Krylov methods for inverse problems: Surveying classical, and introducing new, algorithmic approaches

Silvia Gazzola | Malena Sabaté Landman

Abstract

Large-scale linear systems coming from suitable discretizations of linear inverse problems are challenging to solve. Indeed, since they are inherently ill-posed, appropriate regularization should be applied; since they are large-scale, well-established direct regularization methods (such as Tikhonov regularization) cannot often be straightforwardly employed, and iterative linear solvers should be exploited. Moreover, every regularization method crucially depends on the choice of one or more regularization parameters, which should be suitably tuned. The aim of this paper is twofold: (a) survey some well-established regularizing projection methods based on Krylov subspace methods (with a particular emphasis on methods based on the Golub-Kahan bidiagonalization algorithm), and the so-called hybrid approaches (which combine Tikhonov regularization and projection onto Krylov subspaces of increasing dimension); (b) introduce a new principled and adaptive algorithmic approach for regularization similar to specific instances of hybrid methods. In particular, the new strategy provides reliable parameter choice rules by leveraging the framework of bilevel optimization, and the links between Gauss quadrature and Golub-Kahan bidiagonalization. Numerical tests modeling inverse problems in imaging illustrate the performance of existing regularizing Krylov methods, and validate the new algorithms.

KEYWORDS

hybrid methods, imaging problems, Krylov subspace methods, large-scale linear inverse problems, regularization parameter choice rules, Tikhonov regularization

1 | INTRODUCTION

This paper considers linear, large-scale, discrete ill-posed problems of the form

\[ Ax_{\text{true}} + e = b_{\text{true}} + e = b, \]  

where the matrix \( A \in \mathbb{R}^{m \times n} \) represents a forward mapping that is ill-conditioned with ill-determined rank (ie, the singular values of \( A \) decay and cluster at zero without an evident gap between two consecutive ones), \( x_{\text{true}} \in \mathbb{R}^n \) is the desired solution, and \( e \in \mathbb{R}^m \) is some unknown noise that affects the data \( b \in \mathbb{R}^m \). Systems like (1) typically stem from the discretization of first-kind Fredholm integral equations, and model inverse problems arising in a variety of applications, such...
as (just to name a few): computed tomography in Medicine and Industry, image deblurring in Astronomy and Biology, inverse scattering in Geophysics, and parameter identification (see [1-5] and the references therein for more details). In this paper the noise \( e \) is assumed to be Gaussian white; although this may appear like a limitation, strategies for whitening the noise or approximating actual noise (even realized from distributions other than Gaussian) with Gaussian white noise are available in the literature; see, for instance, [1, Chapter 3] and [6,7]. In general, the linear system \( Ax = b \) might not be consistent, essentially because of the presence of noise in \( b \). However, even the noise-free linear system \( Ax_{\text{true}} = b_{\text{true}} \) may not be consistent, essentially because one should not assume a perfect agreement between the adopted (discretized) model and the true underlying physical model. In a discrete setting, the properties of inverse problems are best analyzed considering the singular value decomposition (SVD) of \( A [8, \text{Chapter 2}] \), that is,

\[
A = \hat{U} \Sigma \hat{V}^T, \quad \text{for} \quad \hat{U} = [\hat{u}_i]_{i=1,\ldots,m} \in \mathbb{R}^{m \times m}, \hat{V} = [\hat{v}_i]_{i=1,\ldots,n} \in \mathbb{R}^{n \times n}, \quad \text{and} \quad \Sigma = \text{diag}(\hat{\sigma}_1, \ldots, \hat{\sigma}_{\min{[m,n]}}) \in \mathbb{R}^{m \times n}.
\]

(2)

Here the columns \( \hat{u}_i \) and \( \hat{v}_i \) of the square orthogonal matrices \( \hat{U} \) and \( \hat{V} \) are the left and right singular vectors of \( A \), respectively, and the diagonal entries \( \hat{\sigma}_1 \geq \ldots \geq \hat{\sigma}_{\min{[m,n]}} \geq 0 \) of the rectangular matrix \( \hat{\Sigma} \) are its singular values. Depending on the decay of the singular values of \( A \), (1) is regarded as severely ill-posed (exponential decay) or moderately and mildly ill-posed (power decay); see [9, Chapter 2]. Let \( A^\dagger \) denote the Moore-Penrose pseudoinverse of \( A [8, \text{Chapter 5}] \). Because of the ill-conditioning of the coefficient matrix \( A \), and the noise in the right-hand side vector \( b \), the (unregularized) solution \( x = A^\dagger b \) to (1) is not guaranteed to be a meaningful approximation to \( x_{\text{true}} \). Indeed, assuming that the discrete Picard condition [10] holds for (1) (i.e., the spectral coefficients \( \hat{u}_i^T b_{\text{true}} \) decay on average faster than \( \hat{\sigma}_i \)), and expressing \( x \) with respect to the SVD of \( A \), leads to

