Received: 27 January 2020 | Accepted: 17 July 2020 | Published on: 20 September 2020

DOI: 10.1002/gamm.202000016

ORIGINAL PAPER

GAMM Mitteilungen

A survey of subspace recycling iterative methods

Kirk M. Soodhalter[1] | **Eric de Sturler**[2] | **Misha E. Kilmer**[3]

[1]School of Mathematics, Trinity College Dublin, Dublin, Ireland

[2]Department of Mathematics, Virginia Tech, Blacksburg, Virginia, USA

[3]Department of Mathematics, Tufts University, Medford, Massachussetts, USA

Correspondence
Kirk M. Soodhalter, School of Mathematics, Trinity College Dublin, College Green, Dublin 2, Ireland. Email: ksoodha@maths.tcd.ie

Abstract

This survey concerns *subspace recycling methods*, a popular class of iterative methods that enable effective reuse of subspace information in order to speed up convergence and find good initial vectors over a sequence of linear systems with slowly changing coefficient matrices, multiple right-hand sides, or both. The subspace information that is recycled is usually generated during the run of an iterative method (usually a Krylov subspace method) on one or more of the systems. Following introduction of definitions and notation, we examine the history of early augmentation schemes along with deflation preconditioning schemes and their influence on the development of recycling methods. We then discuss a general residual constraint framework through which many augmented Krylov and recycling methods can both be viewed. We review several augmented and recycling methods within this framework. We then discuss some known effective strategies for choosing subspaces to recycle before taking the reader through more recent developments that have generalized recycling for (sequences of) shifted linear systems, some of them with multiple right-hand sides in mind. We round out our survey with a brief review of application areas that have seen benefit from subspace recycling methods.

KEYWORDS

augmentation, deflation, Krylov subspaces, recycling

1 | INTRODUCTION

In many applications in the computational sciences, there is a need to solve many hundreds or thousands of large-scale linear systems of the form

$$\mathbf{A}^{(i)}\mathbf{x}_{\ell}^{(i)} = \mathbf{b}_{\ell}^{(i)} \quad i = 1, 2, \ldots \quad \ell = 1, 2, \ldots \tag{1}$$

where the systems indexed by i are available in sequence rather than simultaneously, and for each i all right-hand sides indexed by ℓ are available simultaneously. We consider the case where consecutive coefficient matrices are sufficiently "closely related" to exploit the relationship between them. Such sequences of problems arise in a diverse array of applications; see Section 8. The coefficient matrices are often sparse or otherwise allow efficient matrix-vector multiplication in a matrix-free fashion. Generally, the dimension of the matrix is so large that matrix-free iterative methods (eg, Krylov

Abbreviations: CG, conjugate gradients; GKB, Golub-Kahan bidiagonalization; GMRES, generalized minimum residual method; HPD, Hermitian positive-definite; LSQR, GKB-based least-squares iterative solver for tall rectangular problems.

subspace methods or multigrid) are the most viable choice for these problems. With such an iterative method in hand, the most straightforward approach would be to apply it to each consecutive linear system, with no consideration of the relationship between systems. However, speedups in convergence and good initial guesses can be achieved by exploiting the closeness of consecutive coefficient matrices.

This survey concerns *(Krylov) subspace recycling methods*, a popular class of iterative methods enabling effective reuse of subspace information generated during the run of an iterative method (usually a Krylov subspace method) applied to $\mathbf{A}^{(i)}\mathbf{x}^{(i)} = \mathbf{b}^{(i)}$, for reuse either after a cycle of iterations (for the same system) or during the iteration applied to $\mathbf{A}^{(i+1)}\mathbf{x}^{(i+1)} = \mathbf{b}^{(i+1)}$.

1.1 | Organization of the survey

In the next section, we briefly review some basic information about Krylov subspaces methods. We include some important concepts related to residual projection methods that assist in understanding recycling strategies. In Section 3, we discuss a number of precursors and related techniques. Some of these are direct forbearers to the current recycling methods, while others were proposed with different theoretical/practical concerns in mind, but they can be interpreted in the same mathematical framework. In Section 4, we give an overview of the state-of-the-art augmentation-based recycling methods. We describe a general framework, extending those proposed in [69,70,76], that can be used to described the majority of recycling approaches. In Section 5, we describe generic examples of methods in this general framework for linear systems with coefficient matrices of various structure and for different residual constraints, and we discuss effective strategies for choosing a subspace to recycle in Section 6. In Section 7, we discuss strategies to take advantage of recycling for families of systems with additional structure, in particular, solving multiple shifted systems for each coefficient matrix $\mathbf{A}^{(i)}$. We then briefly discuss in Section 8 a variety of scientific, computational, and engineering applications which have benefited from incorporating a recycling strategy into their solvers. To conclude, we discuss some challenges which remain.

1.2 | Notation

Boldface capital letters denote matrices. Boldface lowercase letters denote vectors. Nonboldface letters (Latin and Greek) denote scalar quantities; and, when necessary, we use matching nonboldface letters to denote entries of a matrix or vector denoted by the same letter (eg, $h_{ij} \in \mathrm{C}$ denotes the entries of the matrix $\mathbf{H}$). The matrices $\mathbf{P}$ and $\mathbf{Q}$ are used to denote specific projectors that arise in residual projection methods. Calligraphic letters denote subspaces, and we often use a matching uppercase boldface letter to denote a matrix having that subspace as the span of its columns (eg, the columns of $\mathbf{U} \in \mathrm{C}^{n\times k}$ span $\mathcal{U} \subset \mathbb{C}^n$ which has dimension k). When these subspaces are without a tilde above (eg, $\mathcal{U}$ and $\mathcal{V}_j$), they denote correction spaces from which updates to solution approximations are drawn. When such subspaces are written with tildes above (eg, $\tilde{\mathcal{U}}$ and $\tilde{\mathcal{V}}_j$), they denote residual constraint spaces used to determine which element is drawn from the correction space, that is, by enforcing that the new residual should be orthogonal to the constraint spaces, cf (21). All norms are assumed to be the 2-norm unless otherwise indicated.

2 | ITERATIVE PROJECTION METHODS—KRYLOV SUBSPACE METHODS

Krylov subspace iterative methods are a well-known class of methods for computing an approximation $\mathbf{t}_j$ to the initial error, $\mathbf{t}$, such that for an initial guess, $\mathbf{x}_0$, the solution satisfies $\mathbf{x} = \mathbf{x}_0 + \mathbf{t} \approx \mathbf{x}_0 + \mathbf{t}_j$, that is, we are approximating $\mathbf{t}$ which solves

$$\mathbf{A}(\mathbf{x}_0 + \mathbf{t}) = \mathbf{b}, \quad \mathbf{A} \in \mathbb{C}^{n\times n}, \quad \mathbf{b} \in \mathbb{C}^n. \tag{2}$$

For solving a linear system of the form (2) with $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$, one builds the Krylov subspace

$$\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0) = \{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ \dots \ , \mathbf{A}^{j-1}\mathbf{r}_0\}$$

iteratively (at the cost of one matrix-vector product per iteration). At iteration j, $\mathbf{t}_j \in \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$ is selected according to some constraint on the residual $\mathbf{r}_j$ and $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{t}_j$ is the jth approximation. We call $\mathbf{t}_j$ the *correction* and the space from which it is drawn the *correction space*.

2.1 | An example: The generalized minimum residual (GMRES) method

To illustrate how these methods work, we focus briefly on GMRES [128], in which we select

$$\mathbf{t}_j \in \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0) \text{ such that } \mathbf{r}_j \perp \mathbf{A}\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0). \tag{3}$$

Methods with such a residual orthogonality constraint are called *residual projection methods* because the constraint defines the projection (oblique or orthogonal) of the residual onto $\mathcal{K}_{j+1}(\mathbf{A}, \mathbf{r}_0)$, and we call $\mathbf{A}\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$ here the *constraint space*. Characterization of these methods according to residual constraints is helpful in understanding the general structure of recycling methods. This particular constraint is equivalent to solving the residual minimization problem

$$\mathbf{t}_j = \underset{\boldsymbol{\tau} \in \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)}{\operatorname{argmin}} \|\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \boldsymbol{\tau})\|, \tag{4}$$

and leads to the approximation

$$\mathbf{t}_j = \mathbf{P}_{\mathcal{K}_j}\mathbf{t} \text{ and } \mathbf{r}_j = (\mathbf{I} - \mathbf{Q}_{\mathcal{K}_j})\mathbf{r}_0, \tag{5}$$

where $\mathbf{P}_{\mathcal{K}_j}$ is the $(\mathbf{A}^*\mathbf{A})$-orthogonal projector onto $\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$, and $\mathbf{Q}_{\mathcal{K}_j}$ is the orthogonal projector onto $\mathbf{A}\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$.

2.2 | The Arnoldi orthogonalization procedure leads to a practical implementation

The Arnoldi process builds an orthonormal basis for $\mathcal{K}_{j+1}(\mathbf{A}, \mathbf{r}_0)$. Set $\mathbf{v}_1 = \beta^{-1}\mathbf{r}_0$ with $\beta = \|\mathbf{r}_0\|$. At iteration j, we compute

$$\mathbf{v}_{j+1}h_{j+1,j} = \mathbf{A}\mathbf{v}_j - \sum_{i=1}^{j}\mathbf{v}_i h_{i,j}, \quad \text{with} \quad h_{i,j} = \mathbf{v}_i^*\mathbf{A}\mathbf{v}_j \text{ and } h_{j+1,j} = \|\mathbf{A}\mathbf{v}_j - \sum_{i=1}^{j}\mathbf{v}_i h_{i,j}\|, \tag{6}$$

so that $\mathbf{v}_{j+1}$ is a unit vector. The process computes the matrices

$$\mathbf{V}_{j+1} = \begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_{j+1}\end{bmatrix} \in \mathbb{C}^{n\times(j+1)} \text{ and } \underline{\mathbf{H}}_j \in \mathbb{C}^{(j+1)\times j}, \tag{7}$$

such that $\mathbf{V}_{j+1}^*\mathbf{V}_{j+1} = \mathbf{I}$, $\operatorname{range}(\mathbf{V}_{j+1}) = \mathcal{K}_{j+1}(\mathbf{A}, \mathbf{r}_0)$, $\underline{\mathbf{H}}_j$ is an upper Hessenberg matrix with components $h_{i,j}$, and

$$\mathbf{A}\mathbf{V}_j = \mathbf{V}_{j+1}\underline{\mathbf{H}}_j = \mathbf{V}_j\mathbf{H}_j + h_{j+1,j}\mathbf{v}_{j+1}\mathbf{e}_j^*, \tag{8}$$

where $\mathbf{H}_j \in \mathbb{C}^{j\times j}$ is simply the first j rows of $\underline{\mathbf{H}}_j$. Using (8), one can reduce the minimization (4) to a smaller $(j+1)\times j$ least-squares minimization problem

$$\mathbf{y}_j = \underset{\mathbf{y}\in\mathbb{C}^j}{\operatorname{argmin}}\|\underline{\mathbf{H}}_j\mathbf{y} - \beta\mathbf{e}_1\| \text{ and } \mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j. \tag{9}$$

2.3 | The Hermitian Lanczos process

In this case, we can use the Arnoldi process above, while exploiting that $\mathbf{A}$ is Hermitian and hence $\mathbf{H}_j = \mathbf{V}_j^*\mathbf{A}\mathbf{V}_j$ is itself Hermitian. It follows that $\mathbf{H}_j$ is tridiagonal and $\mathbf{V}_j$ can be computed efficiently with the three-term recurrence,

$$\mathbf{v}_{j+1}h_{j+1,j} = \mathbf{A}\mathbf{v}_j - \mathbf{v}_j h_{j,j} - \mathbf{v}_{j-1}h_{j-1,j}, \quad \text{and} \quad \mathbf{v}_2 h_{2,1} = \mathbf{A}\mathbf{v}_1 - \mathbf{v}_1 h_{1,1} \tag{10}$$

with the $h_{i,j}$ as in (6) and $h_{j-1,j} = h_{j,j-1}$ available from the previous iteration. This recurrence is called the Lanczos recurrence, leading to the Lanczos relation (with **T** for tridiagonal)

$$\mathbf{A}\mathbf{V}_j = \mathbf{V}_{j+1}\underline{\mathbf{T}}_j = \mathbf{V}_j\mathbf{T}_j + \mathbf{v}_{j+1}h_{j+1,j}\mathbf{e}_j^*. \tag{11}$$

The short recurrence (10) leads to a great reduction in the memory requirements as we only need to store the two most recently generated Lanczos vectors. Using the thin QR-decomposition $\underline{\mathbf{T}}_j = \underline{\mathbf{G}}_j^{(j+1)\times j}\mathbf{R}_j$ to solve (9), we can write the solution update at step j as $\mathbf{t}_j = \mathbf{V}_j\mathbf{R}_j^{-1}\underline{\mathbf{G}}_j^*\mathbf{e}_1\beta$. Since the coordinate vector $\tilde{\mathbf{y}} = \underline{\mathbf{G}}_j^*\mathbf{e}_1\beta$ changes only in its last coefficient from one iteration to the next, this leads to an efficient update procedure as follows. Performing the *change of basis* $\mathbf{W}_j = \mathbf{V}_j\mathbf{R}_j^{-1}$ leads to the additional 3-term recurrence $\mathbf{w}_j = r_{j,j}^{-1}(\mathbf{v}_j - \mathbf{w}_{j-1}r_{j-1,j} - \mathbf{w}_{j-2}r_{j-2,j})$ and the update $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{t}_j = \mathbf{x}_0 + \mathbf{W}_j\tilde{\mathbf{y}} = \mathbf{x}_{j-1} + \mathbf{w}_j\tilde{y}_j$. The resulting method is called MINRES [113].

If we assume additionally that **A** is Hermitian positive definite (HPD), we can minimize the error in the **A**-norm, $\|\mathbf{t} - \mathbf{t}_j\|_{\mathbf{A}}$, which corresponds to enforcing the residual constraint $\mathbf{r}_j \perp \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$ [127, sections 6.4 and 6.7.1]. Considering again the Lanczos relation (11), we have that $\mathbf{T}_j$ is also HPD and allows the LU decomposition $\mathbf{T}_j = \mathbf{L}_j\mathbf{U}_j$ with unit bidiagonal $\mathbf{U}_j$. Now taking the change of basis $\mathbf{W}_j = \mathbf{V}_j\mathbf{U}_j^{-1}$ and $\tilde{\mathbf{y}} = \mathbf{L}_j^{-1}\mathbf{e}_1$ and eliminating the explicit Lanczos recurrence using that $\mathbf{v}_{j+1}$ is just a normalization of the residual $\mathbf{r}_j$, we obtain the celebrated method of conjugate gradients [80].

2.4 | The non-Hermitian Lanczos process

For general non-Hermitian linear systems, there exist short recurrence methods, though they generally do not lead to an orthonormal basis for the Krylov subspace [55]. Instead, one simultaneously generates dual bases for the subspaces $\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$ and $\mathcal{K}_j(\mathbf{A}^*, \hat{\mathbf{r}}_0)$ where $\hat{\mathbf{r}}_0$ is either the initial residual of a dual problem involving $\mathbf{A}^*$, $\mathbf{r}_0$ itself, or some other nonzero vector. The biorthogonal Lanczos process is a short recurrence for iteratively generating these bases simultaneously at the cost of one application of **A** and one of $\mathbf{A}^*$ per iteration,

$$\operatorname{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_j\} = \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0) \quad \text{and} \quad \operatorname{span}\{\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \ldots, \hat{\mathbf{v}}_j\} = \mathcal{K}_j(\mathbf{A}^*, \hat{\mathbf{r}}_0). \tag{12}$$

The basis vectors are constructed to satisfy $\mathbf{v}_\ell^*\hat{\mathbf{v}}_k = \delta_{\ell k}$ (hence they are *biorthogonal*). The biorthogonal Lanczos relations are as follows, where $\mathbf{T}_j$, $\hat{\mathbf{T}}_j$ are tridiagonal:

$$\mathbf{A}\mathbf{V}_j = \mathbf{V}_j\mathbf{T}_j + h_{j+1,j}\mathbf{v}_{j+1}\mathbf{e}_j^*, \quad \text{and} \quad \mathbf{A}^*\hat{\mathbf{V}}_j = \hat{\mathbf{V}}_j\hat{\mathbf{T}}_j + \hat{h}_{j+1,j}\hat{\mathbf{v}}_{j+1}\mathbf{e}_j^*. \tag{13}$$

The Petrov-Galerkin condition $\mathbf{r}_j \perp \mathcal{K}_j(\mathbf{A}^*, \tilde{\mathbf{r}}_0)$, giving rise to the biconjugate gradient (BiCG) method [60,94], requires

$$\text{Solve} \quad \mathbf{T}_j\mathbf{y}_j = \beta\mathbf{e}_1 \quad \text{and set} \quad \mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j. \tag{14}$$

Similar to (8) for Hermitian systems, (13) allow us to define efficient short recurrence iterative methods for non-Hermitian systems using the LU decomposition $\mathbf{T}_j = \mathbf{L}_j\mathbf{U}_j$, change of basis $\mathbf{W}_j = \mathbf{V}_j\mathbf{U}_j^{-1}$, and setting $\tilde{\mathbf{y}} = \mathbf{L}_j^{-1}(\beta\mathbf{e}_1)$ (in general, $\mathbf{T}_j$ and $\hat{\mathbf{T}}_j$ are closely related, and only one LU-decomposition is computed). An analog to GMRES/MINRES also exists in this setting, called the quasi minimum residual (QMR) method [64]. At iteration j, this is leads to

$$\text{Minimize} \quad \|\underline{\mathbf{T}}_j\mathbf{y}_j - \beta\mathbf{e}_1\|, \quad \text{and set} \quad \mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j. \tag{15}$$

In cases where the action of $\mathbf{A}^*$ is unavailable, so-called *transpose-free* variants are available [63,141,150]. The BiCGStab(ℓ) [140] method was introduced to stabilize the oscillatory convergence pattern often exhibited by BiCG [150]. This is accomplished by alternating between ℓ steps of BiCG and an ℓ-cycle of GMRES, effectively building a hybrid residual polynomial from the BiCG and GMRES polynomials. Another related method is IDR(s) [142], which was shown to fit into the Petrov-Galerkin residual projection framework [139] and also to be a generalization of BiCGStab(ℓ).

3 | SUBSPACE AUGMENTATION AND THE STRATEGY OF DEFLATION

Subspace recycling extends ideas from the last few decades for the preservation of information between iteration cycles or between different linear systems. This is done to mitigate the effects of discarding basis vectors due to memory requirements as well as to accelerate the convergence of an iterative method. A number of acceleration strategies occupy the same or a similar theoretical framework as augmentation-based subspace recycling, and we briefly touch upon these as well.

What we mean by augmentation: In this survey, we use the terms *subspace augmentation* and *augmented method* to refer to any iterative method (here, a Krylov subspace method) which uses a sum correction space $\mathcal{U} + \mathcal{V}_j$, where $\mathcal{V}_j$ is the space generated by the iterative method, and $\mathcal{U}$ is the fixed *augmentation space*. Later, we discuss cases where $\mathcal{V}_j$ is a Krylov subspace, and one can consider the case in which the Krylov subspace is generated by the coefficient matrix/residual pair $(\mathbf{A}, \mathbf{r}_0)$ vs cases in which the Krylov subspace is generated by a projected pair of the form $((\mathbf{I}-\mathbf{Q})\,\mathbf{A}, (\mathbf{I}-\mathbf{Q})\,\mathbf{r}_0)$, cf Section 4.

