$$\Delta\widetilde{A} \equiv \Delta\widetilde{A}^{(1)} + \Delta\widetilde{A}^{(2)} \qquad \text{and} \qquad \Delta\widetilde{b} \equiv \Delta\widetilde{b}^{(1)} + \Delta\widetilde{b}^{(2)}.$$

In addition, the errors $\Delta\widetilde{A}$ and $\Delta\widetilde{b}$ satisfy from the bounds (3.13), (3.27), and (3.28)

$$\begin{aligned}\|\Delta\widetilde{A}\|_F &\lesssim c(n,k)(\alpha\varepsilon_c + \alpha\varepsilon_b + \alpha\varepsilon_{\text{ls}} + \lambda\varepsilon_x)\|\widetilde{A}\|_F,\\ \|\Delta\widetilde{b}\|_F &\lesssim c(n,k)(\alpha\varepsilon_c + \beta\varepsilon_b + \beta\varepsilon_{\text{ls}})\|\widetilde{b}\|_2,\end{aligned}$$

with

$$\alpha = \sigma_{\min}^{-1}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}, \quad \beta = \max\left(1, \sigma_{\min}^{-1}(Z_k)\frac{\|\widetilde{A}Z_k\|_F}{\|\widetilde{A}\|_F}\right), \quad \lambda = \sigma_{\min}^{-1}(Z_k)\|Z_k\|_F.$$

The backward error of the preconditioned system therefore satisfies the bound

$$\frac{\|\widetilde{A}\widehat{x}_k - \widetilde{b}\|_2}{\|\widetilde{A}\|_F\|\widehat{x}_k\|_2 + \|\widetilde{b}\|_2} \lesssim c(n,k)\xi, \tag{3.29}$$

with

$$\xi = \alpha\varepsilon_c + \beta\varepsilon_b + \beta\varepsilon_{\text{ls}} + \lambda\varepsilon_x.$$

4. Forward and backward errors of the original linear system. The last part of the proof consists in deriving bounds on the forward and backward errors of the original system. We can bound the forward error with the backward error (3.29) and the condition number of the preconditioned system. We obtain

$$\frac{\|\widehat{x}_k - x\|_2}{\|x\|_2} \lesssim c(n,k)\xi\kappa_F(\widetilde{A}). \tag{3.30}$$

Additionally, the backward error of the original system can be bounded by using $b - A\widehat{x}_k = M_L(\widetilde{b} - \widetilde{A}\widehat{x}_k)$; we have

$$\begin{aligned}\|b - A\widehat{x}_k\|_2 &\lesssim c(n,k)\xi\|M_L\|_F(\|\widetilde{b}\|_2 + \|\widetilde{A}\|_F\|\widehat{x}_k\|_2)\\ &\leq c(n,k)\xi\left(\kappa_F(M_L)\|b\|_2 + \frac{\|M_L\|_F\|\widetilde{A}\|_F}{\|A\|_F}\|A\|_F\|\widehat{x}_k\|_2\right).\end{aligned} \tag{3.31}$$

This last bound in addition to the fact that $\|M_L\|_F\|\widetilde{A}\|_F/\|A\|_F \leq \kappa_F(M_L)$ implies in particular

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)\xi\kappa_F(M_L). \tag{3.32}$$

which ends the proof. □

A few points are worth noticing from the forward and backward error bounds (3.9) and (3.10) in Theorem 3.1. First, these bounds depend on the accuracy parameters ε_c, ε_b, ε_{ls}, and ε_x associated with each operation in Algorithm 1. While this conclusion was expected, it is interesting to confirm that the errors made in each of the four operations at lines 1 to 4 play a relatively identical role in the final attainable errors of the computed solution to (1.1) and that, therefore, none of them should be neglected. Second, right- and left-preconditioning badly affect the bounds for both forward and backward errors. Indeed, the terms $\kappa_F(M_L)$ and $\sigma_{\min}^{-1}(Z_k)\|Z_k\|_F$ can be substantially larger than 1 and increase the bounds (3.9) and (3.10). This key observation that preconditioning deteriorates the attainable accuracies might sound counterintuitive. This is because preconditioning is generally associated with improved numerical behavior in the sense that it accelerates convergence if the preconditioners are well-chosen. This benefit still applies in the finite precision world but comes at the cost of a loss of stability: we converge faster but not further. To illustrate this loss of stability, let us assume that the accuracy parameters are of order the unit roundoff of the machine precision (i.e., $\varepsilon_c \equiv \varepsilon_b \equiv \varepsilon_{\text{ls}} \equiv \varepsilon_x \equiv c(n,k)u$), that the absence of preconditioning implies $M_L \equiv I$ and $Z_k \equiv \widehat{V}_k$ where $\widehat{V}_k$ is the near orthogonal computed Arnoldi Krylov basis, and that we meet all the required assumptions of Theorem 3.1. The bound on the backward error becomes

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u, \tag{3.33}$$

and the process is therefore backward stable. Now consider that we apply in a split-preconditioning fashion a left- and a right-preconditioner such that $M_L \neq I$ and $Z_k \equiv \text{fl}(M_R^{-1}\widehat{V}_k) \approx M_R^{-1}\widehat{V}_k$, where $M_L, M_R \in \mathbb{R}^{n\times n}$ are nonsingular. Observing that $\sigma_{\min}^{-1}(Z_k)\|Z_k\|_F \lesssim \kappa_F(M_R)$, the backward error bound becomes

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u\kappa_F(M_R)\kappa_F(M_L),$$

which is substantially higher than the previous bound (3.33). In particular, we lost the backward stability property since the backward error is not guaranteed anymore to be of order u. Fortunately, this problem can be overcome, and we explain in section 4 how a restart process can recover the backward stability.

4. Backward error analysis of restarted MOD-GMRES

The cost in execution time and memory consumption of GMRES algorithms grows with the size k of the basis Z_k. By reframing GMRES to make use of multiple smaller bases, restarted GMRES algorithms intend to bound this cost while still providing a good approximation to the solution of (1.1). For instance, restarting can be very convenient or even necessary for solving extremely large sparse linear systems where only a few dense vectors can be stored in memory at once.

The MOD-GMRES framework alone, presented and studied in section 3, cannot cover restarted variants of GMRES. Therefore, for our framework to account for restarted GMRES

algorithms, we introduce and study a new abstract algorithm called restarted MOD-GMRES represented by Algorithm 2. Roughly, the algorithm consists of the successive applications of MOD-GMRES where the $(i+1)$th call of MOD-GMRES uses the solution of the ith call as a starting vector. It can also be interpreted as the computation of successive corrections d_i obtained as the solutions of the linear systems $Ad_i = r_i$ solved through MOD-GMRES. We repeat the process until we are satisfied with the quality of the computed solution. Note that with restarted MOD-GMRES the bases $Z_{k_i}^{(i)} \in \mathbb{R}^{n\times k_i}$, their sizes k_i, the left-preconditioners $M_L^{(i)} \in \mathbb{R}^{n\times n}$, and the accuracy parameters $\varepsilon_c^{(i)}$, $\varepsilon_b^{(i)}$, $\varepsilon_{\text{ls}}^{(i)}$, and $\varepsilon_x^{(i)}$ are allowed to differ from a restart iteration to another.

Algorithm 2 Restarted MOD-GMRES

Input: a matrix $A \in \mathbb{R}^{n\times n}$, a right-hand side $b \in \mathbb{R}^n$, a set of bases $(Z_{k_i}^{(i)})_i$ of size $n \times k_i$, and a set of left-preconditioners $(M_L^{(i)})_i$ of size $n \times n$.
Output: a computed solution to $Ax = b$.

1: Initialize x_0
2: **repeat**
3: Compute the residual $r_i = b - Ax_i$.
4: Compute $C_{k_i}^{(i)} = \widetilde{A}^{(i)} Z_{k_i}^{(i)}$ where $\widetilde{A}^{(i)} = (M_L^{(i)})^{-1} A$.
5: Compute $\widetilde{r}_i = (M_L^{(i)})^{-1} r_i$.
6: Solve $y_i = \arg\min_y \|\widetilde{r}_i - C_{k_i}^{(i)} y\|_2$.
7: Compute the correction $d_i = Z_{k_i}^{(i)} y_i$.
8: Compute the next iterate $x_{i+1} = x_i + d_i$.
9: $i = i + 1$
10: **until** convergence

Equivalently as for MOD-GMRES, deriving a backward error analysis for restarted MOD-GMRES requires some assumptions on the operations in Algorithm 2. Lines 4 to 7 in Algorithm 2 correspond to the applications of MOD-GMRES to the linear systems $Ad_i = r_i$. We need these lines to satisfy the assumptions (3.1) to (3.8) described in section 3 at a key dimension k_i for all restart iterations $i \geq 1$ and for given accuracy parameters $\varepsilon_c^{(i)}$, $\varepsilon_b^{(i)}$, $\varepsilon_{\text{ls}}^{(i)}$, and $\varepsilon_x^{(i)}$. In addition, we require the following two assumptions for, respectively, the computation of the residual at line 3 and the computation of the next iterate at line 8.

For $b, \widehat{x}_i \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$ we suppose that the computed residual $\widehat{r}_i$ at line 3 of Algorithm 2 satisfies for all $i \geq 1$

$$\widehat{r}_i = b - A\widehat{x}_i + \Delta r_i, \quad \|\Delta r_i\|_2 \leq \varepsilon_r(\|b\|_2 + \|A\|_F \|\widehat{x}_i\|_2), \tag{4.1}$$

where $\Delta r_i \in \mathbb{R}^n$ is the error introduced while computing the matrix–vector product and the vector subtraction and ε_r is a parameter bounding the magnitude of this error.

In addition, for $\widehat{x}_i, \widehat{d}_i \in \mathbb{R}^n$ we suppose that the computation of the next iterate $\widehat{x}_{i+1}$ at line 8 of Algorithm 2 yields for all $i \geq 1$

$$\widehat{x}_{i+1} = \widehat{x}_i + \widehat{d}_i + \Delta x_i, \quad \|\Delta x_i\|_2 \leq \varepsilon_u \|\widehat{x}_{i+1}\|_2, \tag{4.2}$$

where $\Delta x_i \in \mathbb{R}^n$ is the computing error generated from the vector addition and ε_u is a parameter bounding the magnitude of this error. It is very likely that $\varepsilon_u \equiv u$ where u is the unit roundoff of the arithmetic precision used to compute line 8. We are not aware of a relevant implementation of restarted MOD-GMRES that would provide a different outcome for ε_u, but we let our framework the possibility to handle this eventuality.

Finally, for the same reasons we require condition (3.6) for MOD-GMRES, we also require

$$0 \leq \varepsilon_r, \varepsilon_u \ll 1. \tag{4.3}$$

Under these previous assumptions, we can guarantee that restarted MOD-GMRES will provide a solution whose backward and forward errors are bounded by functions of the accuracy parameters ε_r and ε_u. We summarize this result in the following Theorem 4.1.

Theorem 4.1. *Consider the solution of $Ax = b$ with Algorithm 2. Suppose that for all $i \geq 1$ lines 3 and 8 of Algorithm 2 satisfy conditions (4.1) to (4.3) for given parameters ε_r and ε_u, and lines 4 to 7 satisfy conditions (3.1)–(3.8) of Theorem 3.1 for given parameters $\varepsilon_c^{(i)}$, $\varepsilon_b^{(i)}$, $\varepsilon_{ls}^{(i)}$, and $\varepsilon_x^{(i)}$. Then as long as*

$$\Lambda_1^{(i)} = c(n,k)\xi^{(i)}\|M_L^{(i)}\|_F\|\widetilde{A}^{(i)}\|_F\|A^{-1}\|_F \ll 1 \tag{4.4}$$

and

$$\Lambda_2^{(i)} = c(n,k)\xi^{(i)}\kappa_F(\widetilde{A}^{(i)}) \ll 1, \tag{4.5}$$

the backward and forward errors are reduced at the iteration i, respectively, by factors (at least) $\Lambda_1^{(i)}$ and $\Lambda_2^{(i)}$ until they satisfy

$$\frac{\|b - A\widehat{x}\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}\|_2} \lesssim c(n,k)\varepsilon_r + \varepsilon_u \tag{4.6}$$

and

$$\frac{\|\widehat{x} - x\|_2}{\|x\|_2} \lesssim c(n,k)\varepsilon_r\kappa_F(A) + \varepsilon_u, \tag{4.7}$$

where

$$\xi^{(i)} = \alpha^{(i)}\varepsilon_c^{(i)} + \beta^{(i)}\varepsilon_b^{(i)} + \beta^{(i)}\varepsilon_{ls}^{(i)} + \lambda^{(i)}\varepsilon_x^{(i)} \tag{4.8}$$

with

$$\begin{gathered}\alpha^{(i)} = \sigma_{\min}^{-1}(Z_{k_i}^{(i)})\frac{\|\widetilde{A}^{(i)}Z_{k_i}^{(i)}\|_F}{\|\widetilde{A}^{(i)}\|_F}, \quad \beta^{(i)} = \max\left(1, \sigma_{\min}^{-1}(Z_{k_i}^{(i)})\frac{\|\widetilde{A}^{(i)}Z_{k_i}^{(i)}\|_F}{\|\widetilde{A}^{(i)}\|_F}\right), \\ \lambda^{(i)} = \sigma_{\min}^{-1}(Z_{k_i}^{(i)})\|Z_{k_i}^{(i)}\|_F,\end{gathered} \tag{4.9}$$

and where $c(n,k)$ accounts for polynomials in n and k of low degrees.

Proof. The proof consists in noticing that Algorithm 2 is an iterative refinement process since it can be directly rewritten as the repetition of the three following steps:

1: Compute the residual $r_i = b - Ax_i$.
2: Solve $Ad_i = r_i$ with MOD-GMRES for given $M_L^{(i)}$ and $Z_{k_i}^{(i)}$.
3: Update the solution $x_{i+1} = x_i + d_i$.

Hence, iterative refinement backward error analyses are applicable to Algorithm 2. For this proof, we use the backward error analysis of Carson and Higham [14]. More specifically, we shall apply [14, Thm. 3.2] and [14, Thm. 4.1] on Algorithm 2 to prove that the restarted MOD-GMRES process delivers the attainable backward and forward errors (4.6) and (4.7) under the convergence conditions (4.4) and (4.5).

Technically, the results of these theorems are based on the assumption that the residual at line 3 is computed through a standard matrix–vector product in precision of unit roundoff u_r and satisfies

$$\widehat{r}_i = b - A\widehat{x}_i + \Delta r_i, \quad |\Delta r_i| \leq \gamma_n^r(|b| + |A||\widehat{x}_i|). \tag{4.10}$$

Identically, the computation of the next iterate at line 8 is assumed to be computed in precision of unit roundoff u and satisfies

$$\widehat{x}_{i+1} = \widehat{x}_i + \widehat{d}_i + \Delta x_i, \quad |\Delta x_i| \leq u|\widehat{x}_{i+1}|. \tag{4.11}$$

In addition, these theorems deliver convergence conditions and bounds on the attainable errors in the infinity norm. This departs from our conditions (4.1) and (4.2) on the computation of the residual and the next iterate which are normwise instead of componentwise, and with accuracy parameters ε_r and ε_u instead of unit roundoffs u_r and u. It also departs from our resulting convergence conditions (4.4) and (4.5), and our bounds on the attainable errors (4.6) and (4.7) which are in 2-norm and Frobenius norm instead of infinity norm. Nevertheless, the theorems and the analysis of [14] can be straightforwardly adapted to the case where (4.10) and (4.11) are exchanged with (4.1) and (4.2), and to the case where Frobenius norm is used instead of infinity norm. We assume these adjustments in the following.

