RANGE RESTRICTED ITERATIVE METHODS
FOR LINEAR DISCRETE ILL-POSED PROBLEMS[*]

ALESSANDRO BUCCINI[†], LUCAS ONISK[‡], AND LOTHAR REICHEL[‡]

Abstract. Linear systems of equations with a matrix whose singular values decay to zero with increasing index number, and without a significant gap, are commonly referred to as linear discrete ill-posed problems. Such systems arise, e.g., when discretizing a Fredholm integral equation of the first kind. The right-hand side vectors of linear discrete ill-posed problems that arise in science and engineering often represent an experimental measurement that is contaminated by measurement error. The solution to these problems typically is very sensitive to this error. Previous works have shown that error propagation into the computed solution may be reduced by using specially designed iterative methods that allow the user to select the subspace in which the approximate solution is computed. Since the dimension of this subspace often is quite small, its choice is important for the quality of the computed solution. This work describes algorithms for three iterative methods that modify the GMRES, block GMRES, and global GMRES methods for the solution of appropriate linear systems of equations. We contribute to the work already available on this topic by introducing two block variants for the solution of linear systems of equations with multiple right-hand side vectors. The dominant computational aspects are discussed, and software for each method is provided. Additionally, we illustrate the utility of these iterative subspace methods through numerical examples focusing on image reconstruction. This paper is accompanied by software.

Key words. ill-posed problems, iterative method, Arnoldi process, block Arnoldi process, global Arnoldi process

AMS subject classifications. 65F10, 65F22

1. Introduction. We are interested in the approximate solution of linear systems of equations

$$(1.1) \qquad \boldsymbol{Ax} = \boldsymbol{b}^{\delta},$$

where $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is a large matrix whose singular values decay to zero with increasing index number without a significant gap. This makes the matrix $\boldsymbol{A}$ severely ill-conditioned and possibly rank-deficient. Linear systems of equations with such a matrix are commonly referred to as linear discrete ill-posed problems. They arise, for instance, from the discretization of a Fredholm integral equation of the first kind; see, e.g., Engl et al. [15] and Hansen [17] for discussions on ill-posed and linear discrete ill-posed problems. In particular, discrete deconvolution problems that arise in image restoration give rise to linear discrete ill-posed problems; see [20]. We will illustrate the performance of the methods discussed with a focus on image restoration.

The right-hand side vector $\boldsymbol{b}^{\delta} \in \mathbb{R}^n$ in linear discrete ill-posed problems typically arises from measured data from experiments in science and engineering. These data are frequently contaminated by a measurement error $\boldsymbol{e} \in \mathbb{R}^n$. Let $\boldsymbol{b} \in \mathbb{R}^n$ denote the unknown error-free vector associated with $\boldsymbol{b}^{\delta}$. Then

$$\boldsymbol{b}^{\delta} = \boldsymbol{b} + \boldsymbol{e}.$$

We assume that the exact right-hand side $\boldsymbol{b}$ is in the range of $\boldsymbol{A}$. This is required when using the discrepancy principle (see below) to determine how many steps of the chosen iterative

[†]Department of Mathematics and Computer Science University of Cagliari, 09124 Cagliari, Italy (alessandro.buccini@unica.it).

[‡]Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA (lonisk@kent.edu, reichel@math.kent.edu).

methods to carry out. We are interested in computing an accurate approximation of the solution $\boldsymbol{x}^\dagger$ of minimal Euclidean norm of the unavailable linear system of equations

$$(1.2) \qquad\qquad \boldsymbol{Ax} = \boldsymbol{b}.$$

We note that because of the ill-conditioning of $\boldsymbol{A}$ and the error $\boldsymbol{e}$ in $\boldsymbol{b}^\delta$, the least-squares solution of minimal Euclidean norm of (1.1) generally does not provide a useful approximation of $\boldsymbol{x}^\dagger$.

A popular and simple approach to determine a useful approximation of $\boldsymbol{x}^\dagger$ is to apply an iterative solution method to (1.1) and carry out sufficiently few iterations. Let $\boldsymbol{x}_0 = \boldsymbol{0}$ be the initial iterate used by an iterative method. The successive iterates $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ will approach $\boldsymbol{x}^\dagger$ with increasing index number when the index numbers are small enough. However, because of the ill-conditioning of $\boldsymbol{A}$ and the error $\boldsymbol{e}$ in $\boldsymbol{b}^\delta$, iterates $\boldsymbol{x}_k$ with large indices will diverge from $\boldsymbol{x}^\dagger$. This behavior of the iterates is known as *semiconvergence*. Thus, there is at least one optimal iterate $\boldsymbol{x}_p$ such that

$$\left\|\boldsymbol{x}_p - \boldsymbol{x}^\dagger\right\| \leq \left\|\boldsymbol{x}_k - \boldsymbol{x}^\dagger\right\|, \qquad k = 0, 1, \ldots$$

Throughout this paper $\|\cdot\|$ denotes either the Euclidean vector norm or the Frobenius norm depending on the context.

The software accompanying this paper seeks to compute an accurate approximation of $\boldsymbol{x}^\dagger$ by determining the first iterate $\boldsymbol{x}_p$ that satisfies the *discrepancy principle*. This approach can be used when a bound

$$\|\boldsymbol{e}\| \leq \delta$$

is known. The discrepancy principle prescribes that an iterative method for the approximate solution of (1.1) should be terminated as soon as an iterate $\boldsymbol{x}_p$ that satisfies

$$\left\|\boldsymbol{Ax}_p - \boldsymbol{b}^\delta\right\| \leq \tau\delta$$

has been found. Here $\tau > 1$ is a user-specified constant that is independent of δ; see [15] for a discussion on the discrepancy principle. Note that the iterate $\boldsymbol{x}_p$ determined by the discrepancy principle depends on δ; generally, the number of iterations p increases when δ decreases.

When an upper bound for $\|\boldsymbol{e}\|$ is not available, the discrepancy principle cannot be applied unless such a bound can be estimated. It is possible to compute a fairly accurate estimate of $\|\boldsymbol{e}\|$ in some situations. This is illustrated in [22, 23] when the error $\boldsymbol{e}$ can be modeled by additive white Gaussian noise of unknown variance. An application of the discrepancy principle also requires the linear systems of equations (1.2) to be consistent. This property is important to be able to show that the iterates $\boldsymbol{x}_p$ converge to $\boldsymbol{x}^\dagger$ when δ converges to zero. In applications, the consistency of (1.2) is particularly important when the amount of error $\boldsymbol{e}$ in $\boldsymbol{b}^\delta$ is small.

There are many other approaches available to identify a suitable iterate when the discrepancy principle cannot be applied, including the L-curve criterion and generalized cross validation; see, e.g., [7, 8, 19, 27, 28, 31, 32]. These methods are referred to as "heuristic"; they generally perform well, but may fail for certain problems; see Kindermann [27] for an insightful discussion on heuristic methods. Some of the references listed above are concerned with Tikhonov regularization, which is an alternative to the truncated iteration described above. Many methods that use Tikhonov regularization quite easily can be modified to instead apply a truncated iteration. Conversely, the iterative methods described in this paper can easily be applied together with Tikhonov regularization; see, e.g., [24]. While the algorithms of this paper implement stopping criteria based on the discrepancy principle, they can easily be modified to be applicable with a heuristic rule for determining a suitable iterate.

The generalized minimal residual (GMRES) method is a popular iterative method for the solution of linear systems of algebraic equations with a large non-symmetric matrix that stem from the discretization of linear well-posed problems such as elliptic partial differential equations with Dirichlet boundary conditions [34]. The application of GMRES to the solution of linear discrete ill-posed problems is described in, e.g., [9, 10, 11, 18]. Let the initial iterate be $\boldsymbol{x}_0 = \boldsymbol{0}$. Then the pth iterate, $\boldsymbol{x}_p$, computed by GMRES applied to the solution of (1.1) satisfies

$$\left\| \boldsymbol{A}\boldsymbol{x}_p - \boldsymbol{b}^\delta \right\| = \min_{\boldsymbol{x} \in \mathbb{K}_p(\boldsymbol{A}, \boldsymbol{b}^\delta)} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^\delta \right\|,$$

where

$$\mathbb{K}_p(\boldsymbol{A}, \boldsymbol{b}^\delta) = \operatorname{span} \left\{ \boldsymbol{b}^\delta, \boldsymbol{A}\boldsymbol{b}^\delta, \dots, \boldsymbol{A}^{p-1}\boldsymbol{b}^\delta \right\}$$

is a Krylov subspace. We tacitly assume that p is small enough so that $\dim\left(\mathbb{K}_p\left(\boldsymbol{A}, \boldsymbol{b}^\delta \right) \right) = p$. Typically, this condition holds for linear discrete ill-posed problems when the iterations are terminated by the discrepancy principle.

In applications where the desired solution $\boldsymbol{x}^\dagger$ of (1.2) is the discretization of a smooth function, it has been illustrated in, e.g., [9, 18], that a variant of GMRES, known as range restricted GMRES (rrGMRES), often delivers more accurate approximations of $\boldsymbol{x}^\dagger$ than the standard GMRES method when applied to (1.1). The name of the method derives from the fact that the computed iterates, $\boldsymbol{x}_p$, live in $\mathcal{R}(\boldsymbol{A})$. Here and throughout this paper $\mathcal{R}(\boldsymbol{M})$ denotes the range of the matrix $\boldsymbol{M}$. With initial iterate $\boldsymbol{x}_0 = \boldsymbol{0}$, the pth iterate, $\boldsymbol{x}_p$, determined by rrGMRES satisfies

$$\left\| \boldsymbol{A}\boldsymbol{x}_p - \boldsymbol{b}^\delta \right\| = \min_{\boldsymbol{x} \in \mathbb{K}_p(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{b}^\delta)} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^\delta \right\|,$$

where

(1.3) $$\boldsymbol{x}_p \in \mathbb{K}_p\left(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{b}^\delta \right) = \operatorname{span} \left\{ \boldsymbol{A}\boldsymbol{b}^\delta, \boldsymbol{A}^2\boldsymbol{b}^\delta, \dots, \boldsymbol{A}^p\boldsymbol{b}^\delta \right\}.$$

rrGMRES was first used in [9]. Properties and applications of this method are described in [2, 3, 4, 33]. The implementation used in these references demands reorthogonalization when computing an orthonormal basis for the Krylov subspace (1.3), because the Fourier coefficients of $\boldsymbol{b}^\delta$ with respect to this basis have to be computed. An implementation that does not require reorthogonalization was subsequently described in [29, 30].

It is natural to extend rrGMRES to the ℓ-shifted GMRES method, which solves the minimization problem

(1.4) $$\left\| \boldsymbol{A}\boldsymbol{x}_p^{(\ell)} - \boldsymbol{b}^\delta \right\| = \min_{\boldsymbol{x} \in \mathbb{K}_p\left(\boldsymbol{A}, \boldsymbol{A}^\ell \boldsymbol{b}^\delta \right)} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}^\delta \right\|,$$

where

$$\boldsymbol{x}_p^{(\ell)} \in \mathbb{K}_p\left(\boldsymbol{A}, \boldsymbol{A}^\ell \boldsymbol{b}^\delta \right) = \operatorname{span} \left\{ \boldsymbol{A}^\ell \boldsymbol{b}^\delta, \boldsymbol{A}^{\ell+1}\boldsymbol{b}^\delta, \dots, \boldsymbol{A}^{\ell+p-1}\boldsymbol{b}^\delta \right\}, \quad \text{for } \ell = 0, 1, \dots, \ell_{\max},$$

with $\ell_{\max}$ typically small (usually 3–4). The superscript of $\boldsymbol{x}_p^{(\ell)}$ indicates the subspace in which this vector lives, specifically $\boldsymbol{x}_p^{(\ell)} \in \mathcal{R}(A^\ell)$. Thus, rrGMRES corresponds to the shift $\ell = 1$. Preliminary computational results reported in [13] with ℓ-shifted GMRES illustrated that using

a shift $\ell > 1$ may result in more accurate approximations of $\boldsymbol{x}^\dagger$ than those determined by standard GMRES (0-shifted GMRES) and rrGMRES.

It is the purpose of this paper to provide further illustrations of the performance of the ℓ-shifted GMRES method and to discuss some computational aspects, including its efficient implementation and storage requirement. Our implementation of the ℓ-shifted GMRES method does not require much more computer storage than the standard GMRES method. Moreover, the 1-shifted GMRES method described requires less storage than the algorithm provided in [30]. We also develop two block analogues of the ℓ-shifted GMRES method, which are convenient to use when the right-hand side in (1.1) is replaced by a "block vector", i.e., a matrix with a few columns. Linear discrete ill-posed problems of this kind arise, e.g., in color image restoration. MATLAB codes for the ℓ-shifted GMRES method (1.4) and its block analogues, referred to as the ℓ-shifted block GMRES (BGMRES) and ℓ-shifted global GMRES (gl-GMRES) methods, accompany this paper.

The remainder of this paper is organized as follows: Section 2 describes the ℓ-shifted GMRES solution method for the minimization problem (1.4) and discusses its properties. The ℓ-shifted BGMRES method is introduced in Section 3 followed by the ℓ-shifted gl-GMRES method in Section 4. Some background on image deblurring is provided in Section 5, where we also illustrate the performance of our methods with a few computed examples. Section 6 contains concluding remarks.

2. The ℓ-shifted GMRES method. We first review the computation of the rrGMRES iterate, $\boldsymbol{x}_p^{(1)}$, by the method described in [29, 30]. Subsequently, we discuss the computation of the ℓ-shifted GMRES iterate, $\boldsymbol{x}_p^{(\ell)}$, for $\ell = 2, 3, \ldots, \ell_{\max}$, and consider computational aspects. Throughout our derivations, we emphasize the use of elementary reflectors for the efficient upper-triangularization of the matrix of the projected linear systems of equations.

Application of p-steps of the Arnoldi process to the matrix $\boldsymbol{A}$ with initial vector $\boldsymbol{b}^\delta$ generically gives the decomposition

$$(2.1) \qquad\qquad \boldsymbol{A}\boldsymbol{V}_p = \boldsymbol{V}_{p+1}\boldsymbol{H}_{p+1,p},$$

where the matrix $\boldsymbol{V}_{p+1} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{p+1}] \in \mathbb{R}^{n \times (p+1)}$ has orthonormal columns with initial column $\boldsymbol{v}_1 = \boldsymbol{b}^\delta / \left\| \boldsymbol{b}^\delta \right\|$. The matrix $\boldsymbol{V}_p \in \mathbb{R}^{n \times p}$ is made up of the first p columns of $\boldsymbol{V}_{p+1}$, and $\boldsymbol{H}_{p+1,p} \in \mathbb{R}^{(p+1) \times p}$ is an upper Hessenberg matrix with positive subdiagonal entries. The columns of $\boldsymbol{V}_{p+1}$ span the Krylov subspace $\mathbb{K}_{p+1}\left(\boldsymbol{A}, \boldsymbol{b}^\delta\right)$. The decomposition (2.1) is the basis for the implementation of the standard GMRES method; see [34, Chapter 6]. The computation of the decomposition (2.1) requires p matrix-vector product evaluations with $\boldsymbol{A}$. The decomposition is conveniently computed with the Arnoldi process, which applies the modified Gram-Schmidt method for orthogonalization of the columns of $\boldsymbol{V}_{p+1}$. The Arnoldi process is described by Algorithm 1. A discussion of this algorithm can be found, e.g., in [34].

Introduce the QR factorization of the upper Hessenberg matrix $\boldsymbol{H}_{p+1,p}$ on the right-hand side of (2.1),

$$(2.2) \qquad\qquad \boldsymbol{H}_{p+1,p} = \boldsymbol{Q}_{p+1}^{(1)}\boldsymbol{R}_{p+1,p}^{(1)},$$

where the matrix $\boldsymbol{Q}_{p+1}^{(1)} \in \mathbb{R}^{(p+1) \times (p+1)}$ is orthogonal and $\boldsymbol{R}_{p+1,p}^{(1)} \in \mathbb{R}^{(p+1) \times p}$ has a leading $p \times p$ upper triangular submatrix $\boldsymbol{R}_p^{(1)}$ and a vanishing last row. Because $\boldsymbol{H}_{p+1,p}$ is upper Hessenberg, the matrix $\boldsymbol{Q}_{p+1}^{(1)}$ can be expressed as the product of p elementary reflectors

$$(2.3) \qquad\qquad \boldsymbol{Q}_{p+1}^{(1)} = \boldsymbol{G}_1\boldsymbol{G}_2 \ldots \boldsymbol{G}_p,$$

Algorithm 1: The Arnoldi process.