3.1 | Warm restarting and simple information reuse

The simplest information-reuse approach is called *warm restarting*. As we generate approximate solutions for the sequence of problems (1) we use the approximate solution for $\mathbf{A}^{(i)}\mathbf{x}^{(i)} = \mathbf{b}^{(i)}$ as the initial approximation $\mathbf{x}_0$ for the system $\mathbf{A}^{(i+1)}\mathbf{x}^{(i+1)} = \mathbf{b}^{(i+1)}$. As consecutive systems in (1) are close (eg, consecutive systems in a nonlinear optimization scheme), the approximate solution to system i is likely a high-quality approximation for the solution to the system $i+1$.

For nonsymmetric systems, the first mention of a (nontrivial) strategy to select direction vectors for orthogonalization (to the best of our knowledge) is in [82], where the authors performed a numerical study of algorithms associated with preconditioned conjugate gradients. The authors suggest that for non-symmetric problems and a fixed (but sufficient) length recurrence, convergence is improved by retaining the search directions associated with the (relative) largest orthogonalization coefficients.

3.2 | Augmentation via flexible preconditioning

Augmentation of a Krylov subspace was proposed in a 1997 paper, which presented the idea in the framework of the flexible GMRES method [32]. Flexible GMRES is a modification of right-preconditioned GMRES that accommodates the use of a different preconditioner at each step. Recall that for a fixed right preconditioner $\mathbf{M}$, one applies an iterative solver to the problem $\mathbf{A}\mathbf{M}^{-1}\mathbf{y} = \mathbf{b}$ where $\mathbf{y} = \mathbf{M}\mathbf{x}$, and for an approximation $\tilde{\mathbf{y}}$ one generates the approximation $\tilde{\mathbf{x}} = \mathbf{M}^{-1}\tilde{\mathbf{y}}$.

Flexible preconditioning complicates this situation. In flexible GMRES, one must store the Arnoldi vectors from (7) for a Krylov subspace $\mathcal{K}_j(\mathbf{A}\mathcal{M}^{-1}, \mathbf{r}_0)$, where $\mathcal{M}$ is an unknown implicitly induced preconditioner[1], as well as a set of flexibly preconditioned Arnoldi vectors $\mathbf{Z}_j = \begin{bmatrix}\mathbf{z}_1 & \mathbf{z}_2 & \dots & \mathbf{z}_j\end{bmatrix}$, with $\mathbf{z}_i = \mathbf{M}_i^{-1}\mathbf{v}_i$, spanning $\mathcal{M}^{-1}\mathcal{K}_j(\mathbf{A}\mathcal{M}^{-1}, \mathbf{r}_0)$. One solves the GMRES minimization (9) for $\mathbf{y}_j$ as before but then updates $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{Z}_j\mathbf{y}_j$.

The authors of [32] pointed out that one can use this framework to augment an existing Krylov subspace. Assume we have run $j-k$ iterations of GMRES and $\mathbf{U} = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_k\end{bmatrix} \in \mathbb{C}^{n\times k}$ spans a k-dimensional subspace $\mathcal{U}$ to use in concert with the Krylov subspace for a correction. For the next k iterations, one can implicitly define the action of unspecified flexible preconditioners $\mathbf{M}_{j-k+i}$ with the mappings $\mathbf{M}_{j-k+i}^{-1}\mathbf{v}_{j-k+i} = \mathbf{u}_i$ for $i = 1, 2, \dots, k$. This defines a version of flexible GMRES that minimizes the residual using the constraint space $\mathbf{A}\mathcal{K}_{j-k}(\mathbf{A}, \mathbf{r}_0) + \mathbf{A}\,\mathcal{U}$. This minimization is **not** over a direct sum of spaces. The lack of orthogonality between the subspaces reduces its effectiveness. A similar performance penalty due to lack of orthogonality was the impetus for the development of the GCROT method [39]. Augmentation-type methods for ill-posed problems have been developed that rest on similar ideas [44].

3.3 | Deflation

Deflation-type methods use some of the same underlying mathematical principles as recycling methods, but they are couched in a slightly different language and aimed at different goals. Deflation techniques were originally proposed to

[1]It was pointed out in [110] that flexible preconditioning implicitly induces a fixed preconditioner $\mathcal{M}$ defined by $\mathcal{M}^{-1}\mathbf{v}_i = \mathbf{M}_i^{-1}\mathbf{v}_i$, for $i = 1, 2, \dots, j$.

treat a single system via left preconditioning using a projector onto the orthogonal complement of some subspace [109]. The convergence is then determined by a projected operator and right-hand side, and projecting away from an appropriate subspace (eg, some invariant subspace of the coefficient matrix), leads to faster convergence.

In [53] the authors present a technique to analyze methods such as the deflated restarting method of Morgan [104] (discussed in greater detail in Section 4). They interpret such deflated methods (specifically balancing preconditioners [97]) as being preconditioned with *deflation preconditioners* that project into an associated augmentation space. For example, suppose we are solving (2), where $\mathbf{t}$ represents the initial error. Let $\mathbf{U} \in \mathbb{C}^{n\times k}$ and $\mathcal{U} = \text{range}(\mathbf{U})$ be the subspace of dimension k from which we want to construct an approximation to $\mathbf{t}$. In addition, let $\tilde{\mathbf{U}} \in \mathbb{C}^{n\times k}$ and $\tilde{\mathcal{U}} = \text{range}(\tilde{\mathbf{U}})$ be the constraint space, so that $\mathbf{Us}$ satisfies $\mathbf{r} = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{Us}) \perp \tilde{\mathcal{U}}$ or equivalently $\tilde{\mathbf{U}}^*\mathbf{r} = \tilde{\mathbf{U}}^*(\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{Us})) = \mathbf{0}$. $\mathbf{P}$ denotes the corresponding (oblique or orthogonal) projector onto $\mathcal{U}$, that is, $\mathbf{Pt} = \mathbf{Us}$, which implies $\mathbf{P} = \mathbf{U}(\tilde{\mathbf{U}}^*\mathbf{AU})^{-1}\tilde{\mathbf{U}}^*\mathbf{A}$, and $\mathbf{Q}$ denotes the sibling projector[2] onto $\mathbf{A}\mathcal{U}$ that satisfies

$$\mathbf{QA} = \mathbf{AP}. \tag{16}$$

The deflation-preconditioning process splits the initial error:

$$\mathbf{t} = \mathbf{Pt} + (\mathbf{I} - \mathbf{P})\mathbf{t}. \tag{17}$$

Since $\mathbf{At} = \mathbf{r}_0$, we can directly compute $\mathbf{Pt} = \mathbf{U}(\tilde{\mathbf{U}}^*\mathbf{AU})^{-1}\tilde{\mathbf{U}}^*\mathbf{r}_0$, which can be interpreted as an approximation of the initial error in the space $\mathcal{U}$. It remains then to approximate $(\mathbf{I} - \mathbf{P})\mathbf{t}$, for which we have the following system of equations,

$$\mathbf{A}(\mathbf{x}_0 + \mathbf{t}) = \mathbf{A}(\mathbf{x}_0 + \mathbf{Pt} + (\mathbf{I} - \mathbf{P})\mathbf{t}) = \mathbf{b} \Leftrightarrow \mathbf{A}(\mathbf{I} - \mathbf{P})\mathbf{t} = \mathbf{b} - \mathbf{Ax}_0 - \mathbf{APt} = \mathbf{r}_0 - \mathbf{APt} \quad \Leftrightarrow$$

$$\mathbf{A}(\mathbf{I} - \mathbf{P})\mathbf{t} = \mathbf{r}_0 - \mathbf{QAt} = (\mathbf{I} - \mathbf{Q})\mathbf{r}_0 \Leftrightarrow (\mathbf{I} - \mathbf{Q})\mathbf{At} = (\mathbf{I} - \mathbf{Q})\mathbf{r}_0. \tag{18}$$

Hence, we solve the deflation-preconditioned linear system

$$(\mathbf{I} - \mathbf{Q})\mathbf{At} = (\mathbf{I} - \mathbf{Q})\mathbf{r}_0. \tag{19}$$

If we apply an iterative method to (19), which returns an approximation $\mathbf{t}_j$ at iteration j, then the full approximation at this step is $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{Pt} + (\mathbf{I} - \mathbf{P})\,\mathbf{t}_j$. The residual of this approximation satisfies $\mathbf{r}_j = \mathbf{b} - \mathbf{Ax}_j = (\mathbf{I} - \mathbf{Q})(\mathbf{b} - \mathbf{Ax}_j)$. Thus it is equal to the residual of the projected problem, meaning residual convergence is completely determined by properties of the projected problem (19). In the PhD thesis [69], it was confirmed that GCRO-based recycling schemes are indeed equivalent to particular deflation schemes for certain choices of *orthogonal* projectors $\mathbf{P}$ and $\mathbf{Q}$.

Methods such as those in [2,148] approach the same strategy by performing a one-time projection away from specific accurately computed eigenvectors to remove their influence from the iteration. This does not quite fit into the recycling framework as we have described it in this survey; but, as with deflation, it is based on some of the same ideas.

3.4 | Multigrid and domain decomposition

The correction $\mathbf{Pt} + (\mathbf{I} - \mathbf{P})\,\mathbf{t}_j$ comes from the space $\mathcal{U} + \mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{r}_0)$. This augmented subspace interpretation of deflation-based preconditioning can be widened to also include methods involving restrictions onto and prolongation from subgrids or coarser grids such as domain decomposition and multigrid. This has been observed, for example, in [43]. Consider, a simple restriction-solution-prolongation procedure, such as a one-level multigrid V-cycle or a domain decomposition subdomain solve. Let $\mathfrak{R} \in \mathbb{C}^{k\times n}$ be the restriction operator associated with the procedure, and let $\mathfrak{P} \in \mathbb{C}^{n\times k}$ be the associated prolongation operator. Then the restriction-solution-prolongation procedure applied to (2) can be represented as

$$\hat{\mathbf{x}} = \mathbf{x}_0 + \mathfrak{P}(\mathfrak{R}\mathbf{A}\mathfrak{P})^{-1}\mathfrak{R}\mathbf{r}_0 \quad \text{and} \quad \hat{\mathbf{r}} = \mathbf{r}_0 - \mathbf{A}\mathfrak{P}(\mathfrak{R}\mathbf{A}\mathfrak{P})^{-1}\mathfrak{R}\mathbf{r}_0. \tag{20}$$

[2]We call these sibling projectors because specifying one of them in a projection method determines the other unambiguously. Defining a projection method via a specific projector $\mathbf{P}$ being applied to the error also determines $\mathbf{Q}$. Vice versa, defining a method via the application of a projector $\mathbf{Q}$ to the residual determines $\mathbf{P}$.

If we further assume that $\mathbf{A}$ is Hermitian positive-definite, and that the restriction and prolongation operators satisfy the relationship $\mathfrak{R} = c\mathfrak{P}^*$, where $c \in \mathbb{C}$, then (20) reduces to

$$\hat{\mathbf{x}} \leftarrow \mathbf{x}_0 + \mathbf{P}\mathbf{t} \quad \text{and} \quad \hat{\mathbf{r}} \leftarrow \mathbf{r}_0 - \mathbf{Q}\mathbf{r}_0.$$

where $\mathbf{P}, \mathbf{Q}$ are projectors of the form $\mathbf{P} = \mathfrak{P}(\mathfrak{P}^*\mathbf{A}\mathfrak{P})^{-1}\mathfrak{P}^*\mathbf{A}$ and $\mathbf{Q} = \mathbf{A}\mathfrak{P}(\mathfrak{P}^*\mathbf{A}\mathfrak{P})^{-1}\mathfrak{P}^*$. These do satisfy (16), and can be interpreted in the deflation-augmentation framework, where the augmentation space is now some sort of interpolation or restricted subdomain space. Indeed, it follows from the fact that this restriction-solution-prolongation minimizes the error in the $\mathbf{A}$ -norm that $\mathcal{U} = \text{range}(\mathfrak{P}) = \tilde{\mathcal{U}}$. The projectors $\mathbf{P}$ and $\mathbf{Q}$ are the $\mathbf{A}$-orthogonal projector onto range($\mathfrak{P}$) and the $\mathbf{A}^{-1}$-orthogonal projector onto range($\mathbf{A}\mathfrak{P}$), respectively. For a general matrix $\mathbf{A}$, the procedure becomes nonoptimal with $\mathbf{P}$ being the oblique projector onto $\mathcal{U}$ along $(\mathbf{A}^*\mathcal{U})^\perp$, and $\mathbf{Q}$ is the projector onto $\mathbf{A}\mathcal{U}$ along $\mathcal{U}^\perp$ (see [43] and references therein).

3.5 | Bordering methods

Bordering methods were originally introduced as a means to solve a linear system by augmenting its rows and columns in a particular way to induce a singular consistent problem whose family of solutions have within them the solution to the original problem [11,56]. A straightforward explanation can be found in the presentation [98]. Looking at the details, one sees elements of something that looks like augmentation and is also a precursor to deflation preconditioning.

3.6 | Polynomial preconditioning

Some types of polynomial preconditioning and hybrid approaches [54,86,107,137] have a strong connection to deflation methods and augmentation for a single linear system. In such methods, one runs a cycle of GMRES for a given system. After this cycle, a fixed polynomial is constructed implicitly using information from the Krylov subspace. This polynomial can either be used with a stationary iterative method or as a preconditioner for an iterative method such as GMRES.

4 | SUBSPACE RECYCLING

4.1 | When is an augmented method subspace recycling?

In this survey, we focus on a specific type of augmented Krylov subspace methods called *Krylov subspace recycling methods*, a moniker that communicates that the additional correction subspace was *recycled* from a previously generated but discarded subspace for its importance in obtaining a fast convergence rate or good initial vectors. For example, at the end of a GMRES cycle, before restarting, retaining a subspace of the generated Krylov space may accelerate convergence in the next cycle. Or, the recycled subspace may have been determined to damp the influence of certain parts of the spectrum of the operator [104,115]. *An important aspect of subspace recycling is that, in addition to augmentation, the Krylov space is changed to complement the recycle space.* As with deflation, the new Krylov space is generated by the operator $(\mathbf{I} - \mathbf{Q})\mathbf{A}$ and a projected residual or right-hand side. The importance of this was observed in [38] proposing the GCRO method(s); this generally avoids the (possible) stagnation problems of restarted GMRES, generates the right search space given the existing correction space, and computes the optimal solution over the sum of the correction recycle space and the generated Krylov space.

The main ideas behind Krylov subspace recycling arose from the problem that optimal Krylov methods for a fixed nonsymmetric system need all Krylov iteration (or direction) vectors in each iteration. This leads to excessive storage requirements (linear in the number of iterations) and computational work (quadratic in the number of iterations). A practical solution is to restart every m iterations or to truncate the set of iteration vectors and orthogonalize only against the last m, previous iterations, leading to methods such as GMRES(m) [128], truncated GCR(m) [48], DIOM and DQGMRES [127]. While these strategies often work well, they can lead to very slow convergence.

4.2 | Augmentation based on ideas of deflation

Based on the ideas by Nicolaides for deflated CG [109], several researchers, for example, Morgan [102,103], Karchenko and Yeremin [88], Erhel and collaborators [25,51], Baglama et al [12], and Frank and Vuik [61] proposed to use augmentation with (approximate) eigenvectors to maintain good convergence after restarting. Vuik and collaborators have also considered deflation preconditioned methods based on a priori considerations for good deflation spaces, based on algorithmic or application information [85,132]. These methods may use augmentation or a preconditioning approach to achieve the desired effect. Subsequently, Morgan proposed several improvements to the augmentation approach in [102], leading to the elegant GMRES-DR method that exploits implicit restarting of the Arnoldi recurrence to use deflation [104]. Just as for GCRO, rather than augmenting the standard Krylov subspace generated by GMRES, as in [102], Morgan proposes to augment a Krylov subspace generated by $(\mathbf{I}-\mathbf{Q})\mathbf{A}$, cf (19). Similar ideas based on implicit restarting and deflated restarts have been proposed for Lanczos-based methods [13,27] and Golub-Kahan Bidiagonalization-based methods (GKB) such as LSQR [14,17,18]. There are many other approaches which build on this type of deflated approach which incorporate flexible preconditioning [73], simultaneously treat block Krylov methods [105] with inexact breakdown [3], and use deflation in a flexibly preconditioned CG approach [29].

4.3 | Optimal augmentation built on GCR, that is, GCRO-based methods

In a parallel development, Eirola and Nevanlinna [47] independently proposed a splitting-based iteration in which the splitting is updated in each step with a specially chosen, rank-one, matrix. Although the authors did not state this, with the right choices the method is equivalent to GCR [48], as observed in [152]. The GMRESR method proposed in the latter paper[3] replaces the residual as search direction in GCR with an approximation to the error computed by GMRES (or by another iterative method), leading to an inner-outer iterative method. In [38] it was observed that optimality over the direct sum of the inner and outer correction spaces could be obtained by maintaining the right orthogonality relations. The resulting method is called GCRO. As with GMRESR, the GCRO method can be combined with any inner method, such as BiCGStab [38]. The augmentation space for GCRO is spanned by the corrections computed by the sequence of inner iterations. The GCROT method extends GCRO by computing an optimal subspace to recycle for subsequent iterations. The optimality is based on considering the canonical angles between the subspaces generated by restarted GMRES [39]. An extension of GCRO for a sequence of linear systems with a fixed matrix and multiple right-hand sides was presented in [37]. In [115], this was further extended to a sequence of linear systems where the matrix changes slowly with right-hand sides that may or may not be close, leading to the recycling GCROT and GCRODR methods. In [90], extensions include (a) a recycling version of MINRES, (b) utilizing approximate solutions in the recycle space to get good initial vectors, and (c) using recycling for a sequence of matrices with a number of shifts for each matrix. Additional innovations for multiple shifted systems are discussed in Section 7. More efficient versions of recycling MINRES, especially including efficient ways of computing and updating recycle spaces, were proposed in [100,153]. Extensions for recycling in BiCG [60], BiCGStab [150], and IDR(s) [142] have also been proposed [5,6,108]. The idea to use projection on a search space of old solutions for initial vectors was also proposed in [59], but this paper does not involve recycling or augmentation.

4.4 | Analysis

There has not been a great deal of convergence analysis of subspace recycling methods. However, several papers illuminate certain aspects of the behavior of these methods and related augmentation and acceleration strategies. An early analysis focuses on the nonoptimal augmentation strategy [126]. An analysis of acceleration strategies based on the principle angles between subspaces provides a comparison of strategies but not quantitative convergence bounds can be found in [45]. There has been some analysis of using approximate invariant subspaces to accelerate convergence. This was touched upon in a presentation at the 2012 Householder Symposium and the associated abstract [40], and it will be elaborated further in our review paper [41]. Another important way to address these questions might be to use the analysis of the onset of superlinear convergence in a GMRES iteration [138]. Another justification supported by analytic results can be found in [90]. See Section 6 for further details. There is also analysis from the point of view of deflation preconditioning [53,69],

[3]The general scheme is referred to as GMRES* in [151] and only as GMRESR if GMRES(m) is used as the inner method.

and there has been a related analysis of the use of approximate deflation preconditioners constructed using previously generated Arnoldi vectors [133].

