SIAM J. Sci. Comput.
Vol. 38, No. 1, pp. A302–A324

BLOCK KRYLOV SUBSPACE RECYCLING FOR SHIFTED SYSTEMS WITH UNRELATED RIGHT-HAND SIDES[*]

KIRK M. SOODHALTER[†]

Abstract. Many Krylov subspace methods for shifted linear systems take advantage of the invariance of the Krylov subspace under a shift of the matrix. However, exploiting this fact in the non-Hermitian case introduces restrictions; e.g., initial residuals must be collinear and this collinearity must be maintained at restart. Thus we cannot simultaneously solve shifted systems with unrelated right-hand sides using this strategy, and all shifted residuals cannot be simultaneously minimized over a Krylov subspace such that collinearity is maintained. It has been shown that this renders them generally incompatible with techniques of subspace recycling [K. M. Soodhalter, D. B. Szyld, and F. Xue, *Appl. Numer. Math.*, 81C (2014), pp. 105–118]. This problem, however, can be overcome. By interpreting a family of shifted systems as one Sylvester equation, we can take advantage of the known "shift invariance" of the Krylov subspace generated by the Sylvester operator. Thus we can simultaneously solve all systems over one block Krylov subspace using the full orthogonalization method or GMRES-type methods, even when they have unrelated right-hand sides. Because residual collinearity is no longer a requirement at restart, these methods are fully compatible with subspace recycling techniques. Furthermore, we realize the benefits of block sparse matrix operations which arise in the context of high-performance computing applications. In this paper, we discuss exploiting this Sylvester equation point of view which has yielded methods for shifted systems which are compatible with unrelated right-hand sides. From this, we propose a recycled GMRES method for simultaneous solution of shifted systems. Numerical experiments demonstrate the effectiveness of the methods.

Key words. Krylov subspace methods, shifted systems, subspace recycling, Sylvester equations, block Krylov methods, high-performance computing, augmentation, deflation

AMS subject classifications. 6F510, 65F08

DOI. 10.1137/140998214

1. Introduction.

For a given coefficient matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$, a problem which often arises in applied mathematics is to solve multiple linear systems in which the coefficient matrix of each system differs from $\mathbf{A}$ by a scalar multiple of the identity, i.e., we must solve

$$(\mathbf{A} + \sigma_i \mathbf{I})\mathbf{x}(\sigma_i) = \mathbf{b}(\sigma_i) \quad \text{with} \quad i = 1, 2, \ldots, L, \quad \text{where} \quad \{\sigma_i\}_{i=1}^{L} \subset \mathbb{C}. \tag{1.1}$$

Such families arise in applications such as Tikhonov–Phillips regularization, lattice quantum chromodynamics, rational Krylov subspaces, diffuse optical tomography, etc. When the coefficient matrix is large and sparse, matrix-free iterative methods such as Krylov subspace methods are of interest. When we are solving multiple shifted linear systems, such as those in (1.1), Krylov subspace methods are particularly attractive because, under certain assumptions, the Krylov subspace $\mathcal{K}_j(\mathbf{A}, \mathbf{u})$ (cf. (2.2)), is invariant under scalar shift of the coefficient matrix. Specifically, we have that

$$\mathcal{K}_j(\mathbf{A} + \sigma_{i_1}\mathbf{I}, \mathbf{u}) = \mathcal{K}_j(\mathbf{A} + \sigma_{i_2}\mathbf{I}, \widetilde{\mathbf{u}}) \tag{1.2}$$

as long as $\widetilde{\mathbf{u}} = \beta\mathbf{u}$, where $\beta \in \mathbb{C} \setminus \{0\}$. This shift invariance has led to numerous methods for solving the systems in (1.1) over a single Krylov subspace; see, e.g.,

[*] Submitted to the journal's Methods and Algorithms for Scientific Computing section December 1, 2014; accepted for publication (in revised form) November 20, 2015; published electronically January 28, 2016.
http://www.siam.org/journals/sisc/38-1/99821.html

[†] Industrial Mathematics Institute, Johannes Kepler University, Altenbergerstraße 69, A-4040 Linz, Austria (kirk.soodhalter@indmath.uni-linz.ac.at).

[2, 12, 21, 22, 35, 39, 52, 54]. However, this *collinearity requirement* means, in general, we cannot use the invariance (1.2) when the right-hand sides $\{\mathbf{b}_{(\sigma_i)}\}_{i=1}^{L}$ are unrelated; and in any case, for GMRES-type methods, we cannot simultaneously minimize all residuals while maintaining collinearity [22]. Furthermore, it has been shown that the collinearity requirement causes great difficulty when incorporating shifted system solvers into the subspace recycling framework [62].

Therefore, we propose an alternative. In this paper, we recall that we can still exploit the shift invariance of the Krylov subspace but avoid the collinearity restriction by exploiting the invariance of a block Krylov subspace generated by a related Sylvester operator [17, 46, 55, 53]. By collecting initial residuals for all systems in (1.1) as columns of $\mathbf{R}_0^{\boldsymbol{\sigma}} \in \mathbb{C}^{n\times L}$ and building a block Krylov subspace, we can construct approximations for each shifted system over one block Krylov subspace according to a Petrov–Galerkin condition on each residual (e.g., GMRES [50] or full orthogonalization method (FOM) [48]). By building the block Krylov subspace from all the residuals we avoid the above discussed problems arising from a lack of collinearity. Building upon block Krylov subspace technology allows us to use existing, well-tested implementation strategies with minor modifications. By avoiding the restrictive collinearity requirement, this produces methods which can be incorporated into the subspace recycling framework [44]. Furthermore, building methods from block Krylov subspace techniques allows us to realize the benefits in communication efficiency which have been observed for block Krylov methods with their sparse block operations; see, e.g., [11, 32, 40, 45].

For clarity, it should be noted, the methods presented in this paper are *not* extensions of the shifted GMRES method [22] or the shifted FOM [54] method to block Krylov subspace to solve problems with multiple right-hand sides. Extensions of these methods to the case of multiple right-hand sides can be derived and indeed do already exist; see, e.g., [12, 66]. However, such methods still require that the columns of the block residuals for each shifted system span the same subspace (a generalization of the collinearity requirement). Thus they are not applicable to problems with unrelated right-hand sides. Here, we are treating the problem of solving multiple shifted systems by taking advantage of its equivalence to a Sylvester equation and using what has already been proven in that context (see, e.g., [46]) to build solvers satisfying the requirements necessary for compatibility with the subspace recycling framework.

The rest of this paper is organized as follows. In the next section, we review methods for shifted linear systems based upon the invariance (1.2) and discuss their restrictions. In section 3, we review the Sylvester equation formulation of a family of shifted systems and show how through an associated block Krylov subspace invariance, we get a shifted Arnoldi relation for each individual shifted system in (1.1); we further discuss how the block shift invariance leads to GMRES- and FOM-like methods which allow simultaneous projection of all residuals according to a Petrov–Galerkin condition over the block Krylov subspace from which we can build subspace recycling methods. We also discuss strategies when initial residuals are not linearly independent. In section 4, we propose a recycled GMRES method for simultaneous solution of multiple shifted systems. Further algorithmic details are also discussed. In section 5, we discuss performance of these new Sylvester-based methods as well as how one selects recycled augmentation subspaces. Numerical results demonstrating proof of concept and effectiveness of these methods are presented in section 6.

2. Preliminaries.

We begin with a brief review of Krylov subspace methods as well as techniques for solving shifted linear system and of subspace recycling techniques. Recall that in many Krylov subspace iterative methods for solving the unshifted system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{2.1}$$

with $\mathbf{A} \in \mathbb{C}^{n\times n}$, we generate an orthonormal basis for

$$\mathcal{K}_j(\mathbf{A}, \mathbf{u}) = \operatorname{span}\left\{\mathbf{u}, \mathbf{A}\mathbf{u}, \ldots, \mathbf{A}^{j-1}\mathbf{u}\right\} \tag{2.2}$$

with the Arnoldi process, where $\mathbf{u}$ is some starting vector. Let $\mathbf{V}_j \in \mathbb{C}^{n\times j}$ be the matrix with orthonormal columns generated by the Arnoldi process spanning $\mathcal{K}_j(\mathbf{A}, \mathbf{u})$. Then we have the Arnoldi relation

$$\mathbf{A}\mathbf{V}_j = \mathbf{V}_{j+1}\overline{\mathbf{H}}_j, \tag{2.3}$$

where $\overline{\mathbf{H}}_j \in \mathbb{C}^{(j+1)\times j}$ is upper Hessenberg; see, e.g., [49, section 6.3] and [59]. Let $\mathbf{x}_0$ be an initial approximation to the solution of (2.1) and $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ be the initial residual. At iteration j, we choose $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{t}_j$ with $\mathbf{t}_j \in \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$. The vector $\mathbf{t}_j$ is called a *correction* and the subspace from which it is drawn (in this case a Krylov subspace) is called a *search space*. In GMRES [50], $\mathbf{t}_j$ satisfies

$$\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{t}_j) \perp \mathbf{A}\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0),$$

where $\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{t}_j)$ is the jth residual which is equivalent to

$$\mathbf{t}_j = \underset{\mathbf{t}\in\mathcal{K}_j(\mathbf{A},\mathbf{r}_0)}{\operatorname{argmin}} \left\|\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{t})\right\|,$$

which is itself equivalent to solving the $(j+1)\times j$ minimization problem

$$\mathbf{y}_j = \underset{\mathbf{y}\in\mathbb{C}^j}{\operatorname{argmin}} \left\|\overline{\mathbf{H}}_j\mathbf{y} - \|\mathbf{r}_0\|\,\mathbf{e}_1^{(j+1)}\right\|, \tag{2.4}$$

where $\mathbf{e}_J^{(i)}$ denotes the Jth Cartesian basis vector in $\mathbb{C}^i$. We then set $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{V}_j\mathbf{y}_j$. Recall that in restarted GMRES, often called GMRES(m), we run an m-step cycle of the GMRES method and compute an approximation $\mathbf{x}_m$. We halt the process, discard $\mathbf{V}_m$, and restart with the new residual. This process is repeated until we achieve convergence.

A similar derivation leads to the related FOM [48]. Here we enforce the condition that

$$\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{t}_j) \perp \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$$

which is equivalent to solving the $j\times j$ linear system

$$\mathbf{H}_j\mathbf{y}_j = \beta\mathbf{e}_1^{(j)}, \tag{2.5}$$

where $\mathbf{H}_j \in \mathbb{C}^{j\times j}$ is simply the matrix obtained by deleting the last row of $\overline{\mathbf{H}}_j$. The iterates produced by the GMRES and FOM algorithms are closely related; see, e.g., [49, section 6.5.7] as well as [10]. As with GMRES, a restarted version of the FOM method has been proposed called FOM(m).

One downside to the restarting strategy for GMRES and FOM is that one discards all the information generated in the Krylov subspace $\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$. This leads to a characteristic delay in convergence of restarted methods [6, 34], as well as at times unpredictable convergence [18]. Krylov subspace augmentation/deflation techniques have been proposed to allow for the inclusion of vectors to the search space in addition to the Krylov subspace [5, 9, 14, 15, 19, 27, 41, 42, 35, 44, 51, 65]. Augmentation allows the user to select a subspace of $\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$ to include in the subsequent cycle. Furthermore, if one is solving a sequence of problems $\mathbf{A}_i\mathbf{x}_i = \mathbf{b}_i$, where $\Delta\mathbf{A}_i \;=\; \mathbf{A}_{i+1} \;-\; \mathbf{A}_i$

is in some sense "not large," one can use vectors generated for solving system i for augmentation when solving system $i+1$.

In this paper, we focus on the augmentation technique called subspace recycling, specifically GMRES with subspace recycling (rGMRES or GCRO-DR) [44], and we describe this method using the framework developed by Gaul in his thesis [24] which has overlap with related work undertaken with colleagues and published earlier in [25]. In principle, when one views subspace recycling methods in this framework, one sees that the framework can be wrapped around many iterative methods (though what results may not be easily implemented). Suppose we have a k-dimensional subspace $\mathcal{U}$ of vectors which is to be part of the search space. Then for an appropriately chosen projector $\mathbf{P}$, we use the Krylov subspace iteration to solve the projected subproblem

$$\mathbf{PA}\widehat{\mathbf{x}} = \mathbf{Pb}. \tag{2.6}$$

If $\mathbf{x}_0$ is an initial approximation for the full problem, then one can compute from this an initial approximation $\widehat{\mathbf{x}}_0$ for (2.6). From the iteration, we get an approximation $\widehat{\mathbf{x}}_j = \widehat{\mathbf{x}}_0 + \widehat{\mathbf{t}}_j$, where $\widehat{\mathbf{t}}_j \in \mathcal{K}_j(\mathbf{PA}, \widehat{\mathbf{r}}_0)$. To get an approximation for the original problem, we compute a correction $\widehat{\mathbf{s}}_j \in \mathcal{U}$ and set $\mathbf{x}_j = \widehat{\mathbf{x}}_j + \widehat{\mathbf{s}}_j$. How the terms $\mathbf{P}$, $\widehat{\mathbf{x}}_0$, and $\widehat{\mathbf{s}}_j$ are computed is determined by $\mathcal{U}$ and the iterative method being applied to the projected problem.