To apply [14, Thm. 3.2] and [14, Thm. 4.1], we need to show that the computed correction $\widehat{d}_i$ by MOD-GMRES satisfies, for all $i \geq 1$,

$$\begin{aligned} \widehat{d}_i &= (I + u_i E_i) d_i, \quad u_i \|E_i\|_F < 1, \\ \|\widehat{r}_i - A\widehat{d}_i\|_2 &\leq u_i(\omega_i \|A\|_F \|\widehat{d}_i\|_2 + w_i \|\widehat{r}_i\|_2), \end{aligned} \tag{4.12}$$

for given u_i, E_i, ω_i, and w_i; see conditions [14, eqs. (2.3) and (2.4)]. Under these conditions, supposing in addition that $\omega_i \kappa_F(A) u_i \ll 1$, and using a quantity μ_i defined as $\|A(x - \widehat{x}_i)\|_2 = \mu_i \|A\|_F \|x - \widehat{x}_i\|_2$, [14, Thm. 3.2] guarantees that the computed solution $\widehat{x}_{i+1}$ at the $(i+1)$th restart satisfies

$$\begin{aligned} \|x - \widehat{x}_{i+1}\|_2 &\leq \Lambda_2^{(i)} \|x - \widehat{x}_i\|_2 + \lambda_2^{(i)}, \\ \Lambda_2^{(i)} &= 2u_i \kappa_F(A) \mu_i + u_i \|E_i\|_F, \\ \lambda_2^{(i)} &= 2(1 + u_i) \varepsilon_r \kappa_F(A)(\|x\|_2 + \|\widehat{x}_i\|_2) + \varepsilon_u \|\widehat{x}_{i+1}\|_2, \end{aligned} \tag{4.13}$$

and [14, Thm. 4.1] guarantees

$$\begin{aligned} \|b - A\widehat{x}_{i+1}\|_2 &\leq \Lambda_1^{(i)} \|b - A\widehat{x}_i\|_2 + \lambda_1^{(i)}, \\ \Lambda_1^{(i)} &= u_i \left(1 + (1 + u_i) \frac{\omega_i \kappa_F(A) + w_i}{1 - \omega_i \kappa_F(A) u_i}\right), \\ \lambda_1^{(i)} &= \left(1 + \frac{u_i(\omega_i \kappa_F(A) + w_i)}{1 - \omega_i \kappa_F(A) u_i}\right)(1 + u_i) \varepsilon_r (\|b\|_2 + \|A\|_F \|\widehat{x}_i\|_2) + \varepsilon_u \|A\|_F \|\widehat{x}_{i+1}\|_2. \end{aligned} \tag{4.14}$$

It is explained in [14] or [47, sect. 4.2.2] that μ_i is expected to be small and that $2u_i \kappa_F(A) \mu_i$ is negligible in front of $u_i \|E_i\|_F$ in the expression of $\Lambda_2^{(i)}$ in (4.13). Assuming that $u_i \ll 1$ for all $i \geq 1$ and by dropping second order terms in (4.13) and (4.14), we guarantee that if for all $i \geq 1$

$$\Lambda_2^{(i)} \approx u_i \|E_i\|_F \ll 1 \qquad \text{and} \qquad \Lambda_1^{(i)} \approx u_i(\omega_i \kappa_F(A) + w_i) \ll 1, \tag{4.15}$$

then the forward and backward errors are improved respectively by factors (at least) $\Lambda_2^{(i)}$ and $\Lambda_1^{(i)}$ at each iteration i until they reach their bounds on the attainable errors

$$\frac{\|\widehat{x} - x\|_2}{\|x\|_2} \lesssim c(n,k) \varepsilon_r \kappa_F(A) + \varepsilon_u \qquad \text{and} \qquad \frac{\|b - A\widehat{x}\|_2}{\|b\|_2 + \|A\|_F \|\widehat{x}\|_2} \lesssim c(n,k) \varepsilon_r + \varepsilon_u.$$

From (3.30), (3.31), and (3.32) we conclude that condition (4.12) is met for

$$\begin{aligned} u_i \equiv \xi^{(i)}, \quad \|E_i\|_F \equiv c(n,k) \kappa_F(\widetilde{A}^{(i)}), \quad \omega_i \equiv c(n,k) \frac{\|M_L^{(i)}\|_F \|\widetilde{A}^{(i)}\|_F}{\|A\|_F}, \\ w_i \equiv c(n,k) \kappa_F(M_L^{(i)}), \end{aligned}$$

where $\xi^{(i)}$ is defined by (4.8). Hence, by using these values of u_i, E_i, ω_i, and w_i in (4.15) and by observing that $\omega_i \kappa_F(A) \geq w_i$, we identify

$$\Lambda_2^{(i)} \approx c(n,k) \xi^{(i)} \kappa_F(\widetilde{A}^{(i)}) \ll 1 \qquad \text{and} \qquad \Lambda_1^{(i)} \approx c(n,k) \xi^{(i)} \|M_L^{(i)}\|_F \|\widetilde{A}^{(i)}\|_F \|A^{-1}\|_F \ll 1,$$

which ends the proof. □

The result of Theorem 4.1 can be interpreted as follows: if at each restart MOD-GMRES can compute a correction $\widehat{d}_i$ with a few correct digits, Algorithm 2 will eventually improve the computed solution $\widehat{x}_i$ to its maximal attainable accuracy defined by (4.6) and (4.7). In addition, because the model allows $M_L^{(i)}$, $Z_{ki}^{(i)}$, $\varepsilon_c^{(i)}$, $\varepsilon_b^{(i)}$, $\varepsilon_{\text{ls}}^{(i)}$, and $\varepsilon_x^{(i)}$ to change from an iteration to another, the theorem is applicable to algorithms where the GMRES variant can be modified between each restart iteration (e.g., switch from MGS-GMRES to FGMRES after a restart).

It is essential to remark that the bounds on the attainable errors of MOD-GMRES ((3.9) and (3.10)) and restarted MOD-GMRES ((4.6) and (4.7)) are different. Namely, compared with MOD-GMRES, the bounds of restarted MOD-GMRES depends solely on the accuracy

parameters ε_r and ε_u, and not on $M_L^{(i)}$, $Z_{ki}^{(i)}$, $\varepsilon_c^{(i)}$, $\varepsilon_b^{(i)}$, $\varepsilon_{\text{ls}}^{(i)}$, and $\varepsilon_x^{(i)}$ which only affect the convergence conditions (4.4) and (4.5). This has major implications, namely if ε_r and ε_u are of order the unit roundoff of the machine precision u and if the convergence condition (4.4) as well as the other assumptions of Theorem 4.1 are met, restarted MOD-GMRES is backward stable regardless of the used preconditioners and the accuracies at which lines 4 to 7 are computed. In other words, restarting can make a non-backward stable GMRES variant backward stable. One can also exploit this property to enhance the computing performance of GMRES algorithms. For instance, computing lines 4 to 7 in cheaper IEEE fp32 single precision while computing lines 3 and 8 in IEEE fp64 double precision would enable the algorithm to deliver a double precision accuracy solution in potentially less time and memory, regardless of the fact that most of the flops are carried out in low accuracy IEEE fp32 arithmetic. This type of mixed precision approach has been proposed by Turner and Walker [46] and we provide more details on how to apply our framework to this particular mixed precision GMRES in section 6.3.

The statement that restarted GMRES is more stable than GMRES without restart might appear contradictory at first. Indeed, compared with restarted GMRES, GMRES can build higher dimensional Krylov subspace better fitted to deal with numerically difficult problems. Moreover, it is well-known that implementations of restarted GMRES imposing a fixed maximum size m ($k_i \leq m \ll n$) on the bases $Z_{ki}^{(i)}$ are not always able to provide correct solutions for those difficult problems due to the limited size of the Krylov spaces used. It is however important to understand that our claim is different and does not invalidate the previous statement. In particular, those "classical" restarted GMRES implementations using a fixed maximum number of iterations m as a stopping criterion do not meet the key dimension conditions (3.7) and (3.8) and ultimately do not enjoy the stability results of Theorem 4.1. The reason stems from the fact that we cannot guarantee the solution to be improved under a given fixed number of iterations since "any nonincreasing convergence curve is possible for GMRES" [23]. However, with slightly less practical restart criteria, we can build stable restarted versions of GMRES. It is the case, for example, when the restart criterion is based on a tolerance τ on the backward error of the (left-preconditioned) linear system, that is, we restart when we have computed $\widehat{d}_i$ satisfying

$$\frac{\|\widetilde{r}_i - \widetilde{A}^{(i)}\widehat{d}_i\|_2}{\|\widetilde{r}_i\|_2 + \|\widetilde{A}^{(i)}\|_F\|\widehat{d}_i\|_2} \leq \tau;$$

see [47, Thm. 5.3] for an example of stability result with a restarted left-preconditioned GMRES with tolerance. While this criterion cannot ensure that the size of the bases will stay bounded, the tolerance τ can be set to a very large value such that the bases can still be expected to remain empirically small. Overall, while our result cannot be directly applied to implementations of restarted GMRES using a fixed maximum number of iterations m as a restart criterion, it instead participates in the better understanding of those methods. In particular, it can give stability results for a very close restarted GMRES algorithm where this criterion is relaxed.

5. Application and compatibility of the framework with previous error analyses

In this section, we show how our framework can be used to derive backward error analyses for HH-GMRES, MGS-GMRES, and FGMRES mentioned in section 1 and for which we already have analyses in the literature. Our aim is twofold: firstly, we want to demonstrate pedagogically how Theorem 3.1 can be used and, secondly, we want to show that our framework is compatible with all the major existing analyses, namely it delivers (almost) the same error bounds under (almost) the same conditions up to some differences in the constants related to the problem dimension n and the size of the basis k.

5.1. On the application of the MOD-GMRES framework.

In order to apply the MOD-GMRES framework, the basis Z_k, the left-preconditioner M_L, and the operations at

lines 1 to 4 of Algorithm 1 have to be specialized to describe the GMRES algorithm of interest. Once MOD-GMRES is specialized, we have access to more information on Z_k, M_L, and how the operations are computed, and we can determine the accuracy parameters ε_c, ε_b, ε_{ls}, and ε_x. The task of determining these accuracy parameters requires backward error analysis results for the given specialized operations used at each line in Algorithm 1. Note that, since rounding error analyses of the most "standard" matrix–matrix product, least squares solver, and matrix–vector product algorithms already exist in the literature, this process can be relatively straightforward. With these accuracy parameters determined, we can check whether the framework assumptions (3.1) to (3.8) are met. Under those assumptions, Theorem 3.1 holds and can be used to derive bounds for the attainable backward and forward errors of the GMRES algorithm of interest. To this end the modular backward and forward error bounds (3.9) and (3.10) of Theorem 3.1 are to be specialized by using the determined ε_c, ε_b, ε_{ls}, ε_x, Z_k and M_L.

Overall, our framework reduces the process of deriving new error bounds for GMRES to the four following tasks: (1) specialize the four operations of MOD-GMRES to describe the algorithm of interest; (2) determine the parameters ε_c, ε_b, ε_{ls}, and ε_x for each of these operations; (3) verify that the assumptions (3.1) to (3.8) are met; (4) apply Theorem 3.1 and specialize the backward and forward error bounds (3.9) and (3.10). The approach is identical for the restarted MOD-GMRES framework.

5.2. **HH-GMRES.** As in [18], we study HH-GMRES run in precision of unit roundoff u. To apply our framework, we can specialize MOD-GMRES to HH-GMRES as follows. We specify $M_L \equiv I$ and $Z_k \equiv \widehat{V}_k$ since we assume no preconditioning. The matrix $\widehat{V}_k$ is the Krylov basis computed by the Arnoldi process using the Householder orthogonalization. In exact arithmetic, the kth step of the Arnoldi process can be viewed as a column-oriented Householder QR factorization of the matrix $[b, AV_k]$ delivering the following recurrence

$$\big[b, AV_k\big] = V_{k+1}R_{k+1}, \quad R_{k+1} = \big[\beta e_1, \bar{H}_k\big], \tag{5.1}$$

where $\bar{H}_k \in \mathbb{R}^{(k+1)\times k}$ is upper Hessenberg and $\beta = \|b\|_2$. The least squares problem at line 3 of Algorithm 1 is then solved by computing the solution of the transformed least squares problem $\min_y \|\beta e_1 - \bar{H}_k y\|_2$ with Givens rotations. Assuming that $\widehat{V}_k$ is explicitly formed and stored in memory, the products at lines 1 and 4 are made from standard matrix–vector products with A and $\widehat{V}_k$, respectively. Using our framework, we recover the result of Drkošová et al. [18] and show, in particular, that HH-GMRES is backward stable; we summarize our conclusion by the following theorem.

Theorem 5.1. *Consider solving $Ax = b$ with HH-GMRES run in precision of unit roundoff $u \ll 1$. As long as the system is not numerically singular, that is,*

$$\sigma_{\min}(A) \gg u\|A\|_F, \tag{5.2}$$

then there exists an iteration $k \leq n$ such that

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u, \tag{5.3}$$

where $c(n,k)$ is a polynomial in n and k of low degree.

Proof. To use Theorem 3.1 to derive (5.3) under (5.2), we need to show that the conditions (3.1) to (3.8) are met for given accuracy parameters ε_c, ε_b, ε_{ls}, and ε_x at a given iteration k. We will first show that conditions (3.1) to (3.6) and (3.8) are met for all iterations $k \leq n$ and for accuracy parameters that we will identify. Subsequently, we will demonstrate that at $k = n$ we meet condition (3.7) and Theorem 3.1 is applicable.

1. Orthogonality of $\widehat{V}_k$. First and foremost, we need to exploit one major property of the Householder orthogonalization: it preserves the orthogonality of the computed basis $\widehat{V}_k$ for all $k \leq n$. From [25, Thm. 19.4] and the rest of the comments in [25, p. 360] the basis $Z_k \equiv \widehat{V}_k$, which corresponds to the computed "reduced-size" Q-factor by the Householder

orthogonalization process, satisfies for all $k \leq n$

$$\widehat{V}_k = \widetilde{V}_k(I_k + \Delta I_k), \quad \|\Delta I_k e_j\|_2 \leq \widetilde{\gamma}_{n^2}, \quad \forall j = 1:k, \tag{5.4}$$

where $\widetilde{V}_k$ is an exactly orthogonal matrix and $I_k, \Delta I_k \in \mathbb{R}^{k\times k}$. It follows that the smallest singular value of $\widehat{V}_k$ stays close to 1. More precisely we have

$$\begin{aligned} \sigma_{\min}(\widehat{V}_k) &= \min_{\|x\|_2=1} \|\widetilde{V}_k(I_k + \Delta I_k)x\|_2 \geq \sigma_{\min}(\widetilde{V}_k) - \|\widetilde{V}_k \Delta I_k\|_F \geq 1 - n^{\frac{1}{2}}\widetilde{\gamma}_{n^2}, \\ \sigma_{\min}(\widehat{V}_k) &\leq \sigma_{\min}(\widetilde{V}_k) + \|\widetilde{V}_k \Delta I_k\|_F \leq 1 + n^{\frac{1}{2}}\widetilde{\gamma}_{n^2}, \end{aligned} \tag{5.5}$$

from which we deduce, by dropping second order terms,

$$\forall k \leq n \quad \sigma_{\min}(\widehat{V}_k) \approx 1, \quad \|\widehat{V}_k\|_F \approx k^{\frac{1}{2}}, \quad \text{and} \quad \|A\widehat{V}_k\|_F \approx \|A\|_F. \tag{5.6}$$

2. Identifying ε_c. Let us now consider the standard matrix–matrix product $\widehat{C}_k = \mathrm{fl}(A\widehat{V}_k)$ corresponding to line 1 of Algorithm 1. We note $\dot{V}_k = [\dot{v}_1, \ldots, \dot{v}_k]$ which is $\widehat{V}_k$ with its columns correctly normalized; that is, for $j \leq k$,

$$\begin{aligned} \widehat{v}_j &= \dot{v}_j + \Delta v_j, \quad \|\Delta v_j\|_2 \leq \widetilde{\gamma}_n, \\ \widehat{V}_k &= \dot{V}_k + \Delta V_k, \quad \Delta V_k = [\Delta v_1, \ldots, \Delta v_k], \end{aligned} \tag{5.7}$$

where Δv_j is the error for the normalization of $\widehat{v}_j$ and ΔV_k is the accumulated error for the normalization of the basis at step k. By [25, eq. (3.11)] and (5.7), we have

$$\widehat{c}_j = \mathrm{fl}(A\widehat{v}_j) = (A + \Delta A)\widehat{v}_j = A\widehat{v}_j + \Delta_{c_j},$$

where

$$\|\Delta_{c_j}\|_2 \leq \widetilde{\gamma}_n \|A(\dot{v}_j + \Delta v_j)\|_2 \lesssim \widetilde{\gamma}_n \|A\|_F$$

since $\|\dot{v}_j\|_2 = 1$. We therefore obtain for all $k \leq n$

$$\widehat{C}_k = A\widehat{V}_k + \Delta_c, \qquad \|\Delta_c\|_F \lesssim k^{\frac{1}{2}}\widetilde{\gamma}_n \|A\|_F, \tag{5.8}$$

where Δ_c is the error on the matrix–matrix product at line 1. From (5.8), we identify $\varepsilon_c \equiv k^{1/2}\widetilde{\gamma}_n \|A\|_F / \|A\widehat{V}_k\|_F$ for which assumption (3.1) is satisfied.

3. Identifying ε_b. In HH-GMRES, no left-preconditioner is used (i.e., $M_L = I$). Therefore, there is no error in forming the left-preconditioned right-hand side at line 2. We have $\Delta_b = 0$ and $\varepsilon_b \equiv 0$, and assumption (3.2) is straightforwardly satisfied.

4. Identifying $\varepsilon_{\mathrm{ls}}$. We turn our attention to the error generated while solving the least squares problem $\min_y \|b - \widehat{C}_k y\|_2$ at line 3 with the Householder Arnoldi algorithm. For conciseness and readability, we present the outcome of the backward error analysis of this process in Theorem A.1 in the appendix. For all $k \leq n$, accounting for (5.2) and (5.6), $\widehat{C}_k$ satisfies condition (A.2) of Theorem A.1 which is therefore applicable. Hence, the computed solution $\widehat{y}_k$ satisfies

$$\begin{aligned} \widehat{y}_k &= \arg\min_y \|(b + \Delta^b_{\mathrm{ls}}) - (\widehat{C}_k + \Delta^c_{\mathrm{ls}})y\|_2, \\ \|[\Delta^b_{\mathrm{ls}}, \Delta^c_{\mathrm{ls}}]e_j\|_2 &\lesssim \widetilde{\gamma}_{nk+2(n+k)-2}\|[b, \widehat{C}_k]e_j\|_2, \quad j \leq k+1, \end{aligned}$$

from which we identify $\varepsilon_{\mathrm{ls}} \equiv \widetilde{\gamma}_{nk+2(n+k)-2}$ such that assumption (3.3) is satisfied.

5. Identifying ε_x. Then, we consider the error made while computing the action of the basis $\widehat{V}_k$ on $\widehat{y}_k$ at line 4. Depending on the implementation of HH-GMRES, this operation might be computed from the Householder vectors without the need to store the basis $\widehat{V}_k$ explicitly in memory. However, for simplicity we consider the case where the computed basis $\widehat{V}_k$ is formed explicitly and where line 4 is computed by a standard matrix–vector product with $\widehat{V}_k$. In this case, from [25, eq. (3.11)] the product $\widehat{V}_k\widehat{y}_k$ satisfies for all $k \leq n$

$$\mathrm{fl}(\widehat{V}_k\widehat{y}_k) = (\widehat{V}_k + \Delta V_k)\widehat{y}_k, \quad \|\Delta V_k\|_F \leq \gamma_n \|\widehat{V}_k\|_F,$$

and we identify $\Delta_x \equiv \Delta V_k \widehat{y}_k$ and $\varepsilon_x \equiv \gamma_n$ for which assumption (3.4) is satisfied.