4.5 | Alternative approaches

Independently of augmentation-based subspace recycling, recycling of information was the primary point of *seed-based methods*. These are not deflation methods. Rather, they use the whole Krylov space generated for one system to then solve multiple simultaneously available systems. The block seed methods in [30,31,91,125,137] use a block Krylov subspace to solve a subset of the systems and update the remaining systems using the just-generated block Krylov subspace according to some projection or minimization. It should be noted, though, that one strategy pursued for combining subspace recycling with strategies for solving multiple shifted linear systems (cf Section 7.1.3) built upon these methods and the analysis thereof [145].

For groups of HPD matrices that are all close to one another, a method was proposed which is close in spirit to a recycled CG method [123]. The authors propose to reuse the entire Krylov space generated for one coefficient matrix as the augmentation space for the next, which is computationally quite expensive, and the authors suggest to restart once memory has been exhausted. The method was extended with the use of converged dominant Ritz vectors (related to the largest eigenvalues) [122] resulting in modest improvements. Both methods, as proposed, require an expensive full recursion in spite of the operator being HPD. Later, alternative subspaces for augmentation using the theory in [149] were proposed [74], following the recycling approach.

4.6 | A framework

We now briefly present a general residual constraint framework through which many augmented Krylov subspace methods can be viewed. For a more complete view of this framework, see [41] which builds an understanding of these methods in terms of residual constraints on top of the existing work in [69,70,76]. Consider two subspaces $\mathcal{U}$, $\mathcal{V}_j \subset \mathbb{C}^n$, where $\mathcal{U}$ has fixed dimension k and $\mathcal{V}_j$ is generated by an iterative process such that at step j, $\dim \mathcal{V}_j = j$. We take $\mathcal{U} + \mathcal{V}_j$ as our correction space. We can similarly construct a constraint space $\tilde{\mathcal{U}} + \tilde{\mathcal{V}}_j$, with $\tilde{\mathcal{U}}$, $\tilde{\mathcal{V}}_j \subset \mathbb{C}^n$ being of fixed dimension k and iteratively generated with dimension j at iteration j, respectively. We assume for simplicity that $\dim(\mathcal{U} + \mathcal{V}_j) = \dim(\tilde{\mathcal{U}} + \tilde{\mathcal{V}}_j) = k + j$ (direct sums). Then the general augmented projection method becomes

$$\text{select } \mathbf{s}_j \in \mathcal{U} \text{ and } \mathbf{t}_j \in \mathcal{V}_j \text{ such that } \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j) \perp (\tilde{\mathcal{U}} + \tilde{\mathcal{V}}_j). \tag{21}$$

We represent these subspaces with the four matrices $\mathbf{U}$, $\tilde{\mathbf{U}} \in \mathbb{C}^{n\times k}$ and $\mathbf{V}_j$, $\tilde{\mathbf{V}}_j \in \mathbb{C}^{n\times j}$, such that $\text{range}(\mathbf{U}) = \mathcal{U}$, $\text{range}(\tilde{\mathbf{U}}) = \tilde{\mathcal{U}}$, and so on. Associated with every such augmented projection method is a pair of projectors: $\mathbf{P} = \mathbf{U}(\tilde{\mathbf{U}}^*\mathbf{A}\mathbf{U})^{-1}\tilde{\mathbf{U}}^*\mathbf{A}$, the projector onto $\mathcal{U}$ along $(\mathbf{A}^*\tilde{\mathbf{U}})^\perp$, and $\mathbf{Q} = \mathbf{A}\mathbf{U}(\tilde{\mathbf{U}}^*\mathbf{A}\mathbf{U})^{-1}\tilde{\mathbf{U}}^*$, the projector onto $\mathbf{A}\,\mathcal{U}$ along $\tilde{\mathcal{U}}^\perp$. Analogous to the derivation of (18), see also [41], the constraint equation (21) can be reformulated as the projected approximation problem whereby we take $\mathbf{V}_j\mathbf{y}_j$ as the approximate solution of

$$(\mathbf{I} - \mathbf{Q})\mathbf{A}\mathbf{t} = (\mathbf{I} - \mathbf{Q})\mathbf{r}_0 \tag{22}$$

that satisfies, for the residual of (22), $\hat{\mathbf{r}}_j$, the orthogonality condition

$$\hat{\mathbf{r}}_j = (\mathbf{I} - \mathbf{Q})(\mathbf{r}_0 - \mathbf{A}\mathbf{V}_j\mathbf{y}_j) \perp \tilde{\mathcal{V}}_j \quad \text{and so} \quad \mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j,$$

and we use $\mathbf{P}$ to compute

$$\mathbf{s}_j = \mathbf{P}\mathbf{t} - \mathbf{P}\mathbf{t}_j \text{ where } \mathbf{P}\mathbf{t} = \mathbf{U}(\tilde{\mathbf{U}}^*\mathbf{A}\mathbf{U})^{-1}\tilde{\mathbf{U}}^*\mathbf{r}_0,$$

which (when we substitute in the expression for $\mathbf{P}$ and $\mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j$) leads to the construction of the full approximation at iteration j

$$\mathbf{x}_j = x_0 + \mathbf{U}(\tilde{\mathbf{U}}^*\mathbf{A}\mathbf{U})^{-1}\tilde{\mathbf{U}}^*\mathbf{r}_0 + \mathbf{V}_j\mathbf{y}_j - \mathbf{U}\mathbf{B}_j\mathbf{y}_j \quad \text{where } \mathbf{B}_j = (\tilde{\mathbf{U}}^*\mathbf{A}\mathbf{U})^{-1}\tilde{\mathbf{U}}^*\mathbf{A}\mathbf{V}_j. \tag{23}$$

We note that the matrix $\mathbf{B}_j$ is generally built column-by-column iteratively and that the residual of the full problem and the projected subproblem are, in fact, equal (so the residual norm is known without making the full update), that is,

$$\mathbf{r}_j = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j) = \hat{\mathbf{r}}_j. \tag{24}$$

This means properties of the projected coefficient matrix in (22) and the iterative method we use to approximate its solution (ie, our choices of $\mathcal{V}_j$ and $\tilde{\mathcal{V}}_j$) will dictate the convergence behavior of the augmented method. Properties such as (23) and (24) are common to all subspace recycling methods. These and other characteristics which can be gleaned from the framework allow us to much more systematically design and implement a subspace recycling method.

Remark 1. Although one can choose any pair of subspaces $\mathcal{V}_j$, $\tilde{\mathcal{V}}_j$, it is important that the choice results in an efficient method. The review [41] details how to leverage this theory to obtain customized recycling methods. The greatest strength of viewing recycling methods through the lens of this framework is that it decouples the choice of the iteratively generated correction and constraint spaces $\mathcal{V}_j$ and $\tilde{\mathcal{V}}_j$ from the projected operator induced by enforcing the residual constraint via (22). Previously, these methods have been considered in a context where the Krylov subspace is generated by the projected operator and an appropriate right-hand side. However, this excludes useful augmentation schemes that can be described in this framework [41]. It is this observation which enables greater latitude in the systematic design of new customized recycling methods.

Example: Recycled FOM

With the above framework and the techniques for building new methods thereof, it becomes much more straightforward to build new, customized recycling methods using specific subspaces and residual constraints. Following [41], we demonstrate this for a recycling FOM method. We show two choices of projectors $\mathbf{Q}$ such that $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$ leads to a viable implementation of a recycling FOM method: in one case $\mathbf{Q}$ is the oblique projector onto $\mathbf{A}\,\mathbf{U}$ along $\mathcal{U}^\perp$, in the other case $\mathbf{Q}$ is the orthogonal projector onto $\mathbf{A}\,\mathcal{U}$.

The first choice of $\mathbf{Q}$ is explored in detail to produce a full implementation. In this case, we have $\mathbf{Q} = \mathbf{AU}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*$, and as a consequence of (16), $\mathbf{P} = \mathbf{U}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{A}$. According to the framework, we apply FOM to the projected problem (19) with the modification that, during the Arnoldi process for generating a basis for $\mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$, we store the coefficients $\mathbf{B}_j = (\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{AV}_j$ which come from applying $\mathbf{Q}$ to $\mathbf{Av}_i$ for each i. Then we solve the FOM linear problem

$$\mathbf{H}_j\mathbf{y}_j = \beta\mathbf{e}_1, \quad \beta = \|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|,$$

and at the end of the iteration we set $\mathbf{x}_j = (\mathbf{x}_0 + \mathbf{V}_j\mathbf{y}_j) + \mathbf{U}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{r}_0 - \mathbf{UB}_j\mathbf{y}_j$.

5 | PRACTICAL REALIZATIONS OF THE RECYCLING FRAMEWORK

Although the framework described in Section 4.6 can be used to understand the vast majority of subspace recycling methods, most methods were not derived this way. There are some exceptions, notably those arising from earlier proposed recycling frameworks [69,70,76] such as [71] and the methods proposed in the forthcoming review [41]. In this section, we give an overview of existing methods and put them into context of the framework in Section 4.6. We note that when discussing practical implementations of these methods, one must consider the developments in two research communities with overlapping interests, those for solving linear systems arising from discretizations of *well-posed problems* (which may be ill-conditioned due to isolated clusters of relatively small eigenvalues or singular values) and those arising from discretizations of *ill-posed problems* (which are characterized by rapidly decreasing singular values with no large, distinct gaps). There are augmented/recycling-type methods arising from both communities, where similar methods often arose by happenstance almost in parallel.

5.1 | Full basis storage methods

Methods for non-Hermitian systems, which do not take advantage of a short-term recurrences, require that the basis (Arnoldi) vectors from each iteration be stored until the end of the current restart cycle. Thus, recycling for these methods must be between restart cycles for a system associated with coefficient matrix $\mathbf{A}^{(i)}$ as well as between solves for $\mathbf{A}^{(i)}$ and $\mathbf{A}^{(i+1)}$. Indeed, methods such as GMRES-DR [104] can be understood as recycling exclusively between restart cycles using

harmonic Ritz vectors. Thus, as described in Section 4, many of these methods (see Section 5.1.1) can be understood as the confluence of the ideas of reuse of information between cycles and a more general passing of information between different linear systems in a systematic manner.

5.1.1 | Residual minimization methods

Although they are not often explicitly derived as such, recycling (or, more generally, augmentation) methods which minimize the residual over the augmented space $\mathcal{U} + \mathcal{V}_j$ can be characterized with the well-known minimum residual constraint

$$\text{select } \mathbf{s}_j \in \mathcal{U} \text{ and } \mathbf{t}_j \in \mathcal{V}_j \text{ such that } \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j) \perp \mathbf{A}\,(\mathcal{U} + \mathcal{V}_j),$$

which is a specific case of the more general (21). This is the theoretical umbrella which covers all such methods. Two decisions then determine exactly which method (up to implementation) is being proposed: which Krylov subspace is represented by $\mathcal{V}_j$ and what method of subspace downselection is being considered.

GCRO-DR/recycled GMRES

A strength of GCRO-based approaches was that they allowed one to minimize the residual over the combination of an arbitrary subspace $\mathcal{U}$ and a Krylov subspace generated from the projected coefficient matrix and right-hand side from (22). In retrospect, this is one realization of the notion of enforcing a constraint on an augmented subspace, as in Section 4.6. For all GCRO-based methods, we have the common choice that $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$ with $\mathbf{Q}$ being the orthogonal projector onto $\mathbf{A}\,\mathcal{U}$.

Practically speaking, one generates via the Arnoldi process an orthonormal basis for the subspace $\mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$. Let $\mathbf{C} \in C^{n\times k}$, $\mathbf{C}^*\mathbf{C} = \mathbf{I}$, and range$(\mathbf{C}) = \mathbf{A}\,\mathcal{U}$ so that $\mathbf{Q} = \mathbf{C}\mathbf{C}^*$. With this, the Arnoldi relation becomes

$$(\mathbf{I}-\mathbf{Q})\mathbf{A}\mathbf{V}_j = \mathbf{V}_{j+1}\underline{\mathbf{H}}_j \;\Leftrightarrow\; \mathbf{A}\mathbf{V}_j = \mathbf{C}\mathbf{B}_j + \mathbf{V}_{j+1}\underline{\mathbf{H}}_j \quad \text{where } \mathbf{B}_j = \mathbf{C}^*\mathbf{A}\mathbf{V}_j, \tag{25}$$

and this yields the modified Arnoldi relation

$$\mathbf{A}\begin{bmatrix} \mathbf{U} & \mathbf{V}_j \end{bmatrix} = \begin{bmatrix} \mathbf{C} & \mathbf{V}_{j+1} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{B}_j \\ 0 & \underline{\mathbf{H}}_j \end{bmatrix}. \tag{26}$$

Using, in addition, $\mathbf{r}_0 = \mathbf{Q}\mathbf{r}_0 + (\mathbf{I}-\mathbf{Q})\mathbf{r}_0$ and $(\mathbf{I}-\mathbf{Q})\mathbf{r}_0 = \mathbf{v}_1\|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|$, one can directly derive a practical implementation of the GCRO minimization yielding $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{s}_j + \mathbf{t}_j$ where,

$$(\mathbf{z}_j, \mathbf{y}_j) = \underset{\substack{\mathbf{z}\in\mathbb{C}^k \\ \mathbf{y}\in\mathbb{C}^j}}{\operatorname{argmin}} \left\| \begin{bmatrix} \mathbf{I} & \mathbf{B}_j \\ 0 & \underline{\mathbf{H}}_j \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{y} \end{bmatrix} - \begin{bmatrix} \mathbf{C}^*\mathbf{r}_0 \\ \mathbf{e}_1\beta \end{bmatrix} \right\| \text{ and } \mathbf{s}_j = \mathbf{U}\mathbf{z}_j,\ \mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j, \quad \text{where } \beta = \|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|, \tag{27}$$

where $\mathbf{e}_1 \in C^{k+j+1}$ is the first Euclidean basis vector. In [38], it was also suggested to solve (27) blockwise, first for $\mathbf{y}_j$ and then set $\mathbf{z}_j = \mathbf{C}^*\mathbf{r}_0 - \mathbf{B}_j\mathbf{y}_j$. This is equivalent to applying GMRES directly to the projected problem (19) and storing in $\mathbf{B}_j$ the coefficients obtained from applying $\mathbf{Q}$ during the Arnoldi iteration (25). This also falls directly out of the framework in Section 4.6: one can work out that for initial error $\mathbf{t}$,

$$\mathbf{x}_j = \mathbf{x}_0 + \mathbf{P}\mathbf{t} + (\mathbf{I}-\mathbf{P})\mathbf{t}_j \;\Leftrightarrow\; \mathbf{x}_0 + \mathbf{U}\mathbf{C}^*\mathbf{r}_0 + \mathbf{V}_j\mathbf{y}_j - \mathbf{U}(\mathbf{B}_j\mathbf{y}_j), \tag{28}$$

where $\mathbf{y}_j$ satisfies (9) for $\underline{\mathbf{H}}_j$ generated from (25) and $\beta = \|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|$. The form of the update (28) shows that this method can be implemented as a GMRES iteration applied to an equation of the form (22) where we must simply store $\mathbf{B}_j$ as it is computed. This enables the construction of the full GCRO-DR approximation using the coefficients $\mathbf{y}_j$ obtained from applying GMRES to (22). This is a specific realization of the updating formula (23), reinforcing the notion that all residual constraint, augmentation-based subspace recycling methods share this structure, which allows them to be understood

as known iterative methods being applied to projected linear systems. Multiple authors have made this conclusion for specific subspace recycling algorithms [38,69,70,76,87].

This setup is valid for any GCRO-based recycling method. What differentiates most GCRO-based recycling methods is how we choose a subspace of $\mathcal{U} + \mathcal{V}_j$ to generate and $\mathcal{U}_{new}$, the new recycled subspace. The GCRO-T method [39] is focused on minimizing the penalty for discarding some vectors at the end of a restart cycle in a process called *optimal truncation*. For further details see Section 6. A number of flexible and block variants have also been proposed in the literature [81,101,117].

The GCRO-DR method [115] builds on the same GCRO-type minimization but combined with the deflated-restarting strategies of Morgan [104], wherein the subspace retained between restart cycles is taken to be some harmonic Ritz vectors. There are strong associations between this strategy for solving linear systems and the implicitly restarted Arnoldi strategy for the computation of eigenvalues and eigenvectors of large, sparse matrices. Indeed, as was discussed in Section 4, if one looks at the forbearers of the *-DR strategies, one sees that many elements take inspiration from the implicit restarting techniques for eigen-computations [103]. We discuss this strategy further in the context of its effectiveness as a recycling technique along in a wider discussion of recycling strategies in Section 6. A block version of GMRES-DR has been proposed [105] and the same is true for GCRO-DR [117].

Recycled GMRES with an alternative projector

As noted in [69,70], there is a second variant of augmented GMRES that fits into the augmentation framework that was first proposed in [53] in the context of deflation-type GMRES methods. This variant fits into the framework in Section 4.6. The setup is similar to a GCRO-based method, except that the residual minimization is performed over a obliquely projected Krylov subspace. This second variant was used to propose an alternative type of recycled MINRES [71] than that which is described in Section 5.2.

In this case, let $\mathbf{P}$ be the oblique projector onto the subspace $\mathcal{U}$ along $(\mathbf{A}^* \mathcal{U})^\perp$ and $\mathbf{Q}$ be the sibling projector satisfying (16), the oblique projector onto $\mathbf{A}\mathcal{U}$ along $\mathcal{U}^\perp$. We use $\mathbf{P}$ to split the initial error $\mathbf{t} = \mathbf{P}\mathbf{t} + (\mathbf{I} - \mathbf{P})\mathbf{t}$, observing that as before $\mathbf{P}\mathbf{t}$ can be computed explicitly. We then approximate $(\mathbf{I} - \mathbf{P})\mathbf{t}$ by applying GMRES to (19) obtaining $\mathbf{t}_j$. The full approximation is then $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{P}\mathbf{t} + (\mathbf{I} - \mathbf{P})\ \mathbf{t}_j$. This is equivalent to minimizing the full residual over the subspace $\mathcal{U} + \mathcal{V}_j$ where $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{r}_0)$ where $\mathbf{Q}$ is an oblique projector rather than an orthogonal projector, as in GCRO.