The rGMRES method [44] represents the confluence of two augmentation/deflation approaches, the harmonic Ritz vector deflation approach (GMRES-DR) of Morgan [42] and the GCRO optimal augmentation approach of de Sturler [14]. As described by Gaul [24], this method fits into the general augmentation framework. Let $\mathcal{C} = \mathbf{A} \cdot \mathcal{U}$ be the image of $\mathcal{U}$ under the action of the operator and $\mathbf{P}^{\mathcal{C}}$ be the orthogonal projector onto $\mathcal{C}$. Then rGMRES proceeds by applying a GMRES iteration to the projected subproblem

$$\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right) \mathbf{A}\widehat{\mathbf{x}} = \left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right) \mathbf{b}. \tag{2.7}$$

We compute the initial residual $\widehat{\mathbf{r}}_0 = \left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right) \mathbf{r}_0$ and the associated initial approximation $\widehat{\mathbf{x}}_0 = \mathbf{x}_0 + \mathbf{P}^{\mathcal{U}}_{\mathbf{A}^*\mathbf{A}}(\mathbf{x}_0 - \mathbf{x})$ which is computable without knowing $\mathbf{x}$, where $\mathbf{P}^{\mathcal{U}}_{\mathbf{A}^*\mathbf{A}}$ is the projector onto $\mathcal{U}$ orthogonal with respect to the inner product induced by $\mathbf{A}^*\mathbf{A}$; in other words, $\mathbf{P}^{\mathcal{U}}_{\mathbf{A}^*\mathbf{A}}$ projects onto the subspace $\mathcal{U}$ and maps vectors in $(\mathbf{A}^*\mathbf{A}\mathcal{U})^{\perp}$ to zero.[1] The linear system (2.7) is singular and consistent. At iteration j, GMRES produces the correction $\widehat{\mathbf{t}}_j$ yielding approximate solution $\widehat{\mathbf{x}}_j = \widehat{\mathbf{x}}_0 + \widehat{\mathbf{t}}_j$ to (2.7). To get an approximation for the full problem, we compute the correction $\widehat{\mathbf{s}}_j = \mathbf{P}^{\mathcal{U}}_{\mathbf{A}^*\mathbf{A}}\widehat{\mathbf{t}}_j$ and set $\mathbf{x}_j = \widehat{\mathbf{x}}_j + \widehat{\mathbf{s}}_j$. We have in this case that the residuals for the projected problem (2.7) and the full problem (2.1) are the same, i.e., $\mathbf{r}_j = \widehat{\mathbf{r}}_j$ and that the residual satisfies the Petrov–Galerkin condition

$$\mathbf{r}_j \perp \mathbf{A} \cdot \left\{\mathcal{U} + \mathcal{K}_j(\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathbf{A}, \widehat{\mathbf{r}}_0)\right\} \subset \mathcal{C} \oplus \mathcal{K}_{j+1}(\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathbf{A}, \widehat{\mathbf{r}}_0),$$

where the subspace containment arises from the modified Arnoldi relation in, e.g., [44]. Furthermore, $\widehat{\mathbf{s}}_j$ and $\widehat{\mathbf{t}}_j$ satisfy

$$\left(\widehat{\mathbf{s}}_j, \widehat{\mathbf{t}}_j\right) = \underset{\substack{\widehat{\mathbf{s}} \in \mathcal{U} \\ \widehat{\mathbf{t}} \in \mathcal{K}_j(\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathbf{A}, \widehat{\mathbf{r}}_0)}}{\operatorname{argmin}} \left\| \mathbf{b} - \mathbf{A}(\widehat{\mathbf{x}}_0 + \widehat{\mathbf{s}} + \widehat{\mathbf{t}}) \right\|.$$

Our goal here is to incorporate a strategy to simultaneously solve a family of shifted systems into the rGMRES framework. However, this has been shown to be dif-

[1] If $\mathbf{U} \in \mathbb{C}^{n\times k}$ has columns spanning $\mathcal{U}$, then $\mathbf{P}^{\mathcal{U}}_{\mathbf{A}^*\mathbf{A}} = \mathbf{U}\left(\mathbf{U}^T\mathbf{A}^*\mathbf{A}\mathbf{U}\right)^{-1}(\mathbf{A}^*\mathbf{A}\mathbf{U})^*$.

ficult. Many methods for the simultaneous solution of shifted systems take advantage of the shift invariance (1.2); see, e.g., [12, 21, 22, 23, 35, 36, 54]. In a nonsymmetric method with restarting, collinearity must be maintained at restart. In [62], this was shown to be a troublesome restriction when attempting to extend such techniques to the rGMRES setting. In the Hermitian case, this problem is alleviated as one need not restart, due to the Lanczos three-term recurrence. In this setting, Kilmer and de Sturler proposed a shifted recycled MINRES algorithm [35]. What we will show is that the strategies employed in [35] can be used in the non-Hermitian case when one uses a specific type of block Krylov subspace method to solve the family of shifted systems, such that the collinearity restriction is no longer an issue; cf. section 3.

We also note that the shift invariance (1.2) no longer holds if general preconditioning is used. However, specific polynomial preconditioners can be constructed (see, e.g., [2, 7, 33, 67]) for which shift invariance can be maintained. Since the methods proposed in this paper are built on the shift invariance of a Krylov subspace, they are indeed compatible with this type of polynomial preconditioner. However, in this paper, we treat only the unpreconditioned problem, as in, e.g., [12, 22, 54].

It should also be noted that methods have been proposed which do not rely on the shift invariance property of Krylov subspace methods. Kressner and Tobler treated the more general situation of parameter dependent linear systems where dependence on the parameter of the matrix and right-hand sides is sufficiently smooth [37]. In [60], the relationship between the shifted coefficient matrices is exploited without using the shift invariance by solving one system and projecting the other residuals in a Lanczos–Galerkin procedure.

We end this section by briefly reviewing the restarted GMRES method for shifted systems of Frommer and Glässner [22] and the restarted FOM method for shifted systems of Simoncini [54], both developed to solve (1.1).

Frommer and Glässner proposed a restarted GMRES method to solve (1.1) in the case that the initial residuals are collinear (which for clarity we denote in this paper as "sGMRES [22]"). Within a cycle, the residual for one system from (1.1) is minimized. We call this the *base residual*. Approximations for all other systems are chosen such that their residuals are collinear with the base residual. This procedure reduces to solving $L-1$ small $(m+1)\times(m+1)$ linear systems at the end of each cycle. Since all residuals are then collinear at the end of the cycle, the shift invariance of the Krylov subspace holds at the beginning of the next cycle. It is not guaranteed for all matrices and all shifts that collinear residuals can be computed; however, conditions are derived for when such residuals can be constructed. Specifically, for a positive-real matrix $\mathbf{A}$ (field of values being contained in the right half-plane), restarted GMRES for shifted linear systems computes solutions at every iteration for all real shifts $\sigma_i > 0$.

Simoncini proposed an algorithm for simultaneously solving the systems in (1.1) based on FOM(m) (which we denote for clarity in this paper as "sFOM [54]"). Due to the properties of the residual produced by the FOM algorithm, the method is conceptually simpler to describe. For each cycle, the common shift-invariant Krylov subspace is generated. For each shifted system, the approximation is computed according to the Petrov–Galerkin condition which defines FOM. Residuals produced by the FOM Petrov–Galerkin condition at step m are always collinear with the $(m+1)$st Arnoldi vector $\mathbf{v}_{m+1}$. Therefore, FOM for shifted systems produces collinear residuals by default, and the Krylov subspace remains invariant after restart. Thus, as long as the initial residuals for all shifted systems in (1.1) are collinear, the shifted FOM algorithm is applicable without modification. However, if the right-hand sides are in general unrelated, we cannot use this method to simultaneously solve all linear systems in (1.1).

3. A generalization of shifted systems.

The main tool in this paper which allows us to achieve our goals is to observe that a generalization of shifted systems has been observed and exploited in the literature [46, 53, 55], and this allows one to solve such families over block Krylov subspaces.

3.1. Sylvester equation interpretation.

Solving the family of shifted systems (1.1) is equivalent to solving the Sylvester equation

$$\mathbf{A}\mathbf{X}^{\boldsymbol{\sigma}} - \mathbf{X}^{\boldsymbol{\sigma}}\mathbf{D} = \mathbf{B}^{\boldsymbol{\sigma}}, \tag{3.1}$$

where $\mathbf{D} = \operatorname{diag}(\sigma_1, \ldots, \sigma_L) \in \mathbb{C}^{L\times L}$ and $\mathbf{X}^{\boldsymbol{\sigma}}, \mathbf{B}^{\boldsymbol{\sigma}} \in \mathbb{C}^{n\times L}$ are the block vectors containing the solutions and right-hand sides of each shifted system from (1.1), respectively. This observation has been previously discussed; see, e.g., [46, 53, 55]. Sylvester equations of this form are in many ways a natural generalization of a single shifted system. As shown in [46], for *any* $\widetilde{\mathbf{D}} \in \mathbb{C}^{L\times L}$, if for $\mathbf{F} \in \mathbb{C}^{n\times L}$ we define the Sylvester operator as $\mathcal{T} : \mathbf{F} \mapsto \mathbf{A}\mathbf{F} + \mathbf{F}\widetilde{\mathbf{D}}$ and define powers of the operator by $\mathcal{T}^i\mathbf{F} = \mathcal{T}(\mathcal{T}^{i-1}\mathbf{F})$ with $\mathcal{T}^0\mathbf{F} = \mathbf{F}$, then it has been shown that the block Krylov subspace (cf. (3.3)), generated by $\mathcal{T}$ and $\mathbf{F}$ is equivalent to that generated by the matrix $\mathbf{A}$ and $\mathbf{F}$, i.e.,

$$\mathbb{K}_j(\mathcal{T}, \mathbf{F}) = \mathbb{K}_j(\mathbf{A}, \mathbf{F}). \tag{3.2}$$

The case in which $\widetilde{\mathbf{D}} = \mathbf{D}$ is diagonal is simply a special case of this more general result. Let $\mathbf{X}_0^{\boldsymbol{\sigma}}$ be an initial approximate solution to (3.1), and $\mathbf{R}_0^{\boldsymbol{\sigma}} = \mathbf{B}^{\boldsymbol{\sigma}} - \mathbf{A}\mathbf{X}_0^{\boldsymbol{\sigma}} - \mathbf{X}_0^{\boldsymbol{\sigma}}\mathbf{D}$ the associated residual. Then one can generate the block Krylov subspace $\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$ and project (3.1) onto $\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$ and solve a projected Sylvester equation or apply some block GMRES/FOM residual constraint. This is described in, e.g., [17, 46, 55, 53]. By exploiting the shifted system structure, we also can solve a FOM- or GMRES-type subproblem individually for each shifted system over $\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$, a procedure proposed in [53] and described below. In the case $\widetilde{\mathbf{D}} = \mathbf{D}$, this is equivalent to applying the FOM/GMRES methods for Sylvester equations to (3.1).

3.2. Derivation of sbFOM and sbGMRES.

The block Krylov subspace $\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$ is a generalization of the definition of a Krylov subspace, i.e.,

$$\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}}) = \operatorname{span}\left\{\mathbf{R}_0^{\boldsymbol{\sigma}}, \mathbf{A}\mathbf{R}_0^{\boldsymbol{\sigma}}, \mathbf{A}^2\mathbf{R}_0^{\boldsymbol{\sigma}}, \ldots \mathbf{A}^{j-1}\mathbf{R}_0^{\boldsymbol{\sigma}}\right\}, \tag{3.3}$$

where the span of a sequence of block vectors is simply the span of the columns of all the blocks combined. It is straightforward to show that this definition is equivalent to

$$\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}}) = \mathcal{K}_j(\mathbf{A}, \mathbf{r}_{0(\sigma_1)}) + \mathcal{K}_j(\mathbf{A}, \mathbf{r}_{0(\sigma_2)}) + \cdots + \mathcal{K}_j(\mathbf{A}, \mathbf{r}_{0(\sigma_L)}). \tag{3.4}$$

Except in section 5.4, we assume throughout this paper that $\dim \mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}}) = jL$, which implies that the block size (i.e., the number of linearly independent right-hand sides) is the same as the number of shifts. Following the description in [49, section 6.12], we represent $\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$ in terms of the block Arnoldi basis $\{\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_j\}$, where $\mathbf{V}_i \in \mathbb{C}^{n\times L}$ has orthonormal columns and each column of $\mathbf{V}_i$ is orthogonal to all columns of $\mathbf{V}_j$ for all $j \neq i$. We obtain $\mathbf{V}_1$ via the reduced QR-factorization $\mathbf{R}_0^{\boldsymbol{\sigma}} = \mathbf{V}_1\mathbf{S}_0$, where $\mathbf{S}_0 \in \mathbb{C}^{L\times L}$ is upper triangular. Let

$$\mathbf{W}_j = \begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_2 \cdots \mathbf{V}_j\end{bmatrix} \in \mathbb{C}^{n\times jL}.$$

Let $\overline{\mathbf{H}}_j = (\mathbf{H}_{i\ell}) \in \mathbb{C}^{(j+1)L\times jL}$ be the block upper Hessenberg matrix generated by the block Arnoldi method, where $\mathbf{H}_{i\ell} \in \mathbb{C}^{L\times L}$. This yields the block Arnoldi relation

$$\mathbf{A}\mathbf{W}_j = \mathbf{W}_{j+1}\overline{\mathbf{H}}_j. \tag{3.5}$$

A straightforward generalization of GMRES for block Krylov subspaces (called block GMRES) has been described [64] and a block FOM method has been similarly derived in the context of evaluating matrix exponentials [66].