6. Existence of the key dimension. Condition (3.5) is guaranteed by (5.6). Using (5.6) again we obtain for all $k \leq n$,

$$\varepsilon_c \equiv k^{\frac{1}{2}}\widetilde{\gamma}_n \|A\|_F / \|A\widehat{V}_k\|_F \approx k^{\frac{1}{2}}\widetilde{\gamma}_n \ll 1,$$

which guarantees that condition (3.6) is met. Finally, since $\sigma_{\min}(A\widehat{V}_k) \geq \sigma_{\min}(A)\sigma_{\min}(\widehat{V}_k) \approx \sigma_{\min}(A)$ from (5.6), condition (3.8) is met for all $k \leq n$ under assumption (5.2). Moreover, $[b\phi, A\widehat{V}_k]$ is rank deficient for $k = n$ since it is composed of $n+1$ vectors of dimension n, and condition (3.7) is trivially met.

7. Application of Theorem 3.1. Therefore, at the iteration $k = n$ HH-GMRES meets all the conditions of Theorem 3.1 which is applicable for $\varepsilon_c \approx k^{1/2}\widetilde{\gamma}_n$, $\varepsilon_b \equiv 0$, $\varepsilon_{\text{ls}} \equiv \widetilde{\gamma}_{nk+2(n+k)-2}$, and $\varepsilon_x \equiv \gamma_n$. Using (5.6) we identify in (3.11)

$$\alpha \approx \beta \approx 1, \qquad \lambda \approx k^{\frac{1}{2}}, \qquad \text{and} \qquad \xi \approx c(n,k)u,$$

which reduces the backward error bound (3.9) of Theorem 3.1 to (5.3) and ends the proof. □

5.3. **MGS-GMRES.** Compared with HH-GMRES, MGS-GMRES uses the MGS orthogonalization instead of the Householder one to solve the least squares problem at line 3 of Algorithm 1 and, apart from this slight variation, the operations of MGS-GMRES are identical to those of HH-GMRES described in section 5.2. However, this change is not insignificant since, unlike HH-GMRES, the computed Krylov basis $\widehat{V}_k$ by MGS-GMRES faces loss of orthogonality. This phenomenon can be explained as follows. The Krylov basis computed by MGS-GMRES satisfies the Arnoldi iterative process (5.1), where $\widehat{V}_{k+1}$ is the Q-factor of the QR decomposition of $[b, \widehat{C}_k]$ computed by MGS. Assuming the MGS orthogonalization is run in precision of unit roundoff u, [25, Thm. 19.13] ensures that the computed $\widehat{V}_{k+1}$ satisfies

$$\|I - \widehat{V}_{k+1}^T \widehat{V}_{k+1}\|_F \leq c(n,k)u\kappa_F\big([b, \widehat{C}_k]\big), \tag{5.9}$$

whereas the Householder orthogonalization provides

$$\|I - \widehat{V}_{k+1}^T \widehat{V}_{k+1}\|_F \leq c(n,k)u. \tag{5.10}$$

As can be seen, the orthogonality of $\widehat{V}_{k+1}$ is dependent on the condition number of $[b, \widehat{C}_k]$ with MGS. Unfortunately, as we get closer to the solution the right-hand side b lies more and more in the range of AZ_k, the matrix $[b, \widehat{C}_k]$ becomes nearly singular, its condition number grows, and the upper bound in (5.9) becomes very large. Overall, as MGS-GMRES converges to the solution its computed basis will most likely fully lose its orthogonality. This phenomenon is a major challenge to proving the backward stability of MGS-GMRES. To relate to the proof of Theorem 5.1 on the backward stability of HH-GMRES, the loss of orthogonality invalidates the statement (5.4) and ultimately (5.6) is not guaranteed to hold anymore for all $k \leq n$. As a result, the reasoning carried out for HH-GMRES does not extend straightforwardly to MGS-GMRES.

In the remainder of this section, we apply our framework on MGS-GMRES and we recover the backward stability result of Paige et al. [36]. In particular, we demonstrate how we can account for the loss of orthogonality and prove the following theorem on the backward stability of MGS-GMRES, which embodies the conclusion of [36].

Theorem 5.2. *Consider solving $Ax = b$ with MGS-GMRES run in precision of unit roundoff $u \ll 1$. As long as the system is not numerically singular, that is,*

$$\sigma_{\min}(A) \gg u\|A\|_F, \tag{5.11}$$

then there exists an iteration $k \leq n$ such that

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u, \tag{5.12}$$

where $c(n,k)$ is a polynomial in n and k of low degree.

Proof. Similarly to the proof of Theorem 5.1 for HH-GMRES, we shall demonstrate that conditions (3.1) to (3.8) are met at a certain iteration k and for certain accuracy parameters in order to apply Theorem 3.1. As in this previous proof, conditions (3.1), (3.2), and (3.4) are met for $\varepsilon_c \equiv k^{1/2}\widetilde{\gamma}_n\|A\|_F/\|A\widehat{V}_k\|_F$, $\varepsilon_b \equiv 0$, and $\varepsilon_x \equiv \gamma_n$ for all $k \leq n$ because the implementations of lines 1, 2, and 4 are left unchanged compared with HH-GMRES. The remainder of this proof consists in demonstrating that the rest of the conditions are still met.

1. Identifying ε_{ls}. The least squares problem $\min_y \|b - \widehat{C}_k y\|_2$ at line 3 is solved with the MGS Arnoldi algorithm. Using the result developed in [36, sect. 7], and more particularly referring to [36, eq. (7.13)] which is a similar result to (A.3) for Householder Arnoldi, we have

$$\begin{aligned} \widehat{y}_k &= \arg\min_y \|(b + \Delta^b_{\text{ls}}) - (\widehat{C}_k + \Delta^c_{\text{ls}})y\|_2, \\ \|\left[\Delta^b_{\text{ls}}, \Delta^c_{\text{ls}}\right]e_j\|_2 &\lesssim j\widetilde{\gamma}_n\|\left[b, \widehat{C}_k\right]e_j\|_2, \quad j \leq k+1, \end{aligned} \tag{5.13}$$

and condition (3.3) is met for $\varepsilon_{\text{ls}} \equiv \widetilde{\gamma}_{n(k+1)}$ as long as $\widehat{C}_k$ is numerically full-rank, namely $\sigma_{\min}(\widehat{C}_k) \gg u\|\widehat{C}_k\|_F$. This full-rank condition is satisfied at the specific iteration k we will define below.

As a first remark, Theorem A.1, which assesses the error on the least squares problem solved by Householder Arnoldi, could be adapted relatively straightforwardly to MGS Arnoldi. It would require a few modifications in its proof, namely exchanging (A.5) with [25, Thm. (19.13)], adapting the constants, and adapting the text since MGS computes the "reduced-size" QR factorization instead of the "full-size" one as for the Householder orthogonalization. As a second remark, meeting (5.13) either in [36, sect. 7] or from adapting Theorem A.1, is substantially simplified by the MGS's Householder equivalence developed in [9] which leads to the existence of a perfectly orthogonal matrix $\widetilde{V}_{k+1}$ associated with the computed R-factor $[\widehat{\beta}e_1, \widehat{H}_k]$ of $[b, \widehat{C}_k]$ by MGS such that

$$\left[b, \widehat{C}_k\right] + \left[\Delta^b_{\text{qr}}, \Delta^c_{\text{qr}}\right] = \widetilde{V}_{k+1}[\widehat{\beta}e_1, \widehat{H}_k], \quad \|\left[\Delta^b_{\text{qr}}, \Delta^c_{\text{qr}}\right]e_j\|_F \leq c(n,k)\|\left[b, \widehat{C}_k\right]e_j\|_2, \quad j \leq k+1; \tag{5.14}$$

see also [25, Thm. (19.13)]. Under this result, the difference between the original least squares problem and the Arnoldi transformed least squares problem does not suffer from the loss of orthogonality of $\widehat{V}_{k+1}$ and we have

$$\begin{aligned} \min_y \|\left[b, \widehat{C}_k\right]\begin{bmatrix}1\\-y\end{bmatrix}\|_2 &= \min_y \|\left(\widetilde{V}_{k+1}\left[\widehat{\beta}e_1, \widehat{H}_k\right] - \left[\Delta^b_{\text{qr}}, \Delta^c_{\text{qr}}\right]\right)\begin{bmatrix}1\\-y\end{bmatrix}\|_2 \\ &= \min_y \|(\widehat{\beta}e_1 - \widetilde{V}^T_{k+1}\Delta^b_{\text{qr}}) - (\widehat{H}_k - \widetilde{V}^T_{k+1}\Delta^c_{\text{qr}})y\|_2. \end{aligned}$$

2. Addressing the loss of orthogonality. We now identify a key iteration $k \leq n$ for which conditions (3.7) and (3.8) are met. This is where the loss of orthogonality brings some challenges. A successful approach proposed in [36] consists in exploiting [20, Thm. 3.1] that assesses that as long as $[b, A\widehat{V}_k]$ is not nearly singular to machine precision, the set of computed vectors $\widehat{V}_{k+1}$ by MGS, which forms the next basis, is very well-conditioned. This result stems from the observation that a growing condition number for $\widehat{V}_{k+1}$ is associated with a full loss of orthogonality. In our own proof, we will use the reworking of this theorem by [36, sect. 6] and make a slight simplification. Namely, the results in [36, sect. 6] are derived for $\dot{V}_k$ which is $\widehat{V}_k$ with its columns correctly normalized, but for the sake of conciseness we consider that these results hold for $\widehat{V}_k$; it accounts for ignoring second order terms which are harmless to our analysis.

First, we consider the case where we never fully lose the orthogonality of the basis; that is, for all $k \leq n$, we keep $\kappa_2(\widehat{V}_k) \leq 4/3$ and, since $\sigma_{\min}(\widehat{V}_k) \leq \|\widehat{v}_1\|_2 \approx 1 \leq \sigma_{\max}(\widehat{V}_k)$, we have

$$\kappa_2(\widehat{V}_k) \leq 4/3, \quad \sigma^{-1}_{\min}(\widehat{V}_k) \leq 4/3, \quad \sigma_{\max}(\widehat{V}_k) \leq 4/3. \tag{5.15}$$

In this case, similarly to HH-GMRES, conditions (3.7) and (3.8) are met for $k = n$ since $\sigma_{\min}(A\widehat{V}_n) \geq 3\sigma_{\min}(A)/4 \gg u\|A\|_F$ by assumption (5.11) and $[b\phi, A\widehat{V}_n]$ is singular for

all $\phi > 0$. Conversely, consider that there is an iteration where the basis fully loses its orthogonality and that, therefore, $\kappa_2(\widehat{V}_k) \leq 4/3$ is not valid for all $k \leq n$. For the first $k \leq n$ such that $\kappa_2(\widehat{V}_{k+1}) > 4/3$, we have $\kappa_2(\widehat{V}_k) \leq 4/3$ and, hence, $\widehat{V}_k$ satisfies (5.15). Using [36, sects. 5 and 6] which develop an equivalent result to [20, Thm. 3.1], and more particularly using [36, eqs. (6.2) and (6.3)], we guarantee that since $\kappa_2(\widehat{V}_{k+1}) > 4/3$ then

$$\sigma_{\min}\big(\big[b\phi, \widehat{C}_k\big]\big) < c(n,k)u\|\big[b\phi, \widehat{C}_k\big]\|_F, \quad \forall \phi > 0. \tag{5.16}$$

The assertion (5.16) is stating that if the computed Q-factor $\widehat{V}_{k+1}$ of $[b, \widehat{C}_k]$ by MGS is ill-conditioned, then $[b, \widehat{C}_k]$ itself must be ill-conditioned; this is the converse of [20, Thm. 3.1]. Using (3.1) and (5.16) we obtain for all $\phi > 0$

$$\begin{aligned}\sigma_{\min}\big(\big[b\phi, A\widehat{V}_k\big]\big) = \sigma_{\min}([b\phi, A\widehat{V}_k + \Delta_c - \Delta_c]) &< c(n,k)u\|\big[b\phi, \widehat{C}_k\big]\|_F + \|\Delta_c\|_F \\ &\lesssim c(n,k)(u+\varepsilon_c)\|\big[b\phi, A\widehat{V}_k\big]\|_F,\end{aligned} \tag{5.17}$$

and using (5.15) we obtain

$$\sigma_{\min}(A\widehat{V}_k) \geq \sigma_{\min}(A)\sigma_{\min}(\widehat{V}_k) \geq 3/4\sigma_{\min}(A). \tag{5.18}$$

Hence, at this specific iteration $k \leq n$ condition (3.5) is met by (5.15), and from (5.18) and (5.11) we have

$$\varepsilon_c \equiv k^{\frac{1}{2}}\widetilde{\gamma}_n\|A\|_F/\|A\widehat{V}_k\|_F \leq k^{\frac{1}{2}}\widetilde{\gamma}_n\|A\|_F/\sigma_{\min}(A\widehat{V}_k) \ll 1,$$

and condition (3.6) is met. Moreover, (5.17) and (5.18) guarantee, respectively, that conditions (3.7) and (3.8) are met. Then, at the key iteration k, which is defined as the first iteration where $\kappa_2(\widehat{V}_{k+1}) > 4/3$, the conditions (3.1) to (3.8) are met.

3. Application of Theorem 3.1. Thus, Theorem 3.1 is applicable under conditions that reduce to those of [36]. Observing that

$$\sigma_{\min}^{-1}(\widehat{V}_k)\|A\widehat{V}_k\|_F/\|A\|_F \leq \sigma_{\min}^{-1}(\widehat{V}_k)\|\widehat{V}_k\|_F \leq \kappa_F(\widehat{V}_k) \leq 4k/3$$

by using (5.15), we obtain $\xi \lesssim c(n,k)u$ from which we deduce the backward error bound (5.12) and which concludes the proof. □

5.4. **Flexible GMRES.** Flexible GMRES (FGMRES) is a variant of right-preconditioned GMRES allowing for non-constant preconditioners. More precisely, it uses a basis $Z_k \equiv [\mathrm{fl}(M_{R,1}^{-1}\widehat{v}_1), \ldots, \mathrm{fl}(M_{R,k}^{-1}\widehat{v}_k)]$, where $\widehat{V}_k = [\widehat{v}_1, \ldots, \widehat{v}_k]$ is the computed Krylov basis by the Arnoldi algorithm and $\{M_{R,j}\}_j$ is a set of right-preconditioners; possibly $M_{R,i} \neq M_{R,j}$ for all $i \neq j \leq k$. Backward error analyses of FGMRES using the MGS orthogonalization were carried out by Arioli et al. in [4] and [5], and we are specifically interested in [4, Thm 3.1] which bounds the backward error of FGMRES for an unspecified set of right-preconditioners $\{M_{R,j}\}_j$. This result is subsequently used to prove the backward stability of FGMRES when $M_{R,j} \equiv M_R \equiv \widehat{L}\widehat{U}$ for all $j \leq k$, where $\widehat{L}$ and $\widehat{U}$ are the LU factors of A computed approximately. Except the fact that the basis Z_k is not the computed Krylov basis $\widehat{V}_k$ anymore but is rather unspecified, the operations at lines 1to 4 of Algorithm 1 are identical to those of MGS-GMRES. Accounting for different assumptions that we will comment on later in this section and minor differences in the constants depending on n and k, we recover closely the results of [4, Thm. 3.1] when we apply our framework to FGMRES. We summarize our conclusions in the following theorem.

Theorem 5.3. *Consider solving $Ax = b$ with FGMRES run in precision of unit roundoff $u \ll 1$. As long as the basis Z_k is not numerically singular, that is,*

$$\sigma_{\min}(Z_k) \gg u\|Z_k\|_F, \tag{5.19}$$

and if their exists an iteration $k \leq n$ such that for all $\phi > 0$ we have

$$\sigma_{\min}([b\phi, AZ_k]) \leq c(n,k)u\frac{\|A\|_F\|Z_k\|_F}{\|AZ_k\|_F}\|\big[b\phi, AZ_k\big]\|_F \tag{5.20}$$

and

$$\sigma_{\min}(AZ_k) \gg u\|A\|_F\|Z_k\|_F, \tag{5.21}$$

then, at this iteration k, the backward error of $\widehat{x}_k$ satisfies

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u\sigma_{\min}^{-1}(Z_k)\|Z_k\|_F, \tag{5.22}$$

where $c(n,k)$ accounts for polynomials in n and k of low degrees.

Proof. We proceed identically as for the proofs of Theorem 5.1 and 5.2 where we are looking for meeting the conditions of Theorem 3.1 in order to apply it. For the same reasons as for these previous proofs, conditions (3.2) and (3.4) are met for $\varepsilon_b \equiv 0$ and $\varepsilon_x \equiv \gamma_n$. We shall now demonstrate that the other conditions are met.

1. Identifying ε_c. The matrix–matrix product AZ_k is computed through a succession of standard matrix–vector products Az_j and satisfies

$$\mathrm{fl}(AZ_k) = AZ_k + \Delta_c, \qquad \|\Delta_c\|_F \leq \gamma_n\|A\|_F\|Z_k\|_F.$$

Hence, we meet condition (3.1) for $\varepsilon_c \equiv \gamma_n\|A\|_F\|Z_k\|_F/\|AZ_k\|_F$, and $\varepsilon_c \ll 1$ by assumption (5.21).