Augmented GMRES for ill-posed problems

In [16], the authors propose to reconstruct the solution of a non-Hermitian ill-posed problem of the form (2) over an augmented Krylov subspace as well as over a preselected space. The authors propose to do this by decomposing the problem into two subproblems using projectors. The larger subproblem is posed in a space orthogonal to the augmenting subspace and is solved using an iterative method, and the smaller problem is posed in the augmenting subspace and is solved directly. This strategy can be cast as an augmented method [15]. Indeed, this can be directly related to the technique of splitting the error using a projector (17). Applying GMRES to the projected subproblem leads to a method which is mathematically equivalent to GMRES with recycling, that is, setting $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{r}_0)$. However, in this context, other choices of Krylov subspaces were suggested. Often, it has been shown beneficial to employ a strategy known as range-restriction, wherein one uses a power of the operator times the right-hand side, for example, $\mathcal{V}_j = \mathcal{K}_j((\mathbf{I} - \mathbf{Q})\mathbf{A}, (\mathbf{I} - \mathbf{Q})\mathbf{A}\mathbf{r}_0)$. We note that based on the work in [15], another adaptive augmented method has also been developed [93]. This body of literature is not necessarily concerned with recycling spectral information as much as it is with an augmentation space that encodes certain known features of the solution. However, strategies advocated in [35] show that recycling in the sense of GCRO-DR [115] can also be effective.

Unprojected GMRES with range-restriction

The fact that the modified Arnoldi iteration from [15] indeed is equivalent to a Krylov subspace iteration for the composition of the operator $\mathbf{A}$ and a projector was observed in [44, remark 2.1], and this remark leads the authors to propose a modification of the augmentation method [15]. The authors assert that augmenting an unprojected Krylov subspace allows a residual polynomial approximating the polynomial representation of the true inverse to be constructed. Using the polynomial representation of the inverse may not be the best approach for considering the effectiveness of this method, but we can evaluate this augmentation strategy nonetheless for various situations. They also assert that a poor choice of $\mathcal{U}$ causes the iteration to go awry with no chance of recovery. They propose reordering the steps of the modified Arnoldi process such that a GMRES-type iteration does reconstruct part of the approximation over a Krylov subspace

generated by $\mathbf{A}$ rather than the projected operator of the form in (19). We note that with the reordering of orthogonalization steps, this method is actually closely related to the flexible GMRES-based augmentation scheme of [32] discussed in Section 3.

Rather than generating the residual $(\mathbf{I}-\mathbf{Q})\,\mathbf{r}_0$ and generating a Krylov subspace with respect to the projected operator, in [44], the authors propose to generate $\mathcal{K}_j(\mathbf{A},\mathbf{p}_0)$, where $\mathbf{p}_0 \in \{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0\}$ [4]. After each new Arnoldi vector is generated, it is used to project the image of the augmentation space under the action of the operator away from that vector. In other words, at iteration j, the algorithm generates $\mathbf{W}_j = (\mathbf{I}-\mathbf{V}_{j+1}\mathbf{V}_{j+1}^*)\mathbf{A}\mathbf{U}$ progressively. This leads to

$$\mathbf{A}\begin{bmatrix}\mathbf{V}_j & \mathbf{U}\end{bmatrix} = \begin{bmatrix}\mathbf{V}_{j+1} & \mathbf{W}_j\end{bmatrix}\begin{bmatrix}\underline{\mathbf{H}}_j & \mathbf{L}_j \\ 0 & \mathbf{F}_j\end{bmatrix}, \text{ with } \mathbf{L}_j = \mathbf{V}_{j+1}^*\mathbf{C} \text{ and } \mathbf{F}_j = \mathbf{W}_j^*\mathbf{C}.$$

This leads to a GMRES-like iteration in which one must solve a small least squares problem of the form similar to (9)

$$\begin{bmatrix}\mathbf{y}_j \\ \mathbf{z}_j\end{bmatrix} = \underset{\substack{\mathbf{z}\in\mathbb{C}^k \\ \mathbf{y}\in\mathbb{C}^j}}{\operatorname{argmin}} \left\| \begin{bmatrix}\underline{\mathbf{H}}_j & \mathbf{L}_j \\ 0 & \mathbf{F}_j\end{bmatrix}\begin{bmatrix}\mathbf{y} \\ \mathbf{z}\end{bmatrix} - \begin{bmatrix}\mathbf{V}_{j+1} & \mathbf{W}_j\end{bmatrix}^*\mathbf{r}_0 \right\|, \quad \text{where } \mathbf{s}_j = \mathbf{U}\mathbf{z}_j \quad \text{and} \quad \mathbf{t}_j = \mathbf{V}_j\mathbf{y}_j.$$

They call this method regularized range-restricted GMRES (R3GMRES). It should be noted that this strategy also fits into the general augmentation framework, with $\mathcal{V}_j = \mathcal{K}_j(\mathbf{A},\mathbf{p}_0)$. Further details and elaboration on the relationship of this method to the framework in Section 4.6 can be found in the upcoming paper [146].

5.2 | Short-recurrence-based methods

If the coefficient matrices in (1) are Hermitian or one uses a short recurrence method for non-Hermitian systems, the additional challenges that arise for subspace recycling are different from those for GMRES and other full basis storage methods. Short recurrence methods do not need to restart, so there is no need to recycle for a subsequent restart cycle. The additional challenge is determining how to downselect the constructed basis to recycle. There is no need to recycle at a restart for the current system, but one should select a recycle space for the subsequent problem. The usual strategy is to store a running window of the p most recent (Hermitian or biorthogonal) Lanczos vectors as columns of a matrix $\mathbf{V}_{curr} \in \mathrm{C}^{n\times p}$. Consider that we have a recycled subspace $\mathcal{U}_{curr}$ being used for the current system. We then have a separate recycled subspace $\mathcal{U}_{next}$ which is held for the next system. We initialize $\mathcal{U}_{next} \leftarrow \mathcal{U}_{curr}$. When p vectors have been stored, the existing recycled subspace $\mathcal{U}_{next}$ is overwritten by computing a subspace of $\mathcal{U}_{next} + \text{range}(\mathbf{V}_{curr})$ according to the chosen downselection criteria. The vectors in $\mathbf{V}_{curr}$ are discarded and the matrix is filled again with the next set of p Lanczos vectors. Examples are discussed in [1,6,27,90,153]; a variant of this idea is discussed in [148].

5.2.1 | A-norm optimal error methods for Hermitian positive definite systems

Augmented conjugate gradient-type methods have been proposed independently in both the well- and ill-posed problems communities. Discretizations of many elliptic and parabolic partial differential equations lead to HPD discrete linear problems. For discrete ill-posed problems, one often uses a regularization of the normal equations associated with the linear problem, which generally produces a Hermitian positive-definite problem. [5] An augmented conjugate gradients approach was first proposed in 2000 in [129]. This was then followed up by [52] wherein the method was improved and

[4] The rationale for choosing $\mathbf{p}_0 = \mathbf{A}\mathbf{r}_0$ is that the right-hand side or initial residual for an ill-posed problem is noisy and can corrupt the iterative solution process. Since $\mathbf{A}$ is assumed to be the discretization of an operator which has smoothing properties, $\mathbf{A}\mathbf{r}_0$ is smoother and will contain less noise. In Hanke's monograph [77] on regularization properties of various iterative methods, this was denoted in the context of MINRES by MR2.

[5] It has been shown that CG applied to this problem with an appropriate early stopping rule satisfies the formal definition of regularization; see [50,77] for details. Indeed, augmentation-type methods fitting into the framework in Section 4.6 have been shown to also satisfy the formal definition of a regularization in [121].

it was used to treat a sequence of systems with the same HPD linear system but changing right-hand sides. Some of the residuals from previous systems are proposed to span the augmentation space $\mathcal{U}$. An unpublished manuscript [116] proposes a recycled variant of the augmented CG method that employs the strategy above, storing a fixed window of $\mathbf{A}$-conjugate directions to update the recycled subspace for the next system. In the discrete ill-posed problem setting, a similar method was proposed in [26]. However, the authors go further by modifying the minimization over the augmented space to transform it into a Tikhonov-type penalized minimization. An implicitly restarted Lanczos method for symmetric eigenvalue problems also has elements fitting into this framework [27].

The CG algorithm can be formulated as a highly efficient algorithm by exploiting the fact that an HPD operator can define an inner product. It is important to exploit this property of the linear system also in the recycling version. Several efficient recycling-like algorithms have been proposed doing this [26,69,87,129]. While the framework above allows us to define different recycling methods by distinct choices for the recycle correction and constraint spaces and inner product, we obtain a very efficient method by exploiting the $\mathbf{A}$-inner product and using the Galerkin approach, that is, the constraint space equals the correction space. This leads to $\mathbf{Q} = \mathbf{AU}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*$ and, following (16), $\mathbf{P} = \mathbf{U}(\mathbf{U}^*\mathbf{AU})^{-1}\mathbf{U}^*\mathbf{A}$, which leads to several special properties. By inspection, $\mathbf{Q} = \mathbf{P}^*$ and $(\mathbf{I}-\mathbf{Q})\mathbf{A} = \mathbf{A}(\mathbf{I}-\mathbf{P})$ is Hermitian (the equality holds for any $\mathbf{Q}$ and $\mathbf{P}$ satisfying (16)). Finally, we have that $(\mathbf{I}-\mathbf{Q})\mathbf{A} = \mathbf{A}(\mathbf{I}-\mathbf{P})$ is positive semidefinite, as $\mathbf{A}$ defines an inner product: $\mathbf{w}^*(\mathbf{I}-\mathbf{Q})\mathbf{Aw} = \mathbf{w}^*(\mathbf{I}-\mathbf{Q})(\mathbf{I}-\mathbf{Q})\mathbf{Aw} = \mathbf{w}^*(\mathbf{I}-\mathbf{P})^*\mathbf{A}(\mathbf{I}-\mathbf{P})\mathbf{w} \geq 0$ and $\mathbf{w}^*(\mathbf{I}-\mathbf{Q})\mathbf{Aw} = 0 \Leftrightarrow (\mathbf{I}-\mathbf{P})\mathbf{w} = \mathbf{0}$.

Without loss of generality, we consider henceforth the well-posed problem with $\mathbf{A}$ HPD and $\mathbf{Q}$ and $\mathbf{P}$ as defined above. Since (19) is consistent, we can apply CG directly to this problem. In fact, since $\text{range}((\mathbf{I}-\mathbf{Q})) = \mathcal{U}^\perp$, $\mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0) \subseteq \mathcal{U}^\perp$ and hence for any $\mathbf{w} \in \mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$, $(\mathbf{I}-\mathbf{P})\mathbf{w} \neq \mathbf{0}$. Taking $\mathbf{v}_1 = \beta^{-1}(\mathbf{I}-\mathbf{Q})\mathbf{r}_0$ with $\beta = \|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|$ the Lanczos relation becomes (cf Section 2)

$$(\mathbf{I}-\mathbf{Q})\mathbf{AV}_j = \mathbf{V}_{j+1}\underline{\mathbf{T}}_j = \mathbf{V}_j\mathbf{T}_j + \mathbf{v}_{j+1}h_{j+1,j}\mathbf{e}_j^*.$$

Following the above, $\mathbf{T}_j$ is positive definite and therefore the LU decomposition, $\mathbf{T}_j = \mathbf{L}_j\mathbf{U}_j$ (without) pivoting exists. So, we can again apply the change of basis transformation $\mathbf{W}_j = \mathbf{V}_j\mathbf{U}_j^{-1}$, set $\tilde{\mathbf{y}}_j = \mathbf{L}_j^{-1}\mathbf{e}_1$, and run the standard CG iteration on the projected system. The (full) solution can then be computed according to (23). In this case, by construction $\mathbf{r}_j \perp \{\mathbf{u}_1, \ldots, \mathbf{u}_k, \mathbf{v}_1, \ldots, \mathbf{v}_j\}$, which proves that the error is minimized in the $\mathbf{A}$-norm over the space $\text{range}(\mathbf{U}) \oplus \text{range}(\mathbf{V}_j)$.

In some alternative approaches [26,69,87,129], a change of basis is used that generates an $\mathbf{A}$-orthogonal basis for $\text{range}(\mathbf{U}) \oplus \text{range}(\mathbf{V}_j)$, which could be advantageous for some applications. Other alternative approaches compute more accurate eigenvectors over multiple linear systems, but for each linear system, they deflate these only from the initial residual [1,148].

5.2.2 | Minimum residual methods for Hermitian indefinite systems

Consider (1) with $\mathbf{A}$ Hermitian and indefinite, initial guess $\mathbf{x}_0$, and residual $\mathbf{r}_0$. As in Section 4.6, let $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k]$ define the recycle correction space $\mathcal{U} = \text{range}(\mathbf{U})$, and $\mathbf{AU} = \mathbf{C}$ with $\mathbf{C}^*\mathbf{C} = \mathbf{I}$. The recycle constraint space is chosen to be $\tilde{\mathcal{U}} = \text{range}(\mathbf{C})$. Hence, $\mathbf{Q} = \mathbf{CC}^*$ and $\mathbf{P} = \mathbf{UC}^*\mathbf{A}$, which gives $\mathbf{Pt} = \mathbf{UC}^*\mathbf{r}_0$. Recycling MINRES (rMINRES) [153] works by applying MINRES to approximately solve the system $(\mathbf{I}-\mathbf{Q})\mathbf{At} = (\mathbf{I}-\mathbf{Q})\mathbf{r}_0$ for an approximation to the update $\mathbf{t}$. Since $\mathcal{K}((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0) \subset \text{range}(\mathbf{I}-\mathbf{Q})$, $(\mathbf{I}-\mathbf{Q})\mathbf{A} : \text{range}(\mathbf{I}-\mathbf{Q}) \to \text{range}(\mathbf{I}-\mathbf{Q})$, and $\mathbf{A} = \mathbf{A}^*$, we have for all $\mathbf{z}_1, \mathbf{z}_2 \in \text{range}(\mathbf{I}-\mathbf{Q})$,

$$\langle(\mathbf{I}-\mathbf{Q})\mathbf{Az}_1, \mathbf{z}_2\rangle = \langle(\mathbf{I}-\mathbf{Q})\mathbf{A}(\mathbf{I}-\mathbf{Q})\mathbf{z}_1, \mathbf{z}_2\rangle = \langle\mathbf{z}_1, (\mathbf{I}-\mathbf{Q})\mathbf{A}(\mathbf{I}-\mathbf{Q})\mathbf{z}_2\rangle = \langle\mathbf{z}_1, (\mathbf{I}-\mathbf{Q})\mathbf{Az}_2\rangle.$$

So, $(\mathbf{I}-\mathbf{Q})\mathbf{A}$ is self-adjoint ("Hermitian") over $\mathcal{K}((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$, and we use the Hermitian Lanczos process (10) to get

$$(\mathbf{I}-\mathbf{Q})\mathbf{AV}_j = \mathbf{V}_{j+1}\underline{\mathbf{T}}_j \quad \Leftrightarrow \quad \mathbf{AV}_j = \mathbf{CB}_j + \mathbf{V}_{j+1}\underline{\mathbf{T}}_j, \tag{29}$$

with $\mathbf{B}_j = \mathbf{C}^*\mathbf{AV}_j$. As for standard MINRES, we can apply a change of basis, $\mathbf{W}_j\mathbf{R}_j = \mathbf{V}_j$, using the thin QR decomposition $\mathbf{T}_j = \mathbf{G}_j\mathbf{R}_j$, set $\tilde{\mathbf{y}} = \underline{\mathbf{G}}_j^*\mathbf{e}_1\|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|$, and during the iteration use the (partial) solution update $\mathbf{x}_j = \mathbf{x}_{j-1} + \mathbf{w}_j\tilde{y}_j = \mathbf{x}_0 + \mathbf{W}_j\tilde{\mathbf{y}}_j$,

where $\mathbf{t}_j = \mathbf{W}_j\tilde{\mathbf{y}}_j$ and $\tilde{y}_j$ is the jth component of $\tilde{\mathbf{y}}_j$. At the end of the iteration, we set

$$\mathbf{x}_j = \mathbf{x}_j + \mathbf{P}\mathbf{t} - \mathbf{P}\mathbf{t}_j = \mathbf{x}_0 + \mathbf{W}_j\tilde{\mathbf{y}}_j + \mathbf{U}\mathbf{C}^*\mathbf{r}_0 - \mathbf{U}\mathbf{B}_j(\mathbf{R}^{-1}\tilde{\mathbf{y}}_j). \tag{30}$$

This approach postpones all $\mathbf{U}$ updates during the iteration to a single update at the end, which saves $O(kn)$ work per iteration [153]. Several other efficiency improvements are discussed in [100,153]. This includes a change of basis that allows to discard the columns of $\mathbf{B}_j$ as we go (possibly important if many iterations are required) and very efficient recurrences to compute a recycle space.

5.2.3 | Biorthogonal Lanczos-based methods and transpose-free variants

For non-Hermitian system matrices, we consider BiCG and BiCGstab-based methods [4-6,8]. Obviously, recycling versions of QMR [64] and TFQMR [63] can developed as well. Recently, a recycling IDR(s) variant was developed [108].