A great deal of work on the theory and implementation of block Krylov subspace methods has been published; see, e.g., [29, 31, 30, 57, 56, 60, 64]. The shift invariance properties of Krylov subspaces extend to the block setting. The following straightforward proposition directly follows from their construction.

Proposition 3.1. *The block Krylov subspace is invariant under scalar shifts of the coefficient matrix, i.e.,*

$$\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}}) = \mathbb{K}_j(\mathbf{A} + \sigma\mathbf{I}, \mathbf{R}_0^{\boldsymbol{\sigma}}) \tag{3.6}$$

with $\sigma \in \mathbb{C} \setminus \{0\}$ *and satisfies the shifted block Arnoldi relation*

$$(\mathbf{A} + \sigma\mathbf{I})\mathbf{W}_j = \mathbf{W}_{j+1}\overline{\mathbf{H}}_{j\,(\sigma)}, \tag{3.7}$$

where

$$\overline{\mathbf{H}}_{j\,(\sigma)} = \overline{\mathbf{H}}_j + \sigma \begin{bmatrix} \mathbf{I}_{jL\times jL} \\ \mathbf{0}_{L\times jL} \end{bmatrix}.$$

The block shift invariance has previously been exploited in [12, 46, 53]. It can also be seen as a special case of (3.2) with $\widetilde{\mathbf{D}} = \gamma\mathbf{I}$ for some $\gamma \in \mathbb{C} \setminus \{0\}$. If we build the block Krylov subspace $\mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$, generating the associated $\mathbf{W}_{j+1}$ and $\overline{\mathbf{H}}_j$, then we have for each initial residual the two equalities

$$\mathbf{r}_{0\,(\sigma_i)} = \mathbf{W}_j\mathbf{E}_1^{(j)}\mathbf{S}_0\mathbf{e}_i^{(L)} = \mathbf{W}_{j+1}\mathbf{E}_1^{(j+1)}\mathbf{S}_0\mathbf{e}_i^{(L)}, \tag{3.8}$$

where $\mathbf{R}_0^{\boldsymbol{\sigma}} = \mathbf{V}_1\mathbf{S}_0$ is a QR-factorization as before, $\mathbf{E}_1^{(J)} \in \mathbb{C}^{JL\times L}$ has an $L \times L$ identity matrix in the first L rows and zeros below, and $\mathbf{e}_i^{(L)}$ is the ith column of the $L \times L$ identity matrix.

One can derive a shifted FOM-type method by imposing the FOM Galerkin condition on the Sylvester equation residual. For $\widetilde{\mathbf{D}} = \mathbf{D}$, this reduces to solving a family of small shifted systems. This was shown in [53, Corollary 4.2]. For the ith shifted system, this means we compute $\mathbf{t}_{j\,(\sigma_i)} = \mathbf{W}_j\mathbf{y}_{j\,(\sigma_i)} \in \mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$ such that

$$\mathbf{b}_{(\sigma_i)} - (\mathbf{A} + \sigma_i)(\mathbf{x}_{0\,(\sigma_i)} + \mathbf{t}_{j\,(\sigma_i\mathbf{I})}) \perp \mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}}), \tag{3.9}$$

which is equivalent to solving

$$\mathbf{H}_{j\,(\sigma_i)}\mathbf{y}_{j\,(\sigma_i)} = \mathbf{E}_1^{(j)}\mathbf{S}_0\mathbf{e}_i^{(L)}. \tag{3.10}$$

We must solve L linear systems. Each system can be solved progressively, and the residual norms are available at each iteration without explicit computation of the correction. One can consider this method as a generalization of sFOM [54] since this is built on top of the shift invariance of the Sylvester operator of the block Krylov subspace which is a generalization of the shift invariance of the Krylov subspace generated by one vector. However, we reiterate that this is not simply an extension of sFOM [54] to block Krylov subspaces.

The block GMRES method for Sylvester equations is presented in [17, 46]. In [17, equation 3.1], it is shown that if we compute an approximate correction to the general Sylvester equation $\mathcal{T}(\mathbf{X}_0^{\boldsymbol{\sigma}} + \mathbf{T}^{\boldsymbol{\sigma}}) = \mathbf{B}^{\boldsymbol{\sigma}}$, where $\mathbf{T}^{\boldsymbol{\sigma}} = \mathbf{W}_j \mathbf{Y}_j \in \mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}})$ with $\mathbf{Y}_j \in \mathbb{C}^{jL \times L}$, then the resulting residual has the form

$$\mathbf{R}_j^{\boldsymbol{\sigma}} = \mathbf{B}^{\boldsymbol{\sigma}} - \mathcal{T}(\mathbf{X}_0^{\boldsymbol{\sigma}} + \mathbf{T}_j^{\boldsymbol{\sigma}}) = \mathbf{W}_{j+1} \left(\overline{\mathbf{H}}_j \mathbf{Y}_j - \mathbf{Y}_j \widetilde{\mathbf{D}} - \mathbf{E}_1^{(j)} \mathbf{S}_0 \right).$$

Thus $\|\mathbf{R}_j^{\boldsymbol{\sigma}}\|_F = \|\overline{\mathbf{H}}_j \mathbf{Y}_j - \mathbf{Y}_j \widetilde{\mathbf{D}} - \mathbf{E}_1^{(j)} \mathbf{S}_0\|_F$, and solving for the GMRES minimizer of the Sylvester problem is equivalent to solving the least-squares problem

$$\overline{\mathbf{H}}_j \mathbf{Y}_j - \mathbf{Y}_j \widetilde{\mathbf{D}} \approx \mathbf{E}_1^{(j)} \mathbf{S}_0. \tag{3.11}$$

In our case $\widetilde{\mathbf{D}} = \mathbf{D}$ is diagonal, which means that just as in the case of [53, Corollary 4.2], the residual minimization for the full Sylvester problem decouples into a residual minimization for each shifted system over the block Krylov subspace.

PROPOSITION 3.2. *In the case that* $\widetilde{\mathbf{D}} = \mathbf{D}$ *is diagonal,* (3.11) *decouples such that solving the least-squares problem* (3.11) *is equivalent to solving for each column of* $\mathbf{Y}_j$ *according to*

$$\mathbf{y}_{j(\sigma_i)} = \underset{\mathbf{y} \in \mathbb{C}^{jL}}{\operatorname{argmin}} \left\| \mathbf{E}_1^{(j+1)} \mathbf{S}_0 \mathbf{e}_i^{(L)} - \overline{\mathbf{H}}_{j(\sigma_i)} \mathbf{y} \right\| \tag{3.12}$$

with $\mathbf{Y}_j(:, i) = \mathbf{y}_{j(\sigma_i)}$.

Proof. One observes that (3.11) can be rewritten as a tensor product equation (see, e.g., [63])

$$\left(\mathbf{I}_j \otimes \overline{\mathbf{H}}_j + \mathbf{D} \otimes \mathbf{I}_m \right) \operatorname{vec}(\mathbf{Y}_j) \approx \operatorname{vec}\left(\mathbf{E}_1^{(j)} \mathbf{S}_0 \right).$$

This leads directly to the conclusion due to the diagonal structure of $\mathbf{D}$. ◻

This can be restated according to the minimum residual Petrov–Galerkin condition for the ith shifted system at iteration j, i.e., compute $\mathbf{t}_{j(\sigma_i)} = \mathbf{W}_j \mathbf{y}_{j(\sigma_i)}$ such that

$$\mathbf{b}_{(\sigma_i)} - (\mathbf{A} + \sigma_i \mathbf{I})(\mathbf{x}_{0(\sigma_i)} + \mathbf{t}_{j(\sigma_i)}) \perp (\mathbf{A} + \sigma_i \mathbf{I}) \mathbb{K}_j(\mathbf{A}, \mathbf{R}_0^{\boldsymbol{\sigma}}). \tag{3.13}$$

From (3.7) and (3.8), we have that $\mathbf{y}_{j(\sigma_i)}$ satisfies (3.12).

These least-squares problems can be solved using already well-described techniques for band upper Hessenberg matrices arising in the block Arnoldi algorithm; see, e.g., [29, 30, 31]. We must solve L such least-squares problems. We cannot simultaneously factorize them all, but we can nonetheless efficiently solve each problem at low cost using Householder reflections. As in the block GMRES case, a progressively updated least-squares residual norm is available at each iteration, and the actual correction is only constructed at the end of a cycle or upon convergence.

It should be noted that this method differs from sGMRES [22], because here we minimize every shifted residual. If we begin with collinear initial residuals, we can choose which algorithm to use; cf. section 3.3. We refer to this method as sbGMRES. An outline of sbGMRES and sbFOM is given in Algorithm 3.1.

In the case that the initial residuals are collinear, either of the above derived algorithms may fail or be unstable unless procedures are in place to handle this case.

If all initial residuals are collinear, then one could simply choose to solve (1.1) using one of the methods based on the shifted invariance (1.2), such as sFOM [54] or sGMRES [22]. This may be preferred depending on matrix dimension and structure, which determine the cost of a block matrix-vector product as compared to that of a single-vector matrix-vector product. If some initial residuals are linearly dependent, one can immediately use one of the well-established procedures to gracefully handle this situation; see section 5.4 and references therein. Here, however, we briefly explore some other methods to transform the problem such that the initial residuals are linearly independent. We note that this is an exploration and that Experiment 6.2 demonstrates that transforming the problem to apply one of these block methods may not be the best idea. In Experiment 6.2, we compare the method described in section 3.3.1 with simply using sFOM [54] or sGMRES [22]. Two of the techniques described in section 3.3 involve generating random vectors, and in Experiments 6.4 and 6.5, we investigate the effects of using random vectors.

3.3. When residuals are collinear. In the case of collinear residuals, if we want to use either of our proposed block methods, an initial procedure is necessary to produce noncollinear residuals compatible with our block methods. This can be accomplished in many different ways: by selecting a new $\mathbf{X}_0$ at random so the initial residuals are unrelated, by applying a cycle of GMRES for each shifted system over the common single-vector Krylov subspace they share, or by applying a cycle of block FOM in which the block is constructed with the common residual direction as the first column and random vectors for the other columns. These techniques produce residuals which are not collinear. We briefly expand on the latter two ideas. Choosing a random initial approximation requires no explanation. Applying a cycle of GMRES to each shifted system is deterministic. The other two techniques are built on generating random vectors and can produce different behavior for different sets of random vectors, with variable final outcomes; see Experiments 6.4 and 6.5 for details.

3.3.1. One cycle of single-vector GMRES. In the first cycle, we can generate a single-vector Krylov subspace (due to residual collinearity) and minimize all residuals over this subspace. As long as we avoid stagnation for all right-hand sides, the residuals will not be collinear at the end of the cycle.

3.3.2. sbFOM with random block vectors. We can also use a FOM iteration to obtain noncollinear residuals, but the iteration cannot be over the single-vector Krylov subspace. Shifted FOM naturally produces collinear residuals; thus, we must do something else. Suppose that for each shifted system, we have that the initial residual satisfies $\mathbf{r}_{0(\sigma_i)} = \beta_i \mathbf{v}_1$. Since all initial residuals are collinear, we can build the *block* Krylov subspace

$$\mathbb{K}_m(\mathbf{A}, \widetilde{\mathbf{R}}_0^{\boldsymbol{\sigma}}), \quad \text{where} \quad \widetilde{\mathbf{R}}_0^{\boldsymbol{\sigma}} = \begin{bmatrix} \mathbf{v}_1 & \widetilde{\mathbf{v}}_2 & \cdots & \widetilde{\mathbf{v}}_L \end{bmatrix},$$

from one normalized residual $\mathbf{v}_1$ (since they are all the same except for scaling) and some randomly generated vectors $\{\widetilde{\mathbf{v}}_i\}_{i=2}^{L}$, similarly to procedures for increasing the block size described in [4, 11, 45, 60]. This allows us to still apply the FOM Petrov–Galerkin condition with respect to a block Krylov subspace. Since the collinear residual is the first column of the block, at the end of the cycle, for the ith shifted system, we now solve a problem of the form

$$\mathbf{y}_{m(\sigma_i)} = \beta_i {\mathbf{H}_{m(\sigma_i)}}^{-1} \mathbf{E}_1^{(m)} \mathbf{S}_0 \mathbf{e}_1^{(L)}, \quad \text{where} \quad \mathbf{H}_{m(\sigma_i)} = \mathbf{H}_m + \sigma_i \mathbf{I}$$

and update

$$\mathbf{x}_{m(\sigma_i)} = \mathbf{x}_{0(\sigma_i)} + \mathbf{W}_m \mathbf{y}_{m(\sigma_i)}.$$

After the first cycle, the residuals are no longer collinear, and we proceed as before.