Input: $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{b}^\delta \in \mathbb{R}^n$
Output: $\boldsymbol{V}_{j+1} \in \mathbb{R}^{n \times (j+1)}$ and $\boldsymbol{H}_{j+1,j} \in \mathbb{R}^{(j+1) \times j}$

1   Set $\boldsymbol{v}_1 = \boldsymbol{b}^\delta / \left\| \boldsymbol{b}^\delta \right\|$;
2   **for** $j = 1, 2, \ldots, n$ **do**
3   Compute $\boldsymbol{w}_j = \boldsymbol{A}\boldsymbol{v}_j$;
4   **for** $i = 1, 2, \ldots, j$ **do**
5          $h_{i,j} = (\boldsymbol{w}_j, \boldsymbol{v}_i)$;
6          $\boldsymbol{w}_j = \boldsymbol{w}_j - h_{i,j}\boldsymbol{v}_i$;
7   **end**
8      $h_{j+1,j} = \left\| \boldsymbol{w}_j \right\|$;
9   **if** $h_{j+1,j} = 0$ **then**
10   Stop
11   **end**
12      $\boldsymbol{v}_{j+1} = \boldsymbol{w}_j / h_{j+1,j}$;
13 **end**

---

where the $\boldsymbol{G}_j \in \mathbb{R}^{(p+1) \times (p+1)}$, for $j = 1, 2, \ldots, p$, are symmetric elementary reflectors; specifically, $\boldsymbol{G}_j$ is the identity matrix except for a 2×2 block in the rows and columns j and $j + 1$. It follows from (2.3) that $\boldsymbol{Q}_{p+1}^{(1)}$ is of upper Hessenberg form. The computation of the factorization (2.2) can be carried out in $O(p^2)$ arithmetic floating point operations (flops). For comparison, a structure-ignoring approach such as a standard QR factorization requires $O(p^3)$ flops.

To show that the iterates generated are in the restricted space, consider the "reduced" QR factorization of

$$\boldsymbol{H}_{p+1,p} = \boldsymbol{Q}_{p+1,p}^{(1)} \boldsymbol{R}_p^{(1)},$$

where the matrix $\boldsymbol{Q}_{p+1,p}^{(1)} \in \mathbb{R}^{(p+1) \times p}$ consists of the first p columns of $\boldsymbol{Q}_{p+1}^{(1)}$ and the last row of $\boldsymbol{R}_{p+1,p}^{(1)}$ is removed. Using these matrices, we define

$$\boldsymbol{W}_p^{(1)} = \boldsymbol{V}_{p+1} \boldsymbol{Q}_{p+1,p}^{(1)} \in \mathbb{R}^{n \times p}.$$

From (2.1) and the reduced version of (2.2), it follows that

$$\boldsymbol{W}_p^{(1)} = \boldsymbol{A}\boldsymbol{V}_p \left(\boldsymbol{R}_p^{(1)} \right)^{-1},$$

which shows that $\mathcal{R}\left(\boldsymbol{W}_p^{(1)} \right) = \mathbb{K}\left(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{b}^\delta \right)$. We will assume for the ease of discussion that all Krylov subspaces are of full dimension. This is the generic situation. With this notation the minimization problem (1.4) for $\ell = 1$ may be written as

$$\min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A}\boldsymbol{W}_p^{(1)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\| = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A}\boldsymbol{V}_{p+1} \boldsymbol{Q}_{p+1,p}^{(1)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\|$$

$$= \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{V}_{p+2} \boldsymbol{H}_{p+2,p+1} \boldsymbol{Q}_{p+1,p}^{(1)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\|$$

$$= \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{H}_{p+2,p+1} \boldsymbol{Q}_{p+1,p}^{(1)} \boldsymbol{y} - \left\| \boldsymbol{b}^\delta \right\| \boldsymbol{e}_1 \right\|,$$

where $\boldsymbol{e}_1 = [1, 0, \ldots, 0]^T \in \mathbb{R}^{p+2}$ denotes the first axis vector and the last equality follows from the fact that the first column of $\boldsymbol{V}_{p+2}$ is $\boldsymbol{v}_1 = \boldsymbol{b}^\delta / \left\| \boldsymbol{b}^\delta \right\|$. Throughout, we will use the superscript ^{T} to denote transposition. The matrices $\boldsymbol{H}_{p+2,p+1}$ and $\boldsymbol{V}_{p+1}$ are produced by applying $p + 1$ steps of the Arnoldi process to $\boldsymbol{A}$ with the initial vector $\boldsymbol{b}^\delta$.

Because both $\boldsymbol{H}_{p+2,p+1}$ and $\boldsymbol{Q}_{p+1,p}^{(1)}$ are upper Hessenberg, their product is a "super upper Hessenberg matrix", i.e., it has vanishing entries below the sub-subdiagonal. This allows us to compute a second QR factorization

$$\boldsymbol{H}_{p+2,p+1} \boldsymbol{Q}_{p+1,p}^{(1)} = \boldsymbol{Q}_{p+2}^{(2)} \boldsymbol{R}_{p+2,p}^{(2)}$$

in $O(p^2)$ flops using elementary reflectors. Here the matrix $\boldsymbol{Q}_{p+2}^{(2)} \in \mathbb{R}^{(p+2) \times (p+2)}$ is orthogonal with zero entries below the sub-subdiagonal, and $\boldsymbol{R}_{p+2,p}^{(2)} \in \mathbb{R}^{(p+2) \times p}$ has a leading $p \times p$ upper triangular submatrix, denoted by $\boldsymbol{R}_p^{(2)} \in \mathbb{R}^{p \times p}$, and with two vanishing last rows. With this we have that

$$(2.4) \qquad \begin{aligned} \boldsymbol{A} \boldsymbol{W}_p^{(1)} &= \boldsymbol{A}^2 \boldsymbol{V}_p (\boldsymbol{R}_p^{(1)})^{-1} = \boldsymbol{A} \boldsymbol{V}_{p+1} \boldsymbol{Q}_{p+1,p}^{(1)} \\ &= \boldsymbol{V}_{p+2} \boldsymbol{H}_{p+2,p+1} \boldsymbol{Q}_{p+1,p}^{(1)} = \boldsymbol{V}_{p+2} \boldsymbol{Q}_{p+2}^{(2)} \boldsymbol{R}_{p+2,p}^{(2)}. \end{aligned}$$

It follows that the minimization problem (1.4) for $p = 1$ may be written concisely as

$$(2.5) \qquad \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A} \boldsymbol{W}_p^{(1)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\| = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{R}_{p+2,p}^{(2)} \boldsymbol{y} - \left\| \boldsymbol{b}^\delta \right\| \left(\boldsymbol{Q}_{p+2}^{(2)} \right)^T \boldsymbol{e}_1 \right\|.$$

Let $\boldsymbol{y}_p^{(1)}$ denote the solution of (2.5). Then the solution of the 1-shifted GMRES method is given by $\boldsymbol{x}_p^{(1)} = \boldsymbol{W}_p^{(1)} \boldsymbol{y}_p^{(1)}$.

We provide a brief exposition of the 2-shifted GMRES method. Let the matrix $\boldsymbol{W}_p^{(2)}$ consist of the first p columns of $\boldsymbol{V}_{p+2} \boldsymbol{Q}_{p+2}^{(2)}$. Then

$$\boldsymbol{W}_p^{(2)} = \boldsymbol{A} \boldsymbol{W}_p^{(1)} \left(\boldsymbol{R}_p^{(2)} \right)^{-1}$$

follows from (2.4) and implies that $\mathcal{R} \left(\boldsymbol{W}_p^{(2)} \right) = \mathbb{K} \left(\boldsymbol{A}, \boldsymbol{A}^2 \boldsymbol{b}^\delta \right)$. With this, the minimization problem for the 2-shifted GMRES method can be expressed as

$$(2.6) \qquad \begin{aligned} \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A} \boldsymbol{W}_p^{(2)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\| &= \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A} \boldsymbol{V}_{p+2} \boldsymbol{Q}_{p+2,p}^{(2)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\| \\ &= \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{V}_{p+3} \boldsymbol{H}_{p+3,p+2} \boldsymbol{Q}_{p+2,p}^{(2)} \boldsymbol{y} - \boldsymbol{b}^\delta \right\| \\ &= \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{H}_{p+3,p+2} \boldsymbol{Q}_{p+2,p}^{(2)} \boldsymbol{y} - \left\| \boldsymbol{b}^\delta \right\| \boldsymbol{e}_1 \right\| \\ &= \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{R}_{p+3,p}^{(3)} \boldsymbol{y} - \left\| \boldsymbol{b}^\delta \right\| \left(\boldsymbol{Q}_{p+3}^{(3)} \right)^T \boldsymbol{e}_1 \right\|, \end{aligned}$$

where the matrices $\boldsymbol{V}_{p+3}$ and $\boldsymbol{H}_{p+3,p}$ are determined by applying $p + 2$ steps of the Arnoldi process to $\boldsymbol{A}$ with initial vector $\boldsymbol{b}^\delta$. The matrices $\boldsymbol{Q}_{p+3}^{(3)}$ and $\boldsymbol{R}_{p+3,p}^{(3)}$ stem from the QR factorization

$$(2.7) \qquad \boldsymbol{H}_{p+3,p+2} \boldsymbol{Q}_{p+2,p}^{(2)} = \boldsymbol{Q}_{p+3}^{(3)} \boldsymbol{R}_{p+3,p}^{(3)},$$

where the matrix $\boldsymbol{Q}_{p+3}^{(3)} \in \mathbb{R}^{(p+3)\times(p+3)}$ is orthogonal and the matrix $\boldsymbol{R}_{p+3,p}^{(3)} \in \mathbb{R}^{(p+3)\times p}$ has a leading $p \times p$ upper triangular submatrix and three vanishing last rows. Because of the Hessenberg-type structure of the matrices $\boldsymbol{H}_{p+3,p+2}$ and $\boldsymbol{Q}_{p+2,p}^{(2)}$, the matrix $\boldsymbol{Q}_{p+3}^{(3)}$ vanishes below the sub-sub-subdiagonal. Thus, the factorization (2.7) can be computed in only $O(p^2)$ flops by a suitable choice of elementary reflectors. Letting $\boldsymbol{y}_p^{(2)}$ denote the solution of (2.6), the pth iterate of the 2-shifted GMRES method is given by $\boldsymbol{x}_p^{(2)} = \boldsymbol{W}_p^{(2)}\boldsymbol{y}_p^{(2)}$.

We may proceed in a similar manner to derive matrices $\boldsymbol{W}_p^{(\ell)} \in \mathbb{R}^{n\times p}$ such that $\mathcal{R}\left(\boldsymbol{W}_p^{(\ell)}\right) = \mathbb{K}\left(\boldsymbol{A}, \boldsymbol{A}^\ell \boldsymbol{b}^\delta\right)$. With these matrices we can compute the solution $\boldsymbol{x}_p^{(\ell)}$ of the ℓ-shifted GMRES minimization problem for $\ell = 3, \ldots, \ell_{\max}$. The computation of these solutions requires the evaluation of $p + \ell$ matrix-vector products with $\boldsymbol{A}$. This constitutes the dominant computational work when the matrix $\boldsymbol{A}$ is large and $p + \ell$ is fairly small. The latter is typically the case when solving linear discrete ill-posed problems.

Algorithm 2: ℓ-shifted GMRES ($\ell \geq 1$) with the discrepancy principle.

> **Input:** $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{b}^\delta \in \mathbb{R}^n$, and $\ell \in \{1, 2, 3, \ldots\}$
> **Output:** $\boldsymbol{x}_p^{(\ell)} \in \mathbb{R}^n$
> 1   Set $\boldsymbol{v}_1 = \boldsymbol{b}^\delta / \left\|\boldsymbol{b}^\delta\right\|$ and $\boldsymbol{x}_0^{(\ell)} = \boldsymbol{0}$;
> 2   Compute ℓ steps of Arnoldi: $\boldsymbol{AV}_\ell = \boldsymbol{V}_{\ell+1}\boldsymbol{H}_{\ell+1,\ell}$;
> 3   **for** $p = 1, 2, \ldots$ **do**
> 4   Compute next Arnoldi step: $\boldsymbol{AV}_{\ell+p} = \boldsymbol{V}_{\ell+p+1}\boldsymbol{H}_{\ell+p+1,\ell+p}$;
> 5   Compute QR factorization: $\left[\boldsymbol{Q}_{p+1}^{(1)}, \boldsymbol{R}_{p+1,p}^{(1)}\right] = \boldsymbol{H}_{p+1,p}$;
> 6   **for** $j = 1, 2, \ldots, \ell$ **do**
> 7   Compute QR factorization: $\left[\boldsymbol{Q}_{j+p+1}^{(j+1)}, \boldsymbol{R}_{j+p+1,p}^{(j+1)}\right] = \boldsymbol{H}_{j+p+1,j+p}\boldsymbol{Q}_{j+p,p}^{(j)}$;
> 8   **end**
> 9   Compute $\boldsymbol{y}_p^{(\ell)}$ as the minimizer of $\left\|\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y} - \left\|\boldsymbol{b}^\delta\right\| \left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\right)^T \boldsymbol{e}_1\right\|$;
> 10   Compute $\boldsymbol{x}_p^{(\ell)} = \boldsymbol{V}_{\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}\boldsymbol{y}_p^{(\ell)}$;
> 11   Compute $\|\boldsymbol{r}_p\| = \left\|\boldsymbol{Ax}_p^{(\ell)} - \boldsymbol{b}^\delta\right\|$;
> 12   **if** $\|\boldsymbol{r}_p\| \leq \tau\delta$ **then**
> 13   Stop;
> 14   **end**
> 15 **end**

With the review of the works [13, 29, 30] completed, we turn to some computational aspects of the ℓ-shifted GMRES method. Algorithm 2 provides an implementation of this method that terminates according to the discrepancy principle. The ℓ-shifted GMRES method requires $\ell+p$ matrix-vector product evaluations with $\boldsymbol{A}$ for the Arnoldi iteration in lines **2** and **4**. As mentioned above, these account for the dominant computational work for large problems. The upper triangulation of the Hessenberg-type matrices in lines **5** and **7** may be computed by using elementary reflectors. Since lines **6–8** are executed ℓ times for each one of the p iterations of the algorithm, the total flop count is $O(\ell p^2)$. We remark that a straightforward QR factorization of a $(p + 1) \times p$ matrix requires $O(p^3)$ flops. Using the structure-ignoring approach increases the flop count for lines **6–8** to $O(\ell p^3)$. However, we have found that a

direct implementation of elementary reflectors in MATLAB is computationally slower than using the MATLAB function qr. We comment more on this in Section 5.

Since the discrepancy principle is used as a stopping criterion for Algorithm 2, we require the computation of the norm of the pth residual, $\boldsymbol{r}_p$, in line **11**. In the standard GMRES algorithm for solving the linear system (1.1), the pth residual can be obtained without the use of an additional matrix-vector product; see [34, Proposition 6.9]. We will show this to be true also for the ℓ-shifted GMRES method.

Consider the minimization problem solved by the ℓ-shifted GMRES method

$$(2.8) \qquad \begin{aligned} \min_{\boldsymbol{y}\in\mathbb{R}^p}\left\|\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}\boldsymbol{y}-\boldsymbol{b}^\delta\right\| &= \min_{\boldsymbol{y}\in\mathbb{R}^p}\left\|\boldsymbol{H}_{\ell+p+1,\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}\boldsymbol{y}-\left\|\boldsymbol{b}^\delta\right\|\boldsymbol{e}_1\right\| \\ &= \min_{\boldsymbol{y}\in\mathbb{R}^p}\left\|\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y}-\left\|\boldsymbol{b}^\delta\right\|\left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\right)^T\boldsymbol{e}_1\right\|. \end{aligned}$$

Let $\boldsymbol{y}^{(\ell)}$ denote the solution of (2.8) determined by back substitution. Then the pth iterate of the ℓ-shifted GMRES method is given by $\boldsymbol{x}_p^{(\ell)}=\boldsymbol{W}_p^{(\ell)}\boldsymbol{y}^{(\ell)}$. Since the matrix $\boldsymbol{Q}_{j+p+1}^{(j+1)}$ may be written as a product of $j+p$ elementary reflectors

$$(2.9) \qquad \boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}=\boldsymbol{G}_1\boldsymbol{G}_2\ldots\boldsymbol{G}_{\ell+p},$$

for $j=0,1,\ldots,\ell$, the right-hand side of the last equality of (2.8) can be expressed as

$$(2.10) \qquad \begin{aligned} \left\|\boldsymbol{b}^\delta\right\|\left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\right)^T\boldsymbol{e}_1 &= \boldsymbol{G}_{\ell+p}^T\ldots\boldsymbol{G}_3^T\boldsymbol{G}_2^T\boldsymbol{G}_1^T\left[\left\|\boldsymbol{b}^\delta\right\|,0,\ldots,0\right]^T \\ &= \boldsymbol{G}_{\ell+p}^T\ldots\boldsymbol{G}_3^T\boldsymbol{G}_2^T\left[\gamma_1,\gamma_2,0,\ldots,0\right]^T \\ &= \boldsymbol{G}_{\ell+p}^T\ldots\boldsymbol{G}_3^T\left[\gamma_1,\gamma_2,\gamma_3,0,\ldots,0\right]^T \\ &\vdots \\ &= \left[\gamma_1,\gamma_2,\gamma_3,\ldots,\gamma_{\ell+p},\gamma_{\ell+p+1}\right]^T := \boldsymbol{g}_{\ell+p+1}, \end{aligned}$$

where $\boldsymbol{g}_{\ell+p+1}\in\mathbb{R}^{\ell+p+1}$ and $\boldsymbol{G}_k^T\in\mathbb{R}^{(\ell+p+1)\times(\ell+p+1)}$, for $k=1,\ldots,\ell+p$.