Example: Recycling BiCG

We modify the BiCG algorithm to use recycle spaces. Here, we follow the approach chosen in [6]. Let $\text{range}(\mathbf{U}) = \mathcal{U}$ be the chosen recycle (correction) space for the primary linear system (1) and $\text{range}(\hat{\mathbf{U}}) = \hat{\mathcal{U}}$ be the chosen recycle correction space for the dual system. Moreover, we choose $\mathbf{U}$ and $\hat{\mathbf{U}}$ such that $\mathbf{C} = \mathbf{A}\mathbf{U}$ and $\hat{\mathbf{C}} = \mathbf{A}^*\hat{\mathbf{U}}_j$ satisfy $\hat{\mathbf{C}}^*\mathbf{C} = \mathbf{D}_c$ is real, diagonal, and invertible, that is, $\hat{\mathbf{C}}$ and $\mathbf{C}$ are biorthogonal. This can always be done (eg, using the SVD [6] or an LDU decomposition), although it may require reducing the dimension of the recycle spaces if $\mathbf{D}_c$ had one or more zeros on the diagonal. We define $\mathbf{Q} = \mathbf{C}\mathbf{D}_c^{-1}\hat{\mathbf{C}}^*$ and $\mathbf{P} = \mathbf{U}\mathbf{D}_c^{-1}\hat{\mathbf{C}}^*\mathbf{A}$. Following the discussion on the non-Hermitian Lanczos process in Section 2, we build dual bases for the subspaces $\mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, (\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$ and $\mathcal{K}_j((\mathbf{I}-\mathbf{Q}^*)\mathbf{A}^*, (\mathbf{I}-\mathbf{Q}^*)\hat{\mathbf{r}}_0)$, where $\hat{\mathbf{r}}_0$ is either the initial residual of a dual problem involving $\mathbf{A}^*$, $\mathbf{r}_0$ itself, or some other nonzero vector. This implies that the recycle constraint space is given by $\tilde{\mathcal{U}} = \text{range}(\hat{\mathbf{C}})$, and the recycled constraint space for the dual problem, if defined, is $\text{range}(\mathbf{C})$. Note that $(\mathbf{I}-\mathbf{Q})\mathbf{A} : \text{range}(\hat{\mathbf{C}})^\perp \to \text{range}(\hat{\mathbf{C}})^\perp$ and $(\mathbf{I}-\mathbf{Q}^*)\mathbf{A}^* : \text{range}(\mathbf{C})^\perp \to \text{range}(\mathbf{C})^\perp$, and we have for any $\mathbf{w} \in \text{range}(\hat{\mathbf{C}})^\perp$, $\hat{\mathbf{w}} \in \text{range}(\mathbf{C})^\perp$ that

$$\langle(\mathbf{I}-\mathbf{Q})\mathbf{A}\mathbf{w}, \hat{\mathbf{w}}\rangle = \langle(\mathbf{I}-\mathbf{Q})\mathbf{A}(\mathbf{I}-\mathbf{Q})\mathbf{w}, \hat{\mathbf{w}}\rangle = \langle\mathbf{w}, (\mathbf{I}-\mathbf{Q}^*)\mathbf{A}^*(\mathbf{I}-\mathbf{Q}^*)\hat{\mathbf{w}}\rangle = \langle\mathbf{w}, (\mathbf{I}-\mathbf{Q}^*)\mathbf{A}^*\hat{\mathbf{w}}\rangle.$$

So, $(\mathbf{I}-\mathbf{Q})\mathbf{A}$ and $(\mathbf{I}-\mathbf{Q}^*)\mathbf{A}^*$ act as adjoints over the primary and dual Krylov spaces, and we can again apply a coupled three-term recurrence as in Section 2:

$$(\mathbf{I}-\mathbf{Q})\mathbf{A}\mathbf{V}_j = \mathbf{V}_{j+1}\underline{\mathbf{T}}_j \quad \text{and} \quad (\mathbf{I}-\mathbf{Q}^*)\mathbf{A}^*\hat{\mathbf{V}}_j = \hat{\mathbf{V}}_{j+1}\underline{\hat{\mathbf{T}}}_j. \tag{31}$$

As for the standard BiCG discussion, we consider here the updates for the primary system; the updates for the dual system (if needed) can be computed analogously. We compute (if it exists) the LU-decomposition $\mathbf{T}_j = \mathbf{L}_j\mathbf{U}_j$, use the change of basis $\mathbf{W}_j = \mathbf{V}_j\mathbf{U}_j^{-1}$, and set $\tilde{\mathbf{y}}_j = \mathbf{L}_j^{-1}\mathbf{e}_1\zeta$, where $\zeta = \|(\mathbf{I}-\mathbf{Q})\mathbf{r}_0\|$. This allows us to eliminate the Lanczos vectors and update

$$\mathbf{t}_j = \mathbf{t}_{j-1} + \mathbf{w}_j\tilde{y}_j,$$

where $\tilde{y}_j$ is the jth component of $\tilde{\mathbf{y}}_j$ (the new component). The corresponding full solution is given by

$$\mathbf{x}_j = \mathbf{x}_0 + \mathbf{t}_j + \mathbf{P}\mathbf{t} - \mathbf{P}\mathbf{t}_j = \mathbf{x}_0 + \mathbf{t}_j + \mathbf{U}\mathbf{D}_c^{-1}\hat{\mathbf{C}}^*\mathbf{r}_0 - \mathbf{U}\mathbf{D}_c^{-1}\hat{\mathbf{C}}^*\mathbf{A}\mathbf{W}_j\tilde{\mathbf{y}}_j.$$

Only $\mathbf{t}_j$ is updated during the iteration. The updates in the $\mathbf{U}$ direction are done after the final iteration, while the vector $\mathbf{D}_c^{-1}\hat{\mathbf{C}}^*\mathbf{A}\mathbf{W}_j\tilde{\mathbf{y}}_j$ can be updated during the iteration (without computing $\mathbf{A}\mathbf{W}_j$) [6]. The matrices for the generalized eigenvalue problem that defines these subspaces can be constructed efficiently using recurrences [6].

One can develop a recycling BiCGStab based on recycling BiCG using the polynomials defining the iteration vectors following [150]; see [5]. This leads to a recycling BiCGStab with the matrix $(\mathbf{I}-\mathbf{Q})\mathbf{A}$ and $\mathbf{Q}$ as defined above. However, one can also take $\mathbf{Q}$ as for recycling GCROT above, based on a single recycling correction space $\mathcal{U}$, yielding $\mathbf{Q} = \mathbf{C}\mathbf{C}^*$; see [5,8], which includes an application where this approach is particularly useful.

5.2.4 | LSQR-based methods

This survey has focused on recycling methods for linear problems with square coefficient matrices. However, the framework of Section 4.6 is not restricted to square matrices or even matrix equations. It is shown in [41] that we can extend this framework to Hilbert space operator equations $Tx=y$, where $T : \mathcal{X} \to \mathcal{Y}$ is a linear mapping between two abstract Hilbert spaces. One can approximate the solution using a Petrov-Galerkin residual constraint posed in $\mathcal{Y}$, with the correction spaces $\mathcal{U}$, $\mathcal{V}_j \subset \mathcal{X}$ and the constraint spaces $\tilde{\mathcal{U}}$, $\tilde{\mathcal{V}}_j \subset \mathcal{Y}$. This Hilbert space framework is also used in [121] to prove that augmentation methods satisfy the formal definition of a regularization method in the infinite-dimensional ill-posed problems setting.

One realization of this more general notion of recycling methods arises when we have a linear system with a tall, skinny matrix wherein we are seeking the least-squares solution. Consider the (possibly inconsistent) linear system

$$\mathbf{G}\mathbf{x} \approx \mathbf{f} \quad \text{where } \mathbf{G} \in \mathbb{R}^{m\times n} \quad \text{and} \quad m > n.$$

The LSQR method is a short-recurrence Golub-Kahan bidiagonalization (GKB) method [114] which progressively solves

$$\mathbf{t}_j = \underset{\mathbf{t}\in\mathcal{K}_j(\mathbf{G}^*\mathbf{G},\mathbf{G}^*\mathbf{r}_0)}{\operatorname{argmin}} \|\mathbf{b} - \mathbf{G}(\mathbf{x}_0 + \mathbf{t})\| \quad \text{where } \mathbf{r}_0 = \mathbf{f} - \mathbf{G}\mathbf{x}_0,$$

which can be shown to be equivalent to the residual constraint formulation

$$\text{select } \mathbf{t}_j \in \mathcal{K}_j(\mathbf{G}^*\mathbf{G}, \mathbf{G}^*\mathbf{r}_0) \text{ such that } \mathbf{f} - \mathbf{G}(\mathbf{x}_0 + \mathbf{t}_j) \perp \mathbf{G}\mathbf{G}^*\mathcal{K}_j(\mathbf{G}\mathbf{G}^*, \mathbf{r}_0).$$

The efficient progressive formulation of the algorithm arises from the fact that it is possible to generate simultaneously via short recurrences orthonormal bases for the spaces $\mathcal{K}_j(\mathbf{G}^*\mathbf{G}, \mathbf{G}^*\mathbf{r}_0)$ and $\mathcal{K}_j(\mathbf{G}\mathbf{G}^*, \mathbf{r}_0)$ via the GKB.

With this formulation, one sees that it is possible to impose residual correction/constraint conditions over sums of spaces for this problem just as in the square problem case (21). The main challenge then is to choose projected Krylov subspaces $\mathcal{V}_j$ and $\tilde{\mathcal{V}}_j$ for the iteratively generated parts of the correction and constraint spaces, respectively, which are related such that orthonormal bases can be generated via GKB-type short recurrences. This has not been explored extensively in the literature, to our knowledge. For acceleration of convergence for a single problem, a deflated-restart-type method based on the theory presented for GMRES-DR [104] was proposed [18] . A forthcoming paper for sequences of regularized least-squares problems explores one particular strategy for choosing the projected Krylov spaces $\mathcal{V}_j$ and $\tilde{\mathcal{V}}_j$ appropriately to ensure that the orthonormal bases can be generated via the GKB [35] and is further explored in the upcoming review [41].

6 | WHAT TO RECYCLE

It is difficult to make general prescriptions about which subspaces to recycle because this depends on so many factors connected to the specific problem being solved. We break down the different choices proposed currently in the literature: approximate eigenvector augmentation, POD-type strategies, optimal truncation, and spaces from approximate solutions.

6.1 | Approximate solution augmentation

In [90] solutions from previous nonlinear iterations are recycled to obtain good initial guesses in subsequent problems. The success of this approach is problem dependent. If the right-hand sides do not change (much) from one linear system to the next (as is the case in [90]) and the coefficient matrices are close, this typically produces good initial guesses.

6.2 | Approximate eigenvector augmentation

As discussed in Section 4, there has been limited analysis up to now on *approximate* eigenvector augmentation (see [40,69,90,106] and our forthcoming review [41]), but there is strong evidence of the effectiveness of this strategy based on actual results for application problems. As these methods have often been shown to be equivalent to a deflation strategy, analysis pertaining to exact eigenvector deflation is also useful to study.

One rationale for augmenting with approximate eigenspaces is connected to the convergence theory for Krylov subspace methods. If $\mathcal{U}$ is an approximate eigenspace, then the projector $(\mathbf{I}-\mathbf{Q})$ is an (in this case orthogonal) projector onto $(\mathbf{A}\ \mathcal{U})^{\perp}$, and iterations during a cycle of GCRO-DR take place in $(\mathbf{A}\ \mathcal{U})^{\perp}$. Thus, the projector $(\mathbf{I}-\mathbf{Q})$ may have the effect of damping possible negative influence on convergence speed of the approximated invariant subspace (following from the theory introduced in [138]), leading to an accelerated convergence. Care must be taken, however, as it has been shown that residual convergence need not necessarily be connected to the spectral properties of the coefficient matrix [75].

Additionally, if two matrices $\mathbf{A}^{(i)}$ and $\mathbf{A}^{(i+1)}$ are "close enough," then particular respective invariant subspaces may also be close. For some differential operators (such as an elliptic operator), the higher frequency eigenmodes are associated with the larger eigenvalues, and this property usually is inherited by the discretized matrix. If the sequence of linear systems is induced by local changes in the matrix entries, then the changes to the invariant subspaces associated with the higher frequency eigenvectors (with larger eigenvalues) dominate. The details are quite technical; so we direct the reader to [90] where this was discussed and quantified for the problem under consideration. Conversely, for certain integral operators, one would want to recycle the approximate eigenvectors associated with the larger Ritz values, again to capture the low-frequency eigenvectors.

6.3 | Approximate singular vector augmentation

In [7], it has been recently proposed that one can also observe convergence speedups by recycling approximate singular vectors rather than eigenvectors, based on some analysis in [135] that the residual having large components in the left singular vectors can cause reduced convergence speed. The authors thus propose to use a Ritz-type approximation of the left singular vectors (ie, eigenvectors of $\mathbf{A}^{*}\mathbf{A}$) and recycle some of them.

6.4 | Optimal truncation

In [39], a method is proposed to mitigate the effects of restarting after a cycle of GMRES, wherein the entire Krylov subspace from the previous cycle is discarded. Hence, we are disregarding orthogonality with respect to the discarded subspace. This causes the characteristic reduced convergence rate one sees with restarted GMRES as compared to full GMRES. In [39], the author develops a model to characterize this delay (called "residual error" in the paper). One can again hearken back to the difference between the steepest descent method and CG as in Section 4 to understand this strategy. Restarting necessitates ignoring orthogonality with respect to the previously generated search space, and one way to interpret the so-called residual error caused by this is that the next cycle of GMRES induces a minimization over the new Krylov subspace which, since it is not orthogonalized against the previous, undoes some of the improvement in the residual along directions from the previous Krylov subspace. At worst, this can cause total stagnation and it generally is known to cause a slowdown in convergence. The GCROT method seeks to mitigate this problem with the described strategy, by maintaining orthogonality to some portion of the previously generated Krylov subspace.

This is achieved via the assumption that subspaces which were important (for speed of convergence) to maintain orthogonality against will continue to be important. Thus, for a cycle with length m, one studies the convergence during the last $s<m$ iterations of the cycle and compares that to the slower convergence which would have occurred had a dimension k subspace been neglected during the Arnoldi orthogonalization in those last s iterations. One can determine the dimension k subspace that would have caused the most delay had orthogonality against that space been neglected [39]. This subspace is then chosen to be retained for the next cycle in a process called *optimal truncation*. The model works just as well if we are selecting a subspace of an augmented Krylov subspace of the form $\mathcal{U}+\mathcal{V}_j$, where $\mathcal{V}_j$ is a projected Krylov subspace of the form $\mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A},(\mathbf{I}-\mathbf{Q})\mathbf{r}_0)$. To recycle between consecutive linear systems with respective coefficient matrices $\mathbf{A}^{(i)}$ and $\mathbf{A}^{(i+1)}$ one can also employ this strategy: if consecutive systems are "close enough," typically optimal truncation will still confer benefits associated with maintaining orthogonality to the truncated subspace.

6.5 | Proper orthogonal decomposition (POD)-type strategies

In the setting of recycled GMRES for a non-Hermitian family of shifted systems, the notion of using a POD-type strategy has actually been alluded to in one of the numerical experiments [144, section 6.6]. However, the author did not call this a POD strategy, and this was not expanded upon. This strategy takes advantage of the fact that for the test examples (coming from lattice quantum-chromodynamics application problems) the solutions for all shifted systems with shifts in a positive interval suitably away from zero depend smoothly on the shift. According to the theory of Kressner and Tobler [92], solutions corresponding to all shifts in this positive interval can be well approximated in a low-dimensional subspace; cf Section 7.5 for more details. Thus, a few solutions can be found, and the subspace they span can be used as an effective recycled subspace for solving the rest of the problems.

Specifically, a recycled CG method was proposed in which a subspace of the correction space is selected for retention using a strategy based on techniques from model order reduction [28]. If one is solving a sequence of HPD problems, and $j-1$ systems have already been solved, one should collect specially chosen snapshot vectors accumulated from the first $j-1$ system solves and use them to generate the POD-subspace with which to augment when applying CG to system j. This subspace is generated by approximately minimizing the operator-norm-distance between the true solution and its projection onto the POD subspace.

7 | EXPLOITING OPERATOR STRUCTURE AND MULTIPLE RIGHT-HAND SIDES

Sometimes, the sequence of coefficient matrices $\mathbf{A}^{(i)}$ have additional structure or for each i we actually have a parameterized set of matrices $\mathbf{A}^{(i)}(s)$, each of which is associated with a linear system. Furthermore, there may be many right-hand sides for a single system matrix, or slightly different right-hand sides for a sequence of related matrices. Such problems present additional challenges for adapting recycling. Most often, this takes the form of families of matrices which have some linear shifting structure, although we do consider more general parameter dependence in Section 7.5. The most general formulation of a family of linear systems with linear shift structure is

$$(\mathbf{A}^{(i)}+\gamma_\ell \mathbf{E})\mathbf{x}^{(\ell,i,j)}=\mathbf{b}^{(j,\ell)},\quad \gamma_\ell\geq 0, j=1,\ldots,j_*;\quad k=1,\ldots,k^*;\quad \ell=1,\ldots,L. \tag{32}$$

Consider, for example, that in general, $\mathrm{Range}((\mathbf{A}+\gamma I)\mathbf{U})\neq\mathrm{Range}(\mathbf{AU})$ unless $\mathbf{U}$ is an invariant subspace of $\mathbf{A}$; so even something as seemly innocuous as an identity shift (ie, $\mathbf{E}=\mathbf{I}$) can cause difficulties with any solver that falls into the recycling framework. However, these problems do have structure that can be exploited which allows for *extensions in the spirit of recycling*.

In the following, we split the discussion of recycling for systems with additional structure into scalar identity shifts, scalar nonidentity shifts, and more general continuous parameter dependence. For scalar identity shifts, the discussion is further split into shift-dependent right-hand sides, right-hand sides independent of shifts, and multiple right-hand sides in addition to shifts. Note that these categories are not mutually exclusive, since some methods will transform a family shift-independent right-hand sides into one for which there is shift-dependence. Such cases are noted in Section 7.1 with forward references to relevant sections.

7.1 | Scalar shifted matrices $\{\mathbf{A}+\gamma_\ell\mathbf{I}\}_{\ell=1}^{L}$ whose RHS changes with each shift

In this subsection, we consider $\mathbf{E}=\mathbf{I}$ for a single right-hand side and fixed system matrix in the sequence, which allows us to drop indices i,j in (32). We let $\mathbf{x}_0^{(\ell)}$ be the initial approximation for the shifted system with shift γ_ℓ, $\mathbf{t}_j^{(\ell)}$ to be the correction generated for this shift at iteration j, and $\mathbf{r}_j^{(\ell)}=\mathbf{b}^{(\ell)}-(\mathbf{A}+\gamma_\ell\mathbf{I})(\mathbf{x}_0^{(\ell)}+\mathbf{t}_j^{(\ell)})$ to be the residual for the system associated with shift ℓ. A property of Krylov subspaces which makes them attractive for treating a family of shifted systems is that for a given seed vector, the subspace is invariant with respect to scalar shifts by the identity of the coefficient matrix. More generally,

$$\mathcal{K}_j(\mathbf{A}+\gamma_{\ell_1}\mathbf{I},\mathbf{u})=\mathcal{K}_j(\mathbf{A}+\gamma_{\ell_2}\mathbf{I},\tilde{\mathbf{u}}), \tag{33}$$

where $\tilde{\mathbf{u}} = \omega\mathbf{u}$ for some $\omega \in \mathbb{C} \setminus \{0\}$ and for any nonzero values of γ_{ℓ_1} and γ_{ℓ_2}. If $\mathbf{b}^{(\ell)} = \mathbf{b}$ and $\mathbf{x}_0^{(\ell)} = \mathbf{0}$ for all ℓ, such as with systems arising in lattice quantum-chromodynamics [67] and Tikhonov regularization [68], we can design (nonrecycling based) solvers to take advantage of this shift invariance. In this subsection, we are concerned with reusing information when the right-hand sides do vary with γ_ℓ. Such shift-dependent RHS situations arise naturally in acoustics problems, but also arise when the system (32) for fixed i,j denotes a correction equation, that is, when the right-hand side denotes the initial residual $\mathbf{r}_j^{(\ell)}$.

There are iterative methods that have been tailored to accommodate multiple shifts, and we discuss those briefly here first. Then we move on to discussing impediments to adding recycling *on top of* these shifted system solvers in Section 7.1.2, and methods that are used to overcome these obstacles in Section 7.2.

Iterative methods tailored to shifted systems

For Arnoldi-based methods for large non-Hermitian matrices, it is likely that restarting will be necessary. Enforcing a (Petrov-) Galerkin condition for each shifted system does not guarantee that residuals remain collinear. At the restart stage, then, we are possibly in the position of having to solve systems with multiple shifts, and shift dependent RHS, where the RHS now correspond to the current respective residuals. The necessary conditions for residuals to remain collinear at restart were characterized in [65, theorem 1]. Methods such as the full orthogonalization method (FOM) [136], conjugate gradients [65,68], and biconjugate gradients (BiCG) [65] maintain a natural residual collinearity. Other methods such as GMRES, MINRES, and QMR do not naturally maintain residual collinearity. Since QMR and MINRES use short-term recurrences, this was shown not to be a great obstacle [62].

In [66], a restarted GMRES algorithm to simultaneously solve a family of shifted systems was derived. The key is that the residual of only one system is minimized. The residuals for the other systems are then explicitly forced to be collinear to this minimized residual. This collinearity correction may not always exist, but it was shown in [66] that if $\mathbf{A}$ has field of values in the right half-plane and the shifts are all positive, real numbers, then one can always enforce the residual collinearity.