2. Identifying $\varepsilon_{\mathrm{ls}}$. The least squares problem at line 3 is solved through MGS Arnoldi. As for the proof of Theorem 5.2 on MGS-GMRES, we rely on the analysis of [36, sect. 7]. By assumption (5.21) we have

$$\sigma_{\min}(\widehat{C}_k) \geq \sigma_{\min}(AZ_k) - \|\Delta_c\|_F \gg u\|A\|_F\|Z_k\|_F \gtrsim u\|\widehat{C}_k\|_F,$$

and so $\widehat{C}_k$ is full-rank. The result of [36, sect. 7] are applicable on the least squares problem $\min_y \|b - \widehat{C}_k y\|_2$, and from [36, eq. (7.13)] condition (3.3) is met for $\varepsilon_{\mathrm{ls}} \equiv \widetilde{\gamma}_{n(k+1)}$.

3. Other conditions and application of Theorem 3.1. Condition (3.5) is verified by assumption (5.19), and condition (3.6) is satisfied since all the accuracy parameters are sufficiently less than 1. Finally, by assumptions (5.20) and (5.21) conditions (3.7) and (3.8) are met. Observing that $\sigma_{\min}^{-1}(Z_k)\|AZ_k\|_F/\|A\|_F \leq \sigma_{\min}^{-1}(Z_k)\|Z_k\|_F$, we can apply Theorem 3.1 with $\xi \leq c(n,k)u\sigma_{\min}^{-1}(Z_k)\|Z_k\|_F$ which concludes the proof. □

As Theorem 5.3 slightly differs in its result and assumptions from [4, Thm 3.1], we comment on these differences. The major discrepancy between the two is the presence of assumptions (5.20) and (5.21) in Theorem 5.3 which are not in [4, Thm. 3.1]. These assumptions can hardly be simplified without more knowledge on the basis Z_k. In particular, assumption (5.21) requires

$$\frac{\sigma_{\min}(AZ_k)}{\|A\|_F\|Z_k\|_F} = \frac{\|AZ_k\|_2}{\|A\|_F\|Z_k\|_F}\kappa_2(AZ_k)^{-1} \gg u,$$

which will be met if:

- The right-preconditioners $\{M_{r,j}\}_j$ are good approximations of the inverse of A such that $\kappa_2(AZ_k) \geq 1$ is small enough.
- The cancellation occurring in the product AZ_k is small, that is, $\|AZ_k\|_F/\|A\|_F\|Z_k\|_F \approx 1$. Unfortunately, this tends to conflict with the previous point since, for a good preconditioner, we expect $AM_{r,j} \approx I$ potentially leading to $\|AM_{r_j}\|_F \ll \|A\|_F\|M_{r_j}\|_F$.
- In all cases, if A and Z_k are relatively well-conditioned, and the unit roundoff of the machine precision is small enough, the previous condition should be met.

Nevertheless, assumptions (5.20) and (5.21) are not more restrictive than the ones in [4, Thm. 3.1]. What makes the comparison complex is that the assumption [4, eq. (3.5)] of [4, Thm. 3.1] is too optimistic in general because it requires $[b, AZ_k]$ to be full-rank for all $k \leq n$. As explained in section 3.2, a good approximation to the solution is reached when b lies in the range of AZ_k, and therefore when $[b, AZ_k]$ is nearby rank deficient, which makes assumption [4, eq. (3.5)] unlikely. In particular, this assumption never holds in the case $k = n$, where $[b, AZ_n]$ is always rank deficient.

Another difference between Theorem 5.3 and [4, Thm 3.1] is the form of their error bounds. The error bound provided by [4, Thm. 3.1] is

$$\|b - A\widehat{x}_k\| \lesssim c(n,k)u\big(\|b\|_2 + \|A\|_F\|x_0\| + \|A\|_F\||Z_k||\widehat{y}_k|\| + \|AZ_k\|_F\|\widehat{y}_k\|_2\big), \tag{5.23}$$

where $\widehat{y}_k$ is the computed solution of the least squares problem at line 3 of Algorithm 1 and x_0 is an initial guess of the solution. By using (3.24) in (5.23) and assuming $x_0 = 0$, we recover the error bound (5.22) of Theorem 5.3 and, therefore, (5.23) implies (5.22). In [4], the error bound (5.23) is further specialized for the case where FGMRES uses $M_{R,j} \equiv M_R \equiv \widehat{L}\widehat{U}$ for all $j \le k$, where $\widehat{L}$ and $\widehat{U}$ are the LU factors of A computed in single precision. It is shown, using (5.23) and under good conditions, that this process is backward stable, that is, the backward error is bounded by $c(n,k)u$. This might be a source of confusion since the bound (5.22) resulting from our analysis cannot actually lead to this conclusion; it provides instead

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u\kappa_F(\widehat{L}\widehat{U}),$$

assuming $Z_k \approx \widehat{U}\backslash\widehat{L}\backslash\widehat{V}_k$ and $\|\widehat{V}_k\|_F \approx k^{1/2}$. This difference between the two analyses stems from exploiting the fact that the LU preconditioner computed in single precision is very close to the original matrix A, up to the single precision accuracy if the growth factor is small. This is something that we cannot exploit in our analysis without loss of generality. In addition of the non numerical singularity hypothesis [4, eq. (3.29)] and the initialization of the first guess of the solution to $x_0 = \widehat{U}\backslash\widehat{L}\backslash b$, whereas we use $x_0 = 0$, it leads to substantial simplifications in the reasoning of [4], mainly from [4, eq. (3.27)] to [4, eq. (3.28)], that we cannot do in our own analysis.

6. New backward error analyses of GMRES algorithms

In this section we use our modular framework to derive error bounds on several GMRES algorithms for which a backward error analysis has not yet been proposed. We cover simpler GMRES, CGS2-GMRES, and a mixed precision strategy for restarted GMRES. In addition, we provide insights on how our framework might be used to derive error bounds for deflated GMRES, randomized Gram-Schmidt GMRES, and block GMRES.

6.1. Simpler GMRES.

Simpler GMRES is a variant of GMRES that uses a "simpler" approach to solve the least squares problem at step 3 of Algorithm 1. It has been first described in [49], but we will consider the more general form proposed by Jiránek et al. [30] which uses an unspecified basis Z_k and which is referred in their work as the "generalized simpler approach".

In exact arithmetic, a "standard" GMRES factorizes

$$\big[b, AZ_k\big] = V_{k+1}R_{k+1}, \quad R_{k+1} = \big[\beta e_1, \bar{H}_k\big],$$

and triangularizes the resulting Hessenberg matrix $\bar{H}_k$ with Givens rotations to solve the Arnoldi transformed least squares problem $\min_y \|\beta e_1 - \bar{H}_k y\|_2$ equivalent to the original least squares problem $\min_y \|b - AZ_k y\|_2$. Instead of computing a QR factorization of $[b, AZ_k]$ as for a standard GMRES, simpler GMRES factorizes AZ_k as

$$AZ_k = V_k R_k, \tag{6.1}$$

where $V_k \in \mathbb{R}^{n\times k}$ is orthogonal and $R_k \in \mathbb{R}^{k\times k}$ is triangular. The least squares problem at line 3 of Algorithm 1 is then solved by forming $y_k = R_k^{-1}V_k^T b$. This leads to a more traditional or "simpler" approach to solve the least squares problem $\min_y \|b - AZ_k y\|_2$ that does not involve upper Hessenberg factorization.

To derive a backward error analysis of simpler GMRES, we suppose that there is no left-preconditioner (i.e., $M_L = I$), that lines 1 and 4 of Algorithm 1 are computed from standard matrix–vector products, and that the QR factorization (6.1) is obtained from the MGS orthogonalization. It should be noted that with the MGS method, the least squares problem should not be solved by forming explicitly the products $y_k = \widehat{R}_k^{-1}\widehat{V}_k^T b$ which would

lead to stability issues. Instead, as explained in [25, sect. 20.3], the augmented matrix technique proposed by Björck [8] should be employed. Applying our framework to simpler GMRES yields the following theorem.

Theorem 6.1. *Consider solving $Ax = b$ with simpler GMRES run in precision of unit roundoff $u \ll 1$. Under the same assumptions as Theorem 5.3, the backward error at the key iteration k satisfies*

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}_k\|_2} \lesssim c(n,k)u\sigma_{\min}^{-1}(Z_k)\|Z_k\|_F, \tag{6.2}$$

where $c(n,k)$ is a polynomial in n and k.

Proof. The proof is almost identical to the one of Theorem 5.3 for FGMRES. The only difference lies in the fact that the least squares problem at line 3 of Algorithm 1

$$\min_y \|b - \widehat{C}_k y\|_2 \tag{6.3}$$

is now solved by the MGS QR factorization of $\widehat{C}_k$. The backward error analysis of this process is covered by [25, Thm. 20.3]. Actually, [25, Thm. 20.3] concerns the Householder orthogonalization, but it is explained in [25, sect. 20.3] that it holds for MGS with the augmented matrix technique of Björck [8]. Using this previous result guarantees that the computed solution $\widehat{y}_k$ of the least squares problem satisfies

$$\widehat{y}_k = \arg\min_y \|b + \Delta_{\text{ls}}^b - (\widehat{C}_k + \Delta_{\text{ls}}^c)y\|_2,$$
$$\|\left[\Delta_{\text{ls}}^b, \Delta_{\text{ls}}^c\right]e_j\|_2 \leq c(n,k)u\|\left[b, \widehat{C}_k\right]e_j\|_2, \quad j \leq k+1.$$

Therefore, condition (3.3) is met for $\varepsilon_{\text{ls}} \equiv c(n,k)u$. The rest of the proof is identical to the one of Theorem 5.3 for FGMRES. □

The original form of simpler GMRES, which is described in [49], uses a basis $Z_k \equiv [b/\|b\|_2, \widehat{V}_{k-1}]$ which spans the Krylov subspace $\mathcal{K}_k(A,b)$ and where $\widehat{V}_{k-1}$ are the $k-1$ first Arnoldi vectors computed iteratively through the orthogonalization process (6.1). In this case, the Arnoldi process starts with $v_1 = Ab/\|Ab\|$ and $\widehat{V}_{k-1}$ spans the subspace $A\mathcal{K}_{k-1}(A,b)$.

Unfortunately, for this particular choice of basis Z_k, the key dimension conditions (5.20) and (5.21) will hardly be met. This is because the convergence of the solution amounts to b lying in the range of $\widehat{V}_k$ and, thus, the basis Z_k becomes rank-deficient as we converge to the solution. In particular, it is explained in [30] that $\kappa_F(Z_k)$ is of the same order of magnitude as the ratio $\|b\|_2/\|r_{k-1}\|_2$ (see [30, Thm. 3.2] and comments around). Hence, supposing $\|b\|_2 \approx 1$ and taking the lower bound $\sigma_{\min}(AZ_k) \geq \sigma_{\min}(A)\sigma_{\min}(Z_k)$, we deduce from condition (5.21) the following more stringent condition for the application of Theorem 6.1

$$\sigma_{\min}(A)\|A\|_F^{-1}\sigma_{\min}(Z_k)\|Z_k\|_F^{-1} \geq \kappa_F(A)^{-1}\kappa_F(Z_k)^{-1} \gg u, \tag{6.4}$$

which will break once the residual is of order $u\kappa_F(A)$. Condition (6.4) is not expected to be significantly pessimistic compared with (5.21) since we have no reason to expect large cancellations in the matrix–matrix product AZ_k, and is therefore a good indication of the difficulty to meet condition (5.21). Note that this problem is independent of the orthogonalization process used, and simpler GMRES with Householder or MGS orthogonalization face the same issue.

Jiránek et al. [30] proposed a basis based on the normalized residuals $Z_k \equiv [r_0/\|r_0\|, \ldots, r_{k-1}/\|r_{k-1}\|]$ in exact arithmetic. In particular, it is explained that as long as there is no stagnation of the computed solution, namely $r_j \not\approx r_{j+1}$ for all $j < k$, the vectors of the basis Z_k will stay linearly independent which prevents the previous issue. The conditions of Theorem 6.1 are more likely to be met for this choice of basis under good non-stagnation conditions; note that, however, we do not provide further investigations on the applicability of Theorem 6.1 for this choice of basis Z_k in this article.

6.2. **CGS2-GMRES.** Compared with the MGS or Householder orthogonalization, the classical Gram-Schmidt orthogonalization (CGS) preserves the least the orthogonality of the computed Krylov basis vectors. Indeed, CGS run in precision of unit roundoff u computes $\widehat{V}_{k+1}$ satisfying

$$\|I - \widehat{V}_{k+1}^T \widehat{V}_{k+1}\|_F \leq c(n,k) u \kappa_F\big([b, \mathrm{fl}(A\widehat{V}_k)]\big)^2, \tag{6.5}$$

which is substantially worse than (5.9) and (5.10) for MGS and Householder, respectively; see [21, Thm. 1]. For this reason, the CGS-GMRES variant suffers from stability issues. A common remedy is to reapply the CGS process a second time. The resulting classical Gram-Schmidt with reorthogonalization algorithm (CGS2) has been shown to preserve the orthogonality of the computed vectors close to the machine precision level as long as $[b, \mathrm{fl}(A\widehat{V}_k)]$ is numerically full-rank, namely

$$\|I - \widehat{V}_{k+1}^T \widehat{V}_{k+1}\|_F \leq c(n,k) u; \tag{6.6}$$

see [21, Thm. 2]. We say in this case that $\widehat{V}_{k+1}$ is "orthogonal to machine precision".

Naturally, this increased stability comes at the cost of increased flops. However, it is important to remark that while CGS2 requires twice as many flops as MGS, it can leverage higher-level BLAS kernels and requires less communication in distributed computing. For these reasons, CGS2-GMRES can achieve better overall computing performance than MGS-GMRES depending on the hardware and the problem. For instance, it has been remarked in [33] that CGS2-GMRES is more competitive on GPU accelerators than MGS-GMRES.

Interestingly, while CGS2 better preserves the orthogonality than MGS and while CGS2-GMRES is a popular variant of GMRES often used in practice, the backward stability of CGS2-GMRES has not yet been proven.

In the following we study CGS2-GMRES and prove the backward stability of this algorithm for solving (1.1). To carry out the backward error analysis, we suppose that the operations at lines 1, 2, and 4 of Algorithm 1 are identical to those of MGS-GMRES studied in section 5.3. Compared with this previous algorithm, the only difference is that line 3 is now computed with the CGS2 orthogonalization process. Applying our framework on CGS2-GMRES yields the following theorem.

Theorem 6.2. *Consider solving $Ax = b$ with CGS2-GMRES run in precision of unit roundoff $u \ll 1$. Under the same assumptions as Theorem 5.2, the backward error at the key iteration k satisfies*

$$\frac{\|b - A\widehat{x}_k\|_2}{\|b\|_2 + \|A\|_F \|\widehat{x}_k\|_2} \lesssim c(n,k)u, \tag{6.7}$$

where $c(n,k)$ is a polynomial in n and k of low degree.

Proof. The approach is very similar to proving Theorem 5.2 on the backward stability of MGS-GMRES. Nonetheless, CGS2-GMRES offers new difficulties compared with MGS-GMRES; specifically, the MGS's Householder equivalence [9] does not extend to the CGS2 orthogonalization, and certain useful simplifications made in the proof of Theorem 5.2 are now impossible. In addition, the proof also requires a range of side results on the CGS2 orthogonalization that are developed in the appendix but whose absence in the main text of this proof should not be critical for the well-understanding of our reasoning. Most of these side results are relatively straightforward to obtain but, to our knowledge, are not present in the literature and need proper attention and development. The differences with the proof of MGS-GMRES mostly concern conditions (3.3), (3.7), and (3.8) which we rework as follows.

1. Existence of the key dimension. Demonstrating the existence of a key iteration $k \leq n$ at which conditions (3.7) and (3.8) are met can be done very similarly to the proof of Theorem 5.2 for MGS-GMRES. In this previous proof, we defined the key iteration with the condition number of the computed Arnoldi bases $\widehat{V}_k$. To be more precise, we chose the key iteration to be the first $k \leq n$ such that $\kappa_2(\widehat{V}_{k+1}) > 4/3$ and $\kappa_2(\widehat{V}_k) \leq 4/3$, which amounts to the full loss of orthogonality of $\widehat{V}_{k+1}$. For this proof, our description of the key iteration

is slightly different but conveys the same meaning. We consider the first $k \leq n$ such that the basis $\widehat{V}_{k+1}$ has lost its orthogonality to the level of machine precision, that is,

$$\|I - \widehat{V}_{k+1}^T \widehat{V}_{k+1}\|_F > c(n,k)u \quad \text{and} \quad \|I - \widehat{V}_k^T \widehat{V}_k\|_F \leq c(n,k)u, \tag{6.8}$$

where $c(n,k)$ accounts for polynomials of low degree in n and k; it is not critical for the rest of the reasoning to identify the specific values of these $c(n,k)$.