PROPOSITION 2.1. *During the pth iteration of Algorithm 2, let $\boldsymbol{G}_k$, for $k=1,2,\ldots,\ell+p$, be the reflector matrices used to transform $\boldsymbol{H}_{\ell+p+1,\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}$ in line 8 when $j=\ell$ into upper triangular form $\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}$, and define $\boldsymbol{g}_{\ell+p+1}$ as in (2.10). Let $\boldsymbol{R}_p^{(\ell+1)}$ denote the $p\times p$ upper triangular matrix obtained from $\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}$ by deleting its last $\ell+1$ rows and by $\bar{\boldsymbol{g}}_p$ the p-vector obtained from $\boldsymbol{g}_{\ell+p+1}$ by deleting the last $\ell+1$ components. Then:*

1. *The rank of $\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}$ is equal to the rank of $\boldsymbol{R}_p^{(\ell+1)}$. In particular, if $r_{(p,p)}^{(\ell+1)}$, the last diagonal entry of $\boldsymbol{R}_p^{(\ell+1)}$, is equal to 0, then $\boldsymbol{A}$ is singular.*

2. *The vector $\boldsymbol{y}_p^{(\ell)}$ that minimizes $\left\|\boldsymbol{H}_{\ell+p+1,\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}\boldsymbol{y}-\left\|\boldsymbol{b}^\delta\right\|\boldsymbol{e}_1\right\|$ is given by*

$$\boldsymbol{y}_p^{(\ell)}=\left(\boldsymbol{R}_p^{(\ell+1)}\right)^{-1}\bar{\boldsymbol{g}}_p.$$

3. *The residual vector, $\boldsymbol{r}_p:=\boldsymbol{A}\boldsymbol{x}_p^{(\ell)}-\boldsymbol{b}^\delta$, at step p of Algorithm 2 satisfies*

$$\begin{aligned} \boldsymbol{r}_p &= \boldsymbol{V}_{\ell+p+1}\left(\boldsymbol{H}_{\ell+p+1,\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}\boldsymbol{y}-\left\|\boldsymbol{b}^\delta\right\|\boldsymbol{e}_1\right) \\ &= \boldsymbol{V}_{\ell+p+1}\left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\right)^T\left[0,\ldots,0,\gamma_{p+1},\ldots,\gamma_{\ell+p+1}\right]^T, \end{aligned}$$

and, as a result,

$$\|\boldsymbol{r}_p\| = \left\|\boldsymbol{A}\boldsymbol{x}_p^{(\ell)} - \boldsymbol{b}^\delta\right\| = \left\|[0,\ldots,0,\gamma_{p+1},\ldots,\gamma_{\ell+p+1}]^T\right\| = \sqrt{\sum_{i=p+1}^{\ell+p+1}(\gamma_i)^2}.$$

Proof. To show the first part, we begin by stating the generalized result from (2.4) as

$$\boldsymbol{A}\boldsymbol{W}_p^{(\ell)} = \boldsymbol{V}_{\ell+p+1}\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}.$$

Since $\boldsymbol{V}_{\ell+p+1}\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}$ is orthogonal, the matrices $\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}$ and $\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}$ have the same rank. If the entry $r_{(p,p)}^{(\ell+1)}$ of $\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}$ vanishes, then $\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}$ is of rank $\leq p-1$, and as a result $\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}$ also is of rank $\leq p-1$. We show now that $\boldsymbol{W}_p^{(\ell)}$ is of full rank, which immediately implies that $\boldsymbol{A}$ is singular. We have that

$$\begin{aligned}
\boldsymbol{W}_p^{(\ell)} &= \boldsymbol{A}\boldsymbol{W}_p^{(\ell-1)}\left(\boldsymbol{R}_p^{(\ell)}\right)^{-1}\\
&= \boldsymbol{A}^2\boldsymbol{W}_p^{(\ell-2)}\left(\boldsymbol{R}_p^{(\ell-1)}\right)^{-1}\left(\boldsymbol{R}_p^{(\ell)}\right)^{-1}\\
&\vdots\\
&= \boldsymbol{A}^\ell\boldsymbol{W}_p^{(1)}\left(\boldsymbol{R}_p^{(2)}\right)^{-1}\cdots\left(\boldsymbol{R}_p^{(\ell-1)}\right)^{-1}\left(\boldsymbol{R}_p^{(\ell)}\right)^{-1}\\
&= \boldsymbol{A}^\ell\boldsymbol{A}\boldsymbol{V}_p\left(\boldsymbol{R}_p^{(1)}\right)^{-1}\left(\boldsymbol{R}_p^{(2)}\right)^{-1}\cdots\left(\boldsymbol{R}_p^{(\ell-1)}\right)^{-1}\left(\boldsymbol{R}_p^{(\ell)}\right)^{-1}\\
&= \boldsymbol{A}^{\ell-1}\boldsymbol{A}\boldsymbol{V}_{p+1}\boldsymbol{Q}_{p+1,p}^{(1)}\left(\boldsymbol{R}_p^{(2)}\right)^{-1}\cdots\left(\boldsymbol{R}_p^{(\ell-1)}\right)^{-1}\left(\boldsymbol{R}_p^{(\ell)}\right)^{-1}\\
&\vdots\\
&= \boldsymbol{V}_{\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}.
\end{aligned}$$

Since both matrices $\boldsymbol{V}_{\ell+p}$ and $\boldsymbol{Q}_{\ell+p,p}^{(\ell)}$ are orthogonal, $\boldsymbol{V}_{\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}$ is of full rank. Therefore, so is $\boldsymbol{W}_p^{(\ell)}$.

To prove the second part, we first note that for any vector $\boldsymbol{y} \in \mathbb{R}^p$,

$$\begin{aligned}
\left\|\boldsymbol{H}_{\ell+p+1,\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}\boldsymbol{y} - \left\|\boldsymbol{b}^\delta\right\|\boldsymbol{e}_1\right\| &= \left\|\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y} - \left\|\boldsymbol{b}^\delta\right\|\left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\right)^T\boldsymbol{e}_1\right\|\\
\text{(2.11)} \qquad &= \left\|\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y} - \boldsymbol{g}_{\ell+p+1}\right\|\\
&= \left\|\boldsymbol{R}_p^{(\ell+1)}\boldsymbol{y} - \bar{\boldsymbol{g}}_p\right\| + \left\|[0\ \ldots\ 0\ \gamma_{p+1}\ \ldots\ \gamma_{\ell+p+1}]^T\right\|.
\end{aligned}$$

The minimum of the left-hand side is reached when the first term of the right-hand side of (2.11) is zero. Since $\boldsymbol{R}_p^{(\ell+1)}$ is nonsingular, this will occur when $\boldsymbol{y}_p^{(\ell)} = \left(\boldsymbol{R}_p^{(\ell+1)}\right)^{-1}\bar{\boldsymbol{g}}_p$.

To show the final part of the proposition, we observe that for any $\boldsymbol{x}_p^{(\ell)} = \boldsymbol{W}_p^{(\ell)}\boldsymbol{y}_p^{(\ell)}$,

$$\begin{aligned}
\boldsymbol{A}\boldsymbol{x}_p^{(\ell)} - \boldsymbol{b}^\delta &= \boldsymbol{V}_{\ell+p+1}\left(\boldsymbol{H}_{\ell+p+1,\ell+p}\boldsymbol{Q}_{\ell+p,p}^{(\ell)}\boldsymbol{y} - \boldsymbol{e}_1\left\|\boldsymbol{b}^\delta\right\|\right)\\
\text{(2.12)} \qquad &= \boldsymbol{V}_{\ell+p+1}\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\right)^T\left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y} - \left\|\boldsymbol{b}^\delta\right\|\boldsymbol{e}_1\right)\\
&= \boldsymbol{V}_{\ell+p+1}\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}\left(\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y} - \boldsymbol{g}_{\ell+p+1}\right).
\end{aligned}$$

From the proof of the second part above, the norm of $\boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)}\boldsymbol{y} - \boldsymbol{g}_{\ell+p+1}$ in (2.12) is minimized when $\boldsymbol{y}$ annihilates the first p components of $\boldsymbol{g}_{\ell+p+1}$. What remains are the last $\ell + 1$ entries, and as a result

$$(2.13) \qquad \boldsymbol{A}\boldsymbol{x}_p^{(\ell)} - \boldsymbol{b}^\delta = \boldsymbol{V}_{\ell+p+1}\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}[0,\ldots,0,\gamma_{p+1},\ldots,\gamma_{\ell+p+1}]^T.$$

The result now follows from the orthogonality of the columns of $\boldsymbol{V}_{\ell+p+1}\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)}$. Since there are p leading zeros in the vector (2.13), one only needs to consider the $(p+1)^{st}$ entry to the last:

$$\left\|\boldsymbol{A}\boldsymbol{x}_p^{(\ell)} - \boldsymbol{b}^\delta\right\| = \left\|[0,\ldots,0,\gamma_{p+1},\ldots,\gamma_{\ell+p+1}]^T\right\| = \sqrt{\sum_{i=p+1}^{\ell+p+1}(\gamma_i)^2}. \qquad \square$$

By the third part of Proposition 2.1, the norm of the pth residual in line **11** of Algorithm **2** may be computed without an additional matrix-vector product evaluation with $\boldsymbol{A}$ or the computation of $\boldsymbol{x}_p^{(\ell)}$ at each step. We include this strategy in our implementation of the ℓ-shifted GMRES method. Lines **10** and **11** in Algorithm **2** are presented for clarity only.

We conclude this section with a comment on the storage requirement of Algorithm **2** and note that the matrix $\boldsymbol{W}_p^{(\ell)}$ in the derivation of the ℓ-shifted GMRES method does not have to be stored. Instead, we only store the penultimate $\boldsymbol{Q}$-matrix from the QR factorizations of iteration p from lines **5** and **7**. The orthonormal matrix $\boldsymbol{V}_{\ell+p}$ can be determined from the first $\ell + p$ columns of the already stored matrix $\boldsymbol{V}_{\ell+p+1}$ from the Arnoldi process. Hence, the only additional storage requirement for iteration p of the ℓ-shifted GMRES method over standard GMRES is an $(\ell + p) \times p$ orthogonal matrix, which is usually not very large in practice (see Section **5**).

We finally remark that we will not consider breakdown and restarting of ℓ-shifted methods in this work, as applications to linear discrete ill-posed problems typically only require a few iterations of the Arnoldi process to terminate according to the discrepancy principle. When only a small number of iterations are carried out, reorthogonalization and restarting are typically unnecessary. Indeed, we have not observed the need to carry out a large number of iterations or the occurrence of breakdown in our numerical experiments.

3. The ℓ-shifted block GMRES method. This section reviews the BGMRES method and the structure of the block Arnoldi decomposition. We then derive the pth iterate, $\boldsymbol{X}_p^{(1)}$, computed by the 1-shifted BGMRES algorithm following the same line of reasoning as in Section 2. The ℓ-shifted BGMRES method for $\ell = 2, 3, \ldots, \ell_{\max}$ also is discussed. Subsequently we consider storage requirements and comment on a few computational aspects.

Our aim is to determine an approximate solution of the linear system of equations

$$(3.1) \qquad\qquad\qquad\qquad \boldsymbol{A}\boldsymbol{X} = \boldsymbol{B}^\delta,$$

where $\boldsymbol{X}, \boldsymbol{B}^\delta \in \mathbb{R}^{n \times k}$ and $2 \le k \ll n$. This system differs from (1.1) only in that the right-hand side is a block vector with k columns. These columns are assume to be contaminated by errors. Problems of this kind arise, e.g., in the restoration of color images. In this application $k = 3$; see Section 5.

The BGMRES method determines an approximate solution of (3.1) by computing an approximation of the solution of the minimization problem

$$\min_{\boldsymbol{X} \in \mathbb{R}^{n \times k}} \left\|\boldsymbol{A}\boldsymbol{X} - \boldsymbol{B}^\delta\right\|.$$

Here $\| \cdot \|$ denotes the Frobenius norm. The pth iterate, $\boldsymbol{X}_p$, computed by BGMRES satisfies

$$\left\| \boldsymbol{A}\boldsymbol{X}_p - \boldsymbol{B}^\delta \right\| = \min_{\boldsymbol{X} \in \mathbb{K}_p(\boldsymbol{A},\boldsymbol{B}^\delta)} \left\| \boldsymbol{A}\boldsymbol{X} - \boldsymbol{B}^\delta \right\|,$$

where

$$\mathbb{K}_p(\boldsymbol{A}, \boldsymbol{B}^\delta) = \text{block span}\{\boldsymbol{B}^\delta, \boldsymbol{A}\boldsymbol{B}^\delta, \dots, \boldsymbol{A}^{p-1}\boldsymbol{B}^\delta\}$$
$$= \Big\{ \boldsymbol{X} \in \mathbb{R}^{n \times k} : \boldsymbol{X} = \sum_{i=0}^{p-1} \boldsymbol{A}^i \boldsymbol{B}^\delta \Omega_i, \ \Omega_i \in \mathbb{R}^{k \times k}, \ \text{for } i = 0, 1, \dots, p-1 \Big\}$$

defines a block Krylov subspace of order p; see [14]. Similarly as above, we assume that p is small enough so that $\dim \left(\mathbb{K}_p \left(\boldsymbol{A}, \boldsymbol{B}^\delta \right) \right) = pk$. For full discussions of the BGMRES method, we direct the interested reader to [34] and [36].

The computations are initiated by computing the QR factorization $\boldsymbol{B}^\delta = \hat{\boldsymbol{Q}}\hat{\boldsymbol{R}}$ with $\hat{\boldsymbol{Q}} \in \mathbb{R}^{n \times k}$ having orthonormal columns and $\hat{\boldsymbol{R}} \in \mathbb{R}^{k \times k}$ being upper triangular. If the columns of $\boldsymbol{B}^\delta$ are linearly dependent, then we reduce k so that the range of the so obtained matrix $\hat{\boldsymbol{Q}} \in \mathbb{R}^{n \times k'}$, with $k' < k$, agrees with the range of $\boldsymbol{B}^\delta$. We will for notational simplicity assume that $k' = k$. The pth step of the block Arnoldi iteration applied to the matrix $\boldsymbol{A}$ and initial matrix $\boldsymbol{V}_1 = \hat{\boldsymbol{Q}}$ gives the block Arnoldi decomposition

$$(3.2) \qquad\qquad \boldsymbol{A}\boldsymbol{V}_{pk} = \boldsymbol{V}_{(p+1)k}\boldsymbol{H}_{(p+1)k,pk},$$

where $\boldsymbol{V}_{(p+1)k} \in \mathbb{R}^{n \times (p+1)k}$ has orthonormal columns and $\boldsymbol{H}_{(p+1)k,pk} \in \mathbb{R}^{(p+1)k \times pk}$ is of upper block Hessenberg form with p sub-diagonal blocks of size $k \times k$. We have $\boldsymbol{V}_{pk} = [\boldsymbol{V}_1, \boldsymbol{V}_2, \dots, \boldsymbol{V}_p]$ with each $\boldsymbol{V}_i \in \mathbb{R}^{n \times k}$ having orthonormal columns. Moreover,

$$\boldsymbol{V}_i^T \boldsymbol{V}_j = \begin{cases} \boldsymbol{I}_k & \text{for} \quad i = j, \\ \boldsymbol{O}_k & \text{for} \quad i \neq j, \end{cases}$$

where $\boldsymbol{I}_k$ and $\boldsymbol{O}_k$ denote the identity and zero matrices of order k. The range of the matrix $\boldsymbol{V}_{(p+1)k}$ is the block Krylov subspace $\mathbb{K}_{p+1}(\boldsymbol{A}, \boldsymbol{B}^\delta)$ under the assumption that all upper triangular matrices generated by Algorithm 3 are nonsingular. This algorithm is the foundation for the BGMRES method; see [34].