An extension of BiCGStab(ℓ) for shifted systems has also been proposed [65]. Shifted BiCGStab(ℓ) works by alternating ℓ-cycles of shifted BiCG (which naturally maintains residual collinearity) and of shifted GMRES (which enforces residual collinearity). As with shifted GMRES, shifted BiCGStab(ℓ) inherits the property that it will always be able to generate collinear residuals in the case that the shifts are all positive and the coefficient matrix $\mathbf{A}$ has field of values in the right half-plane.

7.1.1 | Difficulties combining recycling with shifted system solvers

Combining the shifted GMRES method [66] with the GCRO-DR approach [115] was a part of the doctoral dissertation [143], and these results were extended and refined in [147]. In this work, it was shown that for general non-Hermitian coefficient matrices and augmentation subspaces, *it is not possible to embed a shifted restarted GMRES within a subspace recycling framework as described in Section* 4.6. An alternative is proposed, but its effectiveness decreases as the magnitude of the shift increases.

Essentially, for a given augmentation space $\mathcal{U}$ and its image $\mathcal{C}$, each shifted system must be projected as in Section 4.6, and the shifted restarted GMRES then is applied to the projected problems. This leads to three challenges: the projection of the initial residuals, the exploitation of possible shift invariance of the Krylov subspace generated by the projected coefficient matrix, and the enforcement of a residual collinearity condition at the end of each cycle. The correct initial residual projection was not fully treated until [144], and we will defer discussion thereof until Section 7.1.3. Under favorable circumstances in the case of optimal recycling methods for minimum residual Krylov subspace methods, it can be proven that a family of projected shifted matrices still generate the same Krylov subspace. This fact has been used in works as early as [90].

Proposition 1. [147, proposition 3.1] Let $\mathbf{Q}$ be the orthogonal projector onto $\mathcal{C}$. Then for any $\mathbf{v} \in \mathcal{C}^\perp$ we have that

$$\mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, \mathbf{v}) = \mathcal{K}_j((\mathbf{I}-\mathbf{Q})(\mathbf{A}+\gamma\mathbf{I}), \mathbf{v}).$$

This projected operator shift invariance was later extended to Sylvester operators in [144]. What Proposition 1 shows is that when restarting is not required, one can generate one augmented Krylov subspace $\mathcal{U} + \mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, \mathbf{r}_0)$ and compute

minimum residual corrections for each shifted system therefrom. This was used in [90], since the problems were real symmetric; and as was shown, it is compatible in the real case with complex shifts.

The question that remains is whether such a method can be extended to the non-Hermitian case when restarts are required. Ideally, as in [66], for one system we would compute the minimum residual correction, and for all other systems corrections would be computed that give residuals collinear to the minimized residual. However, it was shown in [147, theorem 1] that this is not generally possible. However, in certain situations, the collinear residual was shown to exist. For example, if $\mathcal{U} = C$ is an invariant subspace, then it is possible to compute collinear residuals as in [66]. More generally, if

$$\mathcal{U} + \mathcal{K}_j((\mathbf{I}-\mathbf{Q})\mathbf{A}, \mathbf{r}_0) \subset C + \mathcal{K}_{j+1}((\mathbf{I}-\mathbf{Q})\mathbf{A}, \mathbf{r}_0), \tag{34}$$

then it is possible to enforce residual collinearity. Indeed the shifted GMRES-DR method [36] takes advantage of this relationship, as Morgan previously proved that a Krylov subspace augmented with harmonic Ritz vectors satisfies (34) [104]. Absent this property, however, enforcing shifted residuals to be collinear to the minimized one is not possible [147].

7.1.2 | Effective recycling strategies for shifted systems

The problems discussed in (7.1.2) are impediments to combining these two technologies, but a number of strategies have been proposed which either overcome or work around the problems discussed in Section 7.1.2; for scalar shifted linear systems, there are certain circumstances for which we can take advantage of the shift invariance property and still augment. We note again that these methods treat the case where the right-hand sides/residuals do differ at some point. Even if we begin in the setting that we have the same right-hand side for all shifts, we are dealing with methods which destroy that structure, leaving residuals which are either collinear but not equal or which have no relationship to one another.

GMRES-DR and FOM-DR

As has been mentioned, for solving a single linear system with non-Hermitian coefficient matrix using an Arnoldi-based restarting approach, shifted versions of GMRES-DR and FOM-DR have been proposed and have shown to be effective [36]. Both unshifted algorithms work by retaining some harmonic Ritz or Ritz vectors, respectively, to augment the Krylov subspace generated in the next cycle. It is shown in [104] that in each case, the resulting space is in actuality a Krylov subspace with a different starting vector. In the augmentation language used in this survey, these appended vectors span the subspace $\mathcal{U}$, and the space $\mathcal{U}$ and $C = \mathbf{A}\mathcal{U}$ satisfy (34). Thus, in the case of GMRES-DR, one can enforce residual collinearity, as in the manner of [66]. For FOM-DR, one has the natural collinearity through enforcement of the Galerkin condition due to [65, theorem 1].

Direct projection

In [147], it was shown that one cannot generally enforce shifted residuals to be collinear with the minimized residual of the seed system when minimizing over an augmented Krylov subspace. One can instead simply exploit the shifted system structure directly. In [145], it is proposed to perform a minimum residual projection for all shifted systems over the (augmented) Krylov subspace generated by the base matrix and right-hand side. For nonaugmented Krylov subspaces, one could still exploit the shifted system structure such that the methods was still reasonably efficient, but the version of this method for GCRO-DR is still quite costly in terms of extra floating-point calculations one must carry out for the shifted systems.

Sylvester equation interpretation

It has been observed that a family of shifted systems can instead be interpreted as the Sylvester equations

$$\mathbf{A}\underbrace{\begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_s\end{bmatrix}}_{=:\mathbf{X}} + \begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_s\end{bmatrix}\underbrace{\operatorname{diag}\{\gamma_1, \gamma_2, \dots, \gamma_s\}}_{=:\mathbf{D}} = \underbrace{\begin{bmatrix}\mathbf{b}^{(\gamma_1)} & \mathbf{b}^{(\gamma_2)} & \dots & \mathbf{b}^{(\gamma_s)}\end{bmatrix}}_{=:\mathbf{B}}.$$

In [134], Simoncini showed that one could approximate the solution to this Sylvester equation by generating the block Krylov subspace $\mathrm{K}_j(\mathbf{A}, \mathbf{R}_0)$ generated by the block initial residual $\mathbf{R}_0 = \mathbf{B} - \mathbf{A}\mathbf{X}_0 - \mathbf{X}_0\mathbf{D}$. One can apply a GMRES

minimization over this space simultaneously for all shifted systems without consideration of residual collinearity or any relationship between residuals. Thus, building a subspace recycling method on top of this process suffers from none of the restrictions [144] that were seen in [147].

This method also allows one to take advantage of theory presented in [92], where it was shown that any parameter-dependent family of linear systems has a family of solutions well-approximated in a subspace of small dimension. The smallness of the dimension has an upper bound depending on how smoothly the right-hand side and linear system depend on the parameter. In [144], an experiment is set up such that a family of shifted systems (ie, $\mathbf{A}+\gamma\mathbf{I}$ depends linearly on γ) have right-hand sides $\mathbf{b}^{(\gamma)}$ which have C^∞ dependence on γ. A few of the systems are solved using standard methods. These solutions span a recycling subspace used to rapidly solve all other systems. This structure can be exploited more generally in the context of recycling; cf Section 7.5.

What if the residuals are collinear? If the RHS do not depend on the shifts, we can use the strategies employed in Section 7.2. If the residuals are collinear, one can use, for example, the shifted GMRES method [66], though this cannot take advantage of recycling. Regardless, the Sylvester framework is a viable option. If there is no recycled subspace, one can begin by applying a cycle of GMRES to all shifted systems, generating a single Krylov subspace using the shift invariance, since residuals are collinear. Applying the GMRES minimization to each shifted system renders the residuals noncollinear, and we are then in the general setting and we can apply the Sylvester equation strategy. If there is an initial recycled subspace, one applies the Sylvester equation strategy immediately, as this will destroy residual collinearity anyway.

7.2 | Sequences of shifted systems with shift independent RHS

In the following sections (until Section 7.5), we discuss specific applications where the *coefficient matrices are real-valued* with possible complex scalar shifting. Therefore, Hermitian conjugation is replaced with transposition.

During the course of solving the optimization problem in image reconstruction from diffuse optical tomographic data, the authors of [90] encounter sequences of shifted linear systems of multiple right-hand sides (MRHS) (32) where the RHS do not depend on the shift, and where the *nonzero shifts are pure imaginary*[6] (ie, $\gamma_\ell = \imath\frac{\omega}{v}$):

$$(\mathbf{A}^{(i)}+\gamma_\ell\mathbf{I})\mathbf{x}^{(\ell,i,j)}=\mathbf{b}^{(j)},\quad \gamma_\ell\geq 0, j=1,\ldots,j_*;\quad i=1,\ldots,i^*;\quad \ell=1,\ldots,L, \tag{35}$$

where $\gamma_1=0$. In their application, $\mathbf{A}^{(i)}$ are real and symmetric, though it is possible to extend the idea to nonsymmetric matrices. Right-hand sides do not change as i changes, and are not a function of shift[7].

No shifts, multiple RHS

The contributions contained in the 2006 paper [90] include recycling that takes advantage of similarities of the systems across all three indices, tailored in part to the optimization process. That is, for a given right-hand side, the recycle space consists of the shared recycling basis of approximate eigenvectors, augmented by a very small number of recent, prior solutions for that right-hand side, where the idea of which prior solutions to include depends on where the solve occurs in the optimization process.

Including multiple shifts

To describe their approach for the nonzero shifts we focus on a fixed right-hand side and drop the dependence on i and j. Let $\mathbf{U}$ denote the recycle space for the current i,j. Then the recycling recurrence gives

$$\mathbf{A}\mathbf{V}_m=\mathbf{C}\mathbf{B}_m+\mathbf{V}_{m+1}\underline{\mathbf{T}}_m,\quad \mathbf{B}_m:=\mathbf{C}^T\mathbf{A}\mathbf{V}_m,$$

and as before, an optimal solution is sought in Range($[\mathbf{V}_m,\mathbf{U}]$). After some manipulation, the least squares problem for the solution ultimately leads to the projected problem

$$\min_{\mathbf{y},\mathbf{z}}\left\|\begin{bmatrix}\xi\mathbf{e}_1\\ \mathbf{C}^T\mathbf{b}\\ 0\end{bmatrix}-\begin{bmatrix}\underline{\mathbf{T}}_m+\gamma_\ell\underline{\mathbf{I}}_m & \gamma\mathbf{V}_{m+1}^T\mathbf{U}\\ \mathbf{B}_m & \mathbf{I}+\gamma_\ell\mathbf{C}^T\mathbf{U}\\ 0 & \gamma_\ell\mathbf{N}\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix}\right\|_2\quad\text{or}\quad \min_{\mathbf{y},\mathbf{z}}\left\|\begin{bmatrix}\mathbf{C}^T\mathbf{b}\\ \xi\mathbf{e}_1\end{bmatrix}-\begin{bmatrix}\mathbf{I}+\gamma_\ell(\mathbf{C}^T\mathbf{U}) & \mathbf{B}_m\\ 0 & \underline{\mathbf{T}}_m+\gamma_\ell\underline{\mathbf{I}}_m\end{bmatrix}\begin{bmatrix}\mathbf{z}\\ \mathbf{y}\end{bmatrix}\right\|_2,$$

[6]The approach can be readily modified for a real or complex shift.

[7]In order to generalize to nonsymmetric $\mathbf{A}^{(i)}$, the MINRES solver would need to be replaced with a GMRES solver. If restarts are necessary, then it may be necessary to incorporate components from the previous section as then intermediate residuals would depend on shift value.

where $\mathbf{N}$ arises from orthogonalizing $\mathbf{U}$ against $[\mathbf{V}_{m+1}, \mathbf{C}]$ and the solution is $\mathbf{x} = \mathbf{V}_m\mathbf{y} + \mathbf{U}\mathbf{z}$. Numerical difficulties are observed if Range($\mathbf{U}$) is very close to an invariant subspace of $\mathbf{A}$. The problem on the right is preferred if Range($\mathbf{U}$) is expected to be very close to an invariant subspace of $\mathbf{A}$, or when storage is at a premium, since, in this case, $\mathbf{V}_m\mathbf{y}$ and $\mathbf{B}_m\mathbf{y}$ can be computed via short term recurrences when $\mathbf{A}$ is symmetric, avoiding the storage of $\mathbf{V}_m$.

7.3 | Inner-outer recycling

The concept of inner-outer recycling for systems of the form (32) was first developed in [111] in the context of producing a global basis necessary for creating a reduced order model (ROM) in the diffuse optical tomographic imaging problem. Let the current global basis for creating a reduced order model be the same as the *master recycle space*, Range($\mathbf{U}$). After solving the next set of systems in the sequence, columns may be appended to the global basis. Once the global basis is sufficient, system solves for future i are replaced by system solves with $\mathbf{U}^T\mathbf{A}^{(i)}\mathbf{U}$, the ROM, which has a significantly smaller dimension than that of $\mathbf{A}^{(i)}$. Going forward, the optimization relies on solves with the ROM, so optimization steps are much cheaper. Unfortunately for the number of columns in $\mathbf{U}$ that are needed for a good ROM, orthogonalization would become too costly to use this as a recycle space in the typical setting.

The authors of [111] observed that to enhance the global basis, one need not find an approximation to the system solutions directly, but rather to add information that is not already reconstructible from Range($\mathbf{U}$). For each j, there is a correction equation

$$\mathbf{A}^{(i)}\mathbf{g}^{(j)} = \mathbf{r}^{(j)}, \quad \mathbf{g}^{(j)} := \mathbf{x}^{(j)} - \mathbf{U}\mathbf{C}^T\mathbf{b}^{(j)} \tag{36}$$

(assuming fixed i and $\gamma = 0$) to which the authors look for the optimal $\mathbf{g}^{(j)}$ restricted to a suitable subspace $\mathcal{S}^{(j)}$. Their idea is to define a *local recycling space* Range($\mathbf{U}^{(j)}$) $\subset$ Range($\mathbf{U}$) and use it to apply recycling MINRES to solve (36). Since Range($\mathbf{U}^{(j)}$) $\subset$ Range($\mathbf{U}$) is maintained, a short-term recurrence update is also possible.

Updating across shifted systems

The global basis matrix must provide a suitable reduced transfer function for the 0 *as well as* the pure imaginary shifts γ_ℓ. The authors of [111] observed that in their application, the magnitude of the shifts are small enough that the real parts of the solutions to the shifted systems are not far (in a relative sense) from the corresponding solutions to the 0 shift case. Thus, they initialize the master recycle space $\mathbf{U}$ (and all the local recycle spaces $\mathbf{U}^{(j)}$) with the solutions to the 0 frequency systems for $k = 1$. They also augment $\mathbf{U}$ (and $\mathbf{U}^{(j)}$) by the *imaginary parts of the solutions to the corresponding imaginary shifted systems* for $k = 1$. This ensures that $\mathbf{U}$ and all the $\mathbf{U}^{(j)}$ remain real for all k.

For a $k \geq 2$, the nonshifted systems for $j = 1, \ldots, j_*$ are first solved using the inner-outer recycling described above. Then, the nonzero shifted systems are updated. Assuming $\mathbf{x}^{(\ell,j)} \approx \mathbf{U}\mathbf{q}^{(\ell,j)}$, a Petrov-Galerkin projection is applied so that $\mathbf{C}^T\mathbf{r}^{(\ell,j)} = 0$, giving the solution estimate for the ℓth shift and jth right-hand side as

$$\mathbf{x}^{(\ell,j)} \approx \mathbf{U}(\mathbf{I} + \gamma_\ell\mathbf{C}^T\mathbf{U})^{-1}\mathbf{C}^T\mathbf{b}^{(j)}, \quad \text{where } \mathbf{C}^T\mathbf{U} = \mathbf{U}^T\mathbf{A}^{(i)}\mathbf{U}. \tag{37}$$

No additional correction system is solved for any shifted system; the corrections are found only for the nonshifted systems for all right-hand sides, as before. It should be noted that the elements of the work in [144] were inspired by this strategy.

7.4 | A family of matrices $\{\mathbf{A} + \gamma_\ell\mathbf{E}\}_{\ell=1}^{L}$, $\mathbf{E} \neq \mathbf{I}$

Inner-outer recycling over shifts

In [112], a similar idea of maintaining an outer, master subspace across all shifts, and local, shift-specific recycle spaces is introduced. The method assumes symmetric $\mathbf{A}^{(i)}$, but the shift matrix $\mathbf{E}$ need only be real, and the shifts can be anything. We describe the procedure here for a single RHS, but details for MRHS can be found in [112]. The master subspace, $\mathbf{U}$, is initially seeded with all the solutions across all the shifts for the first system ($i = 1$), as well as approximate invariant subspace information for a fixed shift value (for simplicity, assume this is for $\gamma = 0$). At step $i+1$, the columns of $\mathbf{C}$ provide an orthonormal basis for Range($\mathbf{A}^{(i+1)}\mathbf{U}$). Then, initial guesses to the solutions across all shifts are obtained over

$\mathbf{U}$ via a Petrov-Galerkin constraint on residuals. If the corresponding residual $\mathbf{r}^{(\ell)}$ to the ℓth shift is not small enough, $\mathbf{U}^{(\ell)}$ is selected such that Range($\mathbf{U}^{(\ell)}$) $\subseteq$ Range($\mathbf{U}$). The orthonormal columns of $\mathbf{C}^{(\ell)}$ provide a basis for Range($(\mathbf{A}^{(i+1)} + \gamma_\ell \mathbf{E})\mathbf{U}^{(\ell)}$), and recycling using this shift-specific $\mathbf{U}^{(\ell)}$ and $\mathbf{C}^{(\ell)}$ is used to solve the residual correction equation.

Double shifts, single RHS

In [130], the authors consider an application in hyperspectral DOT, but the system matrix $\mathbf{A}$ remains unchanged and recycling is only over the wavelengths (ie, shifts are real-valued). Here, the system matrices are a small perturbation to systems shifted by the identity, $(\mathbf{A} + \gamma_\ell \mathbf{I} + \mu_\ell \mathbf{E})$ and with a single RHS. It is also assumed $\|\mathbf{E}\|_2 \ll \|\mathbf{A}\|_2$. Thus, the shift invariance property is first used to generate initial guesses to solutions, ignoring $\mathbf{E}$. Recycling is then used on each individual residual correction equation. While the general recycling strategy follows [115], it differs in the way that the recycle and range spaces are updated. This is possible because $\mathbf{A}_\ell := \mathbf{A} + \sigma_\ell \mathbf{I} + \mu_\ell \mathbf{E} = \mathbf{A}_{\ell-1} + \Delta_\ell$, where $\Delta_\ell = (\sigma_\ell - \sigma_{\ell-1})\mathbf{I} + (\mu_\ell - \mu_{\ell-1})\mathbf{E}$. So, once the $\mathbf{U}$ and $\mathbf{C}$ for $\mathbf{A}$ are known, finding $\mathbf{C}_\ell$ for $\mathbf{A}_\ell$ can be done independently across ℓ and parallelized, so all the shifted systems can be handled independently.