To show that we meet conditions (3.7) and (3.8) at this iteration k verifying (6.8), we will show that (5.15) and (5.16) in the MGS-GMRES proof also hold for CGS2-GMRES such that the rest of the reasoning is properly identical to the MGS-GMRES proof. We first use Lemma C.2 developed in the appendix, which guarantees that the computed basis $\widehat{V}_k$ is (very) well-conditioned as long as $\widehat{V}_k$ is orthogonal to machine precision. Namely under (6.8) we have

$$\sigma_{\min}(\widehat{V}_k) \approx \sigma_{\max}(\widehat{V}_k) \approx \kappa_2(\widehat{V}_k) \approx 1; \tag{6.9}$$

from this we recover (5.15). To recover (5.16), we use Corollary C.4 also developed in the appendix, which shows that if CGS2 computes $\widehat{V}_{k+1}$ not orthogonal to the machine precision level then the orthogonalized matrix $[b\phi, \widehat{C}_k]$ must be numerically singular. More precisely, under (6.8) the assumptions of Corollary C.4 are met at the key iteration $k \leq n$ and (C.18) guarantees

$$\sigma_{\min}([b\phi, \widehat{C}_k]) < c(n,k)u\|[b\phi, \widehat{C}_k]\|_F, \quad \forall \phi > 0;$$

thus (5.16) is met. The rest of the reasoning of the paragraph "Addressing the loss of orthogonality" in the proof of Theorem 5.2 holds, we similarly recover (5.17) and (5.18), and conditions (3.7) and (3.8) are satisfied. In the case where we never lose the orthogonality we have $\|I - \widehat{V}_k^T \widehat{V}_k\|_F \leq c(n,k)u$ for all $k \leq n$, we meet condition (3.7) and (3.8) at least for $k = n$ by knowing that $\widehat{V}_k$ satisfies (6.9) and by using the same reasoning as in the proof of Theorem 5.2.

2. Identifying ε_{ls}. The least squares problem $\min_y \|b - \widehat{C}_k y\|_2$ at line 3 is solved with CGS2 Arnoldi. If $\widehat{C}_k$ is numerically full-rank, which is the case at the key iteration k by using (6.9) and assumption (5.11), this process provides a backward stable solution to the least squares problem as for Householder or MGS Arnoldi; namely the computed solution $\widehat{y}_k$ satisfies

$$\begin{aligned} \widehat{y}_k &= \arg\min_y \|(b + \Delta_{\text{ls}}^b) - (\widehat{C}_k + \Delta_{\text{ls}}^c)y\|_2, \\ \|[\Delta_{\text{ls}}^b, \Delta_{\text{ls}}^c] e_j\|_2 &\lesssim c(n,k)u\|[b, \widehat{C}_k] e_j\|_2, \quad j \leq k+1. \end{aligned} \tag{6.10}$$

Proving this statement at the key iteration k verifying (6.8) can be done almost identically as in the proof of Theorem A.1 for Householder Arnoldi. For this reason, we do not provide the full details but rather highlight the main differences. Adapting this proof to CGS2 mainly consists in replacing (A.5), which holds for the Householder orthogonalization, by

$$\begin{aligned} [b, \widehat{C}_k] + [\Delta b, \Delta C_k^{(1)}] &= \widehat{V}_{k+1}[\widehat{\beta} e_1, \widehat{H}_k], \\ \|[\Delta b, \Delta C_k^{(1)}] e_j\|_2 &\leq c(n,k)u\|[b, C_k] e_j\|_2, \quad j \leq k+1, \end{aligned} \tag{6.11}$$

obtained from Theorem B.1 developed in the appendix and which is a columnwise extension of [21, eq. (8)], where $\widehat{\beta} \approx \|b\|_2$ and $\widehat{H}_k \in \mathbb{R}^{(k+1)\times k}$. This result holds for CGS and a fortiori for CGS2. A subtle but yet crucial difference to notice between (A.5) and (6.11) is that, in the former, $\widetilde{V}_{k+1}$ is perfectly orthogonal whereas, in the latter, $\widehat{V}_{k+1}$ is not even orthogonal to the machine precision level. It stems from the fact that the CGS2 orthogonalization does not enjoy the Householder equivalence as for MGS [9]; the benefits of the MGS's Householder equivalence were briefly evoked in the proof of Theorem 5.2 for the backward stability of MGS-GMRES. It makes the proof substantially more difficult since it prevents multiple convenient simplifications in, for instance, (A.6) or in the transition from (A.8) to (A.9).

However, this can be overcome by observing that at the key iteration $k \leq n$ satisfying (6.8), CGS2-GMRES computes

$$\widehat{C}_k + \Delta C_k^{(1)} = \widehat{V}_{k+1}\widehat{H}_k = \widehat{V}_k \widehat{H}_k^\star + \widehat{w}_{k+1} e_k^T \approx \widehat{V}_k \widehat{H}_k^\star, \quad \|\widehat{w}_{k+1}\|_2 < c(n,k)u\|\widehat{c}_k\|_2, \tag{6.12}$$

where $\widehat{H}_k^\star \in \mathbb{R}^{k\times k}$ is $\widehat{H}_k$ with its last row removed, and $\widehat{w}_{k+1}$ is the result of the two consecutive applications of the projection $(I - \widehat{V}_k\widehat{V}_k^T)$ on $\widehat{c}_k = \widehat{C}_k e_k = \mathrm{fl}(A\widehat{v}_k)$ meant to orthonormalize the vector $\widehat{c}_k$ against the vectors of $\widehat{V}_k$; see [42, prop. 6.5]. The key approach to prove (6.12) is to relate the loss of orthogonality of the basis $\widehat{V}_{k+1}$ to a GMRES numerical happy breakdown; that is, at the moment where the orthogonality is lost to machine precision, the vector $\widehat{w}_{k+1}$ vanishes such that $\widehat{H}_k(k+1,k) = \|\widehat{w}_{k+1}\|_2$ is at the machine precision level. To achieve this, we can use another outcome of the application of Corollary C.4 in the appendix which states that if CGS2 does not keep the orthogonality of $\widehat{V}_k$ to machine precision from an iteration k to the next $(k+1)$, then necessarily $\widehat{c}_k$ lies in the range of $\widehat{V}_k$, yielding a very small projection $(I - \widehat{V}_k\widehat{V}_k^T)\widehat{c}_k$ in the orthogonal complement of $\widehat{V}_k$. Under (6.8), the conditions of application of Corollary C.4 are met at the key iteration k, and we conclude from (C.16) that CGS2 computes $\widehat{w}_{k+1}$ satisfying (6.12).

Using (6.12) we rewrite (6.11) as

$$\begin{aligned} \big[b, \widehat{C}_k\big] + \big[\Delta b, \Delta C_k^{(2)}\big] = \widehat{V}_k\big[\widehat{\beta}e_1, \widehat{H}_k^\star\big], \quad \Delta C_k^{(2)} = \Delta C_k^{(1)} - \widehat{w}_{k+1}e_k^T, \\ \|\big[\Delta b, \Delta C_k^{(2)}\big]e_j\|_2 \le c(n,k)u\|\big[b, C_k\big]e_j\|_2, \quad j \le k+1, \end{aligned} \tag{6.13}$$

and replace (A.5) by (6.13) in the proof of Theorem A.1. The rest of the proof of Theorem A.1 can be easily adapted by applying Givens rotations on the square Hessenberg matrix $\widehat{H}_k^\star$ and taking into account the error on the orthogonality of $\widehat{V}_k$ defined by (6.8). For the latter, we can consider the polar decomposition $\widehat{V}_k = UH$, where $U \in \mathbb{R}^{n\times k}$ is orthogonal and $H \in \mathbb{R}^{k\times k}$ is symmetric positive-semidefinite. We can show that the difference $E = \widehat{V}_k - U$ has small norm by using [24, Lem. 5.1] and (6.8), which directly provide the following bound

$$\|E\|_F \le \|I - \widehat{V}_k^T\widehat{V}_k\|_F \le c(n,k)u.$$

Then, we replace $\widehat{V}_k$ by $U+E$ and we account for the error of order $\|E\|_F$ in the proof of Theorem A.1. Hence, an equivalent theorem can be derived for CGS2 Arnoldi, we recover (6.10), and condition (3.3) is met for $\varepsilon_{\mathrm{ls}} \equiv c(n,k)u$. □

6.3. **Mixed precision restarted GMRES.** One of the oldest and most successful mixed precision implementations of GMRES has been described by Turner and Walker [46] and subsequently investigated at great length in, for instance, [14], [3], or [33]. This mixed precision strategy, which we simply refer to as mixed precision GMRES in this article, uses $M_L^{(i)} \equiv I$ and $Z_{ki}^{(i)} \equiv \widehat{V}_{ki}^{(i)}$ for all i in Algorithm 2, where $\widehat{V}_{ki}^{(i)}$ is the Krylov basis computed by MGS Arnoldi at the ith restart and is fully formed and stored in memory. The residual and the update at lines 3 and 8 of Algorithm 2 are computed with standard matrix–vector product and vector addition/subtraction in high precision, while the rest of the operations from lines 4 to 7 are identical to those of MGS-GMRES already described in section 5.3 and are computed in low precision. This process delivers high precision accuracy solution after a suitable number of restarts. Moreover, as the residual and update computed in high precision are expected to be of negligible cost compared with the rest of the operations computed in low precision, mixed precision GMRES can substantially reduce time and memory consumption with respect to a full high precision restarted MGS-GMRES while providing the same solution accuracy.

Backward error analyses providing error bounds for mixed precision restarted left- preconditioned MGS-GMRES can be found in the literature [14], [3], or [15]. Interestingly, while these analyses cover restarted GMRES variants up to five precisions, their bounds are too dependent on the preconditioners used to cover the unpreconditioned case that we analyze in this section.

To carry out the backward error analysis we allow an unbounded number of Arnoldi iterations at each restart. Applying our framework on mixed precision GMRES yields the following theorem.

Theorem 6.3. *Consider solving $Ax = b$ with mixed precision GMRES which computes the residual and update in precision of unit roundoff u_{high} and the rest of its operations in precision of unit roundoff u_{low}. As long as*

$$\Lambda = c(n,k)u_{\text{low}}\kappa_F(A) \ll 1, \tag{6.14}$$

the backward and forward errors reduce at each iteration by a factor (at least) Λ until they reach

$$\frac{\|b - A\widehat{x}\|_2}{\|b\|_2 + \|A\|_F\|\widehat{x}\|_2} \lesssim c(n,k)u_{\text{high}} \quad \text{and} \quad \frac{\|\widehat{x} - x\|_2}{\|x\|_2} \lesssim c(n,k)u_{\text{high}}\kappa_F(A), \tag{6.15}$$

where $c(n,k)$ accounts for polynomials in n and k of low degree.

Proof. Applying Theorem 4.1 on mixed precision GMRES requires conditions (4.1) to (4.3) and (3.1) to (3.8) to be met for all restart iterations i and for given parameters ε_r, ε_u, $\varepsilon_c^{(i)}$, $\varepsilon_b^{(i)}$, $\varepsilon_{\text{ls}}^{(i)}$, and $\varepsilon_x^{(i)}$.

The computation of the residual and update at lines 3 and 8 are standard matrix–vector product and vector addition/subtraction computed in high precision, they satisfy respectively

$$\widehat{r}_i = b - A\widehat{x}_i + \Delta r_i, \quad |\Delta r_i| \leq \gamma_n^{\text{high}}(|b| + |A||\widehat{x}_i|), \tag{6.16}$$

and

$$\widehat{x}_{i+1} = \widehat{x}_i + \widehat{d}_i + \Delta x_i, \quad |\Delta x_i| \leq u_{\text{high}}|\widehat{x}_{i+1}|, \tag{6.17}$$

where we identify $\varepsilon_r \equiv \gamma_n^{\text{high}}$ and $\varepsilon_u \equiv u_{\text{high}}$ and for which conditions (4.1) to (4.3) are met.

The remaining conditions (3.1) to (3.8) concern the computation of the correction $\widehat{d}_i$ by MGS-GMRES in low precision of unit roundoff u_{low}. Fortunately, the work has already been done in the analysis of MGS-GMRES in section 5.3. From the proof of Theorem 5.2 we know that there exists key iterations k_i at which these conditions are met as long as $\sigma_{\min}(A) \gg u_{\text{low}}\|A\|_F$, which is guaranteed by assumption (6.14), and we have

$$\xi^{(i)} = \xi \lesssim c(n,k)u_{\text{low}}. \tag{6.18}$$

Since $M_L^{(i)} \equiv M_L \equiv I$, we can simplify $\Lambda_1^{(i)}$ and $\Lambda_2^{(i)}$ in Theorem 4.1 and obtain

$$\max(\Lambda_1^{(i)}, \Lambda_2^{(i)}) \leq \Lambda = c(n,k)u_{\text{low}}\kappa_F(A)$$

for a given $c(n,k)$. From assumption (6.14) we have $\Lambda \ll 1$, and applying Theorem 4.1 ends the proof. □

6.4. **Discussion around other GMRES algorithms.** To conclude this section we discuss other popular variants of GMRES on which our framework might be conclusive. These discussions do not intend to give error bounds on these algorithms nor give suitable backward error analyses. Instead, we discuss how our framework might be used and aim at identifying potential difficulties in proving error bounds for these algorithms as well as giving a few indications on how these difficulties might be addressed.

Deflated GMRES.. The convergence of GMRES is affected by the distribution of the eigenvalues of A. In particular, deflating small eigenvalues by having its corresponding eigenvector in the subspace $\mathcal{Z}$ spanned by Z_k can substantially improve the convergence rate of the method. This is the base idea of deflated GMRES algorithms. In this paragraph we consider the deflated GMRES algorithm presented by Morgan [35] which can be seen as a variant of restarted GMRES. In a nutshell, the process of deflated GMRES consists in:

- At the end of the $(i-1)$th restart, computing the jth smallest harmonic Ritz pairs.
- Building an orthogonal basis $V_{j+1}^{(i)} \in \mathbb{R}^{n\times(j+1)}$ which is a Krylov subspace containing the smallest Ritz vectors previously computed and its associated matrix $\bar{H}_j^{(i)} \in \mathbb{R}^{(j+1)\times j}$ such that $AV_j^{(i)} = V_{j+1}^{(i)}\bar{H}_j^{(i)}$ holds. It is important to remark that the process does not yield $\bar{H}_j^{(i)}$ as a Hessenberg matrix.
- Restarting GMRES by starting from the $(j+1)$th iteration and computing the remaining $k_i - j - 1$ vectors of the basis $V_{ki}^{(i)}$ with the usual Arnoldi process, where k_i is the size of the basis at the end of the ith restart.

- Solving the least squares problem $\min_y \|(V_{ki+1}^{(i)})^T b - \bar{H}_{ki}^{(i)} y\|$, updating the solution, and repeating the process.

To derive a backward error analysis of deflated GMRES one should use Theorem 4.1. However, showing that we meet the conditions of application of the theorem is not straightforward and, in the following, we briefly identify the difficulties that one should address to use our framework on deflated GMRES.

Compared with restarted MGS-GMRES, deflated GMRES mainly differs in how the Krylov basis $V_{ki}^{(i)}$ is built and how the least squares problem at line 6 of Algorithm 2 is solved. These changes mainly affect conditions (3.3), (3.7), and (3.8) which need further work to be proven. In more detail, the $j+1$ first vectors of the basis at the ith restart are obtained from the deflation process and, therefore, the Arnoldi process does not construct $V_{ki+1}^{(i)}$ as the Q-factor of the matrix $[b, AV_{ki}^{(i)}]$ anymore; it is true in exact arithmetic. In particular, the resulting matrix $\bar{H}_{ki}^{(i)}$ used to solve the transformed least squares problem $\min_y \|(V_{ki+1}^{(i)})^T b - \bar{H}_{ki}^{(i)} y\|$ is not Hessenberg. As a result, the process used to compute a solution to the least squares problem at line 6 is slightly different and, for this reason, it needs its own backward error analysis which would guarantee that condition (3.3) is still met. Moreover, to prove that conditions (3.7) and (3.8) are met, one might want to relate, as in the proof of MGS-GMRES in section 5.3 or the proof of CGS2-GMRES in section 6.2, the loss of orthogonality of the computed basis $\widehat{V}_{ki+1}^{(i)}$ to the near-singularity of $[b\phi, \widehat{C}_{ki}^{(i)}]$. To achieve this, these previous proofs relied on the fact that $\widehat{V}_{ki+1}^{(i)}$ is the computed Q-factor of $[b, \widehat{C}_{ki}^{(i)}]$ by the MGS or CGS2 orthogonalization process. However, since $\widehat{V}_{ki+1}^{(i)}$ is not a Q-factor with deflated GMRES, the reasoning carried out for MGS-GMRES or for CGS2-GMRES has to be adequately revised and, in particular, it needs to consider the first $(j+1)$ vectors obtained through the deflation process.

Randomized Gram-Schmidt GMRES.. A successful implementation of GMRES taking advantage of random sketching techniques is proposed by Balabanov and Grigori [6]. The so-called RGS-GMRES is built upon a randomized Gram-Schmidt process (RGS) and reduces computational resources by computing the expensive inner-products of the Gram-Schmidt orthogonalization in a lower-dimensional space. The resulting basis is orthogonal in the sketched space but no longer orthogonal with respect to the usual ℓ_2-inner-product in exact arithmetic. Namely $(\Theta V_k)^T(\Theta V_k) = I$ but $V_k^T V_k \neq I$ for all $k \leq n$, where $\Theta \in \mathbb{R}^{s\times n}$ is the sketched matrix with $s \ll n$.