We first derive the 1-shifted BGMRES method. Introduce the QR factorization of the upper block Hessenberg matrix $\boldsymbol{H}_{(p+1)k,pk}$ from (3.2),

$$\boldsymbol{H}_{(p+1)k,pk} = \boldsymbol{Q}_{(p+1)k}^{(1)} \boldsymbol{R}_{(p+1)k,pk}^{(1)},$$

where the matrix $\boldsymbol{Q}_{(p+1)k}^{(1)} \in \mathbb{R}^{(p+1)k \times (p+1)k}$ is orthogonal and $\boldsymbol{R}_{(p+1)k,pk}^{(1)} \in \mathbb{R}^{(p+1)k \times pk}$ has a leading $pk \times pk$ upper triangular submatrix denoted by $\boldsymbol{R}_{pk}^{(1)}$ and with vanishing last k rows. Because $\boldsymbol{H}_{(p+1)k,pk}$ is upper block Hessenberg, the matrix $\boldsymbol{Q}_{(p+1)k}^{(1)}$ can be expressed as a product of pk elementary reflectors

$$(3.3) \qquad\qquad \boldsymbol{Q}_{(p+1)k}^{(1)} = \boldsymbol{G}_1 \boldsymbol{G}_2 \dots \boldsymbol{G}_{pk},$$

where $\boldsymbol{G}_j \in \mathbb{R}^{(p+1)k \times (p+1)k}$, for $j = 1, 2, \dots, pk$, is a symmetric elementary reflector in the planes j and $j + 1$. Thus, (3.3) shows that $\boldsymbol{Q}_{(p+1)k}^{(1)}$ is upper block Hessenberg.

Algorithm 3: Block Arnoldi process.

    Input: $A \in \mathbb{R}^{n \times n}$ and $B^\delta \in \mathbb{R}^{n \times k}$
    Output: $V_{(j+1)k} \in \mathbb{R}^{n \times (j+1)k}$ and $H_{(j+1)k,jk} \in \mathbb{R}^{(j+1)k \times jk}$

1 Compute QR factorization $\left[\hat{Q}, \hat{R} \right] = B^\delta$;

2 Set $V_1 = \hat{Q}$;

3 **for** $j = 1, 2, \ldots, n$ **do**

4   Compute $W_j = A V_j$;

5   **for** $i = 1, 2, \ldots, j$ **do**

6         $H_{i,j} = V_i^T W_j$;

7         $W_j = W_j - V_i H_{i,j}$;

8   **end**

9   Compute QR factorization $[V_{j+1}, H_{j+1,j}] = W_j$;

10 **end**

Consider the reduced QR factorization $H_{(p+1)k,pk} = Q^{(1)}_{(p+1)k,pk} R^{(1)}_{pk}$, where the matrix $Q^{(1)}_{(p+1)k,pk} \in \mathbb{R}^{(p+1)k \times pk}$ consists of the first pk columns of $Q^{(1)}_{(p+1)k}$. Define the matrix $W^{(1)}_p = V_{(p+1)k} Q^{(1)}_{(p+1)k,pk} \in \mathbb{R}^{n \times pk}$. Assuming that $R^{(1)}_{pk}$ is nonsingular, it follows from (3.2) and (3.3) that

$$W^{(1)}_p = A V_{pk} \left(R^{(1)}_{pk} \right)^{-1},$$

which shows that $\mathcal{R}\left(W^{(1)}_p \right) = \mathbb{K}\left(A, AB^\delta \right)$. The 1-shifted minimization problem

$$\left\| A X^{(1)}_p - B^\delta \right\| = \min_{X \in \mathbb{K}_p(A, AB^\delta)} \left\| AX - B^\delta \right\|$$

may be written as

$$
\begin{aligned}
\min_{Y \in \mathbb{R}^{pk \times k}} \left\| A W^{(1)}_p Y - B^\delta \right\| &= \min_{Y \in \mathbb{R}^{pk \times k}} \left\| A V_{(p+1)k} Q^{(1)}_{(p+1)k,pk} Y - B^\delta \right\| \\
(3.4) \qquad\qquad &= \min_{Y \in \mathbb{R}^{pk \times k}} \left\| V_{(p+2)k} H_{(p+2)k,(p+1)k} Q^{(1)}_{(p+1)k,pk} Y - B^\delta \right\| \\
&= \min_{Y \in \mathbb{R}^{pk \times k}} \left\| H_{(p+2)k,(p+1)k} Q^{(1)}_{(p+1)k,pk} Y - E_1 \hat{R} \right\|,
\end{aligned}
$$

where $E_1 \in \mathbb{R}^{(p+2)k \times k}$ consists of the first k columns of the identity matrix $I_{(p+2)k}$. The last equality follows from the fact that the leading $n \times k$ submatrix of $V_{(p+2)k}$ is $\hat{Q}$, which is obtained from the initial QR factorization of B^δ and used in the block Arnoldi algorithm.

Since both matrices $H_{(p+2)k,(p+1)k}$ and $Q^{(1)}_{(p+1)k,pk}$ are upper block Hessenberg with $k \times k$ blocks, their product has two $k \times k$ subdiagonal blocks. Therefore, the QR factorization

$$H_{(p+2)k,(p+1)k} Q^{(1)}_{(p+1)k,pk} = Q^{(2)}_{(p+2)k} R^{(2)}_{(p+2)k,pk}$$

can be computed in $O\left((pk)^2 \right)$ flops using elementary reflectors. In this identity the matrix $Q^{(2)}_{(p+2)k} \in \mathbb{R}^{(p+2)k \times (p+2)k}$ is orthogonal with zero entries below the second subdiagonal

$k \times k$ block diagonals and $\boldsymbol{R}_{(p+2)k,pk}^{(2)} \in \mathbb{R}^{(p+2)k \times pk}$ has a leading $pk \times pk$ upper triangular submatrix, denoted by $\boldsymbol{R}_{pk}^{(2)} \in \mathbb{R}^{pk \times pk}$, with the two last block rows vanishing. This gives

$$
\begin{aligned}
(3.5) \qquad \boldsymbol{A}\boldsymbol{W}_p^{(1)} &= \boldsymbol{A}^2 \boldsymbol{V}_{pk} \left(\boldsymbol{R}_{pk}^{(1)} \right)^{-1} = \boldsymbol{A}\boldsymbol{V}_{(p+1)k}\boldsymbol{Q}_{(p+1)k,pk}^{(1)} \\
&= \boldsymbol{V}_{(p+2)k}\boldsymbol{H}_{(p+2)k,(p+1)k}\boldsymbol{Q}_{(p+1)k,pk}^{(1)} = \boldsymbol{V}_{(p+2)k}\boldsymbol{Q}_{(p+2)k}^{(2)}\boldsymbol{R}_{(p+2)k,pk}^{(2)}.
\end{aligned}
$$

Hence, the minimization problem (3.4) may be written as

$$
(3.6) \qquad \min_{\boldsymbol{Y} \in \mathbb{R}^{pk \times k}} \left\| \boldsymbol{A}\boldsymbol{W}_p^{(1)}\boldsymbol{Y} - \boldsymbol{B}^\delta \right\| = \min_{\boldsymbol{Y} \in \mathbb{R}^{pk \times k}} \left\| \boldsymbol{R}_{(p+2)k,pk}^{(2)}\boldsymbol{Y} - \left(\boldsymbol{Q}_{(p+2)k}^{(2)} \right)^T \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\|.
$$

Denote the solution of (3.6) by $\boldsymbol{Y}_p^{(1)}$. Then the approximate solution of (3.1) determined by the 1-shifted BGMRES method is given by $\boldsymbol{X}_p^{(1)} = \boldsymbol{W}_p^{(1)}\boldsymbol{Y}_p^{(1)}$.

Following the same general path as in Section 2 and in the same manner as above for $\ell = 1$, we may derive the appropriate matrices $\boldsymbol{W}_p^{(\ell)} \in \mathbb{R}^{n \times pk}$ with orthonormal columns such that $\mathcal{R}\left(\boldsymbol{W}_p^{(\ell)} \right) = \mathbb{K}\left(\boldsymbol{A}, \boldsymbol{A}^\ell \boldsymbol{B}^\delta \right)$, for $\ell = 2, 3, \ldots, \ell_{\max}$. These matrices are used in the formulation of the ℓ-shifted BGMRES minimization problem

$$
\begin{aligned}
(3.7) \qquad \min_{\boldsymbol{Y} \in \mathbb{R}^{pk \times k}} \left\| \boldsymbol{A}\boldsymbol{W}_p^{(\ell)}\boldsymbol{Y} - \boldsymbol{B}^\delta \right\| &= \min_{\boldsymbol{Y} \in \mathbb{R}^{pk \times k}} \left\| \boldsymbol{H}_{(\ell+p+1)k,(\ell+p)k}\boldsymbol{Q}_{(\ell+p)k,pk}^{(\ell)}\boldsymbol{Y} - \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\| \\
&= \min_{\boldsymbol{Y} \in \mathbb{R}^{pk \times k}} \left\| \boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}\boldsymbol{Y} - \left(\boldsymbol{Q}_{(\ell+p+1)k}^{(\ell+1)} \right)^T \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\|
\end{aligned}
$$

with $\boldsymbol{E}_1 \in \mathbb{R}^{(\ell+p+1)k \times k}$. Denote the solution of (3.7) by $\boldsymbol{Y}_p^{(\ell)}$. Then the solution of the ℓ-shifted BGMRES minimization problem is given by $\boldsymbol{X}_p^{(\ell)} = \boldsymbol{W}_p^{(\ell)}\boldsymbol{Y}_p^{(\ell)}$. Algorithm 4 provides the details of the computations. The iterations are terminated with the discrepancy principle.

The computation of $\boldsymbol{X}_p^{(\ell)}$ requires $(p + \ell)k$ matrix-vector product evaluations with $\boldsymbol{A}$. This constitutes the dominant computational work for large problems. We remark that on many computers the evaluation of one matrix-block-vector product with a block vector with k columns is much faster than the evaluation of k matrix-vector products. Therefore the count of $(p + \ell)k$ matrix-vector product evaluations generally is not an accurate measure of the computing time required for the evaluation of $p + \ell$ matrix-block-vector products with block vectors with k columns.

We briefly comment on a few computational aspects of the ℓ-shifted BGMRES algorithm. The block Arnoldi iteration steps in lines **3** and **5** require a total of $(\ell + p)k$ matrix-vector products. These matrix-vector products are the dominant computational work when the matrix $\boldsymbol{A}$ is large. Algorithm 4 evaluates QR factorizations in lines **6** and **8**. In the pth iteration of the algorithm, the flop count for each QR factorization in line **8** is $O\left((pk)^2\right)$ when using elementary reflectors. Since the lines **7**–**9** are executed ℓ times in iteration p, the total flop count required for evaluating these QR factorizations of the upper block Hessenberg-type matrices is $O\left(\ell(pk)^2\right)$. Application of structure-ignoring QR factorizations would increase the flop count to $O\left(\ell(pk)^3\right)$.

Algorithm 4 requires the computation of the norm of the pth residual in line **12**. The explicit evaluation of the matrix-matrix product $\boldsymbol{A}\boldsymbol{X}_p^{(\ell)}$ can be avoided as follows. First, observe that each matrix $\boldsymbol{Q}_{(j+p+1)k}^{(j+1)}$, for $j = 0, 1, \ldots, \ell$, can be written as a product of $(j + p)k$ elementary reflectors. Therefore, the second term on the right-hand side of (3.7) can

---

Algorithm 4: ℓ-shifted BGMRES ($\ell \geq 1$) with discrepancy principle.

Input: $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, $\boldsymbol{B}^\delta \in \mathbb{R}^{n \times k}$, and $\ell \in \{1, 2, 3, \dots\}$
Output: $\boldsymbol{X}_p^{(\ell)} \in \mathbb{R}^{n \times k}$

1   Compute QR factorization: $\left[\hat{\boldsymbol{Q}}, \hat{\boldsymbol{R}}\right] = \boldsymbol{B}^\delta$;

2   Set $\boldsymbol{V}_1 = \hat{\boldsymbol{Q}}$ and $\boldsymbol{X}_0^{(\ell)} = \boldsymbol{0}$;

3   Compute ℓ steps of block Arnoldi: $\boldsymbol{AV}_{\ell k} = \boldsymbol{V}_{(\ell+1)k} \boldsymbol{H}_{(\ell+1)k,\ell k}$;

4   **for** $p = 1, 2, \dots$ **do**

5   Compute next block Arnoldi step: $\boldsymbol{AV}_{(\ell+p)k} = \boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{H}_{(\ell+p+1)k,(\ell+p)k}$;

6   Compute QR factorization: $\left[\boldsymbol{Q}_{(p+1)k}^{(1)}, \boldsymbol{R}_{(p+1)k,pk}^{(1)}\right] = \boldsymbol{H}_{(p+1)k,pk}$;

7   **for** $j = 1, 2, \dots, \ell$ **do**

8   Compute QR factorization:
        $\left[\boldsymbol{Q}_{(j+p+1)k}^{(j+1)}, \boldsymbol{R}_{(j+p+1)k,pk}^{(j+1)}\right] = \boldsymbol{H}_{(j+p+1)k,(j+p)k} \boldsymbol{Q}_{(j+p)k,pk}^{(j)}$;

9   **end**

10   Compute $\boldsymbol{Y}_p^{(\ell)}$ as the minimizer of $\left\| \boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)} \boldsymbol{Y} - \left(\boldsymbol{Q}_{(\ell+p+1)k}^{(\ell+1)}\right)^T \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\|$;

11   Compute $\boldsymbol{X}_p^{(\ell)} = \boldsymbol{V}_{(\ell+p)k} \boldsymbol{Q}_{(\ell+p)k,pk}^{(\ell)} \boldsymbol{Y}_p^{(\ell)}$;

12   Compute $\|\boldsymbol{r}_p\| = \left\| \boldsymbol{AX}_p^{(\ell)} - \boldsymbol{B}^\delta \right\|$;

13   **if** $\|\boldsymbol{r}_p\| \leq \tau\delta$ **then**

14   Stop;

15   **end**

16 **end**

---

be expressed as

$$\left(\boldsymbol{Q}_{(\ell+p+1)k}^{(\ell+1)}\right)^T \boldsymbol{E}_1 \hat{\boldsymbol{R}} = \boldsymbol{G}_{(\ell+p)k}^T \dots \boldsymbol{G}_3^T \boldsymbol{G}_2^T \boldsymbol{G}_1^T \, \boldsymbol{E}_1 \hat{\boldsymbol{R}}$$

(3.8)
$$= \begin{bmatrix} u_{1,1} & \cdots & u_{1,k} \\ \vdots & \ddots & \vdots \\ u_{pk,1} & \cdots & u_{pk,k} \\ \hline u_{pk+1,1} & \cdots & u_{pk+1,k} \\ \vdots & \ddots & \vdots \\ u_{(\ell+p+1)k,1} & \cdots & u_{(\ell+p+1)k,k} \end{bmatrix} =: \boldsymbol{U}_{(\ell+p+1)k,k},$$

where $\boldsymbol{U}_{(\ell+p+1)k,k} \in \mathbb{R}^{(\ell+p+1)k \times k}$ and $\boldsymbol{G}_i^T \in \mathbb{R}^{(\ell+p+1)k \times (\ell+p+1)k}$, for $i = 1, 2, \dots, (\ell+p)k$.

PROPOSITION 3.1. *At the pth iteration of Algorithm 4, let $\boldsymbol{G}_i$, for $i = 1, 2, \dots, (\ell+p)k$, be the reflector matrices used to transform $\boldsymbol{H}_{(j+p+1)k,(j+p)k} \boldsymbol{Q}_{(j+p)k,pk}^{(j)}$ in line 8 when $j = \ell$ into upper triangular form $\boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}$, and define $\boldsymbol{U}_{(\ell+p+1)k,k}$ as was done in (3.8). Let $\boldsymbol{R}_{pk}^{(\ell+1)}$ denote the leading $pk \times pk$ upper block triangular matrix obtained from $\boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}$ by deleting its last $(\ell+1)k$ rows, and let $\bar{\boldsymbol{U}}$ be the $pk \times k$ matrix obtained from $\boldsymbol{U}_{(\ell+p+1)k,k}$ by deleting the last $(\ell+1)k$ rows. Then:*

1. The rank of $\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}$ is equal to the rank of $\boldsymbol{R}_{pk}^{(\ell+1)}$. In particular, if $r_{(pk,pk)}^{(\ell+1)}$, the last diagonal entry of $\boldsymbol{R}_{pk}^{(\ell+1)}$, vanishes, then $\boldsymbol{A}$ is singular.

