INSTITUTE OF MATHEMATICS

THE CZECH ACADEMY OF SCIENCES

# The stability of block variants of classical Gram-Schmidt

*Erin Carson*
*Kathryn Lund*
*Miroslav Rozložník*

Preprint No. 6-2021

PRAHA 2021

# THE STABILITY OF BLOCK VARIANTS OF CLASSICAL GRAM-SCHMIDT

ERIN CARSON*, KATHRYN LUND*, AND MIROSLAV ROZLOŽNÍK†

**Abstract.** The block version of classical Gram-Schmidt (BCGS) is often employed to efficiently compute orthogonal bases for Krylov subspace methods and eigenvalue solvers, but a rigorous proof of its stability behavior has not yet been established. It is shown that the usual implementation of BCGS can lose orthogonality at a rate worse than $O(\varepsilon)\kappa^2(X)$, where $\varepsilon$ is the unit round-off. A useful intermediate quantity denoted as the Cholesky residual is given special attention and, along with a block generalization of the Pythagorean theorem, this quantity is used to develop more stable variants of BCGS. These variants are proved to have $O(\varepsilon)\kappa^2(X)$ loss of orthogonality with relatively relaxed conditions on the intra-block orthogonalization routine. A variety of numerical examples illustrate the theoretical bounds.

**Key words.** Gram-Schmidt, blocking, block vectors, block Krylov subspace methods, stability, loss of orthogonality

**AMS subject classifications.** 15-02, 15A23, 65-02, 65F05, 65F10, 65F25

**1. Introduction.** Given a matrix $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{m\times n}$, $m \geq n$, we consider the problem of computing its QR factorization $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{Q}}\mathcal{R}$, where $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{m\times n}$ has orthonormal columns and $\mathcal{R} \in \mathbb{R}^{n\times n}$ is upper triangular with positive diagonal entries. In particular, we are concerned with the use of Block Classical Gram-Schmidt (`BCGS`) to accomplish this task, which operates on a block of $s$ vectors each iteration. This corresponds to partitioning $\boldsymbol{\mathcal{X}}$ into a set of $p$ block vectors, each of size $m \times s$:

$$\boldsymbol{\mathcal{X}} = [\boldsymbol{X}_1 \,|\, \boldsymbol{X}_2 \,|\, \cdots \,|\, \boldsymbol{X}_p]. \tag{1.1}$$

`BCGS` uses a block version of the Classical Gram-Schmidt (`CGS`) method to perform *inter-block* orthogonalization. The implementation of a algorithm also requires choosing an `IntraOrtho` routine, which is the method used to perform *intra-block* orthogonalization (also sometimes referred to in the literature as "panel factorization" or "local QR").

With $\bar{\boldsymbol{\mathcal{Q}}} \in \mathbb{R}^{m\times n}$ and $\bar{\mathcal{R}} \in \mathbb{R}^{n\times n}$ denoting the computed analogs of the QR factorization $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{Q}}\mathcal{R}$, this work primarily aims to bound the *loss of orthogonality* in `BCGS`,

$$\left\| I - \bar{\boldsymbol{\mathcal{Q}}}^T \bar{\boldsymbol{\mathcal{Q}}} \right\|,$$

where we use $I$ to denote the identity matrix of appropriate size (here $n \times n$). Another salient quantity is the *residual*

$$\left\| \boldsymbol{\mathcal{X}} - \bar{\boldsymbol{\mathcal{Q}}}\bar{\mathcal{R}} \right\|. \tag{1.2}$$

We will also consider the *Cholesky residual*,

$$\left\| \boldsymbol{\mathcal{X}}^T \boldsymbol{\mathcal{X}} - \bar{\mathcal{R}}^T \bar{\mathcal{R}} \right\|,$$

*Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic, {carson, lundka}@karlin.mff.cuni.cz. Both authors are supported by Charles University PRIMUS project no. PRIMUS/19/SCI/11 and Charles University Research program no. UNCE/SCI/023. The first author is additionally supported by Lawrence Livermore National Security, LLC Subcontract Award B639388 under Prime Contract No. DE-AC52-07NA27344.

†Institute of Mathematics, Czech Academy of Sciences, Prague, Czech Republic, miro@math.cas.cz. Supported by the Grant Agency of the Czech Republic, Grant No. 20-010745 and by the Czech Academy of Science (RVO 67985840).

which emphasizes the connections between BCGS and Cholesky factorization.

Our focus on Gram-Schmidt algorithms for computing the QR factorization is motivated by their prominent use in Krylov subspace iterative methods for solving linear systems, least squares problems, and eigenvalue problems. Due to their reduced synchronization requirements from batching inner products and norms, Block Gram-Schmidt (BGS) variants, and in particular BCGS, play a vital role in communication-avoiding Krylov subspace method variants designed for high-performance computing (HPC), such as $s$-step [13] and enlarged [11] methods.

The development of $s$-step Arnoldi and GMRES algorithms dates back to early work of Walker [19], Joubert and Carey [15], Bai, Hu, and Reichel [2], and others. For a detailed historical background, see [13]. The idea behind the $s$-step approach is that the Krylov subspace is constructed in blocks of $\mathcal{O}(s)$ vectors at a time, and blocks are subsequently orthogonalized using a block orthogonalization routine. We note that the $s$-step approach can be seen as a special case of block Krylov subspace methods but refrain from elaborating on this idea here. In [13, Section 2.4], Hoemmen details the communication cost of BGS methods and shows that BCGS requires a factor of $p$ fewer messages versus standard Block Modified Gram-Schmidt (BMGS) on a parallel machine. For recent performance results of $s$-step GMRES on a distributed memory parallel machine, see, e.g., the work of Yamazaki et al. [20].

Although BCGS is widely used to perform orthogonalization in practical implementations, its stability properties in floating-point arithmetic are not well understood. To date, there have been no rigorous stability studies of BCGS, and there are only a few existing results for other block variants; for a comprehensive overview of what has been studied, see the recent survey [6]. Consequently, theoretical results on the backward stability of block Krylov subspace methods, particularly communication-avoiding variants, are lacking.

The most rigorous recent work is that of Barlow and Smoktunowicz [4], who study a reorthogonalized BGS variant, and Barlow [3], who develops a low-synchronization BGS, consists of best-case-scenario analysis, where the orthogonalization scheme used within blocks (which here we refer to as IntraOrtho) is assumed to have $\mathcal{O}(\varepsilon)$ loss of orthogonality. Here we take a different approach, seeing how much we can relax conditions on the IntraOrtho while still ensuring the desired bounds.

The parent algorithm CGS inspires the approach we take with the stability analysis of BCGS. Prior to work by Giraud et al. [10] and Smoktunowicz et al. [18], the upper bounds on the only loss of orthogonality established for CGS were rather pessimistic [1, 5, 16]. Giraud et al. [10] show that the upper bound is actually $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}})$, as long as $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}) < 1$. Smoktunowicz et al. [18] clarify that while this bound does not hold for CGS, it does hold for CGS-P, a variant that computes the diagonals of the R-factor via the Pythagorean theorem (hence the "P"). The situation turns out to be similar for BCGS, so we develop BCGS-PIP and BCGS-PIO, two variants which use a block vector analogy of the Pythagorean theorem. Both variants can be implemented with favorable communication properties.

The paper is organized as follows. In the next section, we look at the loss of orthogonality of BCGS and prove a block Pythagorean theorem for deriving BCGS-PIP and BCGS-PIO. We prove that these P-variants have $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}})$ loss of orthogonality and $\mathcal{O}(\varepsilon)\,\|\boldsymbol{\mathcal{X}}\|$ residual in Section 3. We demonstrate these bounds with a variety of numerical examples in Section 4 and give conclusions in Section 5.

Unless otherwise noted, $\|\cdot\|$ denotes the Euclidean norm and $\varepsilon$ is a given machine precision. For simplicity, throughout the analysis we use $\mathcal{O}(\varepsilon)$ to denote $\varepsilon$ multiplied by a low degree polynomial in the problem dimensions $m$ and $n$. The $i$th singular value

of a matrix $A$ is denoted as $\sigma_i(A)$, with $\sigma_{\min}(A)$ denoting the smallest. The condition number $\kappa(A) := \frac{\|A\|}{\sigma_{\min}(A)}$, which is equivalent to $\|A\| \left\|A^{-1}\right\|$, when $A$ is square and invertible. We use composition notation $\circ$ to denote a specific implementation of `BCGS`, e.g., `BCGS` $\circ$ `HouseQR` is `BCGS` with Householder QR as the `IntraOrtho`.