Error bounds on the solution computed by RGS-GMRES could be derived using Theorem 3.1. The proof would follow closely, for instance, the one of MGS-GMRES developed in section 5.3. As the operations at lines 1, 2, and 4 of Algorithm 1 are specialized identically as for MGS-GMRES, the conditions associated to those lines still hold. The difference with the MGS-GMRES proof would concern conditions (3.3), (3.7), and (3.8) that are affected by the use of the RGS orthogonalization process. One should use the extensive resources provided in [6] to prove that these conditions are still met. In particular, for the key iteration conditions (3.7) and (3.8), [6, Thm. 3.2] could be used to show that RGS computes a well-conditioned basis $\widehat{V}_{k+1}$ as long as $[b, \widehat{C}_k]$ is not numerically singular. Assuming that this result can be reworked to consider the scaled matrix $[b\phi, \widehat{C}_k]$ instead, we could show that if $\widehat{V}_{k+1}$ is not well-conditioned therefore $[b\phi, \widehat{C}_k]$ is numerically singular, allowing an identical reasoning as in the proof of MGS-GMRES (or CGS2-GMRES). Finally, to prove condition (3.3) one might want to adapt the proof of Theorem A.1 for Householder Arnoldi to the RGS Arnoldi process. In that regard, the backward error result for the RGS QR factorization described in [6, Thm. 3.2] should be made columnwise and substitute (A.5) in the proof of Theorem A.1. In addition, as the basis V_k is not orthogonal anymore in exact arithmetic, the transformed least squares problem $\min_y \|\beta e_1 - \bar{H}_k y\|_2$ is not exactly equivalent to the original one $\min_y \|b - C_k y\|_2$. To adapt the proof of Theorem A.1, which relies on showing that those two least squares problem are near equivalent under rounding errors, one might want to use [6, sect. 4.2] that links the solutions of the RGS Arnoldi transformed least squares problem to the original one.

Block GMRES.. Computing a linear system with multiple right-hand sides $AX = B$ with $B, X \in \mathbb{R}^{n\times b}$ can be done efficiently through block GMRES; see for instance [43, 31, 38, 7]. In exact arithmetic, this algorithm builds at each iteration k the optimal set of approximate solutions $X_k \in \mathbb{R}^{n\times b}$ that minimizes the norm of the residuals associated with each right-hand side $R_k = AX_k - B$ in the block Krylov subspace $\mathcal{K}_k(A, B) = \text{span}\{B, AB, \ldots, A^{k-1}B\}$ spanned by the full-rank block Arnoldi basis $\mathbf{V}_k = [V_1, \ldots, V_k]$ with $V_j \in \mathbb{R}^{n\times b_j}$ and $b_j \leq b$. It has the advantage over a GMRES applied on each individual linear system to rely on BLAS-3 operations that are more cache-friendly and can offer improved performance. In addition, this version of GMRES enables the solution vectors associated with each right-hand side to share their Krylov spaces leading to potentially faster convergence. We explain how one might try to use our framework to derive backward and forward error bounds on each individual solution of the system solved by block GMRES.

We consider a block GMRES implementation using a block modified Gram-Schmidt (BMGS) scheme using the Householder orthogonalization to perform the intra-block QR factorizations; we call this algorithm BMGS-GMRES. Other choices of block Gram-Schmidt schemes or intra-block orthogonalizations could be used for implementing the block Arnoldi process. We refer the reader to [16] for more information relative to block Gram-Schmidt algorithms and their stability; in particular, we refer to [16, sect. 4.5] for a discussion on the stability of BMGS with Householder intra-block orthogonalization.

Theorem 3.1 could be used to bound the errors of the computed solutions associated with each right-hand side individually. Verifying conditions (3.1) and (3.4) should not raise any particular difficulties. We have $Z_k \equiv \widehat{\mathbf{V}}_k$ with block GMRES, and analyzing the numerical errors generated by computing the products $A\widehat{\mathbf{V}}_k$ at line 1 and $\widehat{\mathbf{V}}_k Y_k$ at line 4 of Algorithm 1 is direct if those products are standard matrix–matrix products. To prove condition (3.3), one natural way would be to adapt the proof of Theorem A.1 for Householder Arnoldi to BMGS Arnoldi. To that end, one could inject in (A.5) the backward error result on the computed QR factors by BMGS [29, prop. 4.2], giving

$$[B, \widehat{\mathbf{C}}_k] + \Delta_{\text{qr}} = \widehat{\mathbf{V}}_{k+1}\widehat{\mathbf{R}}, \quad \widehat{\mathbf{C}}_k = \text{fl}(A\widehat{\mathbf{V}}_k), \quad \|\Delta_{\text{qr}}\|_F \leq c(n,k)\|[B, \widehat{\mathbf{C}}_k]\|_F u; \tag{6.19}$$

it might be needed to rework (6.19) in columnwise or block-columnwise form. In [7], the MGS's Householder equivalence [9] is discussed and extended to some BMGS variants. For those variants, (6.19) could take the form of (5.14) which holds for an exactly orthogonal Q-factor $\widetilde{\mathbf{V}}_{k+1}$ instead of the non-orthogonal $\widehat{\mathbf{V}}_{k+1}$ in (6.19). The BMGS's Householder equivalence presented in [9] might be necessary to carry out the proof or, at least, would simplify it. Adapting the remainder of the proof of Theorem A.1 to the "reduced-size" QR factorization (because the proof of Theorem A.1 considers the "full-size" QR factors) and to block operations (e.g., $\widehat{H}_k$ is band-Hessenberg and has b subdiagonals instead of only one, the triangular solve is now applied on a matrix, etc.) might deliver a similar bound to (A.3) on each individual solution of the least squares problem.

Meeting conditions (3.7) and (3.8) is a substantial source of challenges for applying our framework on BMGS-GMRES. The difficulty originates from the variation in convergence rates of the solutions associated with each right-hand side. It means, in particular, that some solutions will reach their attainable accuracies (or user-defined accuracies) earlier than others. Naturally, BMGS-GMRES is completed when all those solutions have converged to their required accuracies. To better understand the source of the problem, we can interpret the block Arnoldi basis $\widehat{\mathbf{V}}_{k+1}$ at the $(k+1)$th iteration as the computed Q-factor of $[B, \text{fl}(A\widehat{\mathbf{V}}_k)]$ by the BMGS QR factorization; see (6.19). Hence, because $[B, A\widehat{\mathbf{V}}_k]$ becomes numerically singular once the first solution has converged to the machine precision accuracy, that is, when the associated right-hand side lies in the range of $A\widehat{\mathbf{V}}_k$, BMGS-GMRES yields potentially numerically singular next computed iterate bases $\widehat{\mathbf{V}}_{\bar{k}}$ for $k < \bar{k}$. This phenomenon is evoked by Langou in his Ph.D. thesis [31, sect. 2.6.6.1.2]. Consequently, after the convergence of the first solution, condition (3.8) becomes impracticable and Theorem 3.1 is not applicable on the remaining solutions. Fortunately, this issue can be prevented by

modifying the Arnoldi procedure to account for the convergence variability of the set of solutions. Those methods generally consist in discarding directions prone to instability during the basis expansion to avoid $\widehat{\mathbf{V}}_k$ becoming nearby rank-deficient. The strategy of Robbé and Sadkane [38], for instance, is popular and has been reused in [1] and [45]. To successfully apply our framework, it is likely that such a variant of BMGS-GMRES has to be considered.

7. Conclusion

We developed a modular framework for the backward error analyses of GMRES algorithms. This framework is made of a set of minimal assumptions under which we obtain modular normwise backward and forward error bounds that can be specialized to any GMRES algorithms meeting the framework assumptions. At the core of the framework are Theorems 3.1 and 4.1 which result from the backward error analyses of the MOD-GMRES and restarted MOD-GMRES abstract algorithms, respectively, and which should be used to derive bounds on the attainable backward and forward errors of a given GMRES algorithm. We dedicated a substantial amount of this article to applying these theorems to a wide range of GMRES algorithms in order to prove our framework's practicality and illustrate how it can be used. To that end, we first assessed the correctness of our framework by showing that it delivers (almost) identical error bounds under (almost) identical conditions as the major already existing backward error analyses of GMRES. Second, we used this framework to derive error bounds for three GMRES algorithms on which, to our knowledge, no conclusive backward error analyses existed. We further discussed how our framework might be used on three other GMRES algorithms without providing complete analyses; we gave insights into the difficulties of analyzing these algorithms and proposed approaches to address them.

We believe that the framework we proposed and the various examples we reviewed can help the community to derive error bounds for current and future GMRES algorithms. We emphasize that many GMRES variants on which the application of our framework can be considered have not been mentioned in this article. We give a quick acknowledgment to some of them here: s-step communication-avoiding GMRES algorithms [28, 12] which are based on block orthogonalization algorithms that we evoked in section 6.4 and on which more details can be found in [16]; the Q-OR algorithm presented in [34] that generates a non-orthogonal Krylov basis; mixed precision GMRES iterative refinement and split-preconditioned FGMRES covered respectively in [3] and [13] and on which we already have backward error analyses; inner-outer FGMRES algorithms which are FGMRES preconditioned by another GMRES algorithm, see for instance [44] or [11].

Acknowledgments

Sadly, the second author of this article is no longer among us. The rest of the authors would like to express their deepest gratitude for his dedication, mentorship, and humanity that pushed us to become better scientists; a gift we will never forget. Thank you Nick.

Funding

The work of the third author was supported by the France 2030 NumPEx Exa-MA (ANR-22-EXNU-0002) project managed by the French National Research Agency (ANR) and the ANR MixHPC project (ANR-23-CE46-0005-01).

The work of the fourth author was supported by the National Natural Science Foundation of China (No. 12288201).

Appendix A. Least squares problem via Householder Arnoldi

Theorem A.1. *Consider the solution of the HH-GMRES least squares problem*

$$\min_y \|b - C_k y\|_2, \quad C_k \in \mathbb{R}^{n\times k}, \quad 0 \neq b \in \mathbb{R}^{n\times n}, \tag{A.1}$$

via the Householder Arnoldi process run in precision of unit roundoff $u \ll 1$. For all $k \leq n$, as long as C_k is numerically full-rank, that is,

$$\sigma_{\min}(C_k) \gg u\|C_k\|_F, \tag{A.2}$$

the computed solution satisfies

$$\begin{aligned} \widehat{y}_k &= \arg\min_y \|(b+\Delta b) - (C_k + \Delta C_k)y\|_2, \\ \|\left[\Delta b, \Delta C_k\right] e_j\|_2 &\lesssim \widetilde{\gamma}_{nk+2(n+k)-2}\|\left[b, C_k\right] e_j\|_2, \quad j \leq k+1. \end{aligned} \tag{A.3}$$

Proof. The least squares problem minimizing the Arnoldi residual solved by HH-GMRES is not directly $\min_y \|b - C_k y\|_2$ but is instead $\min_y \|\beta e_1 - \bar{H}_k y\|_2$, where $\bar{H}_k$ and β are defined in (5.1). In exact arithmetic, the two share the same solution, but the second is computationally less expensive to solve. Thus, we need to show that the computed solution of the second least squares problem in floating-point is a backward stable solution of the first. Proofs already exist in the literature in some forms, see [18]. However, they are not compliant with our notations and not necessarily easy to appreciate by someone reading this article. Therefore, for the sake of completeness, we present what we think is an elegant and compact way to prove it.

As explained in section 5.2, the Householder Arnoldi process can be seen as a column-oriented Householder QR factorization of the matrix $[b, C_k] \in \mathbb{R}^{n\times(k+1)}$. The solution of the least squares problem $\min_y \|\beta e_1 - \bar{H}_k y\|_2$ is then obtained by triangularizing the Hessenberg matrix $\bar{H}_k$ extracted from the resulting R-factor. This triangularization is simply another QR factorization performed with Givens rotations. To carry out the proof, we need to consider separately the cases $k < n$ and $k = n$. This is because in the first case, the matrix $[b, C_k]$ is overdetermined, while in the second, it is underdetermined, leading to QR factors of different shapes.

We first address the case $k < n$. Consider the solution of the least squares problem

$$\min_y \|\widehat{R}\begin{bmatrix} 1 \\ -y \end{bmatrix}\|_2, \tag{A.4}$$

where $\widehat{R} \in \mathbb{R}^{(k+1)\times(k+1)}$ is the computed R-factor from the QR factorization of $[b, C_k]$ using the Householder orthogonalization. From [25, Thm. 19.4], we know that the QR factors of $[b, C_k]$ satisfy

$$\begin{aligned} \left[b, C_k\right] + \left[\Delta b^{(1)}, \Delta C_k^{(1)}\right] &= \left[\widetilde{V}_{k+1}, \widetilde{Q}_2\right]\begin{bmatrix}\widehat{R} \\ 0\end{bmatrix} \\ \|\left[\Delta b^{(1)}, \Delta C_k^{(1)}\right] e_j\|_2 &\leq \widetilde{\gamma}_{n(k+1)}\|\left[b, C_k\right] e_j\|_2, \quad j \leq k+1, \end{aligned} \tag{A.5}$$

where $\widetilde{V}_{k+1} \in \mathbb{R}^{n\times(k+1)}$ and $\widetilde{Q}_2 \in \mathbb{R}^{n\times(n-k-1)}$ are orthogonal. In particular, $\widehat{R}$ is upper triangular and can be decomposed as $\widehat{R} = [\widehat{\beta}e_1, \widehat{H}_k]$, where $\widehat{\beta} \approx \|b\|_2$ and $\widehat{H}_k \in \mathbb{R}^{(k+1)\times k}$ is an upper Hessenberg matrix. Hence, the least squares problem (A.4) is essentially $\min_y \|\beta e_1 - \bar{H}_k y\|_2$, but where the quantities β and $\bar{H}_k$ are replaced by their computed counterparts. To solve (A.4), we transform the upper Hessenberg matrix in a trapezoidal matrix by applying Givens rotations. We then compute the solution of the resulting triangular system. In the following we aim to show that, accounting for the rounding errors, this process provides a backward stable solution for (A.4). From [25, Thm. 19.10], we have

$$Q'^T(\widehat{R} + \Delta R^{(1)}) = \begin{bmatrix} w & \widehat{T} \\ \omega & 0 \end{bmatrix}, \quad \|\Delta R^{(1)} e_j\|_2 \leq \widetilde{\gamma}_{n+k-2}\|\widehat{R}e_j\|_2,$$

where $[\widehat{T}, 0]^T \in \mathbb{R}^{(k+1)\times k}$ is the computed upper triangular R-factor of $\widehat{H}_k$, $Q' \in \mathbb{R}^{(k+1)\times(k+1)}$ is orthogonal, $w \in \mathbb{R}^k$ is a vector, and ω is a scalar such that $[w, \omega]^T = Q'^T(\beta e_1)$. The computed solution of the least squares problem is obtained through a triangular solve with $\widehat{T}$ and satisfies $(\widehat{T} + \Delta T)\widehat{y}_k = w$ where $\|\Delta T e_j\|_2 \leq \gamma_k\|T e_j\|_2 \lesssim \gamma_k\|\widehat{R}e_{j+1}\|_2$ for $j = 1, \ldots, k$, see [25, Thm. 8.5]. The triangular solve is well-defined since $\widehat{T}$ is nonsingular for all k.

Indeed, from (A.5) and because $\widehat{R} = [\widehat{\beta}e_1, \widehat{H}_k]$, we have

$$C_k + \Delta C_k^{(1)} = \widetilde{V}_{k+1}\widehat{H}_k = \widetilde{V}_{k+1}Q'Q'^T\widehat{H}_k = \widetilde{V}_{k+1}Q'(\widehat{T} - Q'^T\Delta R_{2:}^{(1)}). \tag{A.6}$$

As $\widetilde{V}_{k+1}$ and Q' are orthogonal, the rank of $\widehat{T}$ is that of $C_k + \Delta C_k^{(1)} + \widetilde{V}_{k+1}\Delta R_{2:}^{(1)}$, which is full-rank by condition (A.2). Accounting for both errors in the Givens rotations and the triangular solve, we conclude that $\widehat{y}_k$ is the exact solution of the following perturbed least squares problem

$$\widehat{y}_k = \arg\min_y \|(\widehat{R} + \Delta R^{(2)})\begin{bmatrix}1\\-y\end{bmatrix}\|_2, \quad \|\Delta\widehat{R}^{(2)}e_j\|_2 \lesssim \widetilde{\gamma}_{n+2k-2}\|\widehat{R}e_j\|_2, \tag{A.7}$$

which shows that $\widehat{y}_k$ is a backward stable solution of (A.4) for all $k < n$. It remains to show that this is a backward stable solution of the original least squares problem (A.1). From (A.5), we have

$$\widehat{R} + \Delta R^{(2)} = \widetilde{V}_{k+1}^T([b, C_k] + [\Delta b^{(1)}, \Delta C_k^{(1)}] + \widetilde{V}_{k+1}\Delta R^{(2)}), \tag{A.8}$$

which combined with (A.7) gives

$$\begin{gathered}\widehat{y}_k = \arg\min_y \|([b, C_k] + [\Delta b^{(2)}, \Delta C_k^{(2)}])\begin{bmatrix}1\\-y\end{bmatrix}\|_2,\\ \|[\Delta b^{(2)}, \Delta C_k^{(2)}]e_j\|_2 \lesssim \widetilde{\gamma}_{nk+2(n+k)-2}\|[b, C_k]e_j\|_2, \quad j \le k+1,\end{gathered} \tag{A.9}$$

since $\|\widehat{R}e_j\|_2 \approx \|[b, C_k]e_j\|_2$, which ends the proof for $k < n$.