2. The matrix $\boldsymbol{Y}_p^{(\ell)} \in \mathbb{R}^{pk \times k}$, which minimizes

$$\left\| \boldsymbol{H}_{(\ell+p+1)k,(\ell+p)k} \boldsymbol{Q}_{(\ell+p)k,pk}^{(\ell)} \boldsymbol{Y} - \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\|,$$

is given by

$$\boldsymbol{Y}_p^{(\ell)} = \left(\boldsymbol{R}_{pk}^{(\ell+1)} \right)^{-1} \bar{\boldsymbol{U}}_{pk}.$$

3. The residual matrix $\boldsymbol{r}_p := \boldsymbol{A}\boldsymbol{X}_p^{(\ell)} - \boldsymbol{B}^\delta$ in iteration p of Algorithm 4 satisfies

$$\boldsymbol{r}_p = \boldsymbol{V}_{(\ell+p+1)k} \left(\boldsymbol{H}_{(\ell+p+1)k,(\ell+p)k} \boldsymbol{Q}_{(\ell+p)k,pk}^{(\ell)} \boldsymbol{Y} - \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right)$$

$$= \boldsymbol{V}_{(\ell+p+1)k} \left(\boldsymbol{Q}_{(\ell+p+1)k}^{(\ell+1)} \right)^T \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \\ \hline u_{pk+1,1} & \cdots & u_{pk+1,k} \\ \vdots & \ddots & \vdots \\ u_{(\ell+p+1)k,1} & \cdots & u_{(\ell+p+1)k,k} \end{bmatrix}.$$

Therefore,

$$\|\boldsymbol{r}_p\| = \left\| \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \\ \hline u_{pk+1,1} & \cdots & u_{pk+1,k} \\ \vdots & \ddots & \vdots \\ u_{(\ell+p+1)k,1} & \cdots & u_{(\ell+p+1)k,k} \end{bmatrix} \right\|.$$

Proof. The first part can be shown similarly as the analogous result in Proposition 2.1. The generalization of (3.5) for $\ell \geq 1$ can be written as

$$\boldsymbol{A}\boldsymbol{W}_p^{(\ell)} = \boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{Q}_{(\ell+p+1)k}^{(\ell+1)} \boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}.$$

Since $\boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{Q}_{(\ell+p+1)k}^{(\ell+1)}$ has orthonormal columns, the matrices $\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}$ and $\boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}$ have the same rank. If the entry $r_{(pk,pk)}^{(\ell+1)}$ of $\boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}$ vanishes, then $\boldsymbol{R}_{(\ell+p+1)k,pk}^{(\ell+1)}$ is of rank $\leq pk - 1$, and, as a result, $\boldsymbol{A}\boldsymbol{W}_p^{(\ell)}$ also has rank $\leq pk - 1$. Since $\boldsymbol{W}_p^{(\ell)}$ is of full rank, the matrix $\boldsymbol{A}$ is singular.

For the second part, we note that for any matrix $\boldsymbol{Y} \in \mathbb{R}^{pk \times k}$,

$$
\left\| \boldsymbol{H}_{(\ell+p+1)k,(\ell+p)k} \boldsymbol{Q}^{(\ell)}_{(\ell+p)k,pk} \boldsymbol{Y} - \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\|
$$

$$
= \left\| \boldsymbol{R}^{(\ell+1)}_{(\ell+p+1)k,pk} \boldsymbol{Y} - \left(\boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k} \right)^T \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right\|
$$

$$
= \left\| \boldsymbol{R}^{(\ell+1)}_{(\ell+p+1)k,pk} \boldsymbol{Y} - \boldsymbol{U}_{(\ell+p+1)k,k} \right\|
$$

(3.9)

$$
= \left\| \boldsymbol{R}^{(\ell+1)}_{pk} \boldsymbol{Y} - \bar{\boldsymbol{U}}_{pk} \right\| + \left\| \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \\ \hline u_{pk+1,1} & \cdots & u_{pk+1,k} \\ \vdots & \ddots & \vdots \\ u_{(\ell+p+1)k,1} & \cdots & u_{(\ell+p+1)k,k} \end{bmatrix} \right\|.
$$

The minimum of the left-hand side is achieved when the first term on the right-hand side of (3.9) is zero. Since $\boldsymbol{R}^{(\ell+1)}_{pk}$ is nonsingular, this will occur when $\boldsymbol{Y}^{(\ell)}_p = \left(\boldsymbol{R}^{(\ell+1)}_{pk} \right)^{-1} \bar{\boldsymbol{U}}_{pk}$.

To show the last part we note that for any $\boldsymbol{X}^{(\ell)}_p = \boldsymbol{W}^{(\ell)}_p \boldsymbol{Y}^{(\ell)}_p$,

$$
\boldsymbol{A} \boldsymbol{X}^{(\ell)}_p - \boldsymbol{B}^\delta = \boldsymbol{V}_{(\ell+p+1)k} \left(\boldsymbol{H}_{(\ell+p+1)k,(\ell+p)k} \boldsymbol{Q}^{(\ell)}_{(\ell+p)k,pk} \boldsymbol{Y} - \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right)
$$

$$
= \boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k} \left(\boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k} \right)^T \left(\boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k} \boldsymbol{R}^{(\ell+1)}_{(\ell+p+1)k,pk} \boldsymbol{Y} - \boldsymbol{E}_1 \hat{\boldsymbol{R}} \right)
$$

$$
= \boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k} \left(\boldsymbol{R}^{(\ell+1)}_{(\ell+p+1)k,pk} \boldsymbol{Y} - \boldsymbol{U}_{(\ell+p+1)k,k} \right).
$$

The second part above shows that the norm of $\boldsymbol{R}^{(\ell+1)}_{(\ell+p+1)k,pk} \boldsymbol{Y} - \boldsymbol{U}_{(\ell+p+1)k,k}$ is minimized when $\boldsymbol{Y}$ annihilates the upper $pk \times k$ block of $\boldsymbol{U}_{(\ell+p+1)k,k}$. What remains is the lower $(\ell+1)k \times k$ block entries of $\boldsymbol{U}_{(\ell+p+1)k,k}$, which we, for simplicity, denote by $\tilde{\boldsymbol{U}}_{(\ell+1)k,k}$. This gives

$$
\boldsymbol{A} \boldsymbol{X}^{(\ell)}_p - \boldsymbol{B}^\delta = \boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k} \begin{bmatrix} \boldsymbol{0} \\ \tilde{\boldsymbol{U}}_{(\ell+1)k,k} \end{bmatrix}.
$$

The result now follows from the observation that the matrix $\boldsymbol{V}_{(\ell+p+1)k} \boldsymbol{Q}^{(\ell+1)}_{(\ell+p+1)k}$ has orthonormal columns. We obtain

(3.10)
$$
\|\boldsymbol{r}_p\| = \left\| \tilde{\boldsymbol{U}}_{(\ell+1)k,k} \right\|. \qquad \square
$$

Formula (3.10) allows us to evaluate the norm of the pth residual in line **12** of Algorithm **4** without computing an additional matrix-block-vector product with the matrix $\boldsymbol{A}$ or computing $\boldsymbol{X}^{(\ell)}_p$. Regarding the storage requirement of Algorithm 4, we note that the matrix $\boldsymbol{W}^{(\ell)}_p$ does not have to be stored. Instead, we only store the second to last $\boldsymbol{Q}$-matrix (i.e., $\boldsymbol{Q}^{(\ell)}_{(\ell+p)k}$) from the QR factorizations of iteration p from lines **6** and **8**. Thus, the only additional storage requirement for iteration p of the ℓ-shifted BGMRES method over the standard BGMRES method is a $(\ell+p)k \times pk$ orthogonal matrix, which usually is not large.

4. The ℓ-shifted global GMRES method. An alternative approach for solving the block linear system (3.1) is based on the global Arnoldi process. This process requires less storage than Algorithm 3 for the same number of iterations. The global Arnoldi process and the associated global GMRES (gl-GMRES) method were introduced by Jbilou et al. [25, 26]. Among the advantages of the global Arnoldi process, when compared with the standard block Arnoldi process of Section 3 with the same block size, is that the former does not require a special handling of the linear dependence of the vectors in a block; see Baglama et al. [1] for a discussion of linear dependence in the context of the (standard) block Lanczos method. Moreover, the computations are simpler with the global Arnoldi process. A comparison of the convergence properties is provided by Frommer et al. [16].

We begin by defining some notation necessary to discuss the global Arnoldi process and the gl-GMRES method. We then derive the ℓ-shifted gl-GMRES algorithm and discuss some of its computational aspects.

The Kronecker product of two matrices $\boldsymbol{G} = [g_{i,j}] \in \mathbb{R}^{n \times n}$ and $\boldsymbol{H} \in \mathbb{R}^{p \times p}$ is defined as

$$
\boldsymbol{G} \otimes \boldsymbol{H} := \begin{bmatrix} g_{1,1}\boldsymbol{H} & g_{1,2}\boldsymbol{H} & \cdots & g_{1,n}\boldsymbol{H} \\ g_{2,1}\boldsymbol{H} & g_{2,2}\boldsymbol{H} & \cdots & g_{2,n}\boldsymbol{H} \\ \vdots & \vdots & & \vdots \\ g_{n,1}\boldsymbol{H} & g_{n,2}\boldsymbol{H} & \cdots & g_{n,n}\boldsymbol{H} \end{bmatrix} \in \mathbb{R}^{np \times np}.
$$

For general matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ of appropriate sizes, we have

$$
(\boldsymbol{AC}) \otimes (\boldsymbol{BD}) = (\boldsymbol{A} \otimes \boldsymbol{B})(\boldsymbol{C} \otimes \boldsymbol{D}),
$$

where as an immediate consequence

$$
(4.1) \qquad \boldsymbol{A} \otimes \boldsymbol{B} = (\boldsymbol{I} \otimes \boldsymbol{B})(\boldsymbol{A} \otimes \boldsymbol{I}) = (\boldsymbol{A} \otimes \boldsymbol{I})(\boldsymbol{I} \otimes \boldsymbol{B}),
$$

for suitably sized identity matrices $\boldsymbol{I}$. We let $\mathrm{vec}(\cdot)$ denote the operation which transforms a general matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ to a vector $\boldsymbol{a} \in \mathbb{R}^{mn}$ by stacking the columns of $\boldsymbol{A}$ from left to right. We also define the inner product

$$
(4.2) \qquad \langle \boldsymbol{A}, \boldsymbol{B} \rangle_F := \mathrm{tr}(\boldsymbol{A}^T \boldsymbol{B}) = (\mathrm{vec}(\boldsymbol{A}))^T \mathrm{vec}(\boldsymbol{B}),
$$

where $\mathrm{tr}(\cdot)$ denotes the trace.

Let $\boldsymbol{a} = [a_1, a_2, \ldots, a_p]^T \in \mathbb{R}^p$, and introduce the product

$$
\boldsymbol{V}_{pk} * \boldsymbol{a} = \sum_{i=1}^{p} a_i \boldsymbol{V}_i,
$$

where we recall from Section 3 that $\boldsymbol{V}_{pk} = [\boldsymbol{V}_1, \boldsymbol{V}_2, \ldots, \boldsymbol{V}_p] \in \mathbb{R}^{n \times pk}$ and $\boldsymbol{V}_i \in \mathbb{R}^{n \times k}$, for $i = 1, 2, \ldots, p$. We have

$$
(4.3) \qquad \boldsymbol{V}_{pk} * \boldsymbol{H} = [\boldsymbol{V}_{pk} * \boldsymbol{H}_{:,1}, \boldsymbol{V}_{pk} * \boldsymbol{H}_{:,2}, \ldots, \boldsymbol{V}_{pk} * \boldsymbol{H}_{:,p}] = \boldsymbol{V}_{pk} (\boldsymbol{H} \otimes \boldsymbol{I}_k),
$$

where $\boldsymbol{H}_{:,i}$, for $i = 1, 2, \ldots, p$, denotes the ith column of a matrix $\boldsymbol{H} \in \mathbb{R}^{p \times p}$ and $\boldsymbol{I}_k \in \mathbb{R}^{k \times k}$ is the identity matrix. Moreover,

$$
(\boldsymbol{V}_{pk} * \boldsymbol{H}) * \boldsymbol{a} = \boldsymbol{V}_{pk} * \boldsymbol{H} \boldsymbol{a}.
$$

The global Arnoldi process is an alternative to the block Arnoldi process. It uses the inner product (4.2). Algorithm 5 executes p steps of the global Arnoldi process with the matrix

---

Algorithm 5: Global Arnoldi process.

$\qquad$**Input:** $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{B}^\delta \in \mathbb{R}^{n \times k}$

$\qquad$**Output:** $\boldsymbol{V}_{(j+1)k} \in \mathbb{R}^{n \times (j+1)k}$ and $\boldsymbol{H}_{j+1,j} \in \mathbb{R}^{j+1 \times j}$

1 $\quad$ Set $\boldsymbol{V}_1 = \boldsymbol{B}^\delta / \left\| \boldsymbol{B}^\delta \right\|$;

2 $\quad$ **for** $j = 1, 2, \ldots, n$ **do**

3 $\qquad$ Compute $\boldsymbol{W}_j = \boldsymbol{A}\boldsymbol{V}_j$;

4 $\qquad$ **for** $i = 1, 2, \ldots, j$ **do**

5 $\qquad\qquad$ $h_{i,j} = \langle \boldsymbol{W}_j, \boldsymbol{V}_i \rangle_F$;

6 $\qquad\qquad$ $\boldsymbol{W}_j = \boldsymbol{W}_j - h_{i,j}\boldsymbol{V}_i$;

7 $\qquad$ **end**

8 $\qquad$ $h_{j+1,j} = \left\| \boldsymbol{W}_j \right\|$;

9 $\qquad$ $\boldsymbol{V}_{j+1} = \boldsymbol{W}_j / h_{j+1,j}$;

10 **end**

$\boldsymbol{A}$ and an initial block vector $\boldsymbol{V}_1 = \boldsymbol{B}^\delta / \left\| \boldsymbol{B}^\delta \right\|$. Throughout this section $\|\cdot\|$ denotes the Frobenius norm unless indicated otherwise.

As the global Arnoldi process orthogonalizes block columns of the matrix $\boldsymbol{V}_{pk}$, it is not necessary to compute an initial QR factorization as was done in Algorithm 3. At step p, the global Arnoldi process gives the relation

$$(4.4) \qquad\qquad \boldsymbol{A}\boldsymbol{V}_{pk} = \boldsymbol{V}_{(p+1)k} * \boldsymbol{H}_{p+1,p},$$

where we note that the Hessenberg matrix is of dimension $(p+1) \times p$, whereas the analogous matrix in the relation from the block Arnoldi process (3.2) discussed in the previous section has dimension $(p+1)k \times pk$. We assume that all subdiagonal entries of $\boldsymbol{H}_{p+1,p}$ are positive. This is the generic case as otherwise the algorithm breaks down. For linear discrete ill-posed problems, breakdown occurs exceedingly rarely. We therefore will not discuss this issue.

Algorithm 5 determines the matrices

$$\boldsymbol{V}_{pk} = [\boldsymbol{V}_1, \ldots, \boldsymbol{V}_p] \in \mathbb{R}^{n \times pk} \qquad \text{and} \qquad \boldsymbol{V}_{(p+1)k} = [\boldsymbol{V}_1, \ldots, \boldsymbol{V}_{p+1}] \in \mathbb{R}^{n \times (p+1)k},$$

where the block columns $\boldsymbol{V}_i \in \mathbb{R}^{n \times k}$ are F-orthonormal, i.e.,

$$(4.5) \qquad\qquad \langle \boldsymbol{V}_i, \boldsymbol{V}_j \rangle_F = \begin{cases} 1 & \text{for} \quad i = j, \\ 0 & \text{for} \quad i \neq j. \end{cases}$$

From the recursion formulas of Algorithm 5 and the fact that $\boldsymbol{V}_1 = \boldsymbol{B}^\delta / \left\| \boldsymbol{B}^\delta \right\|$, we have

$$(4.6) \qquad\qquad \boldsymbol{V}_j = \mathcal{P}_{j-1}(\boldsymbol{A})\boldsymbol{B}^\delta, \qquad j = 1, 2, \ldots, p,$$

for some block polynomials of degree precisely $j - 1$ defined by

$$\mathcal{P}_{j-1}(\boldsymbol{A})\boldsymbol{B}^\delta = \sum_{i=0}^{j-1} \boldsymbol{A}^i \boldsymbol{B}^\delta \Omega_i, \ \Omega_i \in \mathbb{R}^{k \times k}, \qquad \text{for } i = 0, 1, \ldots, j - 1.$$

Using (4.5) and (4.6) one can show that the set $\left\{ \boldsymbol{V}_j \right\}_{j=1}^p$ forms an orthonormal basis for the block Krylov subspace given by

$$\mathbb{K}_p(\boldsymbol{A}, \boldsymbol{B}^\delta) = \text{block span}\{\boldsymbol{B}^\delta, \boldsymbol{A}\boldsymbol{B}^\delta, \ldots, \boldsymbol{A}^{p-1}\boldsymbol{B}^\delta\}.$$

We assume this space to be of dimension p. This is the generic situation.