Algorithm 3.1. sbGMRES and sbFOM—outline

```
Input  : $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\{\sigma_i\}_{i=1}^{L} \subset \mathbb{C}$, $\mathbf{b}_{(\sigma_1)}, \dots, \mathbf{b}_{(\sigma_L)} \in \mathbb{C}^n$, $\mathbf{x}_{(\sigma_1)}, \dots, \mathbf{x}_{(\sigma_L)} \in \mathbb{C}^n$,
         $\varepsilon > 0$ the convergence tolerance, $m > 0$ a cycle length., $m_{\text{init}} > 0$ an
         initial cycle length
Output : $\mathbf{x}_{(\sigma_1)}, \dots, \mathbf{x}_{(\sigma_L)} \in \mathbb{C}^{n\times p}$ such that $\|\mathbf{b}_{(\sigma_i)} - (\mathbf{A} + \sigma_i \mathbf{I})\mathbf{x}_{(\sigma_i)}\| \le \varepsilon$ for all
         $1 \le i \le L$
for $i = 1, 2, \dots, L$ do
    $\mathbf{r}_{(\sigma_i)} = \mathbf{b}_{(\sigma_i)} - (\mathbf{A} + \sigma_i \mathbf{I})\mathbf{x}_{(\sigma_i)}$
if Initial residuals are collinear with $\mathbf{r}_{(\sigma_i)} = \beta_i \mathbf{r}_{(\sigma_1)}$ then
    Render residuals noncollinear (using a method from section 3.3)
while $\max_{1\le i\le L} \{\|\mathbf{r}_{(\sigma_i)}\|\} > \varepsilon$ do
    $\mathbf{R}^{\boldsymbol{\sigma}} \leftarrow \begin{bmatrix} \mathbf{r}_{(\sigma_1)} & \mathbf{r}_{(\sigma_2)} & \cdots & \mathbf{r}_{(\sigma_L)} \end{bmatrix} \in \mathbb{C}^{n\times L}$
    Build $\mathbb{K}_m(\mathbf{A}, \mathbf{R}^{\boldsymbol{\sigma}})$ using the block Arnoldi method (maintaining linear
    independence according to section 5.4), generating $\mathbf{W}_{m+1} \in \mathbb{C}^{n\times(m+1)L}$
    and $\overline{\mathbf{H}}_m \in \mathbb{C}^{(m+1)L\times mL}$
    for $i = 1, 2, \dots, L$ do
        if GMRES then
            $\widetilde{\mathbf{y}} \leftarrow \underset{\mathbf{y}\in\mathbb{R}^{mL}}{\operatorname{argmin}} \left\| \mathbf{E}_1^{(m+1)} \mathbf{S}_0 \mathbf{e}_i - \overline{\mathbf{H}}_{m(\sigma_i)} \mathbf{y} \right\|$
        else if FOM then
            $\widetilde{\mathbf{y}} \leftarrow \mathbf{H}_{m(\sigma_i)}{}^{-1} \mathbf{E}_1^{m} \mathbf{S}_0 \mathbf{e}_i$
        $\mathbf{x}_{(\sigma_i)} \leftarrow \mathbf{x}_{(\sigma_i)} + \mathbf{W}_m \widetilde{\mathbf{y}}$
        $\mathbf{r}_{(\sigma_i)} \leftarrow \mathbf{r}_{(\sigma_i)} - \mathbf{W}_{m+1} \overline{\mathbf{H}}_{m(\sigma_i)} \widetilde{\mathbf{y}}$
```

4. Recycling for shifted systems.

Using the Sylvester equation interpretation, we now have a method to efficiently generate a Krylov subspace and solve shifted systems without the need to enforce a collinearity restriction on the residuals. Such an iteration is thus a candidate for inclusion into the subspace recycling framework. We focus here on developing a GMRES-type method. We proceed by showing that the block Krylov subspace generated by a projected Sylvester operator has the same shift invariance as (3.2). Then we show that we can minimize each residual for each shifted system individually using the strategy described in [35].

We first observe that the invariance (3.2) holds even when we left multiply by the orthogonal projector $\mathbf{I} - \mathbf{P}^{\mathcal{C}}$. Note, we use $\mathcal{R}(\cdot)$ to denote the range of the argument.

Proposition 4.1. *Let $\mathbf{V} \in \mathbb{C}^{n\times L}$ be any block of vectors such that $\mathbf{P}^{\mathcal{C}}\mathbf{V} = \mathbf{0}$, i.e., $\mathcal{R}(\mathbf{V}) \perp \mathcal{C}$. Then the following invariance holds:*

$$\mathbb{K}_j\left(\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathcal{T}, \mathbf{V}\right) = \mathbb{K}_j\left(\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathbf{A}, \mathbf{V}\right). \tag{4.1}$$

Proof. Since $\mathcal{R}(\mathbf{V}) \perp \mathcal{C}$, we have the equalities

$$\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathcal{T}(\mathbf{V}) = \left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\left(\mathbf{A}\mathbf{V} + \mathbf{V}\widetilde{\mathbf{D}}\right) = \left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathbf{A}\mathbf{V} + \mathbf{V}\widetilde{\mathbf{D}}.$$

This equality is true for any block of L vectors in $\mathcal{C}^{\perp}$. In particular, it holds for all blocks in $\mathcal{R}\left(\left(\mathbf{I}-\mathbf{P}^{\mathcal{C}}\right)\mathcal{T}\right)$ since this projected operator by construction produces blocks orthogonal to $\mathcal{C}$. Thus it follows that

$$\left[\left(\mathbf{I}-\mathbf{P}^{\mathcal{C}}\right)\mathcal{T}\right]^{j}(\mathbf{V})=\mathcal{T}_{\mathcal{C}}^{j}(\mathbf{V}), \tag{4.2}$$

where we define $\mathcal{T}_{\mathcal{C}}:\mathbf{X}\mapsto\left(\mathbf{I}-\mathbf{P}^{\mathcal{C}}\right)\mathbf{A}\mathbf{X}+\mathbf{X}\widetilde{\mathbf{D}}$. We see that $\mathcal{T}_{\mathcal{C}}$ is also a Sylvester operator which has only the trivial null space when applied to $\mathbf{V}$ such that the columns of $\mathbf{V}$ are in $\mathcal{C}^{\perp}$ and it thus follows from (4.2) and (3.2) that

$$\mathbb{K}_{j}\left(\left(\mathbf{I}-\mathbf{P}^{\mathcal{C}}\right)\mathcal{T},\mathbf{V}\right)=\mathbb{K}_{j}(\mathcal{T}_{\mathcal{C}},\mathbf{V})=\mathbb{K}_{j}(\left(\mathbf{I}-\mathbf{P}^{\mathcal{C}}\right)\mathbf{A},\mathbf{V}),$$

which is what we sought to prove. □

It should be noted this proposition and its proof are both generalizations of a similar proposition and proof for one shifted system in [62, Proposition 1].

Thus by viewing (1.1) from the Sylvester equation point of view (3.1), we see from Proposition 4.1 that the projected Sylvester operator is shift invariant in the sense of (3.2). In order to minimize each shifted residual over an augmented Krylov subspace, we propose a hybrid approach. We take advantage of the invariance (4.1) and minimize each shifted residual individually according to the strategy in [35].

Properly generating $\widehat{\mathbf{X}}_{0}^{\boldsymbol{\sigma}}$ requires a bit of thought. In the case of rGMRES, one simply projects the initial residual orthogonally onto $\mathcal{C}^{\perp}$ and updates the initial approximation accordingly. One can do this because the projectors $\mathbf{P}^{\mathcal{C}}$ and $\mathbf{P}_{\mathbf{A}^{*}\mathbf{A}}^{\mathcal{U}}$ are computable using information one has on hand (namely, bases for $\mathcal{U}$ and $\mathcal{C}$) and without solving a linear system. This is not so in the case of a family of shifted systems (or in the more general Sylvester equation case). This was an additional concern touched upon in [62]. We can certainly efficiently apply the projector $\mathbf{I}-\mathbf{P}^{\mathcal{C}}$ to $\mathbf{R}_{0}^{\boldsymbol{\sigma}}$. However, there is no way with the information on hand to then update $\mathbf{X}_{0}^{\boldsymbol{\sigma}}$. However, individually for each shifted system, one can construct an oblique projector onto $\mathcal{C}^{\perp}$ such that an associated projector is computable without solving a linear system which allows us to compute an update of the approximation just for that shifted system. We note that the use of oblique projections in the recycling framework for non-Hermitian systems has been previously advocated by Gutknecht [28] but in a different context than what we present below. We also note that these oblique projectors are applied at most once, at the beginning of a cycle. At each iteration, we are still able to implicitely (through a Gram–Schmidt process) apply the orthogonal projector $\mathbf{I}-\mathbf{P}^{\mathcal{C}}$. Thus, instabilities arising from the repeated use of oblique projectors does not arise here. Also, we introduce the following subspace defiition, which is useful in the following proposition.

Definition 4.2. *Let the k-dimensional subspaces $\mathcal{U}$ and $\mathcal{C}$ have the relationship $\mathbf{A}\mathcal{U}=\mathcal{C}$, where for any basis $\{\mathbf{u}_{1},\ldots,\mathbf{u}_{k}\}$ of $\mathcal{U}$ we denote $\{\mathbf{c}_{1},\ldots,\mathbf{c}_{k}\}$ the basis of $\mathcal{C}$ such that $\mathbf{c}_{i}=\mathbf{A}\mathbf{u}_{i}$ for $1\leq i\leq k$. Then for any $\sigma\in\mathbb{C}\setminus\{0\}$ we define $\mathcal{C}+\sigma\mathcal{U}$ to be the subspace spanned by $\{\mathbf{c}_{1}+\sigma\mathbf{u}_{1},\ldots,\mathbf{c}_{k}+\sigma\mathbf{u}_{k}\}$.*

We note that this definition is well defined and independent of bases chosen for $\mathcal{U}$ and $\mathcal{C}$, as long as they satisfy the relationship in the defintion.

Proposition 4.3. *Let $\mathcal{U}$ and $\mathcal{C}$ be defined as above, and assume we have matrices $\mathbf{U},\mathbf{C}\in\mathbb{C}^{n\times k}$ such that $\mathcal{R}(\mathbf{U})=\mathcal{U}$, $\mathbf{C}=\mathbf{A}\mathbf{U}$, and $\mathbf{C}^{*}\mathbf{C}=\mathbf{I}_{k\times k}$. For any $\sigma\in\mathbb{C}$, suppose we have the shifted system*

$$(\mathbf{A}+\sigma\mathbf{I})\mathbf{x}(\sigma)=\mathbf{b}(\sigma)$$

with initial approximation $\mathbf{x}_0(\sigma)$ *and initial residual* $\mathbf{r}_0(\sigma)$*. If* $\mathbf{Q}_\sigma^{\mathcal{C}^\perp}$ *is the oblique projector onto* $\mathcal{C}+\sigma\mathcal{U}$ *orthogonal to* $\mathcal{C}^\perp$ *and* $\mathbf{Q}_\sigma^{\mathcal{U}}$ *is the oblique projector onto* $\mathcal{U}$ *orthogonal to* $(\mathbf{A}+\sigma\mathbf{I})^*\mathcal{C}$*, then the obliquely projected residual* $\widehat{\mathbf{r}}_0(\sigma) = \mathbf{r}_0(\sigma) - \mathbf{Q}_\sigma^{\mathcal{C}^\perp}\mathbf{r}_0(\sigma)$ *has associated approximation* $\widehat{\mathbf{x}}_0(\sigma) = \mathbf{x}_0(\sigma) + \mathbf{Q}_\sigma^{\mathcal{U}}(\mathbf{x}_0(\sigma) - \mathbf{x}(\sigma))$*, and both projections are computable using only* $\mathbf{U}$*,* $\mathbf{C}$*, and* σ *and with the inversion of a* $k \times k$ *matrix. Furthermore, we have that* $\widehat{\mathbf{r}}_0(\sigma) \perp \mathcal{C}$*.*