The case $k = n$ is slightly different because Householder orthogonalizes an underdetermined system $[b, C_n] \in \mathbb{R}^{n\times(n+1)}$. The application of HH-GMRES at step $k = n$ produces

$$[b, C_n] + [\Delta b^{(1)}, \Delta C_n^{(1)}] = \widetilde{V}_n\widehat{R}, \tag{A.10}$$

where $\widehat{R} = [\widehat{\beta}e_1, \widehat{H}_n] \in \mathbb{R}^{n\times(n+1)}$ is the computed upper trapezoidal R-factor, the errors $[\Delta b^{(1)}, \Delta C_n^{(1)}] \in \mathbb{R}^{n\times(n+1)}$ are equivalently defined as in (A.5), $\widetilde{V}_n \in \mathbb{R}^{n\times n}$ is orthogonal and is close to the computed Q-factor of the first n columns of $[b, C_n]$, and $\widehat{H}_n \in \mathbb{R}^{n\times n}$ is a square Hessenberg matrix which is a subtle difference with the case $k < n$ where $\widehat{H}_k$ is not square. As for the case $k < n$, we apply Givens rotations to $\widehat{H}_n$ followed by a triangular solve to obtain the solution. Carrying the same reasoning as for $k < n$, we can obtain an equivalent result as (A.9) which ends the proof for $k = n$. □

Appendix B. Columnwise backward error result for CGS

Theorem B.1. *Suppose that the CGS method run in floating-point arithmetic with unit roundoff $u \ll 1$ is applied to $B \in \mathbb{R}^{n\times k}$ of rank k, yielding computed matrices $\widehat{Q} \in \mathbb{R}^{n\times k}$ and $\widehat{R} \in \mathbb{R}^{k\times k}$. Then the computed QR factors satisfy*

$$B + \Delta B = \widehat{Q}\widehat{R}, \quad \|\Delta B e_j\|_2 \lesssim c(n,k)u\|Be_j\|_2, \quad j \le k, \tag{B.1}$$

for some polynomial of low degree $c(n,k)$ in n and k.

Proof. In exact arithmetic, the CGS algorithm computes the $Q = [q_1, \ldots, q_k]$ and $R = [r_1, \ldots, r_k]$ factors of $B = [b_1, \ldots, b_k]$ with the following recurrence

$$\begin{aligned} v_j &= [I - Q_{1:j-1}Q_{1:j-1}^T]b_j, \qquad & q_j &= v_j/\|v_j\|_2,\\ r_{1:j-1,j} &= Q_{1:j-1}^T b_j, & r_{j,j} &= \|v_j\|_2,\end{aligned}$$

where $Q_{1:j-1} \in \mathbb{R}^{n\times(j-1)}$ is the matrix composed of the $(j-1)$th vectors of Q and $r_{1:j-1,j} \in \mathbb{R}^{j-1}$ is the vector composed of the $(j-1)$th entries of r_j. Accounting for the floating-point errors in computing the vector subtractions, norms, and ℓ_2-inner-products, the process

satisfies instead for all $j \leq n$

$$\widehat{v}_j = b_j - \sum_{l=1}^{j-1} \widehat{q}_l \widehat{r}_{l,j} + \Delta v_j, \qquad \|\Delta v_j\|_2 \leq c(n,k)u\|b_j\|_2, \tag{B.2}$$

$$\widehat{q}_j = \widehat{v}_j/\|\widehat{v}_j\|_2 + \Delta q_j, \qquad \|\Delta q_j\|_2 \leq c(n,k)u, \tag{B.3}$$

$$\widehat{r}_{l,j} = \widehat{q}_l^T b_j + \delta r_{l,j}, \qquad |\delta r_{l,j}| \leq c(n,k)u\|\widehat{q}_l\|_2\|b_j\|_2, \quad \forall l \leq j-1, \tag{B.4}$$

$$\widehat{r}_{j,j} = \|\widehat{v}_j\|_2 + \delta r_{j,j}, \qquad |\delta r_{j,j}| \leq c(n,k)u\|\widehat{v}_j\|_2. \tag{B.5}$$

Combining (B.3) and (B.5) we obtain

$$\widehat{q}_j\widehat{r}_{j,j} = (\widehat{v}_j/\|\widehat{v}_j\|_2 + \Delta q_j)(\|\widehat{v}_j\|_2 + \delta r_{j,j}) \approx \widehat{v}_j + \Delta q_j\|\widehat{v}_j\|_2 + \delta r_{j,j}\widehat{v}_j/\|\widehat{v}_j\|_2,$$

which, used alongside (B.2), gives

$$b_j + \Delta v_j + \Delta q_j\|\widehat{v}_j\|_2 + \delta r_{j,j}\widehat{v}_j/\|\widehat{v}_j\|_2 \approx \sum_{l=1}^{j} \widehat{q}_l\widehat{r}_{l,j}. \tag{B.6}$$

Using (B.2), (B.3), and (B.4), we have the following bound

$$\|\widehat{v}_j\|_2 \lesssim \|b_j\|_2 + \sum_{l=1}^{j-1} \|\widehat{q}_l\|_2\|\widehat{q}_l^T\|_2\|b_j\|_2 \approx j\|b_j\|_2,$$

which, by defining $\Delta b_j = \Delta v_j + \Delta q_j\|\widehat{v}_j\|_2 + \delta r_{j,j}\widehat{v}_j/\|\widehat{v}_j\|_2$ in (B.6), finally gives

$$b_j + \Delta b_j \approx \widehat{Q}\widehat{r}_j, \quad \|\Delta b_j\|_2 \lesssim c(n,k)u\|b_j\|_2,$$

and completes the proof. □

Appendix C. CGS2 generates a closely-orthogonal set of vectors

In [21], it is shown that CGS2 computes a set of vectors orthogonal to the machine precision level as long as the orthogonalized matrix B is not numerically singular; see the following Theorem C.1. In this appendix, we build on this foundation to derive a set of useful results required to analyze CGS2-GMRES.

Theorem C.1. *(Rewrite of* [21, Thm. 2]*) Suppose CGS2 run in precision of unit roundoff $u \ll 1$ is applied to $B \in \mathbb{R}^{n\times k}$ of rank k yielding a computed Q-factor $\widehat{Q} \in \mathbb{R}^{n\times k}$. Then, as long as $c(n,k)u\kappa_2(B) \leq 1$, $\widehat{Q}$ satisfies*

$$\|I - \widehat{Q}^T\widehat{Q}\|_F \leq c(n,k)u, \tag{C.1}$$

where $c(n,k)$ are polynomials in n and k of low degree.

Lemma C.2. *Consider $\widehat{Q} \in \mathbb{R}^{n\times k}$. If $\|I - \widehat{Q}^T\widehat{Q}\|_F \leq c(n,k)u$, then $\widehat{Q}$ is well-conditioned and we have*

$$\sigma_{\min}(\widehat{Q}) \approx \sigma_{\max}(\widehat{Q}) \approx \kappa_2(\widehat{Q}) \approx 1, \tag{C.2}$$

where $c(n,k)$ are polynomials in n and k of low degree.

Proof. By assumption the vectors of $\widehat{Q}$ are orthogonal to machine precision and we have

$$\|I - \widehat{Q}^T\widehat{Q}\|_F \leq c(n,k)u. \tag{C.3}$$

To evaluate the smallest and largest singular values of $\widehat{Q}$ we consider the polar decomposition $\widehat{Q} = UH$, where $U \in \mathbb{R}^{n\times k}$ is orthogonal and $H \in \mathbb{R}^{k\times k}$ is symmetric positive-semidefinite. Using [24, Lem. 5.1] combined with (C.3) we can bound the distance from $\widehat{Q}$ to U, we obtain

$$\|\widehat{Q} - U\|_F \leq \|I - \widehat{Q}^T\widehat{Q}\|_F \leq c(n,k)u.$$

Using $\widehat{Q} = \widehat{Q} - U + U$ we can write

$$\sigma_{\min}(\widehat{Q}) = \min_x \frac{\|(\widehat{Q} - U + U)x\|_2}{\|x\|_2} \leq \sigma_{\min}(U) + \|\widehat{Q} - U\|_F \leq 1 + c(n,k)u,$$

$$\sigma_{\min}(\widehat{Q}) \geq \sigma_{\min}(U) - \|\widehat{Q} - U\|_F \geq 1 - c(n,k)u,$$

which provides $\sigma_{\min}(\widehat{Q}) \approx 1$. The same reasoning can be used to show that $\sigma_{\max}(\widehat{Q}) \approx 1$ from which we deduce $\kappa_2(\widehat{Q}) \approx 1$ and which ends the proof. □

Condition $c(n,k)u\kappa_2(B) \leq 1$ of Theorem C.1 on the non numerical singularity of B can be exchanged with conditions on the norms of the projections $(I - \widehat{Q}_{j-1}\widehat{Q}_{j-1}^T)b_j$ for $j \leq k$ and where $\widehat{Q}_{j-1} \in \mathbb{R}^{n\times(j-1)}$ is the $(j-1)$ first columns of $\widehat{Q}$. Namely, we will show that if for all $j \leq k$ the norm of the projection $(I - \widehat{Q}_{j-1}\widehat{Q}_{j-1}^T)b_j$, which is the projection of b_j on the orthogonal complement of $\widehat{Q}_{j-1}$ formed by CGS2 at the jth iteration to compute $\widehat{Q}_j$, is large enough, then the conclusion of Theorem C.1 holds. In exact arithmetic, those conditions on the projections $(I - Q_{j-1}Q_{j-1}^T)b_j$ enforce b_j for all $j \leq k$ to never lie in the range of Q_{j-1}, that is, the range of $B_{j-1} = [b_1, \ldots, b_{j-1}]$, and, therefore, enforce the columns of B to be independent so that B is nonsingular.

Theorem C.3. *Suppose that the first $k-1$ iterations of CGS2 run in precision of unit roundoff $u \ll 1$ and applied to $B_{k-1} \in \mathbb{R}^{n\times(k-1)}$ yields a computed Q-factor $\widehat{Q}_{k-1} \in \mathbb{R}^{n\times(k-1)}$ satisfying*

$$\|I - \widehat{Q}_{k-1}^T\widehat{Q}_{k-1}\|_F \leq c(n,k)u. \tag{C.4}$$

Consider the kth iteration of CGS2 applied on $B_k = [B_{k-1},\ b_k]$ of rank k and yielding a computed Q-factor $\widehat{Q}_k = [\widehat{Q}_{k-1},\ \widehat{q}_k]$. As long as $\|(I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 \geq c(n,k)u\|b_k\|_2$, the orthogonality of $\widehat{Q}_k$ is preserved and we have

$$\|I - \widehat{Q}_k^T\widehat{Q}_k\|_F \lesssim c(n,k)u, \tag{C.5}$$

where $c(n,k)$ accounts for polynomials in n and k of low degree.

Proof. The following proof is a direct revisit of [21, Thm. 2]. While our proof is self-contained, it does not provide the level of details and insights present in [21]. Therefore, we strongly recommend reading this work for a deeper understanding of the CGS2 orthogonalization in floating-point.

Under condition (C.4) and since

$$I - \widehat{Q}_k^T\widehat{Q}_k = \begin{bmatrix} I - \widehat{Q}_{k-1}^T\widehat{Q}_{k-1} & -\widehat{Q}_{k-1}^T\widehat{q}_k \\ -\widehat{q}_k^T\widehat{Q}_{k-1} & 1 - \widehat{q}_k^T\widehat{q}_k \end{bmatrix},$$

proving (C.5) reduces to show that $\|\widehat{Q}_{k-1}^T\widehat{q}_k\|_2 \leq c(n,k)u$. We apply the kth iteration of CGS2 on B_k to compute $\widehat{q}_k$. Accounting for the floating point errors, the process yields the following two set of projections

$$\widehat{v}_k = b_k - \sum_{j=1}^{k-1} \widehat{q}_j\widehat{r}_{j,k} + \Delta v_k, \qquad \|\Delta v_k\|_2 \leq c(n,k)u\|b_k\|_2, \tag{C.6}$$

$$\widehat{w}_k = \widehat{v}_k - \sum_{j=1}^{k-1} \widehat{q}_j\widehat{s}_{j,k} + \Delta w_k, \qquad \|\Delta w_k\|_2 \leq c(n,k)u\|\widehat{v}_k\|_2, \tag{C.7}$$

associated, respectively, to the first and second application of the Gram-Schmidt orthogonalization. The computed orthogonalization coefficients $\widehat{r}_{j,k}$ and $\widehat{s}_{j,k}$ for $j \leq k-1$ satisfy

$$\begin{aligned} &\widehat{r}_{j,k} = \widehat{q}_j^T a_k + \delta r_{j,k}, \quad \widehat{s}_{j,k} = \widehat{q}_j^T\widehat{v}_k + \delta s_{j,k}, \quad \widehat{s}_{k,k} = \|\widehat{w}_k\|_2 + \delta s_{k,k}, \\ &|\delta r_{j,k}| \leq c(n,k)u\|\widehat{q}_j^T\|_2\|b_k\|_2, \quad |\delta s_{j,k}| \leq c(n,k)u\|\widehat{q}_j^T\|_2\|\widehat{v}_k\|_2, \quad |\delta s_{k,k}| \leq c(n,k)u\|\widehat{w}_k\|_2, \end{aligned} \tag{C.8}$$

and $\widehat{q}_k$ is finally obtained as

$$\widehat{q}_k = \widehat{w}_k/\|\widehat{w}_k\|_2 + \Delta q_k, \quad \|\Delta q_k\|_2 \leq c(n,k)u, \quad \|\widehat{q}_k\|_2 \leq 1 + c(n,k)u. \tag{C.9}$$

To bound $\|\widehat{Q}_{k-1}^T\widehat{q}_k\|_2$ we first need to derive bounds for $\|\widehat{v}_k\|_2$, $\|\widehat{Q}_{k-1}^T\widehat{v}_k\|_2/\|\widehat{v}_k\|_2$, and $\|\widehat{Q}_{k-1}^T\widehat{w}_k\|_2/\|\widehat{w}_k\|_2$.

We start by providing a lower bound for $\|\widehat{v}_k\|_2$; from (C.6), (C.8), and (C.9) we have

$$\begin{aligned}\|\widehat{v}_k\|_2 &= \|(I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k - \sum_{j=1}^{k-1}\widehat{q}_j\delta r_{j,k} + \Delta v_k\|_2\\ &\geq \|(I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 - \sum_{j=1}^{k-1}\|\widehat{q}_j\|_2|\delta r_{j,k}| - \|\Delta v_k\|_2\\ &\gtrsim \|(I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 - c_1(n,k)u\|b_k\|_2,\end{aligned}$$

where $c_1(n,k)$ is a polynomial in n and k of low degree. Under the assumption

$$\|(I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 \geq c_0(n,k)u\|b_k\|_2 \tag{C.10}$$

of the Theorem, and by choosing $c_0(n,k)$ sufficiently larger than $c_1(n,k)$, we guarantee

$$\|\widehat{v}_k\|_2 \gtrsim \big[c_0(n,k)-c_1(n,k)\big]u\|b_k\|_2 = c(n,k)u\|b_k\|_2. \tag{C.11}$$

We emphasize that we are not interested in determining a specific value for $c_0(n,k)$ in assumption (C.10). Our goal is only to validate that there exists such a polynomial $c_0(n,k)$ of low degree in n and k such that the Theorem holds.

Multiplying the expression (C.6) from the left by $\widehat{Q}_{k-1}^T$ yields

$$\widehat{Q}_{k-1}^T\widehat{v}_k = (I-\widehat{Q}_{k-1}^T\widehat{Q}_{k-1})\widehat{Q}_{k-1}^Tb_k + \widehat{Q}_{k-1}^T\big[-\sum_{j=1}^{k-1}\widehat{q}_j\delta r_{j,k} + \Delta v_k\big],$$

and taking the norm of this expression accounting for the assumption (C.4), the bounds on the errors (C.6) and (C.8), and the fact that $\|\widehat{Q}_{k-1}\|_F \approx (k-1)^{1/2}$ gives the bound

$$\begin{aligned}\|\widehat{Q}_{k-1}^T\widehat{v}_k\|_2 &\leq \|(I-\widehat{Q}_{k-1}^T\widehat{Q}_{k-1})\|_F\|\widehat{Q}_{k-1}^T\|_F\|b_k\|_2 + \|\widehat{Q}_{k-1}^T\|_F\sum_{j=1}^{k-1}\|\widehat{q}_j\|_2|\delta r_{j,k}| + \|\Delta v_k\|_2\\ &\lesssim c_2(n,k)u\|b_k\|_2.\end{aligned}$$

Therefore, combining this previous bound with (C.11) and enforcing $c_0(n,k)$ to be sufficiently larger than $c_2(n,k)+c_1(n,k)$ gives

$$\|\widehat{Q}_{k-1}^T\widehat{v}_k\|_2/\|\widehat{v}_k\|_2 \lesssim c_2(n,k)/\big[c_0(n,k)-c_1(n,k)\big] < 1. \tag{C.12}$$

We now provide a bound for $\|\widehat{Q}_{k-1}^T\widehat{w}_k\|_2/\|\widehat{w}_k\|_2$. We can rewrite (C.7) in the following form

$$\widehat{w}_k = (I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)\widehat{v}_k - \sum_{j=1}^{k-1}\widehat{q}_j\delta s_{j,k} + \Delta w_k$$

from which, accounting for the bounds (C.7), (C.8), and (C.12), we deduce

$$\begin{aligned}\frac{\|\widehat{w}_k\|_2}{\|\widehat{v}_k\|_2} &\geq \frac{\|\widehat{v}_k\|_2}{\|\widehat{v}_k\|_2} - \|\widehat{Q}_{k-1}\|_F\frac{\|\widehat{Q}_{k-1}^T\widehat{v}_k\|_2}{\|\widehat{v}_k\|_2} - \frac{\sum_{j=1}^{k-1}\|\widehat{q}_j\|_2|\delta s_{j,k}|}{\|\widehat{v}_k\|_2} - \frac{\|\Delta w_k\|_2}{\|\widehat{v}_k\|_2}\\ &\gtrsim 1 - \frac{c_3(n,k)}{c_0(n,k)-c_1(n,k)} - c(n,k)u.\end{aligned}$$

Hence, choosing, for instance, $c_0(n,k) \leq 2c_3(n,k)+c_1(n,k)$ leads to

$$\|\widehat{v}_k\|_2/\|\widehat{w}_k\|_2 \lesssim 2. \tag{C.13}$$

In the same fashion as for deriving the bound (C.12), multiplying the expression (C.7) from the left by $\widehat{Q}_{k-1}^T$, taking the norm, and using the bounds (C.4), (C.8), and (C.13) yields

$$\begin{aligned}\frac{\|\widehat{Q}_{k-1}^T\widehat{w}_k\|_2}{\|\widehat{w}_k\|_2} &\leq \|(I-\widehat{Q}_{k-1}^T\widehat{Q}_{k-1})\|_F\|\widehat{Q}_{k-1}^T\|_F\frac{\|\widehat{v}_k\|_2}{\|\widehat{w}_k\|_2} + \|\widehat{Q}_{k-1}^T\|_F\frac{\sum_{j=1}^{k-1}\|\widehat{q}_j\|_2|\delta s_{j,k}| + \|\Delta w_k\|_2}{\|\widehat{w}_k\|_2}\\ &\lesssim c(n,k)u.\end{aligned} \tag{C.14}$$

Finally, taking $\widehat{Q}_{k-1}^T\widehat{q}_k = \widehat{Q}_{k-1}^T\widehat{w}_k/\|\widehat{w}_k\|_2 + \widehat{Q}_{k-1}^T\Delta q_k$ and using (C.9) and (C.14) we write

$$\|\widehat{Q}_{k-1}^T\widehat{q}_k\|_2 \leq \|\widehat{Q}_{k-1}^T\widehat{w}_k\|_2/\|\widehat{w}_k\|_2 + \|\widehat{Q}_{k-1}^T\Delta q_k\|_2 \lesssim c(n,k)u,$$

which ends the proof. □

From Theorem C.3, we can derive the following Corollary where we explain that the loss of orthogonality of the computed vectors of $\widehat{Q}_k$ by CGS2 at the iteration k can only be the consequence of a small projection norm $\|(I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2$.