We apply the gl-GMRES method to determine an approximate solution of the linear system of equations (3.1). Jbliou et al. [25, 26] show that at step p of this method, only a p-dimensional least-squares problem has to be solved. This may result in significant computational savings over the BGMRES method. The global Arnoldi process and the gl-GMRES method are well discussed in [25, 26]. Applications of global methods to the solution of linear systems of equations with multiple right-hand sides are described in [5, 26]. The former reference discusses applications to the restoration of images that have been contaminated by blur and noise.

Similarly to Sections 2 and 3, we now derive the 1-shifted gl-GMRES method. We bypass the discussion of the Hessenberg structured matrices since the results of Section 2 will hold. For the initial iterate $\boldsymbol{X}_p^{(1)} = \boldsymbol{0} \in \mathbb{R}^{n \times k}$, the pth iterate determined by $\boldsymbol{X}_p^{(1)}$ of the 1-shifted gl-GMRES method satisfies (3.4). Analogously as in Section 2, we consider the reduced QR factorization $\boldsymbol{H}_{p+1,p} = \boldsymbol{Q}_{p+1,p}^{(1)} \boldsymbol{R}_p^{(1)}$ of the Hessenberg matrix obtained with the global Arnoldi process (4.4). Using the relations (4.3) and (4.1), we write (4.4) as

$$(4.7) \qquad \boldsymbol{A}\boldsymbol{V}_{pk} = \boldsymbol{V}_{(p+1)k}(\boldsymbol{H}_{p+1,p} \otimes \boldsymbol{I}_k) = \boldsymbol{V}_{(p+1)k}(\boldsymbol{Q}_{p+1,p}^{(1)} \otimes \boldsymbol{I}_k)(\boldsymbol{R}_p^{(1)} \otimes \boldsymbol{I}_k).$$

Introduce

$$\boldsymbol{W}_p^{(1)} = \boldsymbol{V}_{(p+1)k}(\boldsymbol{Q}_{p+1,p}^{(1)} \otimes \boldsymbol{I}_k) \in \mathbb{R}^{n \times pk}.$$

Using (4.7) and the fact that $\boldsymbol{H}_{p+1,p}$ may be written as a product of elementary reflectors (see Section 2), we note that

$$\boldsymbol{W}_p^{(1)} = \boldsymbol{A}\boldsymbol{V}_{pk}(\boldsymbol{R}_p^{(1)} \otimes \boldsymbol{I}_k)^{-1},$$

which shows that $\mathcal{R}\left(\boldsymbol{W}_p^{(1)}\right) = \mathbb{K}\left(\boldsymbol{A}, \boldsymbol{A}\boldsymbol{B}^\delta\right)$. We are lead to the minimization problem

$$
\begin{aligned}
(4.8) \quad & \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A}\boldsymbol{W}_p^{(1)} * \boldsymbol{y} - \boldsymbol{B}^\delta \right\| = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A}\boldsymbol{V}_{(p+1)k}(\boldsymbol{Q}_{p+1,p}^{(1)} \otimes \boldsymbol{I}_k) * \boldsymbol{y} - \boldsymbol{B}^\delta \right\| \\
& = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{V}_{(p+2)k}(\boldsymbol{Q}_{p+2}^{(2)} \otimes \boldsymbol{I}_k)(\boldsymbol{R}_{p+2,p}^{(2)} \otimes \boldsymbol{I}_k) * \boldsymbol{y} - \boldsymbol{B}^\delta \right\| \\
& = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{V}_{(p+2)k} * \left(\boldsymbol{Q}_{p+2}^{(2)} \boldsymbol{R}_{p+2,p}^{(2)} \boldsymbol{y} - \left\| \boldsymbol{B}^\delta \right\| \boldsymbol{e}_1 \right) \right\| \\
& = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{R}_{p+2,p}^{(2)} \boldsymbol{y} - \left\| \boldsymbol{B}^\delta \right\| \left(\boldsymbol{Q}_{p+2}^{(2)} \right)^T \boldsymbol{e}_1 \right\|,
\end{aligned}
$$

where $\boldsymbol{e}_1 \in \mathbb{R}^{p+2}$ denotes the first axis vector. Let $\boldsymbol{y}_p^{(1)}$ denote the solution of (4.8). Then the solution of the 1-shifted gl-GMRES method is given by $\boldsymbol{X}_p^{(1)} = \boldsymbol{W}_p^{(1)} * \boldsymbol{y}_p^{(1)}$.

Following the same line of reasoning as in Sections 2 and 3, we can derive the appropriate matrices $\boldsymbol{W}_p^{(\ell)} \in \mathbb{R}^{n \times pk}$ such that $\mathcal{R}\left(\boldsymbol{W}_p^{(\ell)}\right) = \mathbb{K}\left(\boldsymbol{A}, \boldsymbol{A}^\ell \boldsymbol{B}^\delta\right)$. Let the initial iterate be $\boldsymbol{X}_0^{(\ell)} = \boldsymbol{0} \in \mathbb{R}^{n \times k}$. Then the pth iterate of the ℓ-shifted gl-GMRES method, $\boldsymbol{X}_p^{(\ell)}$, satisfies

$$(4.9) \qquad \left\| \boldsymbol{A}\boldsymbol{X}_p^{(\ell)} - \boldsymbol{B}^\delta \right\| = \min_{\boldsymbol{X} \in \mathbb{K}\left(\boldsymbol{A}, \boldsymbol{A}^\ell \boldsymbol{B}^\delta\right)} \left\| \boldsymbol{A}\boldsymbol{X} - \boldsymbol{B}^\delta \right\|.$$

We may simplify (4.9) as follows

$$
\begin{aligned}
(4.10) \quad & \left\| \boldsymbol{A}\boldsymbol{X}_p^{(\ell)} - \boldsymbol{B}^\delta \right\| = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{A}\boldsymbol{W}_p^{(\ell)} * \boldsymbol{y} - \boldsymbol{B}^\delta \right\| \\
& = \min_{\boldsymbol{y} \in \mathbb{R}^p} \left\| \boldsymbol{R}_{\ell+p+1,p}^{(\ell+1)} \boldsymbol{y} - \left\| \boldsymbol{B}^\delta \right\| \left(\boldsymbol{Q}_{\ell+p+1}^{(\ell+1)} \right)^T \boldsymbol{e}_1 \right\|,
\end{aligned}
$$

where $e_1 \in \mathbb{R}^{\ell+p+1}$ denotes the first axis vector. Denoting the solution of (4.9) by $y_p^{(\ell)}$, the solution of the ℓ-shifted gl-GMRES minimization problem is given by $X_p^{(\ell)} = W_p^{(\ell)} * y_p^{(\ell)}$. Algorithm 6 describes the ℓ-shifted gl-GMRES method. Termination is achieved with the discrepancy principle.

Algorithm 6: ℓ-shifted gl-GMRES ($\ell \geq 1$) with discrepancy principle.

Input: $A \in \mathbb{R}^{n \times n}, ; B^\delta \in \mathbb{R}^{n \times k}$, and $\ell \in \{1, 2, 3, \dots\}$
Output: $X_p^{(\ell)} \in \mathbb{R}^{n \times k}$

1 Set $V_1 = B^\delta / \left\| B^\delta \right\|$ and $X_0^{(\ell)} = \mathbf{0}$;
2 Compute ℓ steps of global Arnoldi: $AV_{\ell k} = V_{(\ell+1)k} * H_{\ell+1,\ell}$;
3 **for** $p = 1, 2, \dots$ **do**
4   Compute next global Arnoldi step: $AV_{(\ell+p)k} = V_{(\ell+p+1)k} * H_{\ell+p+1,\ell+p}$;
5   Compute QR factorization: $\left[Q_{p+1}^{(1)}, R_{p+1,p}^{(1)} \right] = H_{p+1,p}$;
6   **for** $j = 1, 2, \dots, \ell$ **do**
7   Compute QR factorization: $\left[Q_{j+p+1}^{(j+1)}, R_{j+p+1,p}^{(j+1)} \right] = H_{j+p+1,j+p} Q_{j+p,p}^{(j)}$;
8   **end**
9   Compute $y_p^{(\ell)}$ as the minimizer of $\left\| R_{\ell+p+1,p}^{(\ell+1)} y - \left\| B^\delta \right\| \left(Q_{\ell+p+1}^{(\ell+1)} \right)^T e_1 \right\|_2$;
10   Compute $X_p^{(\ell)} = V_{(\ell+p)k} \left(Q_{\ell+p,p}^{(\ell)} y_p^{(\ell)} \otimes I_k \right)$;
11   Compute $\| r_p \| = \left\| AX_p^{(\ell)} - B^\delta \right\|$;
12   **if** $\| r_p \| \leq \tau \delta$ **then**
13   Stop;
14   **end**
15 **end**

The computational work for large problems with Algorithm 6 is dominated by the $\ell + p$ matrix-vector products with A in lines **2** and **4**. Algorithm 6 requires the computation of QR factorizations in lines **5** and **7**. The flop count for these factorization using elementary reflectors is $O(\ell p^2)$. For comparison, a standard QR algorithm would require $O(\ell p^3)$ flops.

Algorithm 6 is terminated with the discrepancy principle. The computation of the norm of the pth residual, r_p, in line **11** can be simplified similarly as described in the previous sections. Below we provide a result analogous to Propositions 2.1 and 3.1.

Because the matrices $Q_{j+p+1}^{(j+1)}$, for $j = 0, 1 \dots, \ell$, may be written as products of elementary reflectors given by (2.9), the right-hand side of the last inequality of (4.10) may be expressed as

$$(4.11) \quad \begin{aligned} \left(Q_{\ell+p+1}^{(\ell+1)} \right)^T e_1 \left\| B^\delta \right\| &= G_{\ell+p}^T \dots G_3^T G_2^T G_1^T \left[\left\| B^\delta \right\|, 0, \dots, 0 \right]^T \\ &= [\gamma_1, \gamma_2, \gamma_3, \dots, \gamma_{\ell+p}, \gamma_{\ell+p+1}]^T =: g_{\ell+p+1}, \end{aligned}$$

where $g_{\ell+p+1} \in \mathbb{R}^{(\ell+p+1)}$ and $G_k^T \in \mathbb{R}^{(\ell+p+1) \times (\ell+p+1)}$, for $k = 1, 2, \dots, \ell + p$.

PROPOSITION 4.1. *Consider the pth iteration of Algorithm 6. For $k = 1, 2, \dots, \ell + p$, let G_k be the reflector matrices used to transform $H_{\ell+p+1,\ell+p} Q_{\ell+p,p}^{(\ell)}$ in line 7 when $j = \ell$*

into upper triangular form $\boldsymbol{R}^{(\ell+1)}_{\ell+p+1,p}$, and define $\boldsymbol{g}_{\ell+p+1}$ similarly as in (4.11). Let $\boldsymbol{R}^{(\ell+1)}_p$ denote the $p \times p$ upper triangular matrix obtained from $\boldsymbol{R}^{(\ell+1)}_{\ell+p+1,p}$ by deleting the last $\ell+1$ rows, and let $\bar{\boldsymbol{g}}_p \in \mathbb{R}^p$ be obtained from $\boldsymbol{g}_{\ell+p+1}$ by deleting the last $\ell+1$ components. Then:

1. The rank of $\boldsymbol{AW}^{(\ell)}_p$ equals the rank of $\boldsymbol{R}^{(\ell+1)}_p$. In particular, if $r^{(\ell+1)}_{(p,p)}$, the last diagonal entry of $\boldsymbol{R}^{(\ell+1)}_p$, vanishes, then $\boldsymbol{A}$ is singular.

2. The vector $\boldsymbol{y}^{(\ell)}_p$ that minimizes $\left\| \boldsymbol{R}^{(\ell+1)}_{\ell+p+1,p} \boldsymbol{y} - \boldsymbol{g}_{\ell+p+1} \right\|$ is given by

$$\boldsymbol{y}^{(\ell)}_p = \left(\boldsymbol{R}^{(\ell+1)}_p \right)^{-1} \bar{\boldsymbol{g}}_p.$$

3. The residual $\boldsymbol{r}_p := \boldsymbol{AX}^{(\ell)}_p - \boldsymbol{B}^\delta$ at step p of Algorithm 6 satisfies

$$\boldsymbol{r}_p = \boldsymbol{V}_{(\ell+p+1)k} * \left(\boldsymbol{R}^{(\ell+1)}_{\ell+p+1,p} \boldsymbol{y} - \left\| \boldsymbol{B}^\delta \right\| \left(\boldsymbol{Q}^{(\ell+1)}_{\ell+p+1} \right)^T \boldsymbol{e}_1 \right).$$

Therefore,

$$\|\boldsymbol{r}_p\| = \left\| \boldsymbol{R}^{(\ell+1)}_{\ell+p+1,p} \boldsymbol{y} - \boldsymbol{g}_{\ell+p+1} \right\|$$

$$= \left\| [0 \ldots, 0, \gamma_{p+1}, \ldots, \gamma_{\ell+p+1}]^T \right\| = \sqrt{\sum_{i=p+1}^{\ell+p+1} (\gamma_i)^2}.$$

The proof of Proposition 4.1 is analogous to the proof of Proposition 2.1. We therefore omit the details. The third result of Proposition 4.1 allows the computation of the norm of the pth residual in line **11** without evaluating an additional matrix-block-vector product with $\boldsymbol{A}$ or computing $\boldsymbol{X}^{(\ell)}_p$. Additionally, while $\boldsymbol{r}_p$ lives in $\mathbb{R}^{n \times k}$, the computation of its norm only requires taking the norm of the last $\ell+1$ entries of an $\ell+p+1$ vector. This results in computational savings when k is large. We include this computational strategy in our implementation of the ℓ-shifted gl-GMRES algorithm.

We conclude this section with some comments on the storage requirement of Algorithm 6. Similarly as for the shifted methods of Sections 2 and 3, we do not require storage of the matrix $\boldsymbol{W}^{(\ell)}_p$. Instead, we only store the penultimate $\boldsymbol{Q}$-matrix from the QR factorization of iteration p from lines **5** and **7**. The F-orthonormal matrix $\boldsymbol{V}_{(\ell+p)k}$ is determined by the first $\ell+p$ block columns of the already stored $\boldsymbol{V}_{(\ell+p+1)k}$-matrix from the global Arnoldi process. Therefore, the only additional storage requirement in iteration p of the ℓ-shifted gl-GMRES method over the standard gl-GMRES method is an $(\ell+p) \times p$ orthogonal matrix.

5. Numerical examples. We illustrate the performance of the ℓ-shifted GMRES, BGM-RES, and gl-GMRES methods with several examples. The algorithms terminate the iterations with the discrepancy principle. To evaluate the quality of computed solutions, we compute the relative reconstructive error (RRE) defined by

$$\text{RRE}(\boldsymbol{x}^{(\ell)}_p) = \frac{\left\| \boldsymbol{x}^{(\ell)}_p - \boldsymbol{x}^\dagger \right\|}{\|\boldsymbol{x}^\dagger\|} \qquad \text{and} \qquad \text{RRE}(\boldsymbol{X}^{(\ell)}_p) = \frac{\left\| \boldsymbol{X}^{(\ell)}_p - \boldsymbol{X}^\dagger \right\|}{\left\| \boldsymbol{X}^\dagger \right\|},$$

where $\boldsymbol{x}^{(\ell)}_p$ and $\boldsymbol{X}^{(\ell)}_p$ denote the approximate solutions determined by the algorithms at iteration p for the linear and block linear systems, respectively. Here, the Euclidean vector and the Frobenius matrix norms are used in their appropriate contexts. We refer to the RRE of

the pth iterate computed by the appropriate ℓ-shifted method as the breakout RRE value. The exact solutions are represented by $\boldsymbol{x}^{\dagger}$ or $\boldsymbol{X}^{\dagger}$.

To provide a measure of the computational effort required by the ℓ-shifted methods in our examples, we also tabulate the number of steps of the appropriate Arnoldi process required by each method to terminate according to the discrepancy principle. The actual number of matrix-vector or matrix-block-vector product evaluations with $\boldsymbol{A}$ may be computed by adding the shifted quantity ℓ to the tabulated number of iterations for any ℓ-shifted method. While shifted variants do require additional matrix-vector or matrix-block-vector product evaluations with $\boldsymbol{A}$, they typically yield computed approximate solutions with lower RRE values.

We begin with a Fredholm integral equation in one space dimension and then proceed to image deblurring problems in two space dimensions. We will restore a gray-scale image to illustrate the performance of the ℓ-shifted GMRES method and a color image to explore the performance of the ℓ-shifted BGMRES and gl-GMRES methods. We also compare the use of elementary reflectors for the upper-triangularization of Hessenberg-type matrices to MATLAB's backslash command.