**2. BCGS and its variants.** Algorithm 2.1 is a typical implementation of `BCGS`, which takes $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{m\times n}$ and returns a matrix $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{m\times n}$ with orthonormal columns and square upper triangular $\mathcal{R} \in \mathbb{R}^{n\times n}$ such that $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{Q}}\mathcal{R}$ in exact arithmetic. We denote entries of block matrices $\mathcal{R}$ by their non-calligraphic counterparts $R_{jk} \in \mathbb{R}^{s\times s}$. When we index $\boldsymbol{\mathcal{Q}}$ or $\mathcal{R}$, we are always referring to the corresponding contiguous block component, be it an $m \times s$ block vector or $s \times s$ block entry, respectively. More clearly, we define

$$\boldsymbol{\mathcal{Q}}_{1:j} := [\boldsymbol{Q}_1 \mid \cdots \mid \boldsymbol{Q}_j] \in \mathbb{R}^{m\times sj},$$

where $1 : j \equiv \{1, 2, \ldots, j\}$ is an indexing vector in the style of MATLAB, and similarly,

$$\mathcal{R}_{1:j,k} := \begin{bmatrix} R_{1,k} \\ R_{2,k} \\ \vdots \\ R_{j,k} \end{bmatrix} \in \mathbb{R}^{sj\times s}.$$

The intra-block orthogonalization routine `IntraOrtho` takes a block vector $\boldsymbol{X} \in \mathbb{R}^{m\times s}$ and returns a matrix $\boldsymbol{Q} \in \mathbb{R}^{m\times s}$ with orthonormal columns and upper triangular $R \in \mathbb{R}^{s\times s}$ such that $\boldsymbol{X} = \boldsymbol{Q}R$.

**Algorithm 2.1** $[\boldsymbol{\mathcal{Q}}, \mathcal{R}] =$ `BCGS`$(\boldsymbol{\mathcal{X}})$

1: $[\boldsymbol{Q}_1, R_{11}] =$ `IntraOrtho` $(\boldsymbol{X}_1)$
2: **for** $k = 1, \ldots, p-1$ **do**
3: $\quad \mathcal{R}_{1:k,k+1} = \boldsymbol{\mathcal{Q}}_{1:k}^T \boldsymbol{X}_{k+1}$
4: $\quad \boldsymbol{W}_{k+1} = \boldsymbol{X}_{k+1} - \boldsymbol{\mathcal{Q}}_{1:k}\mathcal{R}_{1:k,k+1}$
5: $\quad [\boldsymbol{Q}_{k+1}, R_{k+1,k+1}] =$ `IntraOrtho` $(\boldsymbol{W}_{k+1})$
6: **end for**

`BCGS` has one synchronization point (line 1) for the first block column, and two (lines 3 and 5) for successive block columns. Unlike block modified Gram-Schmidt (see, e.g., [14]), `BCGS` has a constant number of synchronization points per step, making it appealing for communication-avoiding methods [13].

Unfortunately, `BCGS` can suffer from severe loss of orthogonality, much like `CGS`, when $\kappa(\boldsymbol{\mathcal{X}})$ is large. In Figure 2.1, we plot loss of orthogonality and relative Cholesky residual for `BCGS` $\circ$ `HouseQR` against a sequence of matrices $\boldsymbol{\mathcal{X}}_t = \boldsymbol{\mathcal{U}}\Sigma_t\mathcal{V}^T \in \mathbb{R}^{m\times ps}$, where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{m\times ps}$ has orthonormal columns, $\Sigma_t \in \mathbb{R}^{ps\times ps}$ is a diagonal matrix whose entries are in the logarithmic interval $10^{[-t,0]}$, and $\mathcal{V} \in \mathbb{R}^{ps\times ps}$ is unitary. In this example, $m = 100$, $p = 20$, and $s = 2$. We refer to such plots as "standard $\kappa$-plots." Figure 2.2 is similar, we but instead consider `glued` $\kappa$-plots, where the matrices are built as the "glued" matrices from [18]; we choose $m = 1000$, $p = 20$, and $s = 2$. All figures are generated with the BlockStab MATLAB package in double precision; for more implementation details, see Section 4.

In Figure 2.1, the loss of orthogonality of `BCGS` is nearly quadratic in $\kappa(\boldsymbol{\mathcal{X}})$, but the relative Cholesky residual reveals a few orders of magnitude difference from $\mathcal{O}(\varepsilon)$, indicating that something is amiss. The `glued` $\kappa$-plots in Figure 2.2 provide a

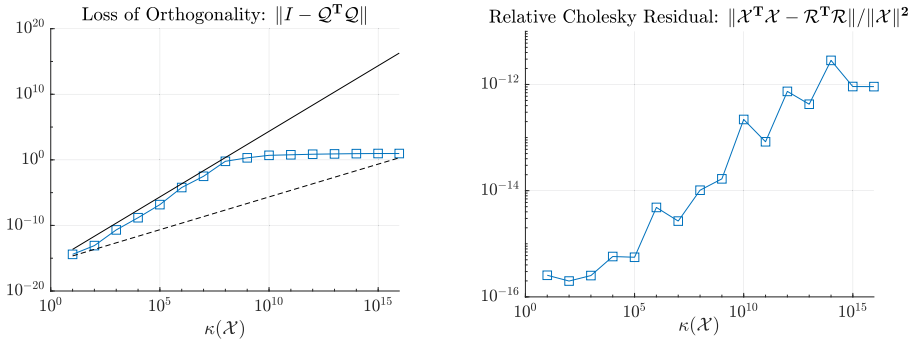


FIG. 2.1. *Standard $\kappa$-plots for* `BCGS` $\circ$ `HouseQR`*, showing loss of orthogonality (left) and the relative Cholesky residual (right). In the left plot, the dashed line marks $\mathcal{O}(\varepsilon)\,\kappa(\boldsymbol{\mathcal{X}})$ and the solid line marks $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}})$.*

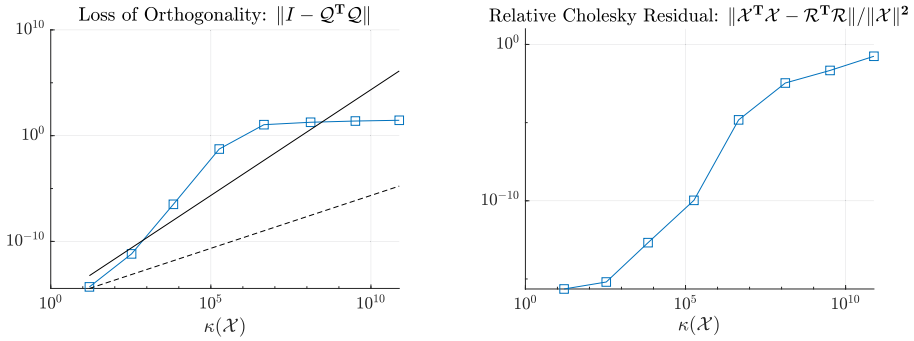


FIG. 2.2. `Glued` *$\kappa$-plots for* `BCGS` $\circ$ `HouseQR`*, showing loss of orthogonality (left) and the relative Cholesky residual (right). In the left plot, the dashed line marks $\mathcal{O}(\varepsilon)\,\kappa(\boldsymbol{\mathcal{X}})$ and the solid line marks $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}})$.*

solid counterexample: not only does the loss of orthogonality noticeably deviate from $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}})$, but also the relative Cholesky residual approaches $O(1)$.

**2.1. A block generalization of `CGS-P`.** To achieve better numerical behavior, we need to improve how `BCGS` computes the block diagonal entries of $\mathcal{R}$, which will improve the size of the Cholesky residual. Taking inspiration from [18], we can achieve this through a generalization of the Pythagorean theorem.

The well-known Pythagorean theorem for vectors says that if for $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^m$, $\boldsymbol{u} = \boldsymbol{v} + \boldsymbol{w}$ and $\boldsymbol{v} \perp \boldsymbol{w}$, then

$$\|\boldsymbol{u}\|^2 = \|\boldsymbol{v}\|^2 + \|\boldsymbol{w}\|^2 .$$

When dealing with block vectors, $\|\cdot\|$ is replaced by the R factor from the QR factorization of a block vector. See [17] for a more concrete sense of the roles that QR and Cholesky factorizations play as "block norms". Suppose we have $\boldsymbol{U} = \boldsymbol{V} + \boldsymbol{W}$, where $\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W} \in \mathbb{R}^{m\times s}$ and $\boldsymbol{V} \perp \boldsymbol{W}$, in the sense that the spaces spanned by the columns of each block vector are perpendicular to each other. Suppose also we have the following QR factorizations for each block vector:

$$\boldsymbol{U} = \boldsymbol{Q_U} R_{\boldsymbol{U}}, \quad \boldsymbol{V} = \boldsymbol{Q_V} R_{\boldsymbol{V}}, \text{ and } \boldsymbol{W} = \boldsymbol{Q_W} R_{\boldsymbol{W}}.$$

Then

$$R_{\boldsymbol{U}}^T R_{\boldsymbol{U}} = \boldsymbol{U}^T \boldsymbol{U} \tag{2.1}$$
$$= \boldsymbol{V}^T \boldsymbol{V} + \boldsymbol{W}^T \boldsymbol{W} \tag{2.2}$$
$$= R_{\boldsymbol{V}}^T R_{\boldsymbol{V}} + R_{\boldsymbol{W}}^T R_{\boldsymbol{W}}, \tag{2.3}$$

implying

$$R_{\boldsymbol{U}} = \texttt{chol}(\boldsymbol{V}^T \boldsymbol{V} + \boldsymbol{W}^T \boldsymbol{W}) = \texttt{chol}(R_{\boldsymbol{V}}^T R_{\boldsymbol{V}} + R_{\boldsymbol{W}}^T R_{\boldsymbol{W}}), \tag{2.4}$$

where `chol` denotes an implementation of Cholesky factorization.