\[
x = \sum_{i=1}^{\min{[m,n]}} \frac{\hat{u}_i^T b}{\hat{\sigma}_i} \hat{v}_i = \sum_{i=1}^{\min{[m,n]}} \frac{\hat{u}_i^T b_{\text{true}}}{\hat{\sigma}_i} \hat{v}_i + \sum_{i=1}^{\min{[m,n]}} \frac{\hat{u}_i^T e}{\hat{\sigma}_i} \hat{v}_i,
\]

(3)

where, on the right-hand-side, the singular components corresponding to the small singular values and the very oscillating right singular vectors in the second sum dominate and spoil \( x \) (since \( |\hat{u}_i^T e| \) is roughly constant because \( e \) is Gaussian white noise); see [1, Chapter 4] and the references therein for more details.

In order to compute a good approximation to \( x_{\text{true}} \), one should therefore regularize (1), that is, replace (1) with a problem closely related to it that is less sensitive to perturbations in the data. In the following, a generic regularized solution of (1), obtained by applying a regularized inverse \( A_{\text{reg}} \dagger \) to the noise-corrupted data \( b \), is denoted by \( x_{\text{reg}} = A_{\text{reg}}^\dagger b \). Although many approaches are possible to achieve regularization, this paper mainly focuses on strategies related to Tikhonov regularization method, which computes a regularized solution \( x_{\text{reg}} = x(\alpha) \) as follows

\[
x(\alpha) = \arg \min_{x \in \mathbb{R}^n} \left\{ \|Ax - b\|^2 + \alpha \|Lx\|^2 \right\}, \quad \text{with either} \quad L = I \quad \text{(standard form)} \quad \text{or} \quad L \neq I \quad \text{(general form)},
\]

(4)

and where \( \| \cdot \| \) denotes the vector 2-norm. In the above formulation, the regularization parameter \( \alpha \geq 0 \) balances the effect of the fit-to-data term \( \|Ax - b\|^2 \) and the regularization term \( \|Lx\|^2 \), which is defined with respect to a regularization matrix \( L \in \mathbb{R}^q \times n \). If \( L \neq I \), typical choices for \( L \) are rescaled finite differences approximations of derivative operators. The choices of both \( \alpha \) and \( L \) contribute to the success of Tikhonov regularization method (4). \( L \) should encode some available information about the solution, in such a way that unwanted behaviors of the regularized solution \( x(\alpha) \) show up in \( Lx(\alpha) \), whose magnitude is penalized: the bigger \( \alpha \), the more penalization, which is typically desirable if one is confident about the choice of \( L \). Many strategies have been proposed to conveniently choose \( L \); see, for instance, [11-14]. This paper will not focus on the choice of \( L \), but rather on parameter choice rules for \( \alpha \). Note that (4) can be equivalently reformulated as

\[
x(\alpha) = \arg \min_{x \in \mathbb{R}^n} \left\| \begin{bmatrix} A & \sqrt{\alpha} L \end{bmatrix} x - \begin{bmatrix} b \\ 0 \end{bmatrix} \right\|, \quad \text{or} \quad x_{\text{reg}} = x(\alpha) = (A^T A + \alpha L^T L)^{-1} A^T b = A_{\text{reg}}^\dagger b.
\]

(5)

where the latter assumes that the null spaces of \( A \) and \( L \) intersect trivially, that is, \( (A) \cap (L) = \{0\} \). Thanks to (5), one can conclude that computing the Tikhonov-regularized solution \( x(\alpha) \) essentially amounts to the solution of a least-squares problem (i.e., the first problem in (5)). An analytical expression for \( x(\alpha) \) is obtained by considering the associated system
of the normal equations (ie, the second equation in (5)). Therefore, at least in theory, any direct solver for least squares problems can be employed to compute \( x(\alpha) \). In order to analyze the behavior of the solution and its dependence on \( \alpha \), the preferred choice is to employ the SVD of \( A \) (defined in (2)) if \( L = I \), or the generalized SVD (GSVD) of \( (A, L) \) (see [8, Chapter 8]) if \( L \neq I \). Indeed, when \( L = I \), convenient expressions of the regularized solution \( x(\alpha) \) and the discrepancy \( b - Ax(\alpha) \) as functions of \( \alpha \) can be obtained using the SVD of \( A \) as follows