Proof. That the resulting residual is orthogonal to $\mathcal{C}$ can be seen by observing that the projector $\mathbf{I} - \mathbf{Q}_\sigma^{\mathcal{C}^\perp}$ projects the residual onto $\mathcal{C}^\perp$ orthogonal to $\mathcal{C}+\sigma\mathcal{U}$. Using the bases we possess for $\mathcal{U}$, $\mathcal{C}$, and $\mathcal{C}+\sigma\mathcal{U}$, we can explicitly construct

$$\mathbf{Q}_\sigma^{\mathcal{C}^\perp} = (\mathbf{C}+\sigma\mathbf{U})\left(\mathbf{C}^*(\mathbf{C}+\sigma\mathbf{U})\right)^{-1}\mathbf{C}^* \quad \text{and} \quad \mathbf{Q}_\sigma^{\mathcal{U}} = \mathbf{U}(\mathbf{C}^*(\mathbf{C}+\sigma\mathbf{U}))^{-1}\mathbf{C}^*(\mathbf{A}+\sigma\mathbf{I}),$$

where we note that $\mathbf{C}^*(\mathbf{C}+\sigma\mathbf{U}) = \mathbf{C}^*(\mathbf{A}+\sigma\mathbf{I})\mathcal{U}$ and that

$$\mathbf{U}(\mathbf{C}^*(\mathbf{C}+\sigma\mathbf{U}))^{-1}\mathbf{C}^*(\mathbf{A}+\sigma\mathbf{I})(\mathbf{x}(\sigma)-\mathbf{x}_0(\sigma)) = \mathbf{U}\left(\mathbf{C}^*(\mathbf{C}+\sigma\mathbf{U})\right)^{-1}\mathbf{C}^*\mathbf{r}_0(\sigma).$$

Thus these projections and updates can be computed using only matrices and vectors which are already available and by inverting $\mathbf{C}^*(\mathbf{C}+\sigma\mathbf{U})$, which completes the proof. □

We now have $\widehat{\mathbf{X}}_0^{\boldsymbol{\sigma}}$ and $\widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}}$ such that $\mathcal{R}(\widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}}) \perp \mathcal{C}$. We can now follow [35] and solve a small least-squares problem which allows us to compute $\widehat{\mathbf{t}}_j(\sigma)$ and $\widehat{\mathbf{s}}_j(\sigma)$ simultaneously, thereby minimizing the full residual of each shifted system over $\mathcal{U} + \mathbb{K}_j((\mathbf{I}-\mathbf{P}^{\mathcal{C}})\mathbf{A}, \widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}})$.

In the proof of Proposition 4.3, we introduced the matrices $\mathbf{U}$ and $\mathbf{C}$, the matrices whose columns are bases for $\mathcal{U}$ and $\mathcal{C}$ such that the columns of $\mathbf{C}$ are orthonormal. Let $\mathbf{W}_{j+1}$ and $\overline{\mathbf{H}}_j$ be as in section 3 but now for $\mathbb{K}_j((\mathbf{I}-\mathbf{P}^{\mathcal{C}})\mathbf{A}, \widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}})$. Furthermore, for this projected Krylov subspace, let $\mathbf{V}_1$ and $\mathbf{S}_0$ be defined as in the unprojected case so that we have the QR-factorization of the initial block residual $\widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}}$. The block version of the rGMRES method [45] is built upon the augmented Arnoldi relation

$$\mathbf{A}\begin{bmatrix}\mathbf{U} & \mathbf{W}_j\end{bmatrix} = \begin{bmatrix}\mathbf{C} & \mathbf{W}_{j+1}\end{bmatrix}\overline{\mathbf{G}}_j, \quad \text{where} \quad \overline{\mathbf{G}}_j = \begin{bmatrix}\mathbf{I} & \mathbf{B}_j \\ & \overline{\mathbf{H}}_j\end{bmatrix} \quad \text{and} \quad \mathbf{B}_j = \mathbf{C}^*\mathbf{A}\mathbf{W}_j,$$

and the recycled MINRES method for Hermitian problems [65] also has this property. In [35], the authors show that with a few modifications, a shifted Arnoldi relation can be developed which allows one to compute efficiently the minimum residual solution over the augmented Krylov subspace. The main issue that must be addressed is that $(\mathbf{A}+\sigma\mathbf{I})\mathbf{U} = \mathbf{C}+\sigma\mathbf{U}$ has columns forming a nonorthonormal basis of $\mathcal{C}+\sigma\mathcal{U}$; and furthermore, these columns are not orthogonal to $\mathbb{K}_j((\mathbf{I}-\mathbf{P}^{\mathcal{C}})\mathbf{A}, \widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}})$. In [35], the QR-factorization

$$\begin{bmatrix}\mathbf{W}_{j+1} & \mathbf{C} & \mathbf{U}\end{bmatrix} = \begin{bmatrix}\mathbf{W}_{j+1} & \mathbf{C} & \widehat{\mathbf{U}}\end{bmatrix}\begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{W}_{j+1}^*\mathbf{U} \\ \mathbf{0} & \mathbf{I} & \mathbf{C}^*\mathbf{U} \\ \mathbf{0} & \mathbf{0} & \mathbf{N}\end{bmatrix}$$

is computed so that by defining

$$\overline{\mathbf{G}}_j(\sigma) = \begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{W}_{j+1}^*\mathbf{U} \\ \mathbf{0} & \mathbf{I} & \mathbf{C}^*\mathbf{U} \\ \mathbf{0} & \mathbf{0} & \mathbf{N}\end{bmatrix}\begin{bmatrix}\overline{\mathbf{H}}_j(\sigma) & \mathbf{0} \\ \mathbf{B}_j & \mathbf{I} \\ \mathbf{0} & \sigma\mathbf{I}\end{bmatrix} = \begin{bmatrix}\overline{\mathbf{H}}_j(\sigma) & \sigma\mathbf{W}_{j+1}^*\mathbf{U} \\ \mathbf{B}_j & \mathbf{I}+\sigma\mathbf{C}^*\mathbf{U} \\ \mathbf{0} & \sigma\mathbf{N}\end{bmatrix}$$

we can derive the shifted augmented Arnoldi relation

$$(\mathbf{A}+\sigma\mathbf{I})\begin{bmatrix}\mathbf{W}_j & \mathbf{U}\end{bmatrix} = \begin{bmatrix}\mathbf{W}_{j+1} & \mathbf{C} & \widehat{\mathbf{U}}\end{bmatrix}\overline{\mathbf{G}}_{j(\sigma)}. \tag{4.3}$$

This relation is used to solve the MINRES shifted residual minimization problem. Due to the fact that the MINRES algorithm is never restarted, the loss of residual collinearity among the shifted residuals is not problematic. Since we are no longer restricted by the collinear residual requirement, we can use (4.3) to compute the minimum residual approximation of each shifted system over the augmented block Krylov subspace efficiently in the non-Hermitian case.

PROPOSITION 4.4. *For the ith shifted system from the family* (1.1), *at iteration j the minimum residual corrections* $\widehat{\mathbf{s}}_{j(\sigma_i)}\in\mathcal{U}$ *and* $\widehat{\mathbf{t}}_{j(\sigma_i)}\in\mathbb{K}_j((\mathbf{I}-\mathbf{P}^{\mathcal{C}})\mathbf{A},\widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}})$ *satisfying*

$$\left(\widehat{\mathbf{s}}_{j(\sigma_i)},\,\widehat{\mathbf{t}}_{j(\sigma_i)}\right) = \underset{\substack{\widehat{\mathbf{s}}\in\mathcal{U}\\ \widehat{\mathbf{t}}\in\mathbb{K}_j\left(\left(\mathbf{I}-\mathbf{P}^{\mathcal{C}}\right)\mathbf{A},\widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}}\right)}}{\operatorname{argmin}}\left\|\mathbf{b}_{(\sigma_i)}-\left(\mathbf{A}+\sigma_i\mathbf{I}\right)\left(\widehat{\mathbf{x}}_{0(\sigma_i)}+\widehat{\mathbf{s}}+\widehat{\mathbf{t}}\right)\right\| \tag{4.4}$$

can be computed by solving the least-squares problem

$$\left(\mathbf{y}_{j(\sigma_i)},\,\mathbf{z}_{j(\sigma_i)}\right) = \underset{y\in\mathbb{C}^{jL},\,\mathbf{z}\in\mathbb{C}^k}{\operatorname{argmin}}\left\|\mathbf{g}_{(\sigma_i)}-\overline{\mathbf{G}}_{j(\sigma_i)}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right\| \tag{4.5}$$

and setting $\widehat{\mathbf{s}}_{j(\sigma_i)}=\mathbf{U}\mathbf{z}_{j(\sigma_i)}$ *and* $\widehat{\mathbf{t}}_{j(\sigma_i)}=\mathbf{W}_j\mathbf{y}_{j(\sigma_i)}$, *where* $\mathbf{g}_{(\sigma_i)}=\widehat{\mathbf{E}}\mathbf{S}_0(:,i)\mathbf{e}_i^{(L)}$, *where* $\widehat{\mathbf{E}}\in\mathbb{C}^{(j+1)L+2k\times L}$ *has the* $L\times L$ *identity matrix and zeros below.*

Proof. We observe first that from the definitions of $\mathbf{Q}_{\sigma_i}^{\mathcal{C}^\perp}$ and $\widehat{\mathbf{x}}_{0(\sigma_i)}$ that

$$\left(\mathbf{A}+\sigma_i\mathbf{I}\right)\widehat{\mathbf{x}}_{0(\sigma_i)} = \left(\mathbf{A}+\sigma_i\mathbf{I}\right)\mathbf{x}_{0(\sigma_i)}+\mathbf{Q}_{\sigma_i}^{\mathcal{C}^\perp}\mathbf{r}_{0(\sigma_i)},$$

and for all possible pairs $\widehat{\mathbf{s}}\in\mathcal{U}$ and $\widehat{\mathbf{t}}\in\mathbb{K}_j((\mathbf{I}-\mathbf{P}^{\mathcal{C}})\mathbf{A},\widehat{\mathbf{R}}_0^{\boldsymbol{\sigma}})$, that $\widehat{\mathbf{s}}+\widehat{\mathbf{t}}=\begin{bmatrix}\mathbf{W}_j & \mathbf{U}\end{bmatrix}\left[\begin{smallmatrix}\mathbf{y}\\ \mathbf{z}\end{smallmatrix}\right]$ for some $\mathbf{y}\in\mathbb{C}^{jL}$ and $\mathbf{z}\in\mathbb{C}^k$. Thus we can rewrite

$$\left\|\mathbf{b}_{(\sigma_i)}-\left(\mathbf{A}+\sigma_i\mathbf{I}\right)\left(\widehat{\mathbf{x}}_{0(\sigma_i)}+\widehat{\mathbf{s}}+\widehat{\mathbf{t}}\right)\right\| = \left\|\widehat{\mathbf{r}}_{0(\sigma_i)}-\left(\mathbf{A}+\sigma_i\mathbf{I}\right)\begin{bmatrix}\mathbf{W}_j & \mathbf{U}\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right\|.$$

Similarly to (3.8), we can decompose the residual. Thus we have the decomposition

$$\widehat{\mathbf{r}}_{0(\sigma_i)} = \begin{bmatrix}\mathbf{W}_{j+1} & \mathbf{C} & \widehat{\mathbf{U}}\end{bmatrix}\begin{bmatrix}\mathbf{S}_0(:,i)\\ \mathbf{0}_{jL}\\ \mathbf{0}_{2k}\end{bmatrix}.$$

This yields the result since

$$\begin{aligned}\left\|\widehat{\mathbf{r}}_{0(\sigma_i)}-\left(\mathbf{A}+\sigma_i\mathbf{I}\right)\begin{bmatrix}\mathbf{W}_j & \mathbf{U}\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right\| &= \left\|\begin{bmatrix}\mathbf{W}_{j+1} & \mathbf{C} & \widehat{\mathbf{U}}\end{bmatrix}\left(\begin{bmatrix}\mathbf{S}_0(:,i)\\ \mathbf{0}_{jL}\\ \mathbf{0}_{2k}\end{bmatrix}-\overline{\mathbf{G}}_{j(\sigma_i)}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right)\right\|\\ &= \left\|\begin{bmatrix}\mathbf{S}_0(:,i)\\ \mathbf{0}_{jL}\\ \mathbf{0}_{2k}\end{bmatrix}-\overline{\mathbf{G}}_{j(\sigma_i)}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right\|,\end{aligned}$$

which allows us to transform (4.4) to (4.5) with $\mathbf{g}_{(\sigma_i)}$ so defined. ∎

We end by briefly observing that performing this minimization for each shifted system can produce residuals which are not orthogonal to $\mathcal{C}$. Thus, at restart, we must perform an additional projection of the residuals back into $\mathcal{C}^\perp$. In a method such as rGMRES [44], this would occur naturally. Here this projection incurs additional computational expense but can be done efficiently through a Gram–Schmidt-type process. We present an outline of this algorithm, called recycled shifted block GMRES (rsbGMRES), as Algorithm 4.1.