We have thus two alternatives for a block version of `CGS-P`, via either (2.2) or (2.3). Both allow for a way to compute $R_{k+1,k+1}$ in Algorithm 2.1. From lines 4 and 5, we have

$$\begin{aligned} \boldsymbol{W}_{k+1} &= \boldsymbol{X}_{k+1} - \boldsymbol{\mathcal{Q}}_{1:k} \mathcal{R}_{1:k,k+1} \\ [\boldsymbol{Q}_{k+1}, R_{k+1,k+1}] &= \texttt{IntraOrtho}\,(\boldsymbol{W}_{k+1})\,. \end{aligned} \tag{2.5}$$

We can rewrite (2.5) as

$$\boldsymbol{X}_{k+1} = \boldsymbol{\mathcal{Q}}_{1:k} \mathcal{R}_{1:k,k+1} + \boldsymbol{W}_{k+1},$$

and let $\mathcal{R}_{1:k,k+1} = \boldsymbol{Q}_{\mathcal{R}} P_{k+1}$ be the QR factorization of $\mathcal{R}_{1:k,k+1}$, where $\boldsymbol{Q}_{\mathcal{R}} \in \mathbb{R}^{sk \times s}$ and $P_{k+1} \in \mathbb{R}^{s \times s}$. Similarly, we let $\boldsymbol{X}_{k+1} = \boldsymbol{Q}_{\boldsymbol{X}} T_{k+1}$ be the QR factorization of $\boldsymbol{X}_{k+1}$, where $\boldsymbol{Q}_{\boldsymbol{X}} \in \mathbb{R}^{m \times s}$ and $T_{k+1} \in \mathbb{R}^{s \times s}$.

From (2.1)-(2.3), the block Pythagorean theorem gives

$$\begin{aligned} T_{k+1}^T T_{k+1} &= \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} \\ &= \boldsymbol{W}_{k+1}^T \boldsymbol{W}_{k+1} + \mathcal{R}_{1:k,k+1}^T \mathcal{R}_{1:k,k+1} && (2.6) \\ &= R_{k+1,k+1}^T R_{k+1,k+1} + P_{k+1}^T P_{k+1}. && (2.7) \end{aligned}$$

Rearranging (2.7), we have

$$R_{k+1,k+1}^T R_{k+1,k+1} = T_{k+1}^T T_{k+1} - P_{k+1}^T P_{k+1},$$

and similar to (2.4), we have

$$R_{k+1,k+1} = \texttt{chol}(T_{k+1}^T T_{k+1} - P_{k+1}^T P_{k+1}),$$

noting that $\boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1}$ is symmetric positive definite since $\boldsymbol{\mathcal{X}}$ has full column rank. The resulting algorithms corresponding to (2.6) and (2.7) are given as Algorithms 2.2 and 2.3, respectively. The acronyms "PIP" and "PIO" denote "Pythagorean Inner Product" and "Pythagorean Intra-Orthogonalization", respectively.

In terms of communication costs, we note that `BCGS` has one `IntraOrtho` at the outset, plus one block inner product and one `IntraOrtho` per iteration. `BCGS-PIP` has only one `IntraOrtho`, plus two block inner products per iteration, the second of which (i.e., $\mathcal{R}_{1:k,k+1}^T \mathcal{R}_{1:k,k+1}$) grows in size as $ks$ approaches $m$. The Cholesky factorization (line 5) and upper triangular inverse (line 7) can be done locally, since they operate on small $s \times s$ matrices. `BCGS-PIO` has as many calls to `IntraOrtho` as `BCGS`– assuming that the `IntraOrtho` is implemented in a "smart" way, so that it decouples the block diagonal matrix in line 5– and the same number of block inner products as `BCGS`. The Cholesky factorization and upper triangular solve can again be executed locally.

**Algorithm 2.2** $[\mathcal{Q}, \mathcal{R}] = \texttt{BCGS-PIP}(\boldsymbol{\mathcal{X}})$

1: $[\boldsymbol{Q}_1, R_{11}] = \texttt{IntraOrtho}\,(\boldsymbol{X}_1)$
2: **for** $k = 1, \ldots, p-1$ **do**
3: $\quad \begin{bmatrix} \mathcal{R}_{1:k,k+1} \\ \mathcal{Z}_{k+1} \end{bmatrix} = [\boldsymbol{\mathcal{Q}}_{1:k}\ \boldsymbol{X}_{k+1}]^T \boldsymbol{X}_{k+1}$
4: $\quad R_{k+1,k+1} = \texttt{chol}(\mathcal{Z}_{k+1} - \mathcal{R}_{1:k,k+1}^T \mathcal{R}_{1:k,k+1})$
5: $\quad \boldsymbol{W}_{k+1} = \boldsymbol{X}_{k+1} - \boldsymbol{\mathcal{Q}}_{1:k} \mathcal{R}_{1:k,k+1}$
6: $\quad \boldsymbol{Q}_{k+1} = \boldsymbol{W}_{k+1} R_{k+1,k+1}^{-1}$
7: **end for**

**Algorithm 2.3** $[\mathcal{Q}, \mathcal{R}] = \texttt{BCGS-PIO}(\boldsymbol{\mathcal{X}})$

1: $[\boldsymbol{Q}_1, R_{11}] = \texttt{IntraOrtho}\,(\boldsymbol{X}_1)$
2: **for** $k = 1, \ldots, p-1$ **do**
3: $\quad \mathcal{R}_{1:k,k+1} = \boldsymbol{\mathcal{Q}}_{1:k}^T \boldsymbol{X}_{k+1}$
4: $\quad \left[\sim, \begin{bmatrix} T_{k+1} & \\ & P_{k+1} \end{bmatrix}\right] = \texttt{IntraOrtho}\left(\begin{bmatrix} \boldsymbol{X}_{k+1} & \mathcal{R}_{1:k,k+1} \end{bmatrix}\right)$
5: $\quad R_{k+1,k+1} = \texttt{chol}(T_{k+1}^T T_{k+1} - P_{k+1}^T P_{k+1})$
6: $\quad \boldsymbol{W}_{k+1} = \boldsymbol{X}_{k+1} - \boldsymbol{\mathcal{Q}}_{1:k} \mathcal{R}_{1:k,k+1}$
7: $\quad \boldsymbol{Q}_{k+1} = \boldsymbol{W}_{k+1} R_{k+1,k+1}^{-1}$
8: **end for**

With regards to floating point operations, both `BCGS-PIO` and `BCGS-PIP` are slightly more expensive than `BCGS`. `BCGS-PIP` may have an advantage over the other two if they are using an expensive `IntraOrtho`. In fact, `BCGS-PIP` was developed independently by Yamazaki et al. as a low-synchronization alternative to $\texttt{BCGS} \circ \texttt{CholQR}$ and has favorable performance [21].

**3. Loss of orthogonality and residual bounds.** The primary goal of this section is to prove bounds on the loss of orthogonality and the residual in `BCGS-PIP` and `BCGS-PIO`. In particular, we prove that if certain constraints on the `IntraOrtho` routine are satisfied, the loss of orthogonality in `BCGS-PIP` and `BCGS-PIO` depends quadratically on $\kappa(\boldsymbol{\mathcal{X}})$ as long as

$$\mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}) < 1/2. \tag{3.1}$$

The final bounds are summarized in Theorem 3.4.

Throughout this section we will make use of standard error results (see, e.g., [12], especially Sections 2.2 and 3.5 and Lemma 6.6) for matrix addition and matrix multiplication, which we state here for completeness. Here and in the remainder of this section, a bar over a quantity means that it is the result of a finite precision computation. For the computation $C = A + B$, with $A, B \in \mathbb{R}^{j \times k}$,

$$\left\| C - \bar{C} \right\| \leq \varepsilon \cdot \min\left(\sqrt{j}, \sqrt{k}\right)\left( \|A\| + \|B\| \right) \leq \mathcal{O}(\varepsilon) \left( \|A\| + \|B\| \right). \tag{3.2}$$

For the computation of the matrix product $C = AB \in \mathbb{R}^{j \times \ell}$, with $A \in \mathbb{R}^{j \times k}$, $B \in \mathbb{R}^{k \times \ell}$,

$$\left\| C - \bar{C} \right\| \leq \varepsilon \cdot 2k \min\left(\sqrt{j}, \sqrt{\ell}\right) \|A\| \, \|B\| \leq \mathcal{O}(\varepsilon) \, \|A\| \, \|B\| \,. \tag{3.3}$$

For simplicity, we make use of a single subroutine, given as Algorithm 3.1, which applies to both `BCGS-PIP` and `BCGS-PIO` variants. The floating point error in Algorithm 3.1 is equivalent to lines 4-7 of Algorithms 2.2 and lines 4-8 of 2.3. We assume

that the `IntraOrtho` handles block diagonal matrices by ignoring the zero off-diagonal blocks and acting on each diagonal block separately.