Corollary C.4. *Suppose that the first $k-1$ iterations of CGS2 run in precision of unit roundoff $u \ll 1$ and applied to $B_{k-1} \in \mathbb{R}^{n\times(k-1)}$ yields a computed Q-factor $\widehat{Q}_{k-1} \in \mathbb{R}^{n\times(k-1)}$ satisfying $\|I - \widehat{Q}_{k-1}^T\widehat{Q}_{k-1}\|_F \leq c(n,k)u$. Consider the kth iteration of CGS2 applied on $B_k = [B_{k-1},\ b_k]$ of rank k and yielding a computed Q-factor $\widehat{Q}_k = [\widehat{Q}_{k-1},\ \widehat{q}_k]$. If $\|I - \widehat{Q}_k^T\widehat{Q}_k\|_F > c(n,k)u$, then*

$$\|(I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 < c(n,k)u\|b_k\|_2, \tag{C.15}$$

the vector $\widehat{w}_k$ resulting from the computation of the two consecutive applications of the projection $(I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)$ yielding in exact arithmetic $\widetilde{w}_k = (I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)(I-\widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k$ satisfies

$$\|\widehat{w}_k\|_2 \lesssim c(n,k)u\|b_k\|_2, \tag{C.16}$$

and B_k is numerically singular

$$\sigma_{\min}(B_k) \leq c(n,k)u\|B_k\|_F, \tag{C.17}$$

where $c(n,k)$ are some polynomials in n and k of low degree.

Proof. The converse of Theorem C.3 implies that if $\|I - \widehat{Q}_k^T\widehat{Q}_k\|_F > c(n,k)u$ but $\|I - \widehat{Q}_{k-1}^T\widehat{Q}_{k-1}\|_F \leq c(n,k)u$, then we must have

$$\|(I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 < c(n,k)u\|b_k\|_2,$$

and we recover (C.15).

From (C.6) and (C.7) we have

$$\widehat{v}_k = (I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k - \sum_{j=1}^{k-1}\widehat{q}_j\delta r_{j,k} + \Delta v_k \quad \text{and} \quad \widehat{w}_k = (I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)\widehat{v}_k - \sum_{j=1}^{k-1}\widehat{q}_j\delta s_{j,k} + \Delta w_k,$$

where $\delta r_{j,k}$, $\delta s_{j,k}$, Δv_k, and Δw_k are defined in (C.6), (C.7), and (C.8). Accounting for (C.9) we have $\|\widehat{q}_j\|_2 \approx 1$, and using in addition (C.15) we deduce

$$\|\widehat{v}_k\|_2 \leq \|(I - \widehat{Q}_{k-1}\widehat{Q}_{k-1}^T)b_k\|_2 + \sum_{j=1}^{k-1}\|\widehat{q}_j\|_2|\delta r_{j,k}| + \|\Delta v_k\|_2 \lesssim c(n,k)u\|b_k\|_2.$$

Moreover, considering $\|\widehat{Q}_{k-1}\|_F \approx (k-1)^{1/2}$, we obtain

$$\|\widehat{w}_k\|_2 \leq (\|I\|_F + \|\widehat{Q}_{k-1}\|_F^2)\|\widehat{v}_k\|_2 + \sum_{j=1}^{k-1}\|\widehat{q}_j\|_2|\delta s_{j,k}| + \|\Delta w_k\|_2 \lesssim c(n,k)\|\widehat{v}_k\|_2 \lesssim c(n,k)u\|b_k\|_2,$$

from which we recover (C.16).

Finally, from Theorem B.1, which holds for CGS and a fortiori for CGS2, the kth first iterations of CGS2 yields computed factors satisfying

$$B_k = [B_{k-1}, b_k] = \left[\widehat{Q}_{k-1}, \widehat{q}_k\right]\begin{bmatrix}\widehat{R}_{k-1} & \widehat{r}_{1:k-1,k}\\ 0 & \|\widehat{w}_k\|_2\end{bmatrix} + \Delta B_k^{(1)}, \quad \|\Delta B_k^{(1)}e_j\|_2 \leq c(n,k)u\|B_ke_j\|_2, \quad j \leq k.$$

Since CGS2 constructs $\widehat{q}_k = \widehat{w}_k/\|\widehat{w}_k\|_2 + \Delta q_k$ with $\|\Delta q_k\|_2 \leq c(n,k)u$, see (C.9), we obtain

$$B_k + \Delta B_k^{(2)} = \widehat{Q}_{k-1}\left[\widehat{R}_{k-1}, \widehat{r}_{1:k-1,k}\right], \quad \Delta B_k^{(2)} = -(\widehat{w}_k + \Delta q_k\|\widehat{w}_k\|_2)e_k^T - \Delta B_k^{(1)},$$

and using (C.16), we bound

$$\|\Delta B_k^{(2)}e_j\|_2 \lesssim c(n,k)u\|B_ke_j\|_2, \quad j \leq k.$$

Because $\left[\widehat{R}_{k-1}, \widehat{r}_{1:k-1,k}\right]$ has rank $k-1$, it is singular and

$$0 = \sigma_{\min}(B_k + \Delta B_k^{(2)}) \geq \sigma_{\min}(B_k) - \|\Delta B_k^{(2)}\|_F$$

leading to

$$\sigma_{\min}(B_k) \lesssim c(n,k)u\|B_k\|_F$$

which proves (C.17) and ends the proof. □

Because the computed $\widehat{Q}$ by CGS2 is invariant by column scaling $B \leftarrow BD$ for all diagonal $D > 0$, at least if D comprises powers of the machine base (see comments in [25, p. 373] and [32, p 502]), (C.17) in Corollary C.4 can be replaced by

$$\sigma_{\min}(BD) < c(n,k)u\|BD\|_F, \quad \text{for all diagonal } D > 0. \tag{C.18}$$

The result remains true even when including the cases where the entries of D are not all powers of the machine base, but for the sake of conciseness we do not attempt a proof of this statement in this article.

References

[1] Emmanuel Agullo, Luc Giraud, and Yan-Fei Jing. Block GMRES Method with Inexact Breakdowns and Deflated Restarting. *SIAM J. Matrix Anal. Appl.*, 35(4):1625–1651, January 2014.

[2] José I Aliaga, Hartwig Anzt, Thomas Grützmacher, Enrique S Quintana-Ortí, and Andrés E Tomás. Compressed basis GMRES on high-performance graphics processing units. *The International Journal of High Performance Computing Applications*, August 2022.

[3] Patrick Amestoy, Alfredo Buttari, Nicholas J. Higham, Jean-Yves L'Excellent, Theo Mary, and Bastien Vieublé. Five-precision GMRES-based iterative refinement. MIMS EPrint 2021.5, Manchester Institute for Mathematical Sciences, The University of Manchester, UK, April 2021. Revised April 2022.

[4] Mario Arioli and Iain S. Duff. Using FGMRES to obtain backward stability in mixed precision. *Electron. Trans. Numer. Anal.*, 33:31–44, 2008.

[5] Mario Arioli, Iain S. Duff, Serge Gratton, and Stéphane Pralet. A Note on GMRES Preconditioned by a Perturbed LDL^T Decomposition with Static Pivoting. *SIAM J. Sci. Comput.*, 29(5):2024–2044, 2007.

[6] Oleg Balabanov and Laura Grigori. Randomized Gram–Schmidt Process with Application to GMRES. *SIAM J. Sci. Comput.*, 44(3):A1450–A1474, June 2022.

[7] Jesse L. Barlow. Block Modified Gram–Schmidt Algorithms and Their Analysis. *SIAM J. Matrix Anal. Appl.*, 40(4):1257–1290, January 2019.

[8] Åke Björck. Solving linear least squares problems by Gram-Schmidt orthogonalization. *BIT*, 7(1):1–21, March 1967.

[9] Åke Björck and Christopher C. Paige. Loss and recapture of orthogonality in the modified gram–schmidt algorithm. *SIAM J. Matrix Anal. Appl.*, 13(1):176–190, January 1992.

[10] Sylvie Boldo, Claude-Pierre Jeannerod, Guillaume Melquiond, and Jean-Michel Muller. Floating-point arithmetic. *Acta Numerica*, 32:203–290, May 2023.

[11] Alfredo Buttari, Jack Dongarra, Jakub Kurzak, Piotr Luszczek, and Stanimir Tomov. Using Mixed Precision for Sparse Matrix Computations to Enhance the Performance while Achieving 64-bit Accuracy. *ACM Trans. Math. Software*, 34(4):1–22, July 2008.

[12] Erin Carson. *Communication-Avoiding Krylov Subspace Methods in Theory and Practice*. PhD thesis, University of California, Berkeley, 2015.

[13] Erin Carson and Ieva Daužickaitė. The stability of split-preconditioned FGMRES in four precisions, 2023.

[14] Erin Carson and Nicholas J. Higham. Accelerating the solution of linear systems by iterative refinement in three precisions. *SIAM J. Sci. Comput.*, 40(2):A817–A847, 2018.

[15] Erin Carson and Noaman Khan. Mixed precision iterative refinement with sparse approximate inverse preconditioning. *SIAM J. Sci. Comput.*, 45(3):C131–C153, June 2023.

[16] Erin Carson, Kathryn Lund, Miroslav Rozložník, and Stephen Thomas. Block Gram-Schmidt algorithms and their stability properties. *Linear Algebra and its Applications*, 638:150–195, April 2022.

[17] Michael P. Connolly, Nicholas J. Higham, and Theo Mary. Stochastic rounding and its probabilistic backward error analysis. *SIAM J. Sci. Comput.*, 43(1):A566–A585, January 2021.

[18] Jitka Drkošová, Anne Greenbaum, Miroslav Rozložník, and Zdeněk Strakoš. Numerical stability of GMRES. *BIT Numerical Mathematics*, 35(3):309–330, September 1995.

[19] Luc Giraud, Serge Gratton, and Julien Langou. A note on relaxed and flexible GMRES. Technical report, Technical Report TR/PA/04/41, CERFACS, Toulouse, France, 2004.

[20] Luc Giraud and Julien Langou. When modified Gram–Schmidt generates a well-conditioned set of vectors. *IMA J. Numer. Anal.*, 22(4):521–528, 10 2002.

[21] Luc Giraud, Julien Langou, Miroslav Rozložník, and Jasper van den Eshof. Rounding error analysis of the classical Gram-Schmidt orthogonalization process. *Numerische Mathematik*, 101(1):87–100, May 2005.
[22] Stef Graillat, Fabienne Jézéquel, Théo Mary, and Roméo Molina. Adaptive precision sparse matrix-vector product and its application to Krylov solvers. working paper or preprint, September 2022.
[23] Anne Greenbaum, Vlastimil Pták, and Zdenvk Strakoš. Any nonincreasing convergence curve is possible for GMRES. *SIAM J. Matrix Anal. Appl.*, 17(3):465–469, July 1996.
[24] Nicholas J. Higham. The matrix sign decomposition and its relation to the polar decomposition. *Linear Algebra and its Applications*, 212-213:3–20, November 1994.
[25] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2002.
[26] Nicholas J. Higham and Theo Mary. A new approach to probabilistic rounding error analysis. *SIAM J. Sci. Comput.*, 41(5):A2815–A2835, 2019.
[27] Nicholas J. Higham and Theo Mary. Sharper Probabilistic Backward Error Analysis for Basic Linear Algebra Kernels with Random Data. *SIAM J. Sci. Comput.*, 42(5):A3427–A3446, 2020.
[28] Mark Hoemmen. *Communication-avoiding Krylov subspace methods*. PhD thesis, University of California, Berkeley, 2010.
[29] William Jalby and Bernard Philippe. Stability Analysis and Improvement of the Block Gram–Schmidt Algorithm. *SIAM J. Sci. Comput.*, 12(5):1058–1073, September 1991.
[30] Pavel Jiránek, Miroslav Rozložník, and Martin H. Gutknecht. How to Make Simpler GMRES and GCR More Stable. *SIAM J. Matrix Anal. Appl.*, 30(4):1483–1499, January 2009.
[31] Julien Langou. *Solving large linear systems with multiple right–hand sides*. PhD thesis, Ph. D. dissertation, INSA Toulouse, 2003.
[32] Steven J. Leon, Åke Björck, and Walter Gander. Gram-Schmidt orthogonalization: 100 years and more. *Numerical Linear Algebra with Applications*, 20(3):492–532, June 2012.
[33] Neil Lindquist, Piotr Luszczek, and Jack Dongarra. Accelerating Restarted GMRES With Mixed Precision Arithmetic. *IEEE Transactions on Parallel and Distributed Systems*, 33(4):1027–1037, April 2022.
[34] Gérard Meurant. An optimal Q-OR Krylov subspace method for solving linear systems. *Electron. Trans. Numer. Anal.*, 27:127–152, 2018.
[35] Ronald B. Morgan. GMRES with Deflated Restarting. *SIAM J. Sci. Comput.*, 24(1):20–37, January 2002.
[36] Christopher C. Paige, Miroslav Rozlozník, and Zdenvek Strakos. Modified Gram-Schmidt (MGS), Least Squares, and Backward Stability of MGS-GMRES. *SIAM J. Matrix Anal. Appl.*, 28(1):264–284, January 2006.
[37] Christopher C. Paige and Zdenek Strakoš. Bounds for the least squares distance using scaled total least squares. *Numerische Mathematik*, 91(1):93–115, March 2002.
[38] Mickaël Robbé and Miloud Sadkane. Exact and inexact breakdowns in the block GMRES method. *Linear Algebra and its Applications*, 419(1):265–285, November 2006.
[39] Miroslav Rozložník. *Numerical Stability of the GMRES Method*. PhD thesis, Institute of Computer Science, Academy of Sciences of the Czech Republic, 1996.
[40] Youcef Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, 1993.
[41] Youcef Saad and Martin H. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856–869, July 1986.
[42] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, second edition, 2003.
[43] Valeria Simoncini and Efstratios Gallopoulos. Convergence properties of block GMRES and matrix polynomials. *Linear Algebra and its Applications*, 247:97–119, November 1996.
[44] Valeria Simoncini and Daniel B. Szyld. Flexible Inner-Outer Krylov Subspace Methods. *SIAM J. Numer. Anal.*, 40(6):2219–2239, January 2002.
[45] Dong-Lin Sun, Ting-Zhu Huang, Bruno Carpentieri, and Yan-Fei Jing. A New Shifted Block GMRES Method with Inexact Breakdowns for Solving Multi-Shifted and Multiple Right-Hand Sides Linear Systems. *Journal of Scientific Computing*, 78(2):746–769, July 2018.
[46] Kathryn Turner and Homer F. Walker. Efficient High Accuracy Solutions with GMRES(m). *SIAM J. Sci. Statist. Comput.*, 13(3):815–825, May 1992.
[47] Bastien Vieublé. *Mixed precision iterative refinement for the solution of large sparse linear systems*. PhD thesis, INP Toulouse, November 2022.
[48] Homer F. Walker. Implementation of the GMRES Method Using Householder Transformations. *SIAM J. Sci. Statist. Comput.*, 9(1):152–163, January 1988.
[49] Homer F. Walker and Lu Zhou. A simpler GMRES. *Numer. Linear Algebra Appl.*, 1(6):571–581, November 1994.
[50] Andrew J. Wathen. Preconditioning. *Acta Numerica*, 24:329–376, 2015.
[51] Qinmeng Zou. GMRES algorithms over 35 years. *Appl. Math. Comput.*, 445:127869, May 2023.