Other methods

There have been other iterative approaches proposed in the literature that deal with shifted systems. In some of these, such as [19], the focus is on preconditioning to convert systems to those for which shift-invariance can be leveraged, but subspace recycling is not explicitly used and therefore we do not review that literature here. Other literature comes closer in spirit: the authors of [99] propose a deflation based Lanczos approach for the symmetric parameterized systems whose matrices correspond to a frequency response function. The method is similar in the sense of projecting out a set of (in this case, generalized) eigenvectors, so some form of deflation that leads to faster convergence. This can be done for multiple shifts. However, since this is based on generalized eigenvectors, this does not give standard spectral deflation in the sense described above, so we do not consider it further here.

7.5 | General parameterized families with continuous parameter dependence

We discussed in Section 7.1.3, if the right-hand sides exhibit sufficiently smooth dependence on the shift for all shifts in some interval, then the solutions associated with all shifts in this interval can be approximated to machine accuracy in a small subspace S which has a dimension dependent on the smoothness of the dependence on the shift. This theory was developed in [92], and it applies more generally to parameter-dependent linear systems of the form

$$\mathbf{A}(s)\mathbf{x}(s) = \mathbf{b}(s) \quad s \in \mathbb{C}. \tag{38}$$

If dependence of $\mathbf{A}(s)$ and $\mathbf{b}(s)$ is sufficiently smooth for all s in some neighborhood $\Omega \subset \mathbb{C}$, then the associated solutions all exist in a small subspace S of dimension d, which depends on the smoothness of the dependence. In [92], upper bounds for d are given in terms of the smoothness of this dependence on s. Thus, one can propose a recycling algorithm to take advantage of this theory. For (38), if we have determined $d < d_M$, then we can build a recycled subspace by choosing $\{s_1, s_2, \ldots, s_{d_M}\} \subset \Omega$ in the neighborhood and use any method to solve the linear systems associated with these shifts. The solutions $\{\mathbf{x}(s_1), \mathbf{x}(s_2), \ldots, \mathbf{x}(s_{d_M})\}$ form the recycled subspace which can be used to solve all systems for other $s \in \Omega$.

8 | USES IN PRACTICAL APPLICATIONS

Large-scale software libraries

Recycling solvers have been implemented in major software/solver libraries, most notably in PETSc [20-22,119,120], see [84] for a discussion on performance and applications in elasticity and electromagnetics, Trilinos [23,79,131], and the DLR-TAU library from the German Aerospace Center [154,155], which also detail several challenging applications in CFD.

Computational scientific and engineering applications

Recycling solvers have been used in a wide range of applications, ranging from calculations for fundamental problems in computational physics to large-scale astrophysical simulations, tomography and medical imaging, and many applications in computational engineering, sometimes with modifications that serve a specific application. Recycling solvers and

closely related approaches have been used in lattice quantum chromodynamics [9,10,24,36,148], in particular, [9] mentions that recycling is important for handling physical regimes with very small eigenvalues. Many problems in design involve sequences of slowly changing linear systems, for which recycling is highly efficient, for example, in topology optimization and other structural optimization problems [28,34,42,153,157], and aerodynamic shape optimization [33,81]. Recycling has also been used to compute reduced order models (for a range of applications) [5,6,57,58,111]. Another important area is nonlinear optimization, such as nonlinear least-squares, for example, in tomography [90,100,111,130] and blind deconvolution [78]. Many applications arise in engineering, such as computational fluid dynamics and nonlinear structural problems [8,74,95,96,124,154,155], acoustics [89,99], and problems from electromagnetics and electrical circuits [49,72,73,84,118,156]. Recycling has found many applications in uncertainty quantification and partial differential equations with stochastic components [46,83].

9 | OUTLOOK AND FUTURE WORK

There are yet many interesting extensions of the work mentioned above. One important area is a better understanding of what type of subspaces to recycle for fast convergence and how to compute such subspaces efficiently, especially in the context of particular applications and in terms of what can be learned from previous iterations/linear systems. A second area is convergence theory related to various recycling approaches, particularly, biorthogonality-based recycling approaches. Third, further work is needed to investigate how to best combine recycling and preconditioning and to determine whether or not the framework outlined here can help in this respect. Finally, we note that there are classes of problems—for example, discrete ill-posed problems—where the convergence needs and problem properties are different; thus, different ways of thinking in this context might be needed.

ACKNOWLEDGEMENTS

The authors wish to thank the ANLA activity group for inviting us to write on this topic. The work by Eric de Sturler was supported in part by the grant NSF DMS 1720305; the work by Misha Kilmer was supported in part by the grant NSF DMS 1720291. The authors would also like to thank Daniel B. Szyld for his comments on the presentation of this survey. The authors would also like to thank the two referees for their helpful comments which improved the quality and flow of the article greatly.

REFERENCES

[1] A. M. Abdel-Rehim et al., Deflated and restarted symmetric Lanczos methods for eigenvalues and linear equations with multiple right-hand sides, SIAM J. Sci. Comput. **32** (2010), no. 1, 129–149. https://doi.org/10.1137/080727361.

[2] A. M. Abdel-Rehim, A. Stathopoulos, and K. Orginos, Extending the eigCG algorithm to nonsymmetric Lanczos for linear systems with multiple right-hand sides, Numer. Linear Algebra Appl. **21** (2014), no. 4, 473–493. https://doi.org/10.1002/nla.1893.

[3] E. Agullo, L. Giraud, and Y.-F. Jing, Block GMRES method with inexact breakdowns and deflated restarting, SIAM J. Matrix Anal. Appl **35** (2014), no. 4, 1625–1651. https://doi.org/10.1137/140961912.

[4] Ahuja, K., Recycling Bi-Lanczos Algorithms: BiCG, CGS, and BiCGSTAB, Master's Thesis, Virginia Polytechnical Institute, Blacksburg, Department of Mathematics, Blacksburg, Virginia, 2009.

[5] K. Ahuja et al., Recycling BiCGSTAB with an application to parametric model order reduction, SIAM J. Sci. Comput. **37** (2015), no. 5, S429–S446. https://doi.org/10.1137/140972433.

[6] K. Ahuja et al., Recycling BiCG with an application to model reduction, SIAM J. Sci. Comput. **34** (2012), no. 4, A1925–A1949. https://doi.org/10.1137/100801500.

[7] H. Al Daas et al., *Recycling Krylov subspaces and reducing deflation subspaces for solving sequence of linear systems*, INRIA Paris, France, 2018.

[8] A. Amritkar et al., Recycling Krylov subspaces for CFD applications and a new hybrid recycling solver, J. Comput. Phys. **303** (2015), 222–237. https://doi.org/10.1016/j.jcp.2015.09.040.

[9] S. Aoki et al., 2+1 flavor lattice QCD toward the physical point, Phys. Rev. D **79** (2009), no. 3, 034503.

[10] S. Aoki et al., Physical point simulation in 2+1 flavor lattice QCD, Phys. Rev. D **81** (2010), no. 7, 074503.

[11] O. Axelsson, M. Neytcheva, and B. Polman, An application of the bordering method to solve nearly singular systems, Vestnik Moskovskogo Universiteta Seria 15, Vychisl. Math. Cybern **1** (1996), 3–25.

[12] J. Baglama et al., Adaptively preconditioned GMRES algorithms, SIAM J. Sci. Comput **20** (1998), no. 1, 243–269.

[13] J. Baglama, D. Calvetti, and L. Reichel, IRBL: An implicitly restarted block-Lanczos method for large-scale Hermitian eigenproblems, SIAM J. Sci. Comput. **24** (2003), no. 5, 1650–1677.https://doi.org/10.1137/S1064827501397949.

[14] J. Baglama and L. Reichel, Augmented implicitly restarted Lanczos bidiagonalization methods, SIAM J. Sci. Comput. **27** (2005), no. 1, 19–42. https://doi.org/10.1137/04060593X.

[15] J. Baglama and L. Reichel, Augmented GMRES-type methods, Numer. Linear Algebra Appl. **14** (2007), no. 4, 337–350. https://doi.org/10.1002/nla.518.

[16] J. Baglama and L. Reichel, Decomposition methods for large linear discrete ill-posed problems, J. Comput. Appl. Math. **198** (2007), no. 2, 332–343. https://doi.org/10.1016/j.cam.2005.09.025.

[17] J. Baglama and L. Reichel, An implicitly restarted block Lanczos bidiagonalization method using Leja shifts, BIT **53** (2013), no. 2, 285–310.

[18] J. Baglama, L. Reichel, and D. Richmond, An augmented LSQR method, Numer. Algor. **64** (2013), no. 2, 263–293. https://doi.org/10.1007/s11075-012-9665-8.

[19] T. Bakhos et al., Multipreconditioned GMRES for shifted systems, SIAM J. Sci. Comput. **39** (2017), no. 5, S222–S247. https://doi.org/10.1137/16M1068694.

[20] S. Balay et al., *PETSc Web page*, 2019, available at https://www.mcs.anl.gov/petsc.

[21] S. Balay et al., *PETSc users manual. ANL-95/11 - Revision 3.12*, Argonne National Laboratory, Illinois, available at, 2019. https://www.mcs.anl.gov/petsc.

[22] S. Balay et al., *Efficient management of parallelism in object oriented numerical software libraries*, in *Modern Software Tools in Scientific Computing*, E. Arge, A. M. Bruaset, and H. P. Langtangen, Eds., Birkhäuser Press, Basel, Switzerland, 1997, 163–202.

[23] Belos Package – Trilinos *Belos: An iterative linear solvers package*. Sandia National Laboratories, Sandia, New Mexico. available at https://docs.trilinos.org/dev/packages/belos/doc/html/index.html. Accessed August 28, 2020.

[24] M. Bolten, N. Božović, and A. Frommer, Preconditioning of Krylov subspace methods using recycling in lattice QCD computations, PAMM **13** (2013), no. 1, 413–414.

[25] K. Burrage and J. Erhel, On the performance of various adaptive preconditioned GMRES strategies, Numer. Linear Algebra Appl. **5** (1998), 101–121.

[26] D. Calvetti, L. Reichel, and A. Shuibi, Enriched Krylov subspace methods for ill-posed problems, Linear Algebra Appl. **362** (2003), 257–273. https://doi.org/10.1016/S0024-3795(02)00533-5.

[27] D. Calvetti, L. Reichel, and D. C. Sorensen, An implicitly restarted Lanczos method for large symmetric eigenvalue problems, Electron. Trans. Numer. Anal. **2** (1994), no. March, 1–21.

[28] K. Carlberg, V. Forstall, and R. Tuminaro, Krylov-subspace recycling via the POD-augmented conjugate-gradient method, SIAM J. Matrix Anal. Appl. **37** (2016), no. 3, 1304–1336. https://doi.org/10.1137/16M1057693

[29] L. M. Carvalho et al., A flexible generalized conjugate residual method with inner orthogonalization and deflated restarting, SIAM J. Matrix Anal. Appl. **32** (2011), no. 4, 1212–1235. https://doi.org/10.1137/100786253.

[30] T. F. Chan and M. K. Ng, Galerkin projection methods for solving multiple linear systems, SIAM J. Sci. Comput. **21** (1999), no. 3, 836–850. https://doi.org/10.1137/S1064827598310227.

[31] T. F. Chan and W. L. Wan, Analysis of projection methods for solving linear systems with multiple right-hand sides, SIAM J. Sci. Comput. **18** (1997), no. 6, 1698–1721.

[32] A. Chapman and Y. Saad, Deflated and augmented Krylov subspace techniques, Numer. Linear Algebra Appl. **4** (1997), no. 1, 43–66. https://doi.org/10.1002/(SICI)1099-1506(199701/02)4:1<43::AID-NLA99>3.3.CO;2-Q.

[33] C.-H. Chen and S. Nadarajah, GCRO with dynamic deflated restarting for solving adjoint systems of equations for aerodynamic shape optimization, Int. J. Numer. Methods Heat Fluid Flow (2019), **29**(7), 2179–2205.

[34] Y. Choi et al., *Accelerating topology optimization using reduced order models*, Lawrence Livermore National Lab.(LLNL), Livermore, CA, 2019.

[35] J. Chung, E. de Sturler, and J. Jiang, Hybrid projection methods with recycling for inverse problems, 2020, arXiv preprints arXiv:2007.00207..

[36] 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), no. 10, 2415–2434. https://doi.org/10.1016/j.laa.2008.04.019.

[37] E. de Sturler, Inner-outer methods with deflation for linear systems with multiple right-hand sides, Proceedings of the XIII Householder Symposium on Numerical Algebra, Pontresina, Switzerland, 1996, pp. 193–196.

[38] E. de Sturler, Nested Krylov methods based on GCR, J. Comput. Appl. Math. **67** (1996), no. 1, 15–41. https://doi.org/10.1016/0377-0427(94)00123-5.

[39] E. de Sturler, Truncation strategies for optimal Krylov subspace methods, SIAM J. Numer. Anal. **36** (1999), no. 3, 864–889. https://doi.org/10.1137/S0036142997315950.

[40] E. de Sturler, Convergence bounds for approximate invariant subspace recycling for sequences of linear systems, Proceedings of the Householder Symposium XVIII on Numerical Linear Algebra, Pontresina, Switzerland, 2011, pp. 51–52.

[41] E. de Sturler, M. Kilmer, and K. M. Soodhalter, Krylov subspace augmentation for the solution of shifted systems: a review, 2020, arXiv preprint arXiv:2001.10347.

[42] E. de Sturler et al., Large scale topology optimization using preconditioned Krylov subspace recycling and continuous approximation of material distribution, Proceedings Multiscale and Functionally Graded Materials 2006 (M&FGM 2006), 15–18 October 2006 (G. H. Paulino, et al Eds.), Oahu Island (Hawaii), 2006, pp. 279–284.

[43] V. Dolean, P. Jolivet, and F. Nataf, *An introduction to domain decomposition methods: algorithms, theory, and parallel implementation*, SIAM, Philadelphia, PA, 2015.

[44] Y. Dong, H. Garde, and P. C. Hansen, R3GMRES: Including prior information in GMRES-type methods for discrete inverse problems, Electron. Trans. Numer. Anal. **42** (2014), 136–146.

[45] M. Eiermann, O. G. Ernst, and O. Schneider, Analysis of acceleration strategies for restarted minimal residual methods, J. Comput. Appl. Math. **123** (2000), no. 1-2, 261–292. https://doi.org/10.1016/S0377-0427(00)00398-8 numerical analysis 2000, Vol. III. Linear algebra.

[46] M. Eiermann, O. G. Ernst, and E. Ullmann, Computational aspects of the stochastic finite element method, Comput. Vis. Sci **10** (2007), no. 1, 3–15. https://doi.org/10.1007/s00791-006-0047-4.

[47] T. Eirola and O. Nevanlinna, Accelerating with rank-one updates, Linear Algebra Appl. **121** (1989), 511–520. https://doi.org/10.1016/0024-3795(89)90719-2.

[48] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, Variational iterative methods for nonsymmetric systems of linear equations, SIAM J. Numer. Anal. **20** (1983), 345–357.

[49] T. A. El-Moselhy, Field solver technologies for variation-aware interconnect parasitic extraction, Ph.D. Thesis, Massachusetts Institute of Technology, 2010.

[50] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of inverse problems. Mathematics and its applications*, Vol **375**, Kluwer Academic Publishers Group, Dordrecht, Netherlands, 1996 viii+321 pp.

[51] J. Erhel, K. Burrage, and B. Pohl, Restarted GMRES preconditioned by deflation, J. Comput. Appl. Math. **69** (1996), 303–318.

[52] J. Erhel and F. Guyomarc'h, An augmented conjugate gradient method for solving consecutive symmetric positive definite linear systems, SIAM J. Matrix Anal. Appl **21** (2000), no. 4, 1279–1299. https://doi.org/10.1137/S0895479897330194.

[53] Y. A. Erlangga and R. Nabben, Deflation and balancing preconditioners for Krylov subspace methods applied to nonsymmetric matrices, SIAM J. Matrix Anal. Appl. **30** (2008), no. 2, 684–699. https://doi.org/10.1137/060678257.

[54] V. Faber et al., Minimal residual method stronger than polynomial preconditioning, SIAM J. Matrix Anal. Appl. **17** (1996), no. 4, 707–729. https://doi.org/10.1137/S0895479895286748.

[55] V. Faber and T. Manteuffel, Necessary and sufficient conditions for the existence of a conjugate gradient method, SIAM J. Numer. Anal. **21** (1984), no. 2, 352–362. https://doi.org/10.1137/0721026.

[56] D. K. Fadeev and V. N. Fadeeva, *Computational methods of linear algebra*, W. H. Freeman & Co, New York, NY, 1963.

[57] L. Feng, P. Benner, and J. G. Korvink, *Parametric model order reduction accelerated by subspace recycling*, Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference, New York, NY, IEEE, 2009, pp. 4328–4333.

[58] L. Feng, P. Benner, and J. G. Korvink, Subspace recycling accelerates the parametric macro-modeling of mems, Int. J. Numer. Methods Eng. **94** (2013), no. 1, 84–110.

[59] P. F. Fischer, Projection techniques for iterative solution of $Ax = b$ with successive right-hand sides, Comput. Methods Appl. Mech. Eng. **163** (1998), no. 1-4, 193–204. https://doi.org/10.1016/S0045-7825(98)00012-7.

[60] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in *Numerical Analysis (Proc 6th Biennial Dundee Conf., Univ. Dundee, Dundee, 1975). Lecture Notes in Mathematics*, Vol **506**, Springer, Berlin, Germany, 1976, 73–89.

[61] J. Frank and K. Vuik, On the construction of deflation-based preconditioners, SIAM J. Sci. Comput. **23** (2001), no. 2, 442–462. https://doi.org/10.1137/S1064827500373231.

[62] R. W. Freund, *Solution of shifted linear systems by quasi-minimal residual iterations*, in *Numerical Linear Algebra (Kent, OH, 1992)*, de Gruyter, Berlin, Germany, 1993, 101–121.

[63] R. W. Freund, A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems, SIAM J. Sci. Comput. **14** (1993), no. 2, 470–482. https://doi.org/10.1137/0914029.

[64] R. W. Freund and N. M. Nachtigal, QMR: a quasi-minimal residual method for non-Hermitian linear systems, Numerische Mathematik **60** (1991), no. 3, 315–339. https://doi.org/10.1007/BF01385726.

[65] A. Frommer, BiCGStab (ℓ) for families of shifted linear systems, Computing **70** (2003), no. 2, 87–109. https://doi.org/10.1007/s00607-003-1472-6.

[66] A. Frommer and U. Glässner, Restarted GMRES for shifted linear systems, SIAM J. Sci. Comput. **19** (1998), no. 1, 15–26. https://doi.org/10.1137/S1064827596304563.

[67] A. Frommer et al., Many masses on one stroke: Economic computation of quark propagators, Int. J. Modern Phys. C **6** (1995), 627–638.

[68] Frommer, A. and P. Maaß, 1999: Fast CG-based methods for Tikhonov-Phillips regularization. SIAM J. Sci. Comput., **20**, no. 5, 1831–1850 (electronic), doi:https://doi.org/10.1137/S1064827596313310.