**Algorithm 3.1** $[\boldsymbol{Q}_{k+1}, R_{k+1,k+1}, \mathcal{R}_{1:k,k+1}] = \texttt{BCGS-P\_step}(\boldsymbol{\mathcal{Q}}_{1:k}, \boldsymbol{X}_{k+1})$

$\mathcal{R}_{1:k,k+1} = \boldsymbol{\mathcal{Q}}_{1:k}^T \boldsymbol{X}_{k+1}$
**if** `BCGS-PIP` **then**
  $C_{k+1} = \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} - \mathcal{R}_{1:k,k+1}^T \mathcal{R}_{1:k,k+1}$
**else if** `BCGS-PIO` **then**
  $\left[\sim, \begin{bmatrix} T_{k+1} & \\ & P_{k+1} \end{bmatrix}\right] = \texttt{IntraOrtho}\left(\begin{bmatrix} \boldsymbol{X}_{k+1} & \mathcal{R}_{1:k,k+1} \end{bmatrix}\right)$
  $C_{k+1} = T_{k+1}^T T_{k+1} - P_{k+1}^T P_{k+1}$
**end if**
$R_{k+1,k+1} = \texttt{chol}(C_{k+1})$
$\boldsymbol{W}_{k+1} = \boldsymbol{X}_{k+1} - \boldsymbol{\mathcal{Q}}_{1:k} \mathcal{R}_{1:k,k+1}$
$\boldsymbol{Q}_{k+1} = \boldsymbol{W}_{k+1} R_{k+1,k+1}^{-1}$
**return** $\boldsymbol{Q}_{k+1}$, $R_{k+1,k+1}$, $\mathcal{R}_{1:k,k+1}$

We first prove a theorem that says that given particular residual bounds, we have a bound on the loss of orthogonality. From now on, to simplify notation we will denote $R_{k+1} := R_{k+1,k+1}$, $\mathcal{R}_{1:k,1:k} := \mathcal{R}_{1:k}$, and similarly for other quantities.

THEOREM 3.1. *Let $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{m \times n}$ be a matrix with block structure given in (1.1) that satisfies (3.1). If*

$$\bar{\mathcal{R}}_{1:k}^T \bar{\mathcal{R}}_{1:k} = \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{\mathcal{X}}_{1:k} + \Delta\mathcal{E}_{1:k}, \quad \|\Delta\mathcal{E}_{1:k}\| \le \mathcal{O}(\varepsilon) \|\boldsymbol{\mathcal{X}}_{1:k}\|^2, \quad \textit{and} \tag{3.4}$$

$$\bar{\boldsymbol{\mathcal{Q}}}_{1:k} \bar{\mathcal{R}}_{1:k} = \boldsymbol{\mathcal{X}}_{1:k} + \Delta\boldsymbol{\mathcal{D}}_{1:k}, \quad \|\Delta\boldsymbol{\mathcal{D}}_{1:k}\| \le \mathcal{O}(\varepsilon) \left( \|\boldsymbol{\mathcal{X}}_{1:k}\| + \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}\right\| \right), \tag{3.5}$$

*then*

$$\left\| I - \bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T \bar{\boldsymbol{\mathcal{Q}}}_{1:k} \right\| \le \frac{\mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}{1 - \mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}, \tag{3.6}$$

$$\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \le \frac{1 + \mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}{1 - \mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k})} \le 3, \quad \textit{and} \tag{3.7}$$

$$\|\Delta\boldsymbol{\mathcal{D}}_{1:k}\| \le \mathcal{O}(\varepsilon) \|\boldsymbol{\mathcal{X}}_{1:k}\|. \tag{3.8}$$

*Proof.* We first note that from (3.4), we have

$$\left\|\bar{\mathcal{R}}_{1:k}\right\|^2 \le (1 + \mathcal{O}(\varepsilon)) \|\boldsymbol{\mathcal{X}}_{1:k}\|^2, \tag{3.9}$$

and a Weyl bound on $\left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2$ follows directly from (3.4) and (3.1):

$$\left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 = \frac{1}{\sigma_{\min}^2(\bar{\mathcal{R}}_{1:k})} \le \frac{1}{\sigma_{\min}^2(\boldsymbol{\mathcal{X}}_{1:k})(1 - \mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k}))}. \tag{3.10}$$

Taking (3.5) and multiplying it by its transpose, we have

$$\bar{\mathcal{R}}_{1:k}^T \bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T \bar{\boldsymbol{\mathcal{Q}}}_{1:k} \bar{\mathcal{R}}_{1:k} = \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{\mathcal{X}}_{1:k} + \boldsymbol{\mathcal{X}}_{1:k}^T \Delta\boldsymbol{\mathcal{D}}_{1:k} + \Delta\boldsymbol{\mathcal{D}}_{1:k}^T \boldsymbol{\mathcal{X}}_{1:k} + \Delta\boldsymbol{\mathcal{D}}_{1:k}^T \Delta\boldsymbol{\mathcal{D}}_{1:k},$$

and then multiplying on the left by $\bar{\mathcal{R}}_{1:k}^{-T}$ and on the right by $\bar{\mathcal{R}}_{1:k}^{-1}$ and using (3.4), we obtain

$$\begin{aligned} \bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T \bar{\boldsymbol{\mathcal{Q}}}_{1:k} = I &- \bar{\mathcal{R}}_{1:k}^{-T} \Delta\mathcal{E}_{1:k} \bar{\mathcal{R}}_{1:k}^{-1} + \bar{\mathcal{R}}_{1:k}^{-T} \boldsymbol{\mathcal{X}}_{1:k}^T \Delta\boldsymbol{\mathcal{D}}_{1:k} \bar{\mathcal{R}}_{1:k}^{-1} \\ &+ \bar{\mathcal{R}}_{1:k}^{-T} \Delta\boldsymbol{\mathcal{D}}_{1:k}^T \boldsymbol{\mathcal{X}}_{1:k} \bar{\mathcal{R}}_{1:k}^{-1} + \bar{\mathcal{R}}_{1:k}^{-T} \Delta\boldsymbol{\mathcal{D}}_{1:k}^T \Delta\boldsymbol{\mathcal{D}}_{1:k} \bar{\mathcal{R}}_{1:k}^{-1}. \end{aligned} \tag{3.11}$$

Ignoring terms of order $\mathcal{O}(\varepsilon^2)$ and using (3.4), (3.5), (3.9), and (3.10), this gives the bound

$$\begin{aligned}\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\|^2 &\leq 1 + \left\|\Delta\mathcal{E}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 + 2\left\|\Delta\boldsymbol{\mathcal{D}}_{1:k}\right\| \left\|\boldsymbol{\mathcal{X}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 \\ &\leq 1 + \mathcal{O}(\varepsilon)\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\|^2 \left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 + \mathcal{O}(\varepsilon)\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}\right\| \\ &\leq \frac{1}{1-\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})} + \frac{\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}{1-\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\|.\end{aligned}$$

Thus altogether we have the quadratic inequality

$$\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\|^2 - \frac{\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}{1-\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| - \frac{1}{1-\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})} \leq 0,$$

which gives

$$\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \leq \frac{1+\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}{1-\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})} \leq \frac{3/2}{1/2} = 3, \tag{3.12}$$

which proves (3.7).

From (3.11), following a similar process, we obtain

$$\left\|I - \bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \leq \mathcal{O}(\varepsilon)\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\|^2 \left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 + \mathcal{O}(\varepsilon)\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}^{-1}\right\|^2 \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}\right\|,$$

and using (3.9), (3.10), and (3.12), we have

$$\left\|I - \bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \leq \frac{\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})}{1-\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}_{1:k})},$$

which proves (3.6).

Finally, combining (3.5) with (3.9) and (3.12),

$$\begin{aligned}\left\|\Delta\boldsymbol{\mathcal{D}}_{1:k}\right\| &\leq \mathcal{O}(\varepsilon)\left(\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\| + \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \left\|\bar{\mathcal{R}}_{1:k}\right\|\right) \\ &\leq \mathcal{O}(\varepsilon)\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\|,\end{aligned}$$

which completes the proof. ∎

Now that Theorem 3.1 is established, it only remains to prove that these residual bounds do indeed hold. For this, we will use an inductive approach on the block vectors of $\boldsymbol{\mathcal{X}}$.

THEOREM 3.2. *Let $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{m\times n}$ be a matrix with block structure given in (1.1) that satisfies (3.1). Further, assume that for all $\boldsymbol{X}$, the following hold for $[\bar{\boldsymbol{Q}}, \bar{R}] =$ `IntraOrtho`$(\boldsymbol{X})$:*

$$\bar{R}^T\bar{R} = \boldsymbol{X}^T\boldsymbol{X} + \Delta E, \quad \|\Delta E\| \leq \mathcal{O}(\varepsilon)\left\|\boldsymbol{X}\right\|^2, \tag{3.13}$$

*and*

$$\bar{\boldsymbol{Q}}\bar{R} = \boldsymbol{X} + \Delta\boldsymbol{D}, \quad \|\Delta\boldsymbol{D}\| \leq \mathcal{O}(\varepsilon)\left(\left\|\boldsymbol{X}\right\| + \left\|\bar{\boldsymbol{Q}}\right\|\left\|\bar{R}\right\|\right). \tag{3.14}$$

*For `BCGS-PIO`, we furthermore require that (3.13) in `IntraOrtho`$(\boldsymbol{X})$ holds for all $\boldsymbol{X}$ regardless of condition number. Then the following hold for $[\bar{\boldsymbol{\mathcal{Q}}}, \bar{\mathcal{R}}] =$ `BCGS-P` $\circ$ `IntraOrtho`$(\boldsymbol{\mathcal{X}})$, where `BCGS-P` is either `BCGS-PIO` or `BCGS-PIP`:*

$$\bar{\mathcal{R}}_{1:k}^T\bar{\mathcal{R}}_{1:k,1:k} = \boldsymbol{\mathcal{X}}_{1:k}^T\boldsymbol{\mathcal{X}}_{1:k} + \Delta\mathcal{E}_{1:k}, \quad \|\Delta\mathcal{E}_{1:k}\| \leq \mathcal{O}(\varepsilon)\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\|^2, \quad \textit{and} \tag{3.15}$$

$$\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\bar{\mathcal{R}}_{1:k} = \boldsymbol{\mathcal{X}}_{1:k} + \Delta\boldsymbol{\mathcal{D}}_{1:k}, \quad \|\Delta\boldsymbol{\mathcal{D}}_{1:k}\| \leq \mathcal{O}(\varepsilon)\left(\left\|\boldsymbol{\mathcal{X}}_{1:k}\right\| + \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\|\left\|\bar{\mathcal{R}}_{1:k}\right\|\right), \tag{3.16}$$

*for all $k = 1, \ldots, p$.*

*Proof.* For the base case $k = 1$, we just run `IntraOrtho` on $\boldsymbol{\mathcal{X}}_1$, and thus (3.15) and (3.16) clearly follow from (3.13) and (3.14).