Algorithm 4.1. Recycled block shifted GMRES—outline.

```
Input  : $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\{\sigma_i\}_{i=1}^{L} \subset \mathbb{C}$, $\mathbf{b}_{(\sigma_1)}, \dots, \mathbf{b}_{(\sigma_L)} \in \mathbb{C}^n$, $\mathbf{x}_{(\sigma_1)}, \dots, \mathbf{x}_{(\sigma_L)} \in \mathbb{C}^n$,
         $\varepsilon > 0$ the convergence tolerance, $m > 0$ a cycle length, $\mathbf{U}, \mathbf{C} \in \mathbb{C}^{n\times k}$
         with $\mathbf{C}^*\mathbf{C} = \mathbf{I}_{k\times k}$
Output : $\mathbf{x}_{(\sigma_1)}, \dots, \mathbf{x}_{(\sigma_L)} \in \mathbb{C}^{n\times p}$ such that $\|\mathbf{b}_{(\sigma_i)} - (\mathbf{A} + \sigma_i\mathbf{I})\mathbf{x}_{(\sigma_i)}\| \le \varepsilon$ for all
         $1 \le i \le L$, updated subspace $\mathcal{U}$
for $i = 1, 2, \dots, L$ do
    $\mathbf{r}_{(\sigma_i)} = \mathbf{b}_{(\sigma_i)} - (\mathbf{A} + \sigma_i\mathbf{I})\mathbf{x}_{(\sigma_i)}$
while $\max_{1\le i\le L}\{\|\mathbf{r}_{(\sigma_i)}\|\} > \varepsilon$ do
    for $i = 1, 2, \dots, L$ do
        $\widehat{\mathbf{r}}_{(\sigma_i)} = \mathbf{r}_{(\sigma_i)} - \mathbf{Q}_{\sigma_i}^{\mathcal{C}^\perp}\mathbf{r}_{(\sigma_i)}$
        $\widehat{\mathbf{x}}_{(\sigma_i)} = \mathbf{x}_{(\sigma_i)} + \mathbf{Q}_{\sigma_i}^{\mathcal{U}}\left(\mathbf{x}_{\sigma_i} - \mathbf{x}_{(\sigma_i)}\right)$
    $\widehat{\mathbf{R}}^{\boldsymbol{\sigma}} \leftarrow \begin{bmatrix}\widehat{\mathbf{r}}_{(\sigma_1)} & \widehat{\mathbf{r}}_{(\sigma_2)} & \cdots & \widehat{\mathbf{r}}_{(\sigma_L)}\end{bmatrix} \in \mathbb{C}^{n\times L}$
    Build $\mathbb{K}_m\left(\left(\mathbf{I} - \mathbf{P}^{\mathcal{C}}\right)\mathbf{A}, \widehat{\mathbf{R}}^{\boldsymbol{\sigma}}\right)$ using the block Arnoldi method, generating
    $\mathbf{W}_{m+1} \in \mathbb{C}^{n\times(m+1)L}$, $\overline{\mathbf{H}}_m \in \mathbb{C}^{(m+1)L\times mL}$, and $\mathbf{B}_j \in \mathbb{C}^{k\times mL}$
    for $i = 1, 2, \dots, L$ do
        Build $\overline{\mathbf{G}}_{j(\sigma_i)}$ and compute $\mathbf{g}_{(\sigma_i)}$
        Compute $(\mathbf{y}_{m(\sigma_i)}, \mathbf{z}_{m(\sigma_i)}) = \underset{y\in\mathbb{C}^{jL},\, \mathbf{z}\in\mathbb{C}^k}{\operatorname{argmin}} \left\|\mathbf{g} - \overline{\mathbf{G}}_{j(\sigma_i)}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right\|$
        Set $\widehat{\mathbf{s}}_{m(\sigma_i)} = \mathbf{U}\mathbf{z}_{m(\sigma_i)}$ and $\widehat{\mathbf{t}}_{m(\sigma_i)} = \mathbf{W}_m\mathbf{y}_{m(\sigma_i)}$
        Set $\mathbf{x}_m = \widehat{\mathbf{x}}_{(\sigma_i)} + \widehat{\mathbf{s}}_{m(\sigma_i)} + \widehat{\mathbf{t}}_{m(\sigma_i)}$
        Set $\mathbf{r}_m = \begin{bmatrix}\mathbf{W}_{j+1} & \mathbf{C} & \widehat{\mathbf{U}}\end{bmatrix}\left(\mathbf{g}_{(\sigma_i)} - \overline{\mathbf{G}}_{j(\sigma_i)}\begin{bmatrix}\mathbf{y}_{m(\sigma_i)}\\ \mathbf{z}_{m(\sigma_i)}\end{bmatrix}\right)$
    Update $\mathcal{U}$ (and $\mathcal{C}$ if we have not converged)
```

5. Performance of the algorithms.

In this section, we discuss performance-related topics: stagnation and the relationship between the sbGMRES and sbFOM methods, residual norms of sbGMRES compared to single-vector GMRES, growth of the block size due to the number of shifts, the occurrence of linear dependence in the block Arnoldi vectors, and selection of appropriate subspaces for recycling.

5.1. Stagnation and the relationship of block GMRES and block FOM.

The sbGMRES and sbFOM are GMRES and FOM-type methods which are defined over the same subspace. The natural question arises, during a cycle, can we relate the approximations produced by sbGMRES and sbFOM in the same way that single-vector GMRES and FOM are related; see, e.g., [49, section 6.5.5]?

In the context of the Sylvester operator interpretation, such analysis has been carried out in [46]. One sees that as with GMRES and FOM, their Sylvester counterparts are closely related, in that if GMRES applied to the Sylvester equations stagnates at iteration j, then the jth iteration for FOM applied to the Sylvester equations does not exist. One can also explore the relationships of the standard block GMRES and block FOM methods. We motivate this assertion by observing that when we apply sbGMRES or sbFOM to (1.1), at iteration j, the approximation $\mathbf{X}_j(:,i)$ for the solution to the ith shifted system is the same approximation that would be produced by applying j iterations of block GMRES or block FOM, respectively, to

$$(\mathbf{A}+\sigma_i\mathbf{I})\widetilde{\mathbf{X}}=\mathbf{R}_0^{\boldsymbol{\sigma}} \tag{5.1}$$

with the single fixed σ_i and initial approximation $\widetilde{\mathbf{X}}_0=\mathbf{0}$ and taking $\mathbf{X}_j(:,i)=\widetilde{\mathbf{X}}_j(:,i)$. Thus the behavior of sbGMRES and sbFOM can also be analyzed by considering the relationship of block GMRES and block FOM when applied to (5.1). Such an analysis was carried out in a companion work to this paper [61], in which the occurrence of stagnation during an iteration of block GMRES (for some or all columns of the approximation) and its relation to block FOM is characterized.

5.2. Comparison of block GMRES to single-vector GMRES. The performance of block methods and comparisons to single-vector counterparts have been well described by many different authors; see, e.g., [31, 45, 57, 60, 64]. In those cases, the analysis assumed one coefficient matrix and multiple right-hand sides. However, much of the convergence analysis does not specifically concern the fact that the block Krylov subspace arises from multiple right-hand sides. The residual bounds in, e.g., [57], which compare a single residual minimized over a block Krylov subspace to the same residual minimized over a single-vector Krylov subspace, are simply derived from the fact that the block GMRES minimization is performed over a larger space. Thus we have the following

$$\left\|\mathbf{B}(:,i)-(\mathbf{A}+\sigma_i\mathbf{I})\mathbf{X}_m(:,i)\right\|\leq\left\|\mathbf{B}(:,i)-(\mathbf{A}+\sigma_i\mathbf{I})\widehat{\mathbf{X}}_m(:,i)\right\|,$$

where $\mathbf{X}_j$ is the approximation resulting from j iterations of sbGMRES and $\widehat{\mathbf{X}}_j$ is the block approximation which results from applying j iterations of a single-vector GMRES algorithm to each shifted system individually.

5.3. Block size growth with the number of shifts. These methods allow us to solve shifted linear systems simultaneously without a collinearity requirement and within a subspace recycling framework, but at a cost. If we assume all shifted residuals are linearly independent, and the block Arnoldi method produces no dependent vectors, we can then observe that the block size is dependent on the number of shifts. As we have stated earlier, the use of a block iteration in this context brings with it the benefits associated with high-performance parallel computing. However, as the number of shifts increases, the block size also increases, and eventually the block size will be large enough that the benefits in data movement efficiency will no longer outweigh the costs of the larger block size. Thus, we must consider what modifications can be made to accommodate this situation. The simplest would be to choose an optimal block size P and solve the shifted systems for P shifts at a time. However, an improvement on this strategy would be to solve P systems at a time and minimize the residuals of the remaining systems according to the strategy advocated in [60].

5.4. Linear dependence of block Arnoldi vectors. As with any iteration built upon a block Krylov subspace, we must address the possibility that during the iteration, a dependent Arnoldi vector may be produced. As was shown in [31], the notion of the grade of a Krylov subspace extends to the block setting. However, unlike the single-vector case, the occurrence of a dependent Arnoldi vector does not indicate that the method has achieved the grade of the block Krylov subspace (which would imply convergence). Many different strategies have been suggested for gracefully handling a dependent Arnoldi vector; see, e.g, [3, 8, 16, 20, 43].

We advocate replacing the dependent Arnoldi vector with a randomly generated vector, as in [38, 45, 47, 60]. This serves the purpose of maintaining the block size in order to continue to realize the data movement efficiencies associated with block methods. Strategies to take advantage of these block method efficiencies have been proposed in other contexts [11]. However, unlike [60], there is no need in the nonsymmetric case to generate these random vectors in advance.

5.5. Selection of recycled subspace. The rGMRES method [44] followed from the GMRES-DR method [42] in that the authors proposed to use harmonic Ritz vectors computed at the end of each cycle to augment at the start of the next cycle. The vectors used are those associated with approximations of small eigenvalues, as it is known that smaller eigenvalues can cause a delay in convergence, and if their influence can be damped, we can achieve accelerated convergence. Here we also attempt to stimulate early onset of superlinear convergence, following from the analysis in [58]. Gaul and Schlömer observed that in the case of MINRES, there seemed to be little difference in the amount of acceleration when Ritz vectors rather than harmonic Ritz are used for augmentation [26]. A criteria for the selection of an optimal subspace for recycling was proposed by de Sturler [15].

In the case of shifted systems, one must consider that we are using one augmentation space to solve a number of shifted problems with the same eigenvectors but different eigenvalues. Thus computing the Ritz or harmonic Ritz vectors associated with small approximate eigenvalues of $\mathbf{A}$ may not yield a subspace $\mathcal{U}$ that damps the influence of the small eigenvalues of $\mathbf{A} + \sigma\mathbf{I}$ for σ enough. It may behoove us to rotate for which matrix we compute Ritz vectors or compute a few vectors for multiple shifts. Also, as observed in experiments, if we compute Ritz vectors associated with the largest Ritz values and harmonic Ritz vectors associated with the smallest harmonic Ritz values with respect to the unshifted matrix $\mathbf{A}$, recycling with the Ritz vectors yields far superior performance, for the set of matrices used in these experiments.

6. Numerical experiments.

We present experiments demonstrating the performance of sbGMRES, sbFOM, and rsbGMRES and compare their performance to sFOM [54], sGMRES [22], and rGMRES. Unless otherwise stated, Ritz vectors associated with the largest Ritz values with respect to the base coefficient matrix were used for recycling in rsbGMRES. Harmonic Ritz vectors associated with the smallest harmonic Ritz values were used for recycling with rGMRES. These experiments were performed on a Mac Pro with two 2.26 GHz, Quad Core Intel Xeon processors, 12 GB of 1066 MHz DDR3 main memory running OS-X 10.10.3 using MATLAB R2014b 64-bit edition.