Our computational work was carried out in MATLAB R2020b on a MacBook Pro laptop running MacOS Catalina with an i5 Dual-Core Intel processor with @2.7 GHz and 8 GB of RAM. The computations were carried out with about 15 significant decimal digits.

Example Shaw. Our first example is a Fredholm integral equation of the first kind discussed by Shaw in [35]. The integral is given by

$$(5.1) \qquad \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \boldsymbol{\kappa}(\omega, \sigma)\,\boldsymbol{x}(\sigma)\,d\sigma = \boldsymbol{b}(\omega), \qquad -\frac{\pi}{2} \le \omega \le \frac{\pi}{2},$$

with kernel

$$\boldsymbol{\kappa}(\omega, \sigma) = (\cos(\sigma) + \cos(\omega)) \left(\frac{\sin(\zeta)}{\zeta}\right)^2, \quad \text{where} \quad \zeta = \pi\left(\sin(\sigma) + \sin(\omega)\right).$$

The matrix $\boldsymbol{A}$ is obtained by discretizing the integral (5.1) using a Nyström method based on a composite trapezoidal quadrature rule with 1000 equidistant nodes [29]. This discretization gives a nonsymmetric matrix $\boldsymbol{A} \in \mathbb{R}^{1000 \times 1000}$. The right-hand side function $\boldsymbol{b}(\omega)$ is chosen so that the solution $\boldsymbol{x}(\sigma)$ is the sum of two Gaussian functions. Application of this equations are described in [35]. The exact solution of the discretized problem satisfies $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x}^{\dagger}$, where $\boldsymbol{x}^{\dagger}, \boldsymbol{b} \in \mathbb{R}^{1000}$. The condition number of the matrix $\boldsymbol{A}$ as determined by the MATLAB function cond is about 10^{16}; thus, $\boldsymbol{A}$ is numerically singular. Its singular values decay without a significant gap and cluster at zero. A vector $\boldsymbol{e} \in \mathbb{R}^{1000}$ is formed with normally distributed random entries with zero mean to simulate noise so that $\boldsymbol{b}^{\delta} = \boldsymbol{b} + \boldsymbol{e}$; the vector $\boldsymbol{e}$ is scaled so as to correspond to a specific percentage noise level

$$v = 100 \left(\frac{\|\boldsymbol{e}\|}{\|\boldsymbol{b}\|}\right).$$

We will refer to v as the noise level, and will consider two noise levels: 1% and 0.1%.

The required number of Arnoldi iterations and RRE values at breakout for the ℓ-shifted GMRES methods for noise levels 1% and 0.1% are displayed in Table 5.1. The best reconstructions with the smallest RRE values among the shifted methods are displayed in Figure 5.1. For 1% noise, the quality of the restoration improves as ℓ is increased from $\ell = 0$ to $\ell = 3$. Carrying out more iterations does not improve the quality of the computed restorations. In the 0.1% case, both 2- and 3-shifted GMRES methods perform better than their lesser shifted

TABLE 5.1

Results for the Shaw *example for the iterative methods for two noise levels. The iteration numbers display the number of Arnoldi iterations necessary for the discrepancy principle to be satisfied. Breakout RRE values presented correspond to the pth iterative solution when the discrepancy principle is satisfied.*

Noise Level	Method	Iterations	Breakout RRE
1%	0-shifted GMRES	6	0.1471
	1-shifted GMRES	6	0.1214
	2-shifted GMRES	9	0.0599
	3-shifted GMRES	9	0.0533
0.1%	0-shifted GMRES	7	0.0553
	1-shifted GMRES	7	0.0560
	2-shifted GMRES	9	0.0525
	3-shifted GMRES	10	0.0525
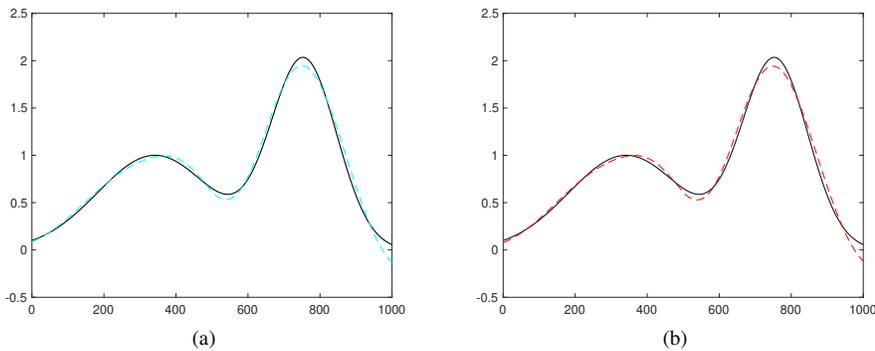
FIG. 5.1. Shaw *example: (a) The highest quality solution for* 1% *noise level for 3-shifted GMRES at breakout is shown by the dashed cyan curve. (b) The highest quality solution for* 0.1% *noise level for 2-shifted GMRES at breakout is shown by the dashed red curve. The desired solutions are represented by the solid black curve.*

counterparts. These RRE comparisons can be seen in Figure 5.2 for both noise levels. The cost of an improved solution here is the additional matrix-vector product evaluation that has to be carried out in the Arnoldi iterations for the ℓ-shifted GMRES algorithms when a larger ℓ-value is used. Relative residual plots for all methods and for both noise levels are displayed in Figure 5.3. The iterations are terminated when the residual error is smaller than or equal to $\tau\delta$ with $\tau = 1.01$ and $\delta = v$.

To provide context to the discussion in Section 2 regarding the use of elementary reflectors versus a standard QR factorization algorithm for the upper triangulation of Hessenberg-type matrices, we provide an experimental result in Table 5.2. Here, we report computational timings and ratios for the 0-shifted GMRES solution for the *Shaw* example with 0.1% noise. The column "Reflection Time" shows timings in seconds required to bring the upper Hessenberg matrix determined by the standard (non-shifted) GMRES method to upper triangular form by an application of successive elementary reflectors and for computing the solution $\boldsymbol{x}_p^{(0)}$ by back substitution. The "Backslash Time" column refers to timings in seconds for solving the least-squares problem with a Hessenberg matrix using the backslash operation in MATLAB. Because the Hessenberg matrix here is non-square, the backslash operation uses QR factorization and computes a least-squares solution. The timings in this table were produced by averaging 100 *Shaw* experiments under the same conditions.
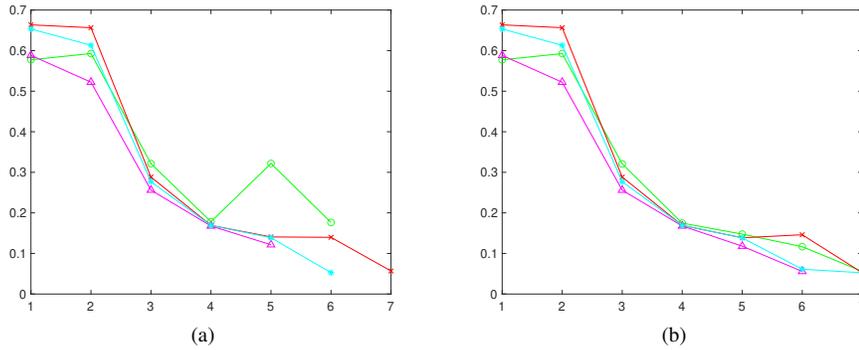
FIG. 5.2. Shaw *example: RRE plots vs. iteration numbers for (a)* 1% *and (b)* 0.1% *noise levels for ℓ-shifted GMRES methods for ℓ* = 0, 1, 2, 3 *at breakout. Line descriptions: green circles:* 0-*shifted GMRES, magenta triangles:* 1-*shifted GMRES, red crosses:* 2-*shifted GMRES, and cyan stars:* 3-*shifted GMRES.*
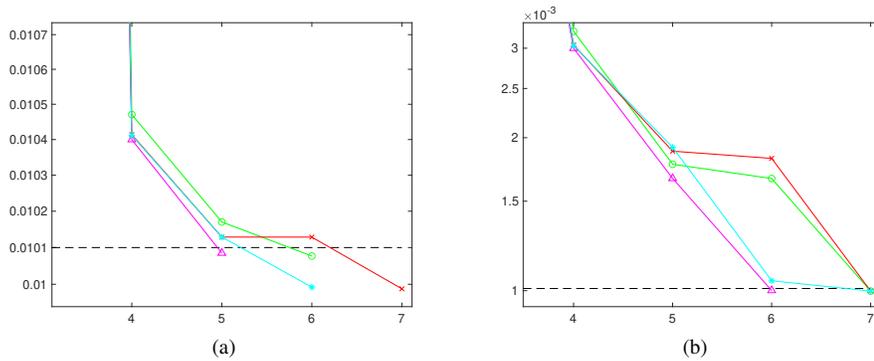
FIG. 5.3. Shaw *example: Relative residual plots vs. iteration numbers for (a)* 1% *and (b)* 0.1% *noise levels for ℓ-shifted GMRES methods for ℓ* = 0, 1, 2, 3 *at breakout. The black dashed horizontal line represents the breakout level according to the discrepancy principle. Line descriptions: green circles:* 0-*shifted GMRES, magenta triangles:* 1-*shifted GMRES, red crosses:* 2-*shifted GMRES, and cyan stars:* 3-*shifted GMRES. (Note that the plots focus on the later iterations for better resolution.)*

The "Reflection-Backslash" column ratios for all iterations are above one. This indicates that using reflectors implemented in MATLAB may be ill-advised. We note that, while one might expect the timings in both columns of Table 5.2 to increase with each iteration, in general these matrices are very small and so a large timing variance can be expected. Finally, its worthwhile to mention that this experiment does not have to be carried out with a shifted method, since these methods would have produced Hessenberg-type matrices with a larger lower bandwidth, which would have required the use of additional reflectors to achieve upper triangulation.

The remaining two examples investigate the shifted methods applied to the restoration of images that have been contaminated by spatially invariant blur and noise. Image deblurring can be modeled by a Fredholm integral equation of the first kind,

$$(5.2) \qquad \int_{\boldsymbol{\Omega}} \boldsymbol{\kappa}(u, s, v, t) \, \boldsymbol{x}(u, v) \, \mathrm{d}u \, \mathrm{d}v = \boldsymbol{b}(s, t), \qquad (s, t) \in \boldsymbol{\Omega},$$

where $\boldsymbol{b}$ represents the blurred image, $\boldsymbol{\kappa}$ is the point spread function (PSF), and $\boldsymbol{\Omega}$ is the domain of the exact image represented by $\boldsymbol{x}$. When the blur is spatially invariant, $\boldsymbol{\kappa}$ in the

TABLE 5.2

Shaw *example: Application of elementary reflectors versus backslash. Timings for the upper triangulation of Hessenberg matrices. The timings were determined by averaging 100 experiments and are given in seconds.*

Iteration Number	Reflection Time	Backslash Time	Reflection-Backslash Ratio
1	8.08×10^{-3}	5.20×10^{-3}	1.55
2	2.38×10^{-2}	1.49×10^{-2}	1.60
3	1.38×10^{-2}	7.17×10^{-3}	1.93
4	1.10×10^{-2}	5.42×10^{-3}	2.04
5	1.28×10^{-2}	5.43×10^{-3}	2.36
6	9.65×10^{-3}	3.92×10^{-3}	2.46
7	1.04×10^{-2}	3.10×10^{-3}	3.37

| (a) | ((b) | ((c) |
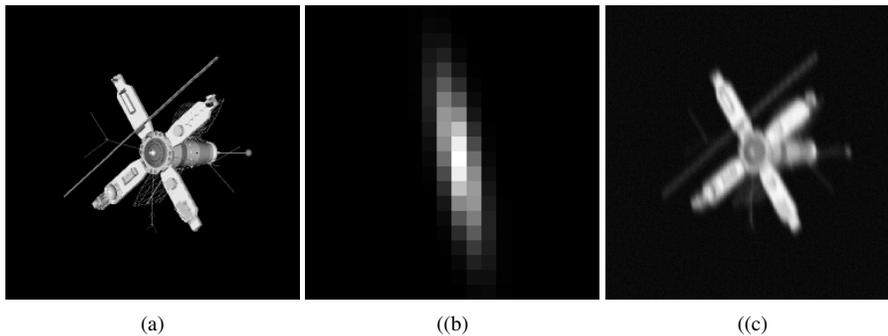
FIG. 5.4. *Satellite example: (a) true image (256 × 256 pixels) , (b) PSF (20 × 20 pixels), (c) blurred and 3% noised image (256 × 256 pixels).*

integral equation is of the form $\boldsymbol{\kappa}(u, s, v, t) = \boldsymbol{\kappa}(u - s, v - t)$. Discretization of (5.2) gives a linear discrete ill-posed problem of the form (1.1), where the structure of $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ depends on the structure of κ and on the boundary condition (BC) of the PSF (see [6, 12, 20] for more details). Some common BCs include periodic, zero, and reflexive. We will use the latter two.

Example Satellite. We consider the restoration of a *satellite* image that has been blurred by a two-dimensional non-symmetric Gaussian function, whose non-zero and non-scaled discrete PSF entries are given by

$$(5.3) \qquad \boldsymbol{P}_{i,j} = \exp\left(-\frac{1}{2} \begin{bmatrix} i - k \\ j - h \end{bmatrix}^T \begin{bmatrix} s_1^2 & \rho^2 \\ \rho^2 & s_2^2 \end{bmatrix}^{-1} \begin{bmatrix} i - k \\ j - h \end{bmatrix}\right),$$

where the parameters s_1, s_2, and ρ define the orientation and width of the PSF centered at (k, h). Gaussian PSFs are well known to produce extremely ill-conditioned matrices. This is evident by the rapid decay of their singular values to zero. Further discussion regarding the above PSF and close variants may be found in [20, 21]. Because of the astronomical nature of the image, we impose zero boundary conditions, and we introduce 3% Gaussian noise in the same manner as in the *Shaw* example. The true image, PSF, and blurred and noised image are displayed in Figure 5.4.

The iteration count and breakout RRE values for the *satellite* example are shown in Table 5.3. Figure 5.5 displays the image reconstructions determined at breakout for each of the shifted GMRES variants. We note the significant RRE improvement when using $\ell \geq 1$ compared to 0-shifted GMRES for this example. However, the improvement for the 2-shifted variant compared to the 1-shifted variant is much smaller than in the *Shaw* example. The

TABLE 5.3

Satellite example: Iteration entries show the number of Arnoldi iterations necessary for the discrepancy principle to be satisfied. Breakout RRE values correspond to the pth iterative solution when the discrepancy principle is satisfied.

Method	Iterations	Breakout RRE
0-shifted GMRES	3	0.3106
1-shifted GMRES	5	0.2511
2-shifted GMRES	8	0.2491
3-shifted GMRES	10	0.2533

(a)   (b)

(c)   (d)
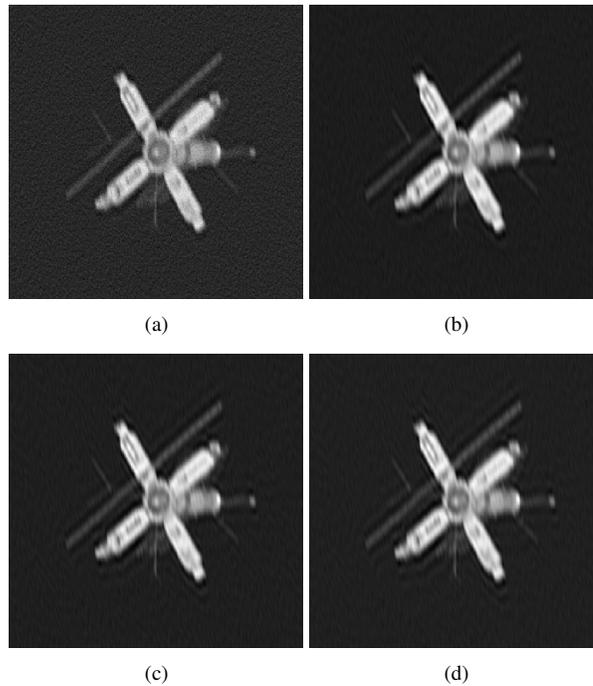
FIG. 5.5. Satellite *example: (a) 0-shifted GMRES, (b) 1-shifted GMRES, (c) 2-shifted GMRES, (d) 3-shifted GMRES reconstructions at breakout.*

iterative evolution of the RREs and the relative residual plots are provided in Figure 5.6. The left-hand plot displays the relative residual progression, and the right-hand plot shows the RRE values for each iteration of the four variants considered.