Now assume that (3.15) and (3.16) hold for some $k \geq 1$. We note that this means that (3.6)-(3.8) from Theorem 3.1 hold for $k$. In particular, we have $\left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \leq 3$, and thus we will absorb this quantity into $\mathcal{O}(\varepsilon)$ terms when applicable. We will now show that (3.15) and (3.16) also hold for $k+1$. We can write $\bar{\mathcal{R}}_{1:k+1}$ as

$$\bar{\mathcal{R}}_{1:k+1} = \begin{bmatrix} \underbrace{\bar{\mathcal{R}}_{1:k}}_{ks\times ks} & \underbrace{\bar{\mathcal{R}}_{1:k,k+1}}_{ks\times s} \\ \underbrace{0}_{s\times ks} & \underbrace{\bar{R}_{k+1}}_{s\times s} \end{bmatrix}.$$

We focus first on (3.15) and look at $\bar{\mathcal{R}}_{1:k+1}^T \bar{\mathcal{R}}_{1:k+1}$ block by block:

$$\bar{\mathcal{R}}_{1:k+1}^T \bar{\mathcal{R}}_{1:k+1} = \begin{bmatrix} \bar{\mathcal{R}}_{1:k}^T \bar{\mathcal{R}}_{1:k} & \bar{\mathcal{R}}_{1:k}^T \bar{\mathcal{R}}_{1:k,k+1} \\ \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k} & \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k,k+1} + \bar{R}_{k+1}^T \bar{R}_{k+1} \end{bmatrix}, \tag{3.17}$$

where $\bar{\mathcal{R}}_{1:k,k+1}$ denotes the computed matrix in line 1 of Algorithm 3.1. For this computed matrix, using (3.3) and eq:Qbound, we can write

$$\bar{\mathcal{R}}_{1:k,k+1} = \bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T \boldsymbol{X}_{k+1} + \Delta\mathcal{R}_{1:k,k+1}, \quad \left\|\Delta\mathcal{R}_{1:k,k+1}\right\| \leq \mathcal{O}(\varepsilon) \left\|\boldsymbol{X}_{k+1}\right\|. \tag{3.18}$$

Note that this also gives

$$\left\|\bar{\mathcal{R}}_{1:k,k+1}\right\| \leq \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\right\| \left\|\boldsymbol{X}_{k+1}\right\| + \left\|\Delta\mathcal{R}_{1:k,k+1}\right\| \leq (3 + \mathcal{O}(\varepsilon)) \left\|\boldsymbol{X}_{k+1}\right\|. \tag{3.19}$$

For the off-diagonal entries in (3.17), using (3.18) and (3.16), we have

$$\begin{aligned} \bar{\mathcal{R}}_{1:k}^T \bar{\mathcal{R}}_{1:k,k+1} &= \bar{\mathcal{R}}_{1:k}^T (\bar{\boldsymbol{\mathcal{Q}}}_{1:k}^T \boldsymbol{X}_{k+1} + \Delta\mathcal{R}_{1:k,k+1}) \\ &= (\bar{\boldsymbol{\mathcal{Q}}}_{1:k} \bar{\mathcal{R}}_{1:k})^T \boldsymbol{X}_{k+1} + \bar{\mathcal{R}}_{1:k}^T \Delta\mathcal{R}_{1:k,k+1} \\ &= (\boldsymbol{\mathcal{X}}_{1:k} + \Delta\boldsymbol{\mathcal{D}}_{1:k})^T \boldsymbol{X}_{k+1} + \bar{\mathcal{R}}_{1:k}^T \Delta\mathcal{R}_{1:k,k+1} \\ &= \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{X}_{k+1} + \underbrace{\Delta\boldsymbol{\mathcal{D}}_{1:k}^T \boldsymbol{X}_{k+1} + \bar{\mathcal{R}}_{1:k}^T \Delta\mathcal{R}_{1:k,k+1}}_{=:\Delta\mathcal{E}_{1:k,k+1}}, \end{aligned} \tag{3.20}$$

with

$$\begin{aligned} \left\|\Delta\mathcal{E}_{1:k,k+1}\right\| &\leq \left\|\Delta\boldsymbol{\mathcal{D}}_{1:k}\right\| \left\|\boldsymbol{X}_{k+1}\right\| + \left\|\bar{\mathcal{R}}_{1:k}\right\| \left\|\Delta\mathcal{R}_{1:k,k+1}\right\| \\ &\leq \mathcal{O}(\varepsilon) \left\|\boldsymbol{\mathcal{X}}_{1:k}\right\| \left\|\boldsymbol{X}_{k+1}\right\|, \end{aligned} \tag{3.21}$$

where we have used (3.18), (3.8), and

$$\left\|\bar{\mathcal{R}}_{1:k}\right\|^2 \leq (1 + \mathcal{O}(\varepsilon)) \left\|\boldsymbol{\mathcal{X}}_{1:k}\right\|^2,$$

which follows from (3.15).

The upper diagonal block of (3.17) can be handled directly with the induction hypothesis (3.15). We now focus on the bottom diagonal entry of (3.17). We first show that the computed arguments of `chol` in line 4 of Algorithm 2.2 and line 5 of Algorithm 2.3 are symmetric positive definitive and numerically nonsingular. We denote both arguments as $C_{k+1}$ and their computed versions as $\bar{C}_{k+1}$.

We begin with BCGS-PIP. Applying (3.3) to the computation of the products $\bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k,k+1}$ and $\boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1}$ each, and (3.2) to the computation of their difference, gives

$$\|\Delta C_{k+1}\| \leq \mathcal{O}(\varepsilon) \|\boldsymbol{X}_{k+1}\|^2 .$$

For BCGS-PIO, we first apply IntraOrtho to $\bar{\mathcal{R}}_{1:k,k+1}$ and $\boldsymbol{X}_{k+1}$ each to obtain Cholesky residual bounds for both:

$$\bar{T}_{k+1}^T \bar{T}_{k+1} = \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} + \Delta T_{k+1}, \quad \|\Delta T_{k+1}\| \leq \mathcal{O}(\varepsilon) \|\boldsymbol{X}_{k+1}\|^2 , \tag{3.22}$$

$$\bar{P}_{k+1}^T \bar{P}_{k+1} = \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k,k+1} + \Delta P_{k+1}, \quad \|\Delta P_{k+1}\| \leq \mathcal{O}(\varepsilon) \left\|\bar{\mathcal{R}}_{1:k,k+1}\right\|^2 . \tag{3.23}$$

The bounds from (3.22) and (3.23) give a relationship for the exact products $\bar{T}_{k+1}^T \bar{T}_{k+1}$ and $\bar{P}_{k+1}^T \bar{P}_{k+1}$. We must still apply (3.2)-(3.3) to the floating-point computation of the products and of their difference, which gives

$$\|\Delta C_{k+1}\| \leq \mathcal{O}(\varepsilon) \|\boldsymbol{X}_{k+1}\|^2 .$$

Thus for both BCGS-PIP and BCGS-PIO, we can write

$$\bar{C}_{k+1} = \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} - \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k,k+1} + \Delta C_{k+1}, \tag{3.24}$$

$$\|\Delta C_{k+1}\| \leq \mathcal{O}(\varepsilon) \|\boldsymbol{X}_{k+1}\|^2 . \tag{3.25}$$