[69] A. Gaul, Recycling Krylov subspace methods for sequences of linear systems: Analysis and applications, Ph.D. Thesis, Technischen Universität Berlin, Germany, 2014.

[70] A. Gaul et al., A framework for deflated and augmented Krylov subspace methods, SIAM J. Matrix Anal. Appl. **34** (2013), no. 2, 495–518. https://doi.org/10.1137/110820713.

[71] A. Gaul and N. Schlömer, Preconditioned recycling Krylov subspace methods for self-adjoint problems, Electron. Trans. Numer. Anal. **44** (2015), 522–547.

[72] L. Giraud, S. Gratton, and E. Martin, Incremental spectral preconditioners for sequences of linear systems, Appl Numer Math **57** (2007), no. 11-12, 1164–1180.

[73] L. Giraud et al., Flexible GMRES with deflated restarting, SIAM J. Sci. Comput. **32** (2010), no. 4, 1858–1878. https://doi.org/10.1137/080741847.

[74] P. Gosselet, C. Rey, and J. Pebrel, Total and selective reuse of Krylov subspaces for the resolution of sequences of nonlinear structural problems, Internat. J. Numer. Methods Engrg. **94** (2013), no. 1, 60–83. https://doi.org/10.1002/nme.4441.

[75] A. Greenbaum, V. Pták, and Z. Strakoš, Any nonincreasing convergence curve is possible for GMRES, SIAM J. Matrix Anal. Appl. **17** (1996), 465–469.

[76] M. H. Gutknecht, Deflated and augmented Krylov subspace methods: A framework for deflated BiCG and related solvers, SIAM J. Matrix Anal. Appl. **35** (2014), no. 4, 1444–1466. https://doi.org/10.1137/130923087.

[77] M. Hanke, *Conjugate gradient type methods for ill-posed problems, volume 327 of pitman research notes in mathematics series*, Longman Scientific & Technical, Harlow, 1995 iv+134pp.

[78] P. Hennig, M. A. Osborne, and M. Girolami, Probabilistic numerics and uncertainty in computations, Proc. Royal Soc. A Math. Phys. Eng. Sci. **471** (2015), no. 2179, 20150142.

[79] M. A. Heroux et al., An overview of the Trilinos project, ACM Trans. Math. Softw. **31** (2005), no. 3, 397–423. https://doi.org/10.1145/1089014.1089021.

[80] Hestenes, M. R. and E. Stiefel, 1952: Methods of conjugate gradients for solving linear systems. J. Res. Nat. Bureau Stand., **49**, 409–436 (1953).

[81] J. E. Hicken and D. W. Zingg, A simplified and flexible variant of GCROT for solving non-symmetric linear systems, SIAM J. Sci. Comput. **32** (2010), no. 3, 1672–1694. https://doi.org/10.1137/090754674.

[82] C. P. Jackson and P. C. Robinson, A numerical study of various algorithms related to the preconditioned conjugate gradient method, Int. J. Numer. Methods Eng **21** (1985), no. 7, 1315–1338. https://doi.org/10.1002/nme.1620210711.

[83] C. Jin, X.-C. Cai, and C. Li, Parallel domain decomposition methods for stochastic elliptic equations, SIAM J. Sci. Comput. **29** (2007), no. 5, 2096–2114.

[84] P. Jolivet, and P.-H. Tournier, *Block iterative methods and recycling for improved scalability of linear solvers*. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis SC'16, New York, NY, IEEE Press, 2016, p. 17, available at https://ieeexplore.ieee.org/abstract/document/7877095.

[85] T. B. Jönsthövel et al., Comparison of the deflated preconditioned conjugate gradient method and algebraic multi-grid for composite materials, Comput. Mech. **50** (2012), no. 3, 321–333.

[86] W. Joubert, A robust GMRES-based adaptive polynomial preconditioning algorithm for nonsymmetric linear systems, SIAM J. Sci. Comput. **15** (1994), no. 2, 427–439. https://doi.org/10.1137/0915029.

[87] K. Kahl and H. Rittich, The deflated conjugate gradient method: Convergence, perturbation and accuracy, Linear Algebra Appl. **515** (2017), 111–129. https://doi.org/10.1016/j.laa.2016.10.027.

[88] S. A. Karchenko and A. Y. Yeremin, Eigenvalue translation based pre-conditioners for the GMRES(k) method, Numer Linear Algebra Appl. **2** (1995), no. 1, 51–77.

[89] S. Keuchel, J. Biermann, and O. von Estorff, A combination of the fast multipole boundary element method and Krylov subspace recycling solvers, Eng. Anal. Bound. Elem. **65** (2016), 136–146.

[90] M. Kilmer and E. de Sturler, Recycling subspace information for diffuse optical tomography, SIAM J. Sci. Comput. **27** (2006), no. 6, 2140–2166. https://doi.org/10.1137/040610271.

[91] M. Kilmer, E. Miller, and C. Rappaport, QMR-based projection techniques for the solution of non-Hermitian systems with multiple right-hand sides, SIAM J. Sci. Comput. **23** (2001), no. 3, 761–780. https://doi.org/10.1137/S1064827599355542.

[92] D. Kressner and C. Tobler, Low-rank tensor Krylov subspace methods for parametrized linear systems, SIAM J. Matrix Anal. Appl. **32** (2011), no. 4, 1288–1316. https://doi.org/10.1137/100799010.

[93] N. Kuroiwa and T. Nodera, The adaptive augmented GMRES method for solving ill-posed problems, ANZIAM J. **50** (2008), no. C, C654–C667.

[94] C. Lanczos, Solution of systems of linear equations by minimized-iterations, J. Res. Nat. Bureau Stand. **49** (1952), 33–53.

[95] S. E. Leon et al., A unified library of nonlinear solution schemes, Appl. Mech. Rev. **64** (2011), no. 4, 040803.

[96] M. Malandain, N. Maheu, and V. Moureau, Optimization of the deflated conjugate gradient algorithm for the solving of elliptic equations on massively parallel machines, J. Comput. Phys. **238** (2013), 32–47. https://doi.org/10.1016/j.jcp.2012.11.046.

[97] J. Mandel, Balancing domain decomposition, Comm. Numer. Methods Eng. **9** (1993), no. 3, 233–241. https://doi.org/10.1002/cnm.1640090307.

[98] Maya Neytcheva, 2019: *Deflation techniques - historical development and advances*, available at https://www.maths.tcd.ie/~ksoodha/beyonddiscrete2019/wp-content/uploads/2019/06/Neytcheva2.pdf. Accessed August 28, 2020.

[99] K. Meerbergen and Z. Bai, The Lanczos method for parameterized symmetric linear systems with multiple right-hand sides, SIAM J. Matrix Anal. Appl. **31** (2009/10), no. 4, 1642–1662. https://doi.org/10.1137/08073144X.

[100] L. S. A. M. Mello et al., Recycling Krylov subspaces for efficient large-scale electrical impedance tomography, Comput. Methods Appl. Mech. Eng. **199** (2010), no. 49-52, 3101–3110. https://doi.org/10.1016/j.cma.2010.06.001.

[101] J. Meng, P.-Y. Zhu, and H.-B. Li, A block GCROT(m,k) method for linear systems with multiple right-hand sides, J. Comput. Appl. Math. **255** (2014), 544–554. https://doi.org/10.1016/j.cam.2013.06.014.

[102] R. B. Morgan, A restarted GMRES method augmented with eigenvectors, SIAM J. Matrix Anal. Appl. **16** (1995), no. 4, 1154–1171. https://doi.org/10.1137/S0895479893253975.

[103] R. B. Morgan, Implicitly restarted GMRES and Arnoldi methods for non-symmetric systems of equations, SIAM J. Matrix Anal. Appl. **21** (2000), no. 4, 1112–1135. https://doi.org/10.1137/S0895479897321362.

[104] R. B. Morgan, GMRES with deflated restarting, SIAM J. Sci. Comput. **24** (2002), no. 1, 20–37. https://doi.org/10.1137/S1064827599364659.

[105] R. B. Morgan, Restarted block-GMRES with deflation of eigenvalues, Appl. Numer. Math. **54** (2005), no. 2, 222–236.

[106] R. B. Morgan et al., Two-grid deflated Krylov methods for linear equations, 2020, arXiv preprints arXiv:2005.03070.

[107] N. M. Nachtigal, L. Reichel, and L. N. Trefethen, A hybrid GMRES algorithm for nonsymmetric linear systems, SIAM J. Matrix Anal. Appl. **13** (1992), no. 3, 796–825.

[108] M. P. Neuenhofen and C. Greif, Mstab: Stabilized induced dimension reduction for Krylov subspace recycling, SIAM J. Sci. Comput. **40** (2018), no. 2, B554–B571.

[109] R. A. Nicolaides, Deflation of conjugate gradients with applications to boundary value problems, SIAM J. Numer. Anal. **24** (1987), no. 2, 355–365. https://doi.org/10.1137/0724027.

[110] Y. Notay, Flexible conjugate gradients, SIAM J. Sci. Comput. **22** (2000), no. 4, 1444–1460. https://doi.org/10.1137/S1064827599362314.

[111] M. O'Connell et al., Computing reduced order models via inner-outer Krylov recycling in diffuse optical tomography, SIAM J. Sci. Comput. **39** (2017), no. 2, B272–B297. https://doi.org/10.1137/16M1062880.

[112] M. J. O'Connell, Advanced techniques in the computation of reduced order models and Krylov recycling for diffuse optical tomography, Ph.D. Thesis, Tufts University, Medford, Massachussets, United States of America, 2016, 115 pp, available at https://dl.tufts.edu/pdfviewer/2227n161k/8c97m229d.

[113] C. C. Paige and M. A. Saunders, Solutions of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. **12** (1975), no. 4, 617–629.

[114] C. C. Paige and M. A. Saunders, LSQR: an algorithm for sparse linear equations and sparse least squares, ACM Trans. Math. Software **8** (1982), no. 1, 43–71. https://doi.org/10.1145/355984.355989.

[115] M. L. Parks et al., Recycling Krylov subspaces for sequences of linear systems, SIAM J. Sci. Comput. **28** (2006), no. 5, 1651–1674. https://doi.org/10.1137/040607277.

[116] Parks, M. L., R. Sampath, and P. K. Nukala, 2013: *Efficient simulation of large-scale 3D fracture networks via Krylov subspace recycling*. Unpublished. Communicated by author Parks to first and second authors.

[117] M. L. Parks, K. M. Soodhalter, and D. B. Szyld, A block Recycled GMRES method with investigations into aspects of solver performance, 2016, Arxiv e-prints 1604.01713.

[118] Z. Peng, X.-C. Wang, and J.-F. Lee, Integral equation based domain decomposition method for solving electromagnetic wave scattering from non-penetrable objects, IEEE Trans. Antennas Propagat. **59** (2011), no. 9, 3328–3338. https://doi.org/10.1109/TAP.2011.2161542.

[119] PETSc - KSPHPDDM, *KSPHPDDM solver interface with the HPDDM library*, Argonna National laboratory in Argonne, Illinois, 2020. available at https://www.mcs.anl.gov/petsc/petsc-current/docs/manualpages/KSP/KSPHPDDM.html. Accessed August 28, 2020.

[120] PETSc Documentation, *PETSc manual and examples*, Argonne National Lab, Argonna National laboratory in Argonne, Illinois, 2020. available at. https://www.mcs.anl.gov/petsc/petsc-current/docs/. Accessed August 28, 2020.

[121] Ramlau, R. and K. M. Soodhalter, 2020: Regularized recycling. *In Preparation*.

[122] C. Rey and F. Risler, A Rayleigh-Ritz preconditioner for the iterative solution to large scale nonlinear problems, Numer. Algor. **17** (1998), no. 3-4, 279–311. https://doi.org/10.1023/A:1016680306741.

[123] F. Risler and C. Rey, *On the reuse of Ritz vectors for the solution to nonlinear elasticity problems by domain decomposition methods*, in *Domain Decomposition Methods, 10 (Boulder, CO, 1997) Volume of Contemporary Mathematics*, American Mathematical Society, Providence, RI, 1998, 334–340.

[124] F. Risler and C. Rey, Iterative accelerating algorithms with Krylov subspaces for the solution to large-scale nonlinear problems, Numer. Algor. **23** (2000), no. 1, 1–30. https://doi.org/10.1023/A:1019187614377.

[125] Y. Saad, On the Lánczos method for solving symmetric linear systems with several right-hand sides, Math. Comp. **48** (1987), no. 178, 651–662. https://doi.org/10.2307/2007834.

[126] Y. Saad, Analysis of augmented Krylov subspace methods, SIAM J. Matrix Anal. Appl. **18** (1997), no. 2, 435–449. https://doi.org/10.1137/S0895479895294289.

[127] Y. Saad, *Iterative methods for sparse linear systems*, 2nd ed., SIAM, Philadelphia, PA, 2003.

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

[129] Y. Saad et al., A deflated version of the conjugate gradient algorithm, SIAM J. Sci. Comput. **21** (2000), no. 5, 1909–1926. https://doi.org/10.1137/S1064829598339761.

[130] A. Saibaba et al., Fast algorithms for hyperspectral diffuse optical tomography, SIAM J. Sci. Comput. **37** (2015), no. 5, B712–B743.

[131] M. Sala, M. A. Heroux, and D. M. Day, *Trilinos tutorial*, Sandial National Laboratories, Sandia, NM. available at, 2007. https://trilinos.github.io/pdfs/Trilinos8.0Tutorial.pdf.

[132] A. H. Sheikh et al., Accelerating the shifted Laplace preconditioner for the Helmholtz equation by multilevel deflation, J. Comput. Phys. **322** (2016), 473–490.

[133] J. A. Sifuentes, M. Embree, and R. B. Morgan, GMRES convergence for perturbed coefficient matrices, with application to approximate deflation preconditioning, SIAM J. Matrix Anal. Appl. **34** (2013), no. 3, 1066–1088. https://doi.org/10.1137/120884328.

[134] V. Simoncini, On the numerical solution of AX− XB= C, BIT **36** (1996), no. 4, 814–830. https://doi.org/10.1007/BF01733793.

[135] V. Simoncini, On the convergence of restarted Krylov subspace methods, SIAM J. Matrix Anal. Appl. **22** (2000), no. 2, 430–452.

[136] V. Simoncini, Restarted full orthogonalization method for shifted linear systems, BIT. Numer. Math. **43** (2003), no. 2, 459–466. https://doi.org/10.1023/A:1026000105893.

[137] V. Simoncini and E. Gallopoulos, An iterative method for nonsymmetric systems with multiple right hand sides, SIAM J. Sci. Comput. **16** (1995), no. 4, 917–933.

[138] V. Simoncini and D. B. Szyld, On the occurrence of superlinear convergence of exact and inexact Krylov subspace methods, SIAM Rev. **47** (2005), no. 2, 247–272. https://doi.org/10.1137/S0036144503424439.

[139] V. Simoncini and D. B. Szyld, Interpreting IDR as a Petrov-Galerkin method, SIAM J. Sci. Comput. **32** (2010), 1898–1912. https://doi.org/10.1137/070685804.

[140] G. L. G. Sleijpen and D. R. Fokkema, BiCGstab(*l*) for linear equations involving unsymmetric matrices with complex spectrum, Electron. Trans. Numer. Anal. **1** (1993), no. Sept., 11–32.

[141] P. Sonneveld, CGS, a fast Lanczos-type solver for nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. **10** (1989), no. 1, 36–52. https://doi.org/10.1137/0910004.

[142] P. Sonneveld and M. B. van Gijzen, IDR(*s*): A family of simple and fast algorithms for solving large nonsymmetric systems of linear equations, SIAM J. Sci. Comput. **31** (2008), no. 2, 1035–1062. https://doi.org/10.1137/070685804.

[143] K. M. Soodhalter, Krylov subspace methods with fixed memory requirements: Nearly Hermitian linear systems and subspace recycling, Ph.D. Thesis, Temple University, 2012.

[144] K. M. Soodhalter, Block Krylov subspace recycling for shifted systems with unrelated right-hand sides, SIAM J. Sci. Comput. **38** (2016), no. 1, A302–A324. https://doi.org/10.1137/140998214.

[145] K. M. Soodhalter, Two recursive GMRES-type methods for shifted linear systems with general preconditioning, Electron. Trans. Numer. Anal. **45** (2016), 499–523, available at. http://arxiv.org/abs/1403.4428.

[146] K. M. Soodhalter, Augmented Arnoldi-Tikhonov methods for ill-posed problems, 2020 (in preparation).

[147] K. M. Soodhalter, D. B. Szyld, and F. Xue, Krylov subspace recycling for sequences of shifted linear systems, Appl. Numer. Math. **81** (2014), 105–118, available at. http://www.sciencedirect.com/science/article/pii/S0168927414000208.

[148] A. Stathopoulos and K. Orginos, Computing and deflating eigenvalues while solving multiple right-hand side linear systems with an application to quantum chromodynamics, SIAM J. Sci. Comput. **32** (2010), no. 1, 439–462. https://doi.org/10.1137/080725532.

[149] A. van der Sluis and H. van der Vorst, The rate of convergence of conjugate gradients, Numerische Mathematik **48** (1986), 543–560.

[150] H. A. van der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. **13** (1992), no. 2, 631–644. https://doi.org/10.1137/0913035.

[151] H. A. van der Vorst, *Iterative Krylov methods for large linear systems volume 13 of Cambridge monographs on applied and computational mathematics*, Cambridge University Press, Cambridge, MA, 2009 xiv+221pp.

[152] H. A. van der Vorst and K. Vuik, GMRESR: A family of nested GMRES methods, Numer. Linear Algebra Appl. **1** (1994), no. 4, 369–386. https://doi.org/10.1002/nla.1680010404.

[153] S. Wang, E. de Sturler, and G. H. Paulino, Large-scale topology optimization using preconditioned Krylov subspace methods with recycling, Int. J. Numer. Methods Eng. **69** (2007), no. 12, 2441–2468. https://doi.org/10.1002/nme.1798.

[154] S. Xu and S. Timme, Robust and efficient adjoint solver for complex flow conditions? Comput Fluids **148** (2017), 26–38. https://doi.org/10.1016/j.compfluid.2017.02.012.

[155] S. Xu, S. Timme, and K. J. Badcock, Enabling off-design linearised aerodynamics analysis using Krylov subspace recycling technique, Comput Fluids **140** (2016), 385–396.

[156] Z. Ye, Z. Zhu, and J. R. Phillips, *Generalized Krylov recycling methods for solution of multiple related linear equation systems in electromagnetic analysis*. Proceedings of the 45th Annual Design Automation Conference on DAC '08, ACM, New York, NY, 2008, pp. 682–687.

[157] X. S. Zhang, E. de Sturler, and A. Shapiro, Topology optimization with many right hand sides using a mirror descent stochastic approximation - reduction from many to a single sample, J. Appl. Mech. **87** (2020), no. 5, 051005. https://doi.org/10.1115/1.4045902.

How to cite this article: Soodhalter KM, de Sturler E, Kilmer ME. A survey of subspace recycling iterative methods. *GAMM-Mitteilungen*. 2020;43:e202000016. https://doi.org/10.1002/gamm.202000016