In all comparison experiments, we judge algorithms by iteration counts rather than timings. The matrices used in the experiments are relatively small. Thus the expense of a matrix-vector product will still be dominated by FLOPS rather than data movement. In data movement costs, it has been shown [45] that block matrix vector products are only slightly more expensive than single matrix-vector products (for moderately sized blocks). Thus for problems with millions or even billions of unknowns, the benefits of using a block method could outweigh the costs. In these

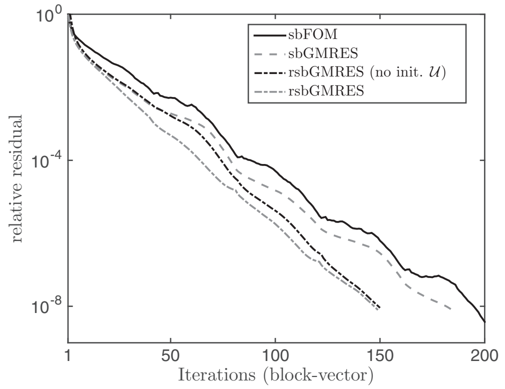

FIG. 1. *Convergence comparison between sbGMRES. sbFOM, and rsbGMRES (with and without an initial $\mathcal{U}$) for restart parameter $m = 40$, recycled subspace dimension $k = 20$, and twelve shifts for the fourth matrix from* `Group 1`*. Initial subspace $\mathcal{U}$ was generated by applying rsbGMRES to a problem with the same matrix but different right-hand side, and saving the outputted $\mathcal{U}$. We began with noncollinear residuals.*

experiments, we will not realize those benefits. Furthermore, all methods were implemented in MATLAB, and thus the overhead costs of MATLAB itself render it difficult to obtain accurate timings for these experiments. Thus, we compare iteration counts in the following experiments. To be fair, though, for Experiments 6.2 and 6.3, for our block methods, we also show the number of iterations multiplied by a *block matvec versus single matvec cost multiplier*. Our chosen number of shifts (and therefore the block size) for these experiments is 5. For matrices with the structure of those used in these experiments, multiplication by a block of 5 vectors takes roughly 3.3 times as long as when multiplying by a single vector (when making timing measurements using compiled Trilinos libraries [1]). Hence a block iteration × 3.3 is roughly comparable to a single vector iteration, and we use this multiplier to give a fairer comparison between the block methods discussed in this paper and the single-vector methods against which we test them.

We performed experiments with sets of matrices coming from a lattice quantum chromodynamics (QCD) application. These matrices, used in Experiments 6.1–6.6, are from two sets of Lattice QCD matrices (`Group 1` and `Group 2`) which are of sizes 3072×3072 and 49152×49152, respectively, available at [13]. With each such matrix $\widehat{\mathbf{A}}$ a number called κ_c is provided such that $\mathbf{I} - \widetilde{\kappa}\widehat{\mathbf{A}}$ is positive real for all $0 \leq \widetilde{\kappa} \leq \kappa_c$. We can equivalently state that the matrix $-\widehat{\mathbf{A}} + \kappa\mathbf{I}$ is positive real for all $\frac{1}{\kappa_c} \leq \kappa < \infty$. For each $\widehat{\mathbf{A}}$, we generate a base matrix $\mathbf{A} = -\widehat{\mathbf{A}} + (10^{-3} + \kappa_c)\mathbf{I}$ and choose only positive shifts to create our shifted family of linear systems. Coefficient matrices with smaller shifts yield more poorly conditioned systems requiring more iterations to solve.

6.1. Convergence curve comparison (Experiment 6.1). In Figure 1, we compare the convergence histories measured with the Frobenius norm of the relative residual of sbFOM, sbGMRES, and rsbGMRES. The matrix used is the fourth from `Group 1`, and the shifts were

$$\{.0001, .0002, .0003, .0004, .001, .002, .003, .004, .01, .02, .03, .04\}.$$

The right-hand side for each shifted system is the same randomly generated vector, and since our initial approximation for each shifted system is the zero vector, we begin

Table 1

Comparison of sbGMRES, sbFOM, and rsbGMRES with the sGMRES [22] *and the sFOM* [54] *for four matrices from* `Group 2`*. For three of the matrices, the proposed methods yield improved matvec counts, but greater costs when the block matrix-vector product cost muliplier is used. This ratio will be different for different matrices and in different computing environments. The other matrices not show yielded similar results. For rsbGMRES, Ritz vectors with respect to* $\mathbf{A}$ *associated with the largest Ritz values were recycled.*

Method	Matrix 1	Matrix 2	Matrix 3	Matrix 4
sbGMRES (mv ×3.3)	499 (1647)	1058 (3491)	442 (1459)	470 (1551)
sGMRES [22]	811	922	657	588
rsbGMRES (mv ×3.3)	457 (1508)	443 (1462)	421 (1389)	415 (1370)
sbFOM (mv ×3.3)	701 (2313)	869 (2868)	622 (2053)	600 (1980)
sFOM [54]	1138	1146	714	652

Table 2

Comparison of block srGMRES, block sGMRES, and block sFOM with their sequentially applied counterparts when each initial residuals of the shifted systems are not collinear for one matrix from `Group 2`*. Both in terms of iteration counts and when multiplying block iterations by the ratio of block matvec cost to single matvec cost, we see that our proposed methods outperform their single-vector counterparts.*

Method	Matvecs	Block Matvecs ×3.3
sbGMRES	1027	3389
rsbGMRES (Ritz)	883	2914
rsbGMRES (harmonic Ritz)	1150	3795
Sequential GMRES	6036	*
Sequential rGMRES	5111	*
sbFOM	1252	4132
Sequential FOM	6647	*

with noncollinear residuals. Observe that initially the convergence of rsbGMRES seems to be accelerated when an initial recycled subspace is given, but in the end it leads to only a one iteration improvement.

6.2. Matvec counts for collinear initial residuals (Experiment 6.2). In Table 1, we compared the performance of all five methods for solving all seven systems from `Group 2`. In this situation, we begin with collinear right-hand sides. For sbFOM and sbGMRES, an initial cycle of GMRES, as described in section 3.3.1, was performed to render the initial residual noncollinear. For rsbGMRES, this occurs naturally due to the initial oblique projections of the residuals. Right-hand sides were generated as in previous experiments, and the shifts were $\{.0001, .0002, .01, .02\}$. In addition to comparing block iteration counts versus single iteration counts, we also provide block iteration counts $\times$ 3.3 as described earlier. Using this metric, the sGMRES [22] and sFOM [54] outperform sbGMRES, sbFOM, and rsbGMRES. For all nonrecycling methods, cycle length $m = 50$. For rsbGMRES cycle length $m = 25$ and $k = 25$.

6.3. Matvec counts for unrelated initial residuals (Experiment 6.3). In Table 2, we compare the performance of the block methods when the right-hand sides are unrelated, i.e., a situation in which sFOM [54] and sGMRES [22] are not applicable. We also compare both methods against simply applying GMRES, FOM, and rGMRES sequentially to each shifted system. The right-hand side for each system is generated randomly, and again the shifts were $\{.0001, .0002, .01, .02\}$. In terms of matrix-vector product counts, the block methods are clearly superior, and this remains

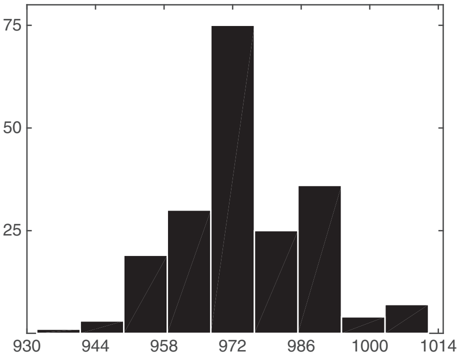

FIG. 2. *Histogram of matvec counts for different runs of sbFOM in which the initial block Krylov vector is composed of one residual and random vectors. Here we investigate the effect on and variation of performance for different random vectors. We have mean* $= 974.57$*, median* $= 972$*, mode* $= 972$*, and standard deviation* $R = 14.344$

the case when we use the block iteration cost multiplier. For unrelated right-hand sides, it is apparent that great speedups can be attained. In particular, recycling of the Ritz vectors associated with the smallest Ritz values yielded the greatest speedups for these problems, while harmonic Ritz recycling actually produced inferior results, increasing the number of iterations. Gaul [24] reported in his thesis instances of inappropriate recycled subspaces slowing down the iterations. We are not sure in this case why Ritz vector recycling so greatly outperforms harmonic Ritz vector recycling. For all methods rsbGMRES, cycle length $m = 50$. For rsbGMRES cycle length $m = 25$, and for both recycling methods $k = 25$.

6.4. Effect of random block in initial cycle of block sFOM (Experiment 6.4). Observe that if we begin with collinear residuals, we described a method in section 3.3.2 in which we apply a block FOM cycle in which the block Krylov subspace is generated by one residual and $s-1$ random vectors. The question arises, how variable is the performance of this method for different sets of random vectors? To shed light on the answer, for the first matrix from `Group 2`, the same shifts as in Experiment 6.3, the same right-hand side generated randomly for all shifted systems, and a zero vector initial approximation for all systems, we applied sbFOM implemented with this random vector strategy to these shifted systems 200 times. Since the initial residuals are collinear, an initial cycle of block FOM with random vectors is executed in this situation. We recorded the number of iterations to convergence for each experiment, each with a different set of random vectors being generated in that initial cycle. In Figure 2 we plot a histogram for the 200 iteration counts. As one can appreciate, there is a large variation in performance (≈ 60 iterations), though a plurality of the iteration counts are clustered near the mean.

6.5. Effect of random initial approximation $\mathbf{X}_0^\sigma$ (Experiment 6.5). Similarly to section 3.3, it is mentioned that one can simply generate a random initial approximation in order to produce noncollinear residuals, in the case that the right-hand sides of (1.1) are collinear. Similarly to Experiment 6.4, for the same matrix, same shifts, and randomly generated by fixed right-hand side, we solved the shifted systems 200 times using sbFOM, each time with a random starting vector, yielding a different initial residual each time. In Figure 3, we plot a histogram of the iteration counts produced by these 200 experiments.

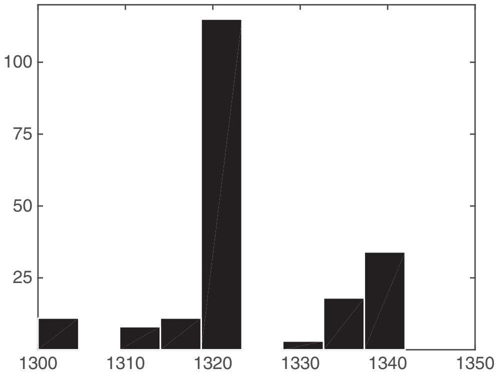

FIG. 3. *Histogram of matvec counts for different runs of sbFOM in which the initial approximation $\mathbf{X}_0^{\boldsymbol{\sigma}}$ is randomly generated. Here we investigate the effect on and variation of performance for different random vectors. We have mean* = 1324.3*, median* = 1322*, mode* = 1322*, and standard deviation* = 9.8651

6.6. Recycling when shifted system solutions depend smoothly on parameter (Experiment 6.6). In this last experiment, we demonstrate that the analysis of Kressner and Tobler [37] on parameter dependent systems for which the dependence is sufficiently smooth offers another option for augmentation subspace for the rsbGMRES method. In [37], it was shown that for a parameter dependent (family of) linear system(s)

$$\mathbf{A}\left(\sigma\right)\mathbf{x}\left(\sigma\right) = \mathbf{b}\left(\sigma\right);$$

if the dependence on the parameter on some interval (i.e., for $\sigma \in [a, b]$) is sufficiently smooth, then for all σ on this interval, $\mathbf{x}\left(\sigma\right)$ is well-approximated in a low-dimensional subspace. Thus, we can solve for a small number of parameter choices and somehow compute the rest of the solutions in this low-dimensional subspace. Some specialized iterative methods for this purpose are proposed in [37]; however, one can also use this low-dimensional subspace as a high-quality augmentation subspace and use the rsbGMRES algorithm to solve the other systems. In principle, this should require only initial residual projections, but if we compute solutions for too few parameter choices and use these solutions to augment, some further iterations for the remaining systems will likely be required.

We constructed an artificial example to test this idea. We selected the first matrix from `Group 1`, and our interval from which we selected 100 random shifts was $[1, 2]$. The shift-dependent right-hand side was constructed to be an infinitely smooth trigonometric function $\mathbf{b}\left(\sigma\right) = (\sin(\sigma\frac{j\pi}{n}))_j \in \mathbb{C}^n$. Due to the small size of $\mathbf{A}$, we can simply solve all 100 shifted systems with the MATLAB function `backslash` and compute the dimension of the subspace they span. We see that for this example, the dimension (computed numerically by taking the singular value decomposition of the matrix containing solution vectors as columns) is 38. However, in this experiment, we demonstrate that one need not compute all 38 solutions. Here, we compute 10 solutions for ten sample shifts, using sGMRES [22]. These solutions are used as an augmentation space, and the rest of the shifted systems are solved in groups of 10. In the results shown in Figure 4, we demonstrate that these initial ten solution vectors serve as an excellent augmentation space, and very few iterations are required to solve the remaining 90 shifted systems to tolerance.