It follows from (3.15), (3.20)-(3.21), and (3.24)-(3.25) that

$$\begin{aligned}
\boldsymbol{\mathcal{X}}_{1:k+1}^T \boldsymbol{\mathcal{X}}_{1:k+1} &= \begin{bmatrix} \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{\mathcal{X}}_{1:k} & \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{X}_{k+1} \\ \boldsymbol{X}_{k+1}^T \boldsymbol{\mathcal{X}}_{1:k} & \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} \end{bmatrix} \\
&= \underbrace{\begin{bmatrix} \bar{\mathcal{R}}_{1:k}^T \bar{\mathcal{R}}_{1:k} & \bar{\mathcal{R}}_{1:k}^T \bar{\mathcal{R}}_{1:k,k+1} \\ \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k} & \bar{C}_{k+1} + \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k,k+1} \end{bmatrix}}_{=:\mathcal{P}_{k+1}} - \underbrace{\begin{bmatrix} \Delta\mathcal{E}_{1:k} & \Delta\mathcal{E}_{1:k,k+1} \\ \Delta\mathcal{E}_{1:k,k+1}^T & \Delta C_{k+1} \end{bmatrix}}_{=:\Delta\widetilde{\mathcal{E}}_{1:k+1}}
\end{aligned}$$

where[1]

$$\begin{aligned}
\left\|\Delta\widetilde{\mathcal{E}}_{1:k+1}\right\| &\leq \left\| \begin{bmatrix} \|\Delta\mathcal{E}_{1:k}\| & \|\Delta\mathcal{E}_{1:k,k+1}\| \\ \|\Delta\mathcal{E}_{1:k,k+1}\| & \|\Delta C_{k+1}\| \end{bmatrix} \right\| \leq \left\| \begin{bmatrix} \|\Delta\mathcal{E}_{1:k}\| & \|\Delta\mathcal{E}_{1:k,k+1}\| \\ \|\Delta\mathcal{E}_{1:k,k+1}\| & \|\Delta C_{k+1}\| \end{bmatrix} \right\|_{\mathrm{F}} \\
&\leq \|\Delta\mathcal{E}_{1:k}\| + 2\|\Delta\mathcal{E}_{1:k,k+1}\| + \|\Delta C_{k+1}\| \\
&\leq \mathcal{O}(\varepsilon) \|\boldsymbol{\mathcal{X}}_{1:k+1}\|^2 ,
\end{aligned}$$

where the final bound follows from (3.15), (3.21), and (3.25).

Clearly $\mathcal{P}_{k+1}$ is symmetric, and by Weyl's inequality it follows that

$$\lambda_{\min}(\mathcal{P}_{k+1}) \geq \sigma_{\min}^2(\boldsymbol{\mathcal{X}}_{1:k+1})\big(1 - \mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k+1})\big) > 0,$$

where $\lambda_{\min}$ denotes the smallest eigenvalue. Therefore $\mathcal{P}_{k+1}$ is symmetric positive definite and, equivalently, so its Schur complement, $\bar{C}_{k+1}$. More specifically, it follows that $\kappa(\bar{C}_{k+1}) \leq \kappa(\mathcal{P}_{k+1})$; see, e.g., [7, Lemma 4.2]. Since

$$\kappa(\mathcal{P}_{k+1}) \leq \frac{(1 + \mathcal{O}(\varepsilon))\kappa^2(\boldsymbol{\mathcal{X}}_{1:k+1})}{1 - \mathcal{O}(\varepsilon)\, \kappa^2(\boldsymbol{\mathcal{X}}_{1:k+1})},$$

[1] See, e.g., P.15.50 in [9].

it is straightforward to see that $\mathcal{O}(\varepsilon)\,\kappa(\bar{C}_{k+1}) < 1$.

We can therefore apply [12, Theorem 10.3] to the Cholesky factorization in line 8 of Algorithm 3.1, and combining this with (3.24) gives

$$\begin{aligned}\bar{R}_{k+1}^T \bar{R}_{k+1} &= \bar{C}_{k+1} + \Delta F_{k+1}, \quad \|\Delta F_{k+1}\| \leq \mathcal{O}(\varepsilon)\left\|\bar{C}_{k+1}\right\| \\ &= \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} - \bar{\mathcal{R}}_{1:k,k+1}^T \bar{\mathcal{R}}_{1:k,k+1} + \Delta S_{k+1},\end{aligned}$$

where $\Delta S_{k+1} = \Delta C_{k+1} + \Delta F_{k+1}$. From (3.19) and (3.25), we can write

$$\|\Delta F_{k+1}\| \leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{X}_{k+1}\|^2,$$

and then with (3.25) we have

$$\|\Delta S_{k+1}\| \leq \|\Delta C_{k+1}\| + \|\Delta F_{k+1}\| \leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{X}_{k+1}\|^2. \tag{3.26}$$

Then using (3.15), (3.20), and applying (3.26) to the bottom diagonal entry of (3.17),

$$\bar{\mathcal{R}}_{1:k+1}^T \bar{\mathcal{R}}_{1:k+1} = \underbrace{\begin{bmatrix} \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{\mathcal{X}}_{1:k} & \boldsymbol{\mathcal{X}}_{1:k}^T \boldsymbol{X}_{k+1} \\ \boldsymbol{X}_{k+1}^T \boldsymbol{\mathcal{X}}_{1:k} & \boldsymbol{X}_{k+1}^T \boldsymbol{X}_{k+1} \end{bmatrix}}_{=\boldsymbol{\mathcal{X}}_{1:k+1}^T \boldsymbol{\mathcal{X}}_{1:k+1}} + \underbrace{\begin{bmatrix} \Delta\mathcal{E}_{1:k} & \Delta\mathcal{E}_{1:k,k+1} \\ \Delta\mathcal{E}_{1:k,k+1}^T & \Delta S_{k+1} \end{bmatrix}}_{=\Delta\mathcal{E}_{1:k+1}}.$$

Using (3.15), (3.21), and (3.26), we can bound $\|\Delta\mathcal{E}_{1:k+1}\|$ much as we did $\left\|\Delta\widetilde{\mathcal{E}}_{1:k+1}\right\|$:

$$\begin{aligned}\|\Delta\mathcal{E}_{1:k+1}\| &\leq \|\Delta\mathcal{E}_{1:k}\| + 2\,\|\Delta\mathcal{E}_{1:k,k+1}\| + \|\Delta S_{k+1}\| \\ &\leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{\mathcal{X}}_{1:k+1}\|^2.\end{aligned}$$

Therefore, (3.15) holds for $k+1$.

We now turn towards proving that (3.16) holds for $k+1$. Let $\bar{\boldsymbol{W}}_{k+1}$ and $\bar{\boldsymbol{Q}}_{k+1}$ denote the computed factors in lines 9-10 of Algorithm 3.1, respectively. Applying (3.2) and (3.3) to the computation of $\boldsymbol{W}_{k+1}$ in line 9 and using (3.7) and (3.19) leads to

$$\bar{\boldsymbol{W}}_{k+1} = \boldsymbol{X}_{k+1} - \bar{\boldsymbol{\mathcal{Q}}}_{1:k}\bar{\mathcal{R}}_{1:k,k+1} + \Delta\boldsymbol{W}_{k+1}, \quad \|\Delta\boldsymbol{W}_{k+1}\| \leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{X}_{k+1}\|. \tag{3.27}$$

We now look at the triangular system solve in line 10. We can apply [12, Theorem 8.5 & Lemma 6.6] to the inversion of the upper triangular Cholesky factor $\bar{R}_{k+1}$ on rows $\bar{\boldsymbol{W}}_{k+1}(j,:)^T$ of $\bar{\boldsymbol{W}}_{k+1}$ to obtain

$$(\bar{R}_{k+1}^T + \Delta\bar{R}_j^T)\bar{\boldsymbol{Q}}_{k+1}(j,:)^T = \bar{\boldsymbol{W}}_{k+1}(j,:)^T, \quad \left\|\Delta\bar{R}_j\right\| \leq \mathcal{O}(\varepsilon)\left\|\bar{R}_{k+1}\right\|,$$

for $j = 1, \ldots, m$. We can then write

$$\bar{\boldsymbol{Q}}_{k+1}\bar{R}_{k+1} = \bar{\boldsymbol{W}}_{k+1} + \Delta\boldsymbol{G}_{k+1}, \tag{3.28}$$

with $\Delta\boldsymbol{G}_{k+1}(j,:)^T := \Delta\bar{R}_j^T\bar{\boldsymbol{Q}}_{k+1}(j,:)^T$ denoting the rows. Then

$$\|\Delta\boldsymbol{G}_{k+1}\|^2 \leq \|\Delta\boldsymbol{G}_{k+1}\|_{\mathrm{F}}^2 \leq \mathcal{O}(\varepsilon^2)\left\|\bar{R}_{k+1}\right\|^2\left\|\bar{\boldsymbol{Q}}_{k+1}\right\|^2,$$

leading to

$$\|\Delta\boldsymbol{G}_{k+1}\| \leq \mathcal{O}(\varepsilon)\left\|\bar{\boldsymbol{Q}}_{k+1}\right\|\left\|\bar{R}_{k+1}\right\|. \tag{3.29}$$

Then combining (3.27) and (3.28), we have

$$\bar{\boldsymbol{Q}}_{k+1}\bar{R}_{k+1} = \boldsymbol{X}_{k+1} - \bar{\boldsymbol{\mathcal{Q}}}_{1:k}\bar{\mathcal{R}}_{1:k,k+1} + \Delta\boldsymbol{Y}_{k+1},$$

where $\Delta\boldsymbol{Y}_{k+1} = \Delta\boldsymbol{W}_{k+1} + \Delta\boldsymbol{G}_{k+1}$, and using (3.27) and (3.29), we have the bound

$$\|\Delta\boldsymbol{Y}_{k+1}\| \leq \mathcal{O}(\varepsilon)\left(\|\boldsymbol{X}_{k+1}\| + \left\|\bar{\boldsymbol{Q}}_{k+1}\right\|\left\|\bar{R}_{k+1}\right\|\right). \tag{3.30}$$