\[
x(\alpha) = \sum_{i=1}^{\min\{m,n\}} \phi_i(\alpha) \frac{\hat{u}_i^T b}{\hat{\sigma}_i} \hat{v}_i, \quad b - Ax(\alpha) = \sum_{i=1}^{\min\{m,n\}} (1 - \phi_i(\alpha)) (\hat{u}_i^T b) \hat{u}_i + \sum_{i=\min\{m,n\}+1}^{m} (\hat{u}_i^T b) \hat{u}_i, \quad \text{where} \quad \phi_i(\alpha) = \frac{\hat{\sigma}_i^2}{\hat{\sigma}_i^2 + \alpha} \leq 1.
\]

(6)

Here the scalars \( \phi_i(\alpha), i = 1, \ldots, \min\{m,n\} \), are the so-called filter factors that dampen the contributions to the regularized solution from the small singular values associated with the high-frequency components: the bigger the regularization parameter \( \alpha \), the more the amount of smoothing. Relations (6) become handy when applying some parameter choice rules. For context, another well-known spectral filtering method worth mentioning is the truncated SVD method [1, Chapter 6]. Thanks to (7), in the following, unless stated otherwise, only Tikhonov regularization in standard form will be formally considered (ie, \( A \) is concerned), \( L = I \) in (4)), with the convention that, with some abuse of notations, \( \bar{A}, \bar{x}, \) and \( \bar{b} \) in (7) may formally replace \( A, x, \) and \( b \), respectively, in (4).

Although all the derivations explained above are useful in understanding the basics behind (Tikhonov) regularization, they are of little practical use when it comes to large-scale problems (like the ones naturally associated, for instance, to imaging problems). In general, in these cases, the matrix \( A \) may be only available as a sparse matrix or in the form of a function that efficiently computes the actions of \( A \) and (possibly) \( A^T \) on vectors (ie, matrix-vector products). In other words, only fully storing and explicitly accessing the matrix \( A \) may require massive memory, let alone the computational cost of solving (5), which is typically of the order of \( O(n^3) \) (unless \( A \) and \( L \) have some special structure that is preserved when computing factorizations and can be exploited; see, for example, [16,17]). In all these situations, the only computationally viable approach to recover a solution of (1) is to apply an iterative linear solver that, as far as \( A \) is concerned, only requires matrix-vector products with \( A \) and (possibly) \( A^T \) (these are commonly referred to as “matrix-free” methods). Note that, for problems originally expressed in general Tikhonov form whose regularization matrix \( L \) does not have an exploitable structure, \( L_A^\dagger \) must also be computed iteratively; this is the case especially for 2D or 3D problems, when, for example, \( L \) is a reweighted finite differences approximation of a derivative operator (see, eg, [18]). Although sometimes it may be more desirable to regularize (1) by applying an iterative solver to (5), in some situations just applying an iterative solver to (1) can lead to a regularized solution. Such linear solvers are commonly dubbed iterative regularization methods.

In the remaining part of the paper, only iterative regularization methods will be considered. More precisely, Section 2 surveys some well-established approaches to iterative regularization, which include stand-alone iterative Krylov solvers for (1) and methods that combine Tikhonov regularization and projection onto Krylov subspaces; a special emphasis will be given on methods based on the Golub-Kahan bidiagonalization algorithm. Sections 3 and 4 deal with practical regularization parameter choice rules that can be efficiently employed when combining Tikhonov regularization and Krylov
projection methods, also introducing a new principled class of algorithms where computations for expanding the approximation (projection) subspace for the solution and choosing the regularization parameter are interlaced. Section 5 displays some numerical results on test problems in image deblurring and computed tomography, where the performances of methods that combine Krylov solvers and Tikhonov regularization are compared. Section 6 presents concluding remarks.

## 2 | REGULARIZING KRYLOV PROJECTION METHODS

As mentioned at the end of the previous section, when dealing with large-scale problems (1), only iterative solvers can generally be employed. The most natural way of computing a regularized solution in this setting is by “early” termination of the iterations of a (convergent, regularizing) solver applied to (1); see, for instance, [1, Chapter 6] and the references therein. Let \( x_k \) denote the approximate solution at the \( k \)th iteration of a linear solver for (1). During the first iterations, that is, for \( k \) “small enough,” \( x_k \) approaches the solution \( x_{\text{true}} \) of the noise-free system, and the relative reconstruction error

\[
\text{RRE}(x_k) = \frac{||x_k - x_{\text{true}}||}{||x_{\text{true}}||} \tag{8}
\]