Example Board. Our final example considers the use of the ℓ-shifted BGMRES and ℓ-shifted gl-GMRES methods to reconstruct the colored *board* image which is blurred by a two-dimensional non-symmetric Gaussian function, whose non-zero PSF entries are given by (5.3). Here we consider a slightly different PSF than in the *satellite* example for expositional purposes. A colored image such as *board* provides a natural way to test block methods since a colored image stored in the RGB format may be thought of as an $n \times n \times 3$ array with each $n \times n$ submatrix corresponding to a color. In order to make it applicable to our methods, we parse the array into three $n \times n$ matrices (one for each color) and store each matrix as a vector with n^2 entries. Thus, each vector corresponds to a color channel; see [20] for details. As the structure beyond the boundary of the image is unknown, we impose reflexive BCs. We also introduce 5% Gaussian noise. The true colored image, PSF, and blurred and noised image are

(a)   (b)
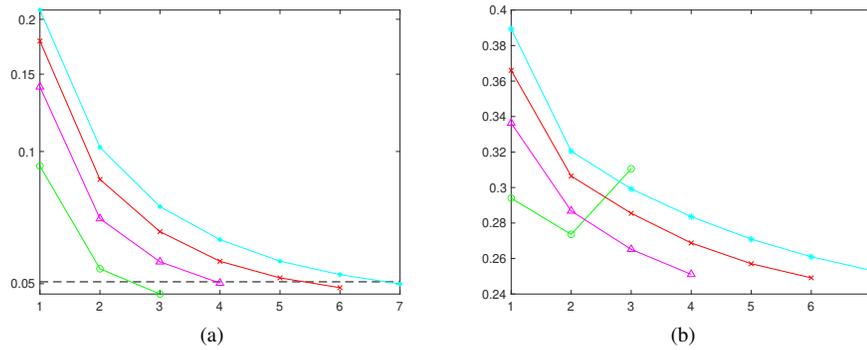
Fig. 5.6. *Satellite example: (a) relative residual plot vs. iteration number and (b) RRE plot vs. iteration numbers for 3% noise for ℓ-shifted GMRES methods for $\ell = 0, 1, 2, 3$ at breakout. The black dashed horizontal line in (a) represents the breakout level according to the discrepancy principle. Graph descriptions: green circles: 0-shifted GMRES, magenta triangles: 1-shifted GMRES, red crosses: 2-shifted GMRES, and cyan stars: 3-shifted GMRES.*

(a)   (b)   (c)
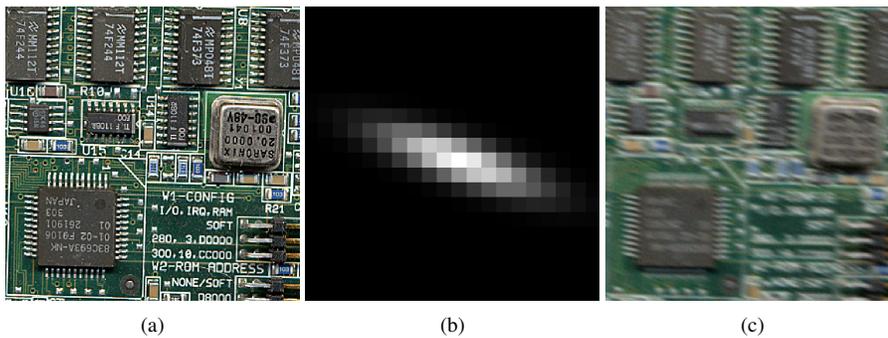
Fig. 5.7. *Board example: (a) true image ($300 \times 300 \times 3$ pixels), (b) PSF (20×20 pixels), (c) blurred and 5% noised image ($300 \times 300 \times 3$ pixels).*

displayed in Figure 5.7.

The iteration count and breakout RRE values for the *board* example are shown in Table 5.4. The best reconstructions in terms of the lowest RRE values among each of the ℓ-shifted BGMRES and ℓ-shifted gl-GMRES variants are displayed in Figure 5.8. Similarly to the previous examples, the ℓ-shifted BGMRES and gl-GMRES methods with $\ell \geq 1$ provide a significant improvement in the quality of the reconstructions determined when compared to reconstructions determined by 0-shifted variants. The 2-shifted BGMRES method yield a slight improvement over its 1-shifted variant.

For this example, we found the ℓ-shifted gl-GMRES methods to perform better than their BGMRES counterparts in terms of the quality of the reconstructed images achieved. Among all variants, the 1-shifted gl-GMRES method achieved the smallest RRE value. The iterative evolution of the RREs and the relative residual plots are provided in Figure 5.9 for the BGMRES methods and in Figure 5.10 for the gl-GMRES methods. The left-hand plots depict the relative residual progression, and the right-hand plots show the RRE values for each iteration. These plots display the same general behavior as for the *satellite* example.

6. Conclusions. This investigation expands upon already available work on the advantages of applying ℓ-shifted Arnoldi-type methods to the solution of linear discrete ill-posed problems. Along with formalizing the ℓ-shifted GMRES method already investigated, we
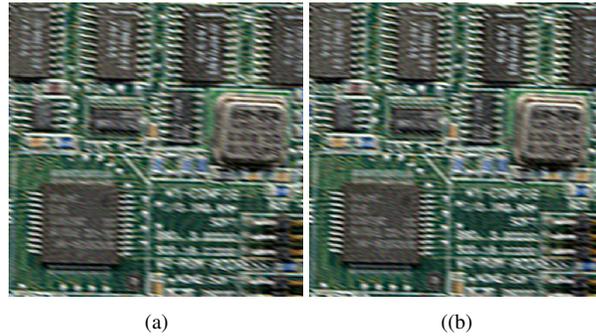
FIG. 5.8. *Board example: Highest reconstructed approximate solutions determined by the (a) 2-shifted BGMRES method and the (b) 1-shifted gl-GMRES method at breakout.*
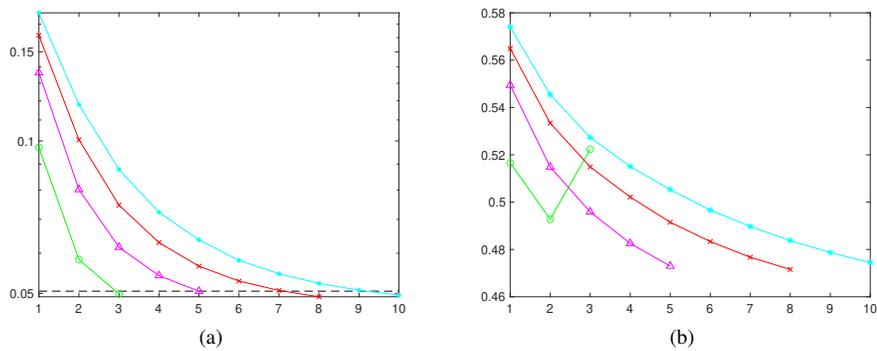
FIG. 5.9. *Board example: (a) relative residual plot against iteration numbers and (b) RRE plot against iteration numbers for 5% noise for ℓ-shifted BGMRES methods for $\ell = 0, 1, 2, 3$ at breakout. The black dashed horizontal line in (a) represents the breakout level according to the discrepancy principle. Graph descriptions: green circles: 0-shifted BGMRES, magenta triangles: 1-shifted BGMRES, red crosses: 2-shifted BGMRES, and cyan stars: 3-shifted BGMRES.*
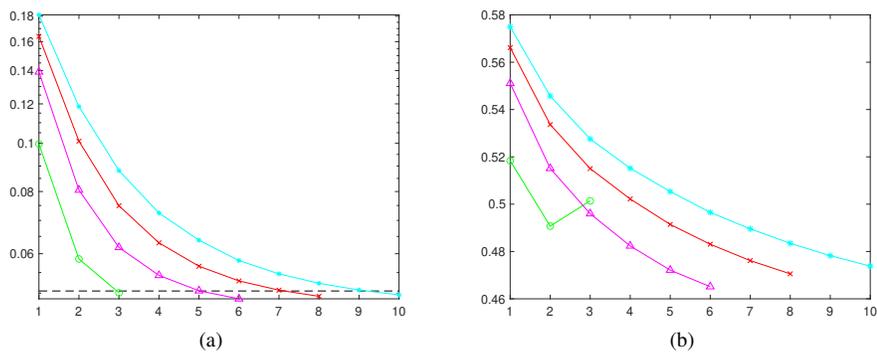
FIG. 5.10. *Board example: (a) relative residual plot vs. iteration numbers and (b) RRE plot vs. iteration numbers for 5% noise for ℓ-shifted gl-GMRES methods for $\ell = 0, 1, 2, 3$ at breakout. The black dashed horizontal line in (a) represents the breakout level according to the discrepancy principle. Graph descriptions: green circles: 0-shifted gl-GMRES, magenta triangles: 1-shifted gl-GMRES, red crosses: 2-shifted gl-GMRES, and cyan stars: 3-shifted gl-GMRES.*

TABLE 5.4
Board example: Iteration entries correspond to the number of block or global Arnoldi iterations necessary for the discrepancy principle to be satisfied. Breakout RRE values presented correspond to the pth approximate solution determined when the discrepancy principle is satisfied.

Method	Iterations	Breakout RRE
0-shifted BGMRES	3	0.5230
1-shifted BGMRES	6	0.4731
2-shifted BGMRES	10	0.4718
3-shifted BGMRES	13	0.4746
0-shifted gl-GMRES	3	0.5014
1-shifted gl-GMRES	7	0.4651
2-shifted gl-GMRES	10	0.4707
3-shifted gl-GMRES	13	0.4738

introduced the ℓ-shifted BGMRES and ℓ-shifted gl-GMRES methods, and provide software capable of running all ℓ-shifted methods discussed. Moreover, we commented on the computational cost and storage requirements of these methods. Examples in Section 5 illustrate that ℓ-shifted methods with $\ell > 1$ can give iterates that approximate the desired solution more accurately than when $\ell = 1$ or $\ell = 0$.

Supplementary material. The accompanying software is available at
https://etna.ricam.oeaw.ac.at/volumes/2021-2030/vol58/addition/p348.php
in form of a compressed file entitled ellShiftedPkg.zip. Installation details are discussed in the file CodePrimer.pdf as well as in the README.md file.

REFERENCES

[1] J. BAGLAMA, *Dealing with linear dependence during the iterations of the restarted block Lanczos methods*, Numer. Algorithms, 25 (2000), pp. 23–36.
[2] J. BAGLAMA AND L. REICHEL, *Augmented GMRES-type methods*, Numer. Linear Algebra Appl., 14 (2007), pp. 337–350.
[3] ———, *Decomposition methods for large linear discrete ill-posed problems*, J. Comput. Appl. Math., 198 (2007), pp. 332–343.
[4] M. BELLALIJ, L. REICHEL, AND H. SADOK, *Some properties of range restricted GMRES methods*, J. Comput. Appl. Math., 290 (2015), pp. 310–318.
[5] A. H. BENTBIB, M. EL GUIDE, K. JBILOU, AND L. REICHEL, *A global Lanczos method for image restoration*, J. Comput. Appl. Math., 300 (2016), pp. 233–244.
[6] S. BERISHA AND J. G. NAGY, *Iterative methods for image restoration*, Tech. Rep., Department of Mathematics and Computer Science, Emory University, Atlanta, 2012.
[7] C. BREZINSKI, M. REDIVO-ZAGLIA, G. RODRIGUEZ, AND S. SEATZU, *Multi-parameter regularization techniques for ill-conditioned linear systems*, Numer. Math., 94 (2003), pp. 203–228.
[8] C. BREZINSKI, G. RODRIGUEZ, AND S. SEATZU, *Error estimates for linear systems with applications to regularization*, Numer. Algorithms, 49 (2008), pp. 85–104.
[9] D. CALVETTI, B. LEWIS, AND L. REICHEL, *On the choice of subspace for iterative methods for linear discrete ill-posed problems*, Int. J. Appl. Math. Comput. Sci., 11 (2001), pp. 1069–1092.
[10] ———, *GMRES, L-curves, and discrete ill-posed problems*, BIT Numer. Math., 42 (2002), pp. 44–65.
[11] ———, *On the regularizing properties of the GMRES method*, Numer. Math., 91 (2002), pp. 605–625.
[12] M. DONATELLI AND L. REICHEL, *Square smoothing regularization matrices with accurate boundary conditions*, J. Comput. Appl. Math., 272 (2014), pp. 334–349.

[13] L. DYKES AND L. REICHEL, *A family of range restricted iterative methods for linear discrete ill-posed problems*, Dolomites Res. Notes Approx., 6 (2013), pp. 27–36.

[14] L. ELBOUYAHYAOUI, A. MESSAOUDI, AND H. SADOK, *Algebraic properties of the block GMRES and block Arnoldi methods*, Electron. Trans. Numer. Anal., 33 (2008/09), pp. 207–220.
https://etna.ricam.oeaw.ac.at/vol.33.2008-2009/pp207-220.dir/pp207-220.pdf

[15] H. W. ENGL, M. HANKE, AND A. NEUBAUER, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996.

[16] A. FROMMER, K. LUND, AND D. B. SZYLD, *Block Krylov subspace methods for functions of matrices*, Electron. Trans. Numer. Anal., 47 (2017), pp. 100–126.
https://etna.ricam.oeaw.ac.at/vol.47.2017/pp100-126.dir/pp100-126.pdf

[17] P. C. HANSEN, *Rank Deficient and Discrete Ill-Posed Problems*, SIAM, Philadelphia, 1998.

[18] P. C. HANSEN AND T. K. JENSEN, *Noise propagation in regularizing iterations for image deblurring*, Electron. Trans. Numer. Anal., 31 (2008), pp. 204–220.
https://etna.ricam.oeaw.ac.at/vol.31.2008/pp204-220.dir/pp204-220.pdf

[19] P. C. HANSEN, T. K. JENSEN, AND G. RODRIGUEZ, *An adaptive pruning algorithm for the discrete L-curve criterion*, J. Comput. Appl. Math., 198 (2007), pp. 483–492.

[20] P. C. HANSEN, J. NAGY, AND D. P. O'LEARY, *Deblurring Images*, SIAM, Philadelphia, 2006.

[21] P. C. HANSEN, J. G. NAGY, AND K. TIGKOS, *Rotational image deblurring with sparse matrices*, BIT Numer. Math., 54 (2014), pp. 649–671.

[22] T. A. HEARN AND L. REICHEL, *Application of denoising methods to regularization of ill-posed problems*, Numer. Algorithms, 66 (2014), pp. 761–777.

[23] ———, *Application of denoising methods to regularization of ill-posed problems*, Numer. Algorithms, 66 (2014), pp. 761–777.

[24] M. E. HOCHSTENBACH, N. MCNINCH, AND L. REICHEL, *Discrete ill-posed least-squares problems with a solution norm constraint*, Linear Algebra Appl., 436 (2012), pp. 3801–3818.

[25] K. JBILOU, A. MESSAOUDI, AND H. SADOK, *Global FOM and GMRES algorithms for matrix equations*, Appl. Numer. Math., 31 (1999), pp. 49–63.

[26] K. JBILOU, H. SADOK, AND A. TINZEFTE, *Oblique projection methods for linear systems with multiple right-hand sides*, Electron. Trans. Numer. Anal., 20 (2005), pp. 119–138.
https://etna.ricam.oeaw.ac.at/vol.20.2005/pp119-138.dir/pp119-138.pdf

[27] S. KINDERMANN, *Convergence analysis of minimization-based noise level-free parameter choice rules for linear ill-posed problems*, Electron. Trans. Numer. Anal., 38 (2011), pp. 233–257.
https://etna.ricam.oeaw.ac.at/vol.38.2011/pp233-257.dir/pp233-257.pdf

[28] S. KINDERMANN AND K. RAIK, *A simplified L-curve method as error estimator*, Electron. Trans. Numer. Anal., 53 (2020), pp. 217–238.
https://etna.ricam.oeaw.ac.at/vol.53.2020/pp217-238.dir/pp217-238.pdf

[29] A. NEUMAN, L. REICHEL, AND H. SADOK, *Algorithms for range restricted iterative methods for linear discrete ill-posed problems*, Numer. Algorithms, 59 (2012), pp. 325–331.

[30] ———, *Implementations of range restricted iterative methods for linear discrete ill-posed problems*, Linear Algebra Appl., 436 (2012), pp. 3974–3990.

[31] L. REICHEL AND G. RODRIGUEZ, *Old and new parameter choice rules for discrete ill-posed problems*, Numer. Algorithms, 63 (2013), pp. 65–87.

[32] L. REICHEL, G. RODRIGUEZ, AND S. SEATZU, *Error estimates for large-scale ill-posed problems*, Numer. Algorithms, 51 (2009), pp. 341–361.

[33] L. REICHEL AND Q. YE, *Breakdown-free GMRES for singular systems*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 1001–1021.

[34] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.

[35] C. B. SHAW, JR., *Improvement of the resolution of an instrument by numerical solution of an integral equation*, J. Math. Anal. Appl., 37 (1972), pp. 83–112.

[36] B. VITAL, *Etude de Quelques Methodes de Resolution de Problemes Lineaires de Grande Taille Sur Multiprocesseur*, PhD. Thesis, Univ. de Rennes, Rennes 1990.