Now, note that

$$\begin{aligned}\bar{\boldsymbol{\mathcal{Q}}}_{1:k+1}\bar{\mathcal{R}}_{1:k+1} &= \begin{bmatrix}\bar{\boldsymbol{\mathcal{Q}}}_{1:k}\bar{\mathcal{R}}_{1:k} & \bar{\boldsymbol{\mathcal{Q}}}_{1:k}\bar{\mathcal{R}}_{1:k,k+1} + \bar{\boldsymbol{Q}}_{k+1}\bar{R}_{k+1}\end{bmatrix} \\ &= \underbrace{\begin{bmatrix}\boldsymbol{\mathcal{X}}_{1:k} & \boldsymbol{X}_{k+1}\end{bmatrix}}_{=\boldsymbol{\mathcal{X}}_{1:k+1}} + \underbrace{\begin{bmatrix}\Delta\boldsymbol{\mathcal{D}}_{1:k} & \Delta\boldsymbol{Y}_{k+1}\end{bmatrix}}_{=\Delta\boldsymbol{\mathcal{D}}_{1:k+1}},\end{aligned}$$

and using (3.30) together with (3.16),

$$\begin{aligned}\|\Delta\boldsymbol{\mathcal{D}}_{1:k+1}\| &\leq \|\Delta\boldsymbol{\mathcal{D}}_{1:k}\| + \|\Delta\boldsymbol{Y}_{k+1}\| \\ &\leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{\mathcal{X}}_{1:k}\| + \mathcal{O}(\varepsilon)\left(\left\|\bar{\boldsymbol{Q}}_{k+1}\right\|\left\|\bar{R}_{k+1}\right\| + \|\boldsymbol{X}_{k+1}\|\right) \\ &\leq \mathcal{O}(\varepsilon)\left(\|\boldsymbol{\mathcal{X}}_{1:k+1}\| + \left\|\bar{\boldsymbol{\mathcal{Q}}}_{1:k+1}\right\|\left\|\bar{\mathcal{R}}_{1:k+1}\right\|\right).\end{aligned}$$

Therefore (3.16) holds for $k+1$ and this completes the proof. ∎

*Remark* 3.3. A particularly nice feature of this approach is that we do not need to make an explicit assumption on $\kappa(\bar{R}_{k+1})$, as is done in, e.g., [4]. Our assumptions are purely *a priori*, meaning we only restrict properties of the input matrix $\boldsymbol{\mathcal{X}}$, its dimensions, and features of the `IntraOrtho`. We also note that the assumptions on the `IntraOrtho` (3.13) and (3.14) are satisfied by most commonly-used orthogonalization algorithms, including `CGS`, `MGS`, Householder QR, and Cholesky QR. One example of an `IntraOrtho` that does not satisfy these assumptions, namely (3.14), is a Cholesky-based approach that computes $A = \boldsymbol{X}^T\boldsymbol{X}$, obtains the $R$ factor through $\texttt{chol}(A)$, and then computes $Q = (\boldsymbol{X}A^{-1})R^T$.

Combining Theorem 3.2 with Theorem 3.1, it is clear that the bound on the loss of orthogonality (3.6) holds for all $k = 1, \ldots, p$. Note also that from (3.8), we also have a bound on the residual that depends only on the norm of $\boldsymbol{\mathcal{X}}$ in contrast with the bound in (3.16). We state these results in the following summarizing theorem.

THEOREM 3.4. *Let $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{m\times n}$ be a matrix with block structure given in (1.1) that satisfies (3.1). Suppose we execute `BCGS-P` $\circ$ `IntraOrtho`$(\boldsymbol{\mathcal{X}})$ on a machine with unit roundoff $\varepsilon$, where `BCGS-P` is either `BCGS-PIO` or `BCGS-PIP`. If for all $\boldsymbol{X}$, `IntraOrtho`$(\boldsymbol{X})$ computes factors $\bar{\boldsymbol{Q}}$ and $\bar{R}$ that satisfy*

$$\begin{aligned}\bar{R}^T\bar{R} &= \boldsymbol{X}^T\boldsymbol{X} + \Delta E, & &\|\Delta E\| \leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{X}\|^2, \quad \textit{and} \\ \bar{\boldsymbol{Q}}\bar{R} &= \boldsymbol{X} + \Delta\boldsymbol{D}, & &\|\Delta\boldsymbol{D}\| \leq \mathcal{O}(\varepsilon)\left(\|\boldsymbol{X}\| + \left\|\bar{\boldsymbol{Q}}\right\|\left\|\bar{R}\right\|\right),\end{aligned}$$

*then the factors $\bar{\boldsymbol{\mathcal{Q}}}$ and $\bar{\mathcal{R}}$ computed by `BCGS-P` $\circ$ `IntraOrtho`$(\boldsymbol{\mathcal{X}})$ satisfy*

$$\begin{aligned}&\left\|I - \bar{\boldsymbol{\mathcal{Q}}}^T\bar{\boldsymbol{\mathcal{Q}}}\right\| \leq \mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}}), \quad \textit{and} \\ &\bar{\boldsymbol{\mathcal{Q}}}\bar{\mathcal{R}} = \boldsymbol{\mathcal{X}} + \Delta\boldsymbol{\mathcal{D}}, \quad \|\Delta\boldsymbol{\mathcal{D}}\| \leq \mathcal{O}(\varepsilon)\,\|\boldsymbol{\mathcal{X}}\|.\end{aligned}$$

**4. Numerical results.** We examine the behavior of `BCGS`, `BCGS-PIP`, and `BCGS-PIO` for different types of `IntraOrtho`s, focusing especially on the interaction between the loss of orthogonality and the residuals. All methods considered exhibit $\mathcal{O}(\varepsilon)\,\|\boldsymbol{\mathcal{X}}\|$ residual (1.2), so we omit those plots and focus instead on the relative Cholesky residual in tandem with loss of orthogonality.

All figures are generated with the BlockStab MATLAB package[2] using double precision in MATLAB 2020a on a machine running 64-bit Windows 10 Pro with an Intel Core i5-8250U CPU and 16GB of RAM. We also use a hand-written Cholesky code based off [12, Algorithm 10.2], since MATLAB's built-in `chol` throws an error for nearly singular matrices, thus limiting observable behavior.

*Example* 4.1. We first consider the "standard" matrices as described in Section 2. We see in Figure 4.1 that the P-variants achieve $\mathcal{O}(\varepsilon)$ relative Cholesky residual but exhibit instability similar to `BCGS` once $\kappa(\boldsymbol{\mathcal{X}})$ exceeds $10^8$. The behavior of an arbitrary variant is nearly identical regardless of the `IntraOrtho`.

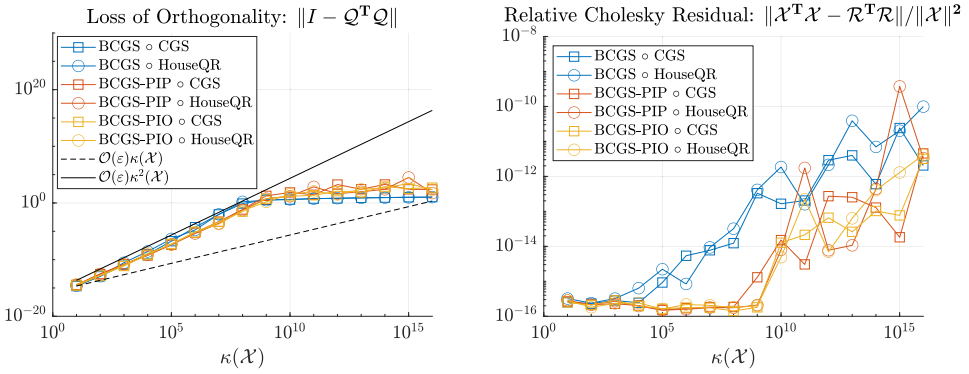


FIG. 4.1. *Standard $\kappa$-plots for different combinations of block variants and* `IntraOrtho`*s.*

*Example* 4.2. Recall that the `glued` matrices were first used by Smoktunowicz et al. [18] to demonstrate how `CGS-P` corrects the loss of orthogonality of `CGS`. In Figure 4.2, we see that the P-variants perfectly satisfy the bounds of Theorems 3.1 and 3.2 until $\kappa(\boldsymbol{\mathcal{X}})$ exceeds $10^8$.

*Example* 4.3. The `monomial` matrices are generated as follows: a diagonal $m \times m$ operator $A$ with evenly distributed eigenvalues in $(\frac{1}{10}, 10)$ is defined, and $q$ vectors $\boldsymbol{v}_k$, $k = 1, \ldots, q$, are randomly generated from the uniform distribution and normalized. The matrix $\boldsymbol{\mathcal{X}}$ is then defined as the concatenation of $q$ block vectors

$$\boldsymbol{X}_k = [\boldsymbol{v}_k \,|\, A\boldsymbol{v}_k \,|\, \cdots \,|\, A^{r-1}\boldsymbol{v}_k],$$

for a given block size $r = n/q$ (which differs from the partitioning parameter specified for the block Gram-Schmidt procedure). By varying $r$, we obtain matrices with a range of condition numbers, with larger $r$ corresponding to larger $\kappa(\boldsymbol{\mathcal{X}})$. In Figure 4.3, $m = 1000$, $p = 120$, and $s = 2$. These matrices highlight similar behavior as the `glued` matrices, but the transition between stable and unstable behavior is much more dramatic.