decreases. However, as \( k \) increases after a certain optimal iteration \( \hat{k} \) (ie, when RRE\((x_k)\) is minimal), RRE\((x_k)\) starts increasing as well and, indeed, \( x_k \) approaches the (unregularized and unwanted) solution \( x = A^T b \) (3). This phenomenon is referred to as “semiconvergence” (after [19, Chapter IV]). In this framework, the number of iterations clearly plays the role of a regularization parameter that must be carefully set according to one or more stopping criteria (acting in this situation as regularization parameter choice rules): performing \( k \ll \hat{k} \) iterations leads to over-regularized solutions, while performing \( k \gg \hat{k} \) iterations leads to under-regularized solutions.

Many well-known iterative solvers (such as Landweber method, Kaczmarz method, and many Krylov subspace methods) are known to semiconverge, and are considered regularization methods according to the following classical definition [5, Chapter 3]

\[
\lim_{\|x\| \to \|b - b_{\text{true}}\|} \sup_{\|r\| \leq \|r\|_0} \|x_{\text{reg}} - x_{\text{true}}\| = 0 , \text{ where } x_{\text{reg}} \text{ is determined according to a given regularization parameter choice rule} \, . \tag{9}
\]

It should be highlighted that not all the parameter-choice rules in use today adhere to this equation; see Section 3 for more details. When employing a regularizing iterative solver, \( x_{\text{reg}} = x_{\hat{k}} \), where \( \hat{k} \) is chosen according to a specified stopping rule (acting as a regularization parameter choice rule).

In a discrete setting one can argue that this definition is not of practical use and, indeed, despite many iterative solvers satisfying (9), the most effective ones are those that, having a comparable computational cost per iteration, reach the optimal iteration \( \hat{k} \) quite “quickly”. Moreover, the quality of the reconstructed solution at iteration \( \hat{k} \) matters. In the framework of this paper, where regularization in norms other than the vectorial 2-norm is not considered, this usually means that the quality of \( x_{\hat{k}} \) is comparable to the quality of a \( x_{\text{reg}} \) that would have been achieved by applying a direct filtering method; see, for example [20]. Indeed, a possible way of exploring the regularizing properties of iterative methods is to formally reformulate them as filtering methods, where the filter factors depend on the iteration number \( k \): this has been done for basic stationary iterative methods, such as Landweber, but also for some Krylov subspace methods, such as CGLS; see [21, Chapter 6] and the references therein.

In the following, only Krylov subspace methods will be considered as iterative solvers for (1) (or (4)). Krylov subspace methods belong to the class of iterative projection methods [22, Chapter 5], that is, at the \( k \)th iteration, and assuming (without loss of generality) a zero initial guess for the solution, an approximate solution \( x_k \) is computed by imposing that \( x_k \) belongs to a \( k \)-dimensional (approximation) subspace \( S_k^{(1)} \) and that the residual vector is orthogonal to a \( k \)-dimensional (constraint) subspace \( S_k^{(2)} \). Both the spaces \( S_k^{(1)} \) and \( S_k^{(2)} \) are typically expanded as the iterations proceed and, for Krylov subspace methods, they are both Krylov subspaces, that is, subspaces of the form

\[
\mathcal{K}_k(G, c) = \text{span}\{ c, Gc, \ldots , G^{k-1}c \} , \tag{10}
\]

where \( G \) is a square matrix and \( c \) is a vector of coherent length; referring to (1) or (4), \( G \) and \( c \) are defined in terms of \( A \) and \( b \). Although, in general, there is no guarantee that the dimension of \( \mathcal{K}_k(G, c) \) is \( k \), in the following it will be assumed...
(without loss of generality) that this is the case. Indeed, the dimension of $\mathcal{K}_k(G, c)$ is $k$ if and only if the degree of the minimal polynomial of $c$ with respect to $G$ [22, Chapter] is greater than or equal to $k$. Krylov subspace methods for system (1) primarily differ for the choice of the two subspaces $S_k^{(1)}$ and $S_k^{(2)}$, but also for specific formulations adopted at implementation stage (methods that share the same $S_k^{(1)}$ and $S_k^{(2)}$ but are implemented differently are indeed mathematically equivalent). When $S_k^{(1)} = A S_k^{(2)} = A \mathcal{K}_k(G, c)$, Krylov subspace methods can be reformulated as constrained optimization methods computing

$$x_k = \arg \min_{x \in \mathcal{K}_k(G, c)} \|Ax - b\| ;$$  \hfill (11)

see, for example, [22, Chapter 5]. In the following, only Krylov methods that, when applied to (1), satisfy (11), will be taken into account. Formally, at iteration $k$, all Krylov subspace methods can be reformulated to update a partial factorization of the form

$$A W_k = Z_{k+1} \overline{R}_k ,$$  \hfill (12)

where $W_k \in \mathbb{R}^{n \times k}$ has orthonormal columns that span the approximation subspace for the solution $\mathcal{K}_k(G, c)$, $Z_{k+1} \in \mathbb{R}^{m \times (k+1)}$ has orthonormal columns, and $\overline{R}_k \in \mathbb{R}^{(k+1) \times k}$. Along with the partial factorization (12), for some Krylov methods another factorization involving $A^T$ is jointly updated at the $k$th iteration (this is, for instance, the case for LSQR method).

Using decomposition (12) and the properties of the matrices appearing therein, problem (11) can be reformulated as follows

$$\text{compute } x_k = W_k y_k, \text{ where } y_k = \arg \min_{y \in \mathbb{R}^k} \| A W_k y - b \| = \arg \min_{y \in \mathbb{R}^k} \| Z_{k+1} (\overline{R}_k y - Z^T_{k+1} b) \| = \arg \min_{y \in \mathbb{R}^k} \| \overline{R}_k y - Z^T_{k+1} b \|. \hfill (13)$$

The rightmost problem appearing in (13) is a projected least squares problem of dimension $k \ll \min \{m, n\}$ and, therefore, when adopting Krylov subspace methods to approximate the solution of (1), the task of solving (regularizing) a full-dimensional system (1) is reduced to the task of solving a sequence of projected least squares problems of the form (13). Linking to the notations introduced in Section 1 to denote a generic regularization method, $x_k$ in (11) and (13) can be expressed as $x_k = x_{\text{reg}} = A_{\text{reg}}^T b$, where $A_{\text{reg}}^T = W_k \overline{R}_k Z^T_{k+1} b$ (note that $x_{\text{reg}}$ depends on the number of iterations $k$, that acts as a regularization parameter). Table 1 lists some relevant Krylov subspace methods, giving some details about $\mathcal{K}_k(G, c)$ (equivalently, $S_k^{(1)}$ and $S_k^{(2)}$) and the matrices appearing in (12).

It must be stressed once more that, with some abuse of notation, here $A$ is assumed to refer to the original coefficient matrix appearing in (1), or to the matrix $\overline{A} = AL^A_j$ appearing in (7). In the latter case, the regularization matrix $L$ affects the approximation subspace for the solution, in that $L^A_j$ can be formally regarded as a right preconditioner for the linear system in (1). Indeed, when computing $x_k \in \mathcal{K}_k(G, c)$ according to (11), (13), and properly accounting for the transformation (7), the following expression for the $k$th approximate solution should be considered

$$x_k \in x_0 + \mathcal{K}_k(G, c), \text{ where } x_0 \in \mathcal{N}(L), G \text{ is defined in terms of } A \text{ and } L^A_j, c \text{ is defined in terms of } b, A, \text{ and } x_0 ; \hfill (14)$$

| Method        | Factorization (12) | $S_k^{(1)}$ | $S_k^{(2)}$ | $W_k$ | $Z_k$ | $\overline{R}_k$ | References          |
|---------------|--------------------|-------------|-------------|-------|-------|-------------------|---------------------|
| GMRES         | Arnoldi            | $\mathcal{K}_k(A, b)$ | $A \mathcal{K}_k(A, b)$ | $V_k$ | $V_k$ | $\overline{H}_k$ | Upper Hessenberg    |
| RR-GMRES      | Arnoldi            | $\mathcal{K}_k(A, Ab)$ | $A \mathcal{K}_k(A, Ab)$ | $V_k$ | $V_k$ | $\overline{H}_k$ | Upper Hessenberg    |
| GMRES         |                    |             |             |       |       |                   |                     |
| LSQR          | equivalent to CGLS | $\mathcal{K}_k(A^T A, A^T b)$ | $A \mathcal{K}_k(A^T A, A^T b)$ | $V_k$ | $U_k$ | $\overline{B}_k$ | Lower bidiagonal     | [28, 29]           |

**Table 1** Examples of Krylov subspace methods commonly employed as iterative regularization methods.
see [1, Chapter 8; 23]. However, \( L_A \) should not be practically regarded as traditional preconditioner: indeed, it does not contribute to speeding up the convergence of the solver (which is undesirable because of semiconvergence [24]) but, on the contrary, most often results in a delayed semiconvergence of the iterative regularization method to a regularized solution of enhanced quality. Indeed, incorporating \( L \neq I \) within the Krylov subspace (10) can largely contribute to building a better approximation subspace for the solution of (1) (and (4)). In other words, similarly to Tikhonov regularization method in general form (4) and to the so-called “truncated” GSVD method [25], including \( L_A^T \) in (14) enforces regularization in the (semi)norm \( \|L \cdot \| \) (even if no penalization appears in (11), (13)).

Much recent (and ongoing) research is studying regularizing properties of Krylov methods of the form (13), expanding on the theoretical definition (9). For instance, [30–32] investigate the quality of the approximation of the SVD of \( A \) obtained when considering the SVD of the projected matrix \( \hat{R}_k \); [33] investigates whether properties like the discrete Picard condition are inherited by the projected problem in (13). The results of these studies often depend on the original problem (1) being severely, moderately, and mildly ill-posed (the first category usually enjoying better regularization properties). Also, more thoughtful investigations of the effect of the noise on the approximation subspace (10) for the solution have been performed in [34–36] for a variety of Krylov subspaces, concluding that, in some cases, an accurate estimate of the amount of noise in the data \( b \) can be also recovered. Thanks to these new results, new insight into the semiconvergence phenomenon for Krylov methods of the form (13) can be gained: namely, these methods semiconverge because of a combination of the noise entering the approximation subspace for the solution \( \mathbf{K}_k \), and the projected matrix \( \hat{R}_k \) inheriting the ill-conditioning of \( A \) as \( k \) increases.

The methods based on the Arnoldi algorithm require only one matrix-vector product with \( A \) per iteration; therefore, assuming that the dominant cost per iteration of an iterative solver is the computation of matrix-vector products with matrices related to \( A \) (which might not be the case for a sparse or structured \( A \)), these methods are the cheapest ones among the Krylov solvers listed in Table 1. However, the use of GMRES as a regularization method is not widespread. Indeed, although some results on the SVD approximation properties of GMRES have been recently obtained (even in a continuous setting [37]), in some situations (e.g., when \( A \) is highly non-normal), GMRES performs worse than other Krylov subspace methods. One of the reasons for the bad performance of GMRES as an iterative regularization method may be that the approximation subspace for the solution explicitly contains the vector \( b \), which is noise-corrupted (indeed, this prompted the introduction of RR-GMRES [26]). Moreover, the approximation subspaces generated by both GMRES and RR-GMRES may not be suitable to represent the exact solution as they basically only contain information about the range of \( A \), or may be affected by a severe ‘mixing’ of the singular value components of the solution (so that GMRES and RR-GMRES cannot be interpreted as filtering methods); see [38]. When \( A \) is symmetric, GMRES (resp. RR-GMRES) reduces to the well-known MINRES [39] (resp. MR-II [40]) method, whose regularization properties are well-studied and well-accepted; see [41]. It should be stressed that, even if the original coefficient matrix \( A \) in (1) is square, if \( L \neq I \) is rectangular, then the preconditioned matrix \( \hat{A} = AL_A^T \) in (14) is not square anymore: in this situation, applying GMRES (and variants thereof) might still be possible by either defining square counterparts of such rectangular matrices (see [42,43]) or by revisiting the standard form transformation (7) (see [44]). When a regularization matrix is incorporated, GMRES applied to the transformed system may perform significantly better than its standard counterpart [45].

When it comes to regularization, LSQR is arguably the most widespread Krylov method among the ones listed in Table 1; in the remaining part of this paper, unless otherwise stated, only LSQR will be considered as a Krylov projection method. More details about LSQR are hereby provided. Given the initial vectors \( v_0 = 0 \) and \( u_1 = b/\|b\| \), and taking \( \zeta_1 = \|b\| \), the \( j \)th iteration of the Golub-Kahan bidiagonalization (GKB) algorithm computes

$$
\rho_j v_j = A^T u_j - \zeta_j v_{j-1}, \quad \zeta_{j+1} u_{j+1} = A v_j - \rho_j u_j, \quad j = 1, 2, \ldots ,
$$

with \( \rho_j \) and \( \zeta_{j+1} \) chosen so that \( \|v_j\| = 1 \) and \( \|u_j\| = 1 \), respectively. Breakdown of GKB happens if \( \rho_j = 0 \) (i.e., range(\( U_k \)) is invariant with respect to \( A A^T \)) or if \( \zeta_{j+1} = 0 \) (i.e., range(\( V_k \)) is invariant with respect to \( A^T A \)): this can only happen if (1) is consistent (which is not necessarily the case considered in this paper) or if \( x_k \) exactly solves the least squares problem \( \min_{x \in \mathbb{R}^n} \|Ax - b\| \) associated with (1) (which is not a desirable situation when performing regularization), respectively; see, for instance, [46] for more details. Otherwise GKB continues until the maximum dimensionality of the problem (i.e, \( \min\{m, n\} \) is reached. The

**Assumption:** The GKB algorithm does not breakdown,
will be made from now on, without loss of generality. Indeed, as already emphasized, LSQR (and, in general, all the regularizing Krylov methods), are meaningful if the total number of iterations $k$ is quite low (i.e., $k \ll \min\{m, n\}$), so that one can usually safely assume that no breakdown will happen. Looking at (15) one can see that each iteration requires computing one matrix-vector product with $A$ and one matrix-vector product with $A^T$. Equivalently, after $k$ iterations of GKB are performed, one can write partial matrix factorizations of the form

$$AV_k = U_k B_k + \zeta_{k+1} u_{k+1} e^T_k = U_{k+1} \overline{B}_k, \quad A^T U_k = V_k B_k^T,$$

where $V_k = [v_1, \ldots, v_k] \in \mathbb{R}^{n \times k}$ and $U_{k+1} = [U_k, u_{k+1}] = [u_1, \ldots, u_k, u_{k+1}] \in \mathbb{R}^{m \times (k+1)}$, are matrices whose orthonormal columns span the Krylov subspaces $\mathcal{K}_k(A^T A, A^T b)$ and $\mathcal{K}_k(A A^T, b)$, respectively; $B_k$ and $\overline{B}_k$ are lower bidiagonal matrices of the form

$$B_k = \begin{bmatrix} 
\rho_1 & \zeta_2 & \rho_2 & \ldots & \rho_{k-1} & \zeta_k & \rho_k \\
\zeta_2 & \rho_2 & \ldots & \rho_{k-1} & \zeta_k & \rho_k \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
\zeta_k & \rho_k 
\end{bmatrix} \in \mathbb{R}^{k \times k}, \quad \overline{B}_k = \begin{bmatrix} B_k \\
\zeta_{k+1} e^T_k 
\end{bmatrix} \in \mathbb{R}^{(k+1) \times k}.$$ (18)

Here and in the following, $e_i$ denotes the $i$th canonical basis vector of $\mathbb{R}^d$, $d \geq i$. Note that the first decomposition appearing in (17) is a specific realization of the more general decomposition in (12).

**Combining Tikhonov regularization and Krylov methods**

Although, by applying a regularizing iterative solver to (1) and stopping the iterations sufficiently early, thanks to semiconvergence one is successful in computing a regularized solution to (1), special attention should be made to select a proper stopping criterion for the iterations. As hinted at the end of Section 1, an alternative approach to apply iterative regularization to (1), which does not rely on semiconvergence, is to apply (till convergence) an iterative solver to the Tikhonov-regularized problem (4) (most often considering its equivalent formulation as a damped least squares problem in (5)). By doing so, if the selected iterative solver is a projection method and, in particular, a Krylov subspace method, one is within the framework of the so-called first-regularize-then-project methods; see [1, Chapter 6]. In other words, by exploiting decomposition (12) and the properties of the matrices appearing therein, the $k$th iteration of these methods computes

$$x_\alpha(\alpha) = \arg \min_{x \in \mathcal{K}_k(G, c)} \{ ||Ax - b||^2 + \alpha ||x||^2 \} = W_k y_k(\alpha), \quad \text{where } y_k(\alpha) = \arg \min_{y \in \mathbb{R}^m} \{ ||\overline{R}_k y - Z_{k+1}^T b||^2 + \alpha ||y||^2 \}. \quad (19)$$

An upside of this approach is that the stopping criterion for the iterations plays a less crucial role in the regularization process. However, a clear downside is that, if a good value for the regularization parameter $\alpha$ is not available a priori (as it is the case), then one should repeatedly fully solve problems of the form (4), one for every considered value of the regularization parameter (these may belong to a discrete set of values chosen a priori, or may be sequentially set by trying to satisfy some well-known parameter choice rules; see Section 3 for more details). Fortunately, when using a Krylov method to solve problem (4) in standard form for a varying $\alpha$, most of the computations performed to determine $x_\alpha(\alpha)$ for a given $\alpha$ can be smartly rearranged or recycled to compute $x_\alpha(\alpha')$ when $\alpha' \neq \alpha_{\alpha+1}$ (exploiting, essentially, the so-called shift-invariance of Krylov subspaces, that is, $\mathcal{K}_k(G + \alpha_r I, c) = \mathcal{K}_k(G + \alpha_{r+1} I, c)$; see [47, 48]). Note that, in general, the number of iterations $k$ for solving (4) depends on $\alpha$.

Another strategy to combine a Krylov (projection) method and Tikhonov method to compute a regularized solution of (1) is to perform a so-called hybrid method; see [4] and the references therein. At the 4th iteration of a hybrid method, one projects problem (4) onto a Krylov subspace $\mathcal{K}_k$ (so that a formulation like the one in (13) is recovered), and then applies some (standard form) Tikhonov regularization to the projected least squares problem. By doing so, one is within the framework of the so-called first-project-then-regularize methods; see [1, Chapter 6].
by using decomposition (12) and the properties of the matrices appearing therein, the kth iteration of these methods computes

\[
x_k(a_k) = \arg \min_{x \in \mathcal{K}_k(G,c)} \{ \|Ax - b\|^2 + \alpha_k \|x\|^2 \} = W_k y_k(a_k), \text{ where } y_k(a_k) = \arg \min_{y \in \mathbb{R}^n} \{ \| \tilde{R}_k y - Z_{k+1}^T b \|^2 + \alpha_k \|y\|^2 \}.
\]  

(20)

By comparing the two formulations (19) and (20) it is evident that, if \(a_k = \alpha\), then these two approaches are equivalent. When adopting hybrid methods (20), the search for a good regularization parameter is done on the fly using well-known parameter choice rules on the projected problem only, which has relatively small dimensions (if \(k \ll \min\{m, n\}\)) and is therefore less computationally demanding than working with the original large-scale problem (4); see [49] and Section 3 for more details. It should also be emphasized that, although \(a_k\) is a suitable regularization parameter for the \(k\)th iteration (ie, it is in principle a local choice, good for the \(k\)th projected Tikhonov problem (20) only), when \(k\) increases \(a_k\) seems to stabilize around a value that is good for the full-dimensional problem, too; see [50]. In the following, among the approaches (19) and (20), only the latter will be considered (unless stated otherwise).

When combining projection methods and Tikhonov regularization, one is effectively dealing with two-parameter approaches, meaning that a good regularized solution to (1) is achieved by a sensible interplay of the number of performed iterations and the chosen Tikhonov regularization parameter. In particular, when comparing these methods to purely iterative Krylov methods (13), setting a good stopping rule for \(k\) is less crucial for recovering a suitable regularized solution, and often a larger approximation subspace \(\mathcal{K}_k\) can be generated, where the regularized solution can be more accurately recovered.

Linking to the notations introduced in Section 1 to denote a generic regularization method, \(x_k(a_k)\) in (20) can be expressed as \(x_{\text{reg}} = A_{\text{reg}}^\dagger\), where

\[
A_{\text{reg}} = W_k^T \tilde{R}_k + a_k I - 1 \tilde{R}_k Z_{k+1}^T b
\]

(note that \(x_{\text{reg}}\) depends on both the number of iterations and the Tikhonov parameter \(a_k\)).

### 3 | EXISTING PARAMETER CHOICE RULES FOR LARGE-SCALE PROBLEMS

Choosing one (or more) suitable regularization parameter(s) is a nontrivial and heavily problem-dependent aspect of every regularization method. For large-scale problems, this has been a widely-studied topic covered in the classical bibliography about Tikhonov, (purely) iterative, and hybrid regularization methods (see, eg, [21, Chapter 7], [51, Chapter 15]), as well as in more or less recent survey papers (see, eg, [49,52,53]). In the hybrid framework, two similar but differently principled approaches can be used: using a projected parameter choice scheme, or using an approximation of the full dimensional parameter choice scheme that only relies on information available within the projected problem. In this section, only a few parameter choice strategies that are well-established in these settings will be reviewed: they can all be formulated as minimization problems, whose objective functions \(P(x_{\text{reg}})\) depend on one (or more) regularization parameter(s) through \(x_{\text{reg}}\) (ie, the regularized solution that is obtained by (4), or (13), or (20)). Table 2 gives more details about the strategies considered in this paper, specified for the Tikhonov regularization case (4), that is, \(x_{\text{reg}} = x(\alpha)\); the information reported there (and in this section) is by no means exhaustive.

If a good estimate of the norm of the error in the right hand side \(\|e\|\) is available, coming from existing knowledge about the problem or recovered by noise estimation algorithms (which may be the same Krylov solver employed for regularization [34,35]), the most widespread and reliable method is the so-called discrepancy principle