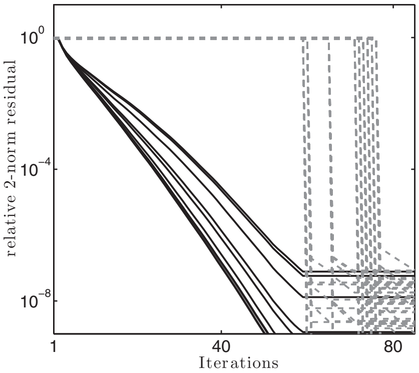

FIG. 4. *Here we have a family of shifted linear systems where the right-hand side (and thus the solution) depends smoothly on the shift. Convergence for* 100 *shifts in the interval* [1, 2]*, where we solve only* 10 *systems and apply the theory of Kressner and Tobler by using these ten solutions as the augmentation subspace for the other* 90*. Horizontal dashed residual curves indicate the algorithm is not acting upon those systems.* Black solid lines *correspond* 10 *systems first solved using sGMRES* [22]*, and* gray dashed lines *to rsbGMRES using the first* 10 *solutions as augmentation vectors. The remaining* 90 *systems were solved ten at a time.*

7. Conclusions.

We have shown that by taking advantage of the Sylvester equation interpretation of the family of shifted systems, we can solve (1.1) with methods not restricted by residual collinearity. Thus we are able to propose a recycled GMRES-type method for the simultaneous solution of the systems in (1.1). This method is built upon the "shift invariance" of the projected Sylvester operator and the shifted system recycling strategy of Kilmer and de Sturler [35]. Furthermore, by basing our new methods on block Krylov subspaces, we realize the benefits in data movement costs associated with block sparse matrix operations. Numerical experiments demonstrate both the validity of the methods and that they can outperform their single-vector counterparts.

Acknowledgments. We thank Jen Pestana for closely reading an earlier version of this manuscript and offering suggestions and corrections. We also thank Martin Gutknecht and Valeria Simoncini, who independently observed the connection between the author's initial work and the Sylvester equation interpretation, and the author thanks Valeria Simoncini for pointers to many relevant references in the literature. Lastly the author thanks the referees for many helpful comments.

REFERENCES

[1] *The Trilinos Home Page*, http://trilinos.org/ (2014).
[2] M. I. Ahmad, D. B. Szyld, and M. B. van Gijzen, *Preconditioned Multishift BiCG for $\mathcal{H}_2$-Optimal Model Reduction*, Technical report 12-06-15, Department of Mathematics, Temple University, Philadelphia, 2012.
[3] J. I. Aliaga, D. L. Boley, R. W. Freund, and V. Hernández, *A Lanczos-type method for multiple starting vectors*, Math. Comp., 69 (2000), pp. 1577–1602.
[4] J. Baglama, *Dealing with linear dependence during the iterations of the restarted block Lanczos methods*, Numer. Algorithms, 25 (2000), pp. 23–36.
[5] J. Baglama and L. Reichel, *Augmented GMRES-type methods*, Numer. Linear Algebra Appl., 14 (2007), pp. 337–350.
[6] A. H. Baker, E. R. Jessup, and T. Manteuffel, *A technique for accelerating the convergence of restarted GMRES*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 962–984.
[7] M. Baumann and M. B. van Gijzen, *Nested Krylov Methods for Shifted Linear Systems*, SISC, 37 (2015), pp. S90–S112.

[8] S. Birk and A. Frommer, *A deflated conjugate gradient method for multiple right hand sides and multiple shifts*, Numer. Algorithms, 67 (2014), pp. 507–529.

[9] M. O. Bristeau and J. Erhel, *Augmented conjugate gradient. Application in an iterative process for the solution of scattering problems*, Numer. Algorithms, 18 (1998), pp. 71–90.

[10] P. N. Brown, *A theoretical comparison of the Arnoldi and GMRES algorithms*, SIAM J. Sci. Statist. Comput., 12 (1991), pp. 58–78.

[11] A. T. Chronopoulos and A. B. Kucherov, *Block-s-step Krylov iterative methods*, Numer. Linear Algebra Appl., 17 (2010), pp. 3–15.

[12] D. Darnell, R. B. Morgan, and W. Wilcox, *Deflated GMRES for systems with multiple shifts and multiple right-hand sides*, Linear Algebra Appl., 429 (2008), pp. 2415–2434.

[13] T. A. Davis and Y. Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Softw., 38 (2011), 1.

[14] E. de Sturler, *Nested Krylov methods based on GCR*, J. Comput. Appl. Math., 67 (1996), pp. 15–41.

[15] E. de Sturler, *Truncation strategies for optimal Krylov subspace methods*, SIAM J. Numer. Anal., 36 (1999), pp. 864–889.

[16] A. A. Dubrulle, *Retooling the method of block conjugate gradients*, Electron. Trans. Numer. Anal., 12 (2001), pp. 216–233.

[17] A. El Guennouni, K. Jbilou, and A. J. Riquet, *Block Krylov subspace methods for solving large Sylvester equations*, Numer. Algorithms, 29 (2002), pp. 75–96.

[18] M. Embree, *The tortoise and the hare restart GMRES*, SIAM Rev., 45 (2003), pp. 259–266.

[19] J. Erhel and F. Guyomarc'h, *An Augmented Subspace Conjugate Gradient*, Research report RR-3278, INRIA, 1997.

[20] R. W. Freund and M. Malhotra, *A block QMR algorithm for non-Hermitian linear systems with multiple right-hand sides*, in Proceedings of the Fifth Conference of the International Linear Algebra Society (Atlanta, GA, 1995), Linear Algebra Appl. 254, North-Holland, Amsterdam, 1997, pp. 119–157.

[21] A. Frommer, BiCGStab(l) *for families of shifted linear systems*, Computing, 70 (2003), pp. 87–109.

[22] A. Frommer and U. Glässner, *Restarted GMRES for shifted linear systems*, SIAM J. Sci. Comput., 19 (1998), pp. 15–26.

[23] A. Frommer, S. Güsken, T. Lippert, B. Nöckel, and K. Schilling, *Many masses on one stroke: Economic computation of quark propagators*, Internat. J. Modern Phys. C, 6 (1995), pp. 627–638.

[24] A. Gaul, *Recycling Krylov Subspace Methods for Sequences of Linear Aystems: Analysis and Applications*, Ph.D. thesis, Technischen Universität Berlin, 2014.

[25] A. Gaul, M. H. Gutknecht, J. Liesen, and R. Nabben, *A framework for deflated and augmented Krylov subspace methods*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 495–518.

[26] A. Gaul and N. Schlömer, *Preconditioned Recycling Krylov Subspace Methods for Self-Adjoint Problems*, Electron. Numer. Anal., 44 (2015), pp. 522–547.

[27] L. Giraud, S. Gratton, X. Pinel, and X. Vasseur, *Flexible GMRES with deflated restarting*, SIAM J. Sci. Comput., 32 (2010), pp. 1858–1878.

[28] M. H. Gutknecht, *Spectral deflation in Krylov solvers: A theory of coordinate space based methods*, Electron. Trans. Numer. Anal., 39 (2012), pp. 156–185.

[29] M. H. Gutknecht and T. Schmelzer, *A QR–decomposition of block tridiagonal matrices generated by the block Lanczos process*, in Proceedings of the 17th IMACS World Congress, Paris, 2005, Ecole Central de Lille, Villeneuve d'Ascq, France, 2005.

[30] M. H. Gutknecht and T. Schmelzer, *Updating the QR decomposition of block tridiagonal and block Hessenberg matrices*, Appl. Numer. Math., 58 (2008), pp. 871–883.

[31] M. H. Gutknecht and T. Schmelzer, *The block grade of a block Krylov space*, Linear Algebra Appl., 430 (2009), pp. 174–185.

[32] M. Hoemmen, *Communication-Avoiding Krylov Subspace Methods*, Ph.D. thesis, University of California Berkeley, Berkeley, CA, 2010.

[33] B. Jegerlehner, *Krylov Space Solvers for Sparse Linear Systems*, Technical report IUHET-353, Indiana University, Bloomington, IN, 1996.

[34] W. Joubert, *On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems*, Numer. Linear Algebra Appl., 1 (1994), pp. 427–447.

[35] M. E. Kilmer and E. de Sturler, *Recycling subspace information for diffuse optical tomography*, SIAM J. Sci. Comput., 27 (2006), pp. 2140–2166.

[36] S. Kirchner, *IDR-Verfahren zur Lösung von Familien geshifteter linearer Gleichungssysteme*, Master's thesis, Bergische Universität Wuppertal, Department of Mathematics, Wuppertal, Germany, 2011.

[37] D. Kressner and C. Tobler, *Low-rank tensor Krylov subspace methods for parametrized linear systems*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 1288–1316.

[38] J. Langou, *Iterative Methods for Solving Linear Systems with Multiple Right-Hand Sides*, Ph.D. thesis, CERFACS, Toulose, France, 2003.

[39] K. Meerbergen and Z. Bai, *The Lanczos method for parameterized symmetric linear systems with multiple right-hand sides*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 1642–1662.

[40] M. Mohiyuddin, M. Hoemmen, J. Demmel, and K. Yelick, *Minimizing communication in sparse matrix solvers*, in Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, SC '09, New York, 2009, ACM, New York, 2009, 36.

[41] R. B. Morgan, *A restarted GMRES method augmented with eigenvectors*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 1154–1171.

[42] R. B. Morgan, *GMRES with deflated restarting*, SIAM J. Sci. Comput., 24 (2002), pp. 20–37.

[43] D. P. O'Leary, *The block conjugate gradient algorithm and related methods*, Linear Algebra Appl., 29 (1980), pp. 293–322.

[44] M. L. Parks, E. de Sturler, G. Mackey, D. D. Johnson, and S. Maiti, *Recycling Krylov subspaces for sequences of linear systems*, SIAM J. Sci. Comput., 28 (2006), pp. 1651–1674.

[45] M. L. Parks, K. M. Soodhalter, and D. B. Szyld, *Block Krylov Subspace Recycling*, manuscript.

[46] M. Robbé and M. Sadkane, *A convergence analysis of GMRES and FOM methods for Sylvester equations*, Numer. Algorithms, 30 (2002), pp. 71–89.

[47] M. Robbé and M. Sadkane, *Exact and inexact breakdowns in the block GMRES method*, Linear Algebra Appl., 419 (2006), pp. 265–285.

[48] Y. Saad, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp., 37 (1981), pp. 105–126.

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

[50] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[51] Y. Saad, M. Yeung, J. Erhel, and F. Guyomarc'h, *A deflated version of the conjugate gradient algorithm*, SIAM J. Sci. Comput., 21 (2000), pp. 1909–1926.

[52] A. K. Saibaba, T. Bakhos, and P. K. Kitanidis, *A flexible Krylov solver for shifted systems with application to oscillatory hydraulic tomography*, SIAM J. Sci. Comput., 35 (2013), pp. A3001–A3023.

[53] V. Simoncini, *On the numerical solution of $AX - XB = C$*, BIT, 36 (1996), pp. 814–830.

[54] V. Simoncini, *Restarted full orthogonalization method for shifted linear systems*, BIT, 43 (2003), pp. 459–466.

[55] V. Simoncini, *Computational Methods for Linear Matrix Equations*, SIAM Rev., to appear.

[56] V. Simoncini and E. Gallopoulos, *An iterative method for nonsymmetric systems with multiple right-hand sides*, SIAM J. Sci. Comput., 16 (1995), pp. 917–933.

[57] V. Simoncini and E. Gallopoulos, *Convergence properties of block GMRES and matrix polynomials*, Linear Algebra Appl., 247 (1996), pp. 97–119.

[58] V. Simoncini and D. B. Szyld, *On the occurrence of superlinear convergence of exact and inexact Krylov subspace methods*, SIAM Rev., 47 (2005), pp. 247–272.

[59] V. Simoncini and D. B. Szyld, *Recent computational developments in Krylov subspace methods for linear systems*, Numer. Linear Algebra Appl., 14 (2007), pp. 1–59.

[60] K. M. Soodhalter, *A block MINRES algorithm based on the banded Lanczos method*, Numer. Algorithms, 69 (2015), pp. 473–494.

[61] K. M. Soodhalter, *Stagnation of block GMRES and its relationship to block FOM*, preprint, arXiv:1411.7801, 2015.

[62] K. M. Soodhalter, D. B. Szyld, and F. Xue, *Krylov subspace recycling for sequences of shifted linear systems*, Appl. Numer. Math., 81C (2014), pp. 105–118.

[63] G. W. Stewart and J. Sun, *Matrix Perturbation Theory*, Academic Press, San Diego, 1990.

[64] B. Vital, *Etude de Quelques Méthodes de Résolution de Problèmes Linéaires de Grande Taille sur Multiprocesseur*, Ph.D. thesis, Université de Rennes, Rennes, France, 1990.

[65] S. Wang, E. de Sturler, and G. H. Paulino, *Large-scale topology optimization using preconditioned Krylov subspace methods with recycling*, Internat. J. Numer. Methods Engrg., 69 (2007), pp. 2441–2468.

[66] G. Wu, H. kui Pang, and J. li Sun, *Preconditioning the Restarted and Shifted Block Fom Algorithm for Matrix Exponential Computation*, preprint, arXiv:1405.0707, arXiv, 2015.

[67] G. Wu, Y.-C. Wang, and X.-Q. Jin, *A preconditioned and shifted GMRES algorithm for the PageRank problem with multiple damping factors*, SIAM J. Sci. Comput., 34 (2012), pp. A2558–A2575.