[2] https://github.com/katlund/BlockStab

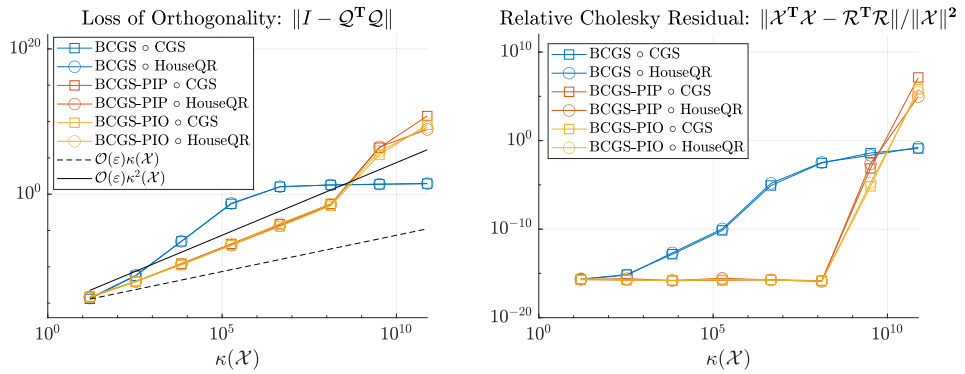


FIG. 4.2. *`Glued` $\kappa$-plots for different combinations of block variants and `IntraOrthos`.*

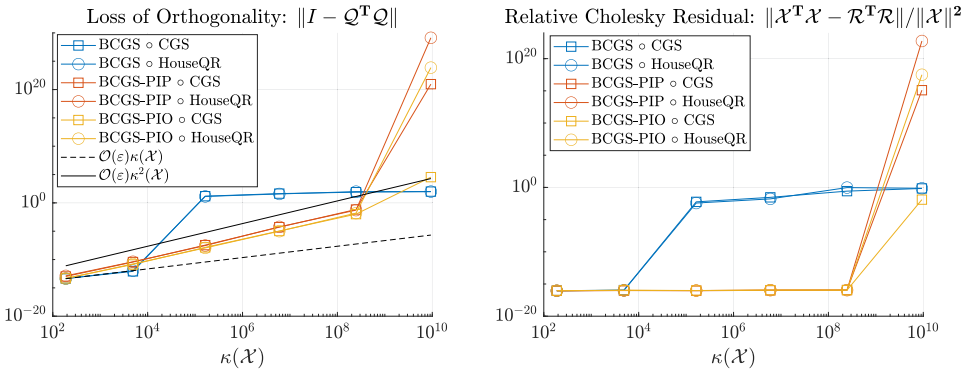


FIG. 4.3. *`Monomial` $\kappa$-plots for different combinations of block variants and `IntraOrthos`.*

**5. Conclusions.** Computing a QR factorization via Gram-Schmidt orthogonalization is a fundamental task in numerical linear algebra. Block variants of the Gram-Schmidt procedure offer potentially increased performance in many scenarios, but many variants lack fundamental results on their stability in finite precision. Our work here aims to help fill this gap. We have developed two new `BCGS` variants, `BCGS-PIO` and `BCGS-PIP`, based on a block analogy of the Pythagorean theorem-based variant of `CGS`. Unlike `BCGS`, for these new block variants the loss of orthogonality is bounded by $\mathcal{O}(\varepsilon)\,\kappa^2(\boldsymbol{\mathcal{X}})$ with a residual bound on the order of $\mathcal{O}(\varepsilon)\,\|\boldsymbol{\mathcal{X}}\|$ as long as relatively mild conditions on the `IntraOrtho` routine are satisfied. These new variants thus offer potentially better numerical behavior than standard `BCGS` without significant cost in terms of performance.

Much open work remains in this area. One avenue to explore is whether the use of a shift within `BCGS-PIO` and `BCGS-PIP` variants can allow the results to extend to more ill-conditioned matrices, as is done in [8]. Another area of interest is the analysis of 'low-sync' block variants, which have been shown experimentally to offer increased stability while maintaining the lower communication cost of `BCGS`; see, e.g., [6] and [21]. The development and analysis of reorthogonalization strategies for the `BCGS-PIO` and `BCGS-PIP` variants is also a possibility. There is also the greater question of whether the results here can be extended to derive results on the stability of block

Arnoldi and block GMRES methods and their communication-avoiding variants.

## REFERENCES


[1] N. N. Abdelmalek, *Round off error analysis for Gram-Schmidt method and solution of linear least squares problems*, BIT, 11 (1971), pp. 345–368, https://doi.org/10.1007/BF01939404.

[2] Z. Bai, D. Hu, and L. Reichel, *A Newton basis GMRES implementation*, IMA Journal of Numerical Analysis, 14 (1994), pp. 563–581.

[3] J. L. Barlow, *Block modified Gram-Schmidt algorithms and their analysis*, SIAM J. Matrix Anal. Appl., 40 (2019), pp. 1257–1290.

[4] J. L. Barlow and A. Smoktunowicz, *Reorthogonalized block classical Gram-Schmidt*, Numer. Math., 123 (2013), pp. 395–423, https://doi.org/10.1007/s00211-012-0496-2.

[5] Å. Björck, *Solving linear least squares problems by Gram-Schmidt orthogonalization*, BIT, 7 (1967), pp. 1–21.

[6] E. Carson, K. Lund, M. Rozložník, and S. Thomas, *An overview of Block Gram-Schmidt methods and their stability properties*, Submitted, (2020).

[7] J. W. Demmel, N. J. Higham, and R. S. Schreiber, *Stability of Block LU Factorization*, Numer. Linear Algebr. with Appl., 2 (1995), pp. 173–190.

[8] T. Fukaya, R. Kannan, Y. Nakatsukasa, Y. Yamamoto, and Y. Yanagisawa, *Shifted Cholesky QR for computing the QR factorization of ill-conditioned matrices*, SIAM J. Sci. Comput., 42 (2020), pp. A477–A503, https://doi.org/10.1137/18M1218212.

[9] S. R. Garcia and R. A. Horn, *A Second Course in Linear Algebra*, Cambridge Mathematical Textbooks, Cambridge University Press, 2017, https://doi.org/10.1017/9781316218419.

[10] L. Giraud, J. Langou, M. Rozložník, and J. Van Den Eshof, *Rounding error analysis of the classical Gram-Schmidt orthogonalization process*, Numer. Math., 101 (2005), pp. 87–100, https://doi.org/10.1007/s00211-005-0615-4.

[11] L. Grigori, S. Moufawad, and F. Nataf, *Enlarged Krylov subspace conjugate gradient methods for reducing communication*, SIAM J. Matrix Anal. Appl., 37 (2016), pp. 744–773, https://doi.org/10.1137/140989492.

[12] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, Society for Industrial and Applied Mathematics, Philadelphia, 2nd ed., 2002.

[13] M. Hoemmen, *Communication-avoiding Krylov subspace methods*, PhD thesis, Department of Computer Science, University of California at Berkeley, 2010.

[14] W. Jalby and B. Philippe, *Stability analysis and improvement of the block Gram-Schmidt algorithm*, SIAM J. Sci. Comput., 12 (1991), pp. 1058–1073, https://doi.org/10.1137/0912056, https://epubs.siam.org/doi/abs/10.1137/0912056.

[15] W. D. Joubert and G. F. Carey, *Parallelizable restarted iterative methods for nonsymmetric linear systems. Part I: Theory*, International Journal of Computer Mathematics, 44 (1992), pp. 243–267.

[16] A. Kiełbasiński, *Analiza numeryczna algorytmu ortogonalizacji Grama-Schmidta*, Ser. III Mat. Stosow. II, (1974), pp. 15–35.

[17] M. Kubínová and K. M. Soodhalter, *Admissible and attainable convergence behavior of block Arnoldi and GMRES*, SIAM J. Matrix Anal. Appl., 41 (2020), pp. 464–486.

[18] A. Smoktunowicz, J. L. Barlow, and J. Langou, *A note on the error analysis of classical Gram-Schmidt*, Numer. Math., 105 (2006), pp. 299–313, https://doi.org/10.1007/s00211-006-0042-1.

[19] H. F. Walker, *Implementation of the GMRES and Arnoldi methods using Householder transformations*, tech. report, Tech. Rep. UCRL-93589, Lawrence Livermore National Laboratory, 1985.

[20] I. Yamazaki, M. Hoemmen, P. Luszczek, and J. Dongarra, *Improving performance of GMRES by reducing communication and pipelining global collectives*, in 2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), IEEE, 2017, pp. 1118–1127.

[21] I. Yamazaki, S. Thomas, M. Hoemmen, E. G. Boman, K. Świrydowicz, and J. J. Elliott, *Low-synchronization orthogonalization schemes for s-step and pipelined Krylov solvers in Trilinos*, Proc. 2020 SIAM Conf. Parallel Process. Sci. Comput., (2020), pp. 118–128, https://doi.org/10.1137/1.9781611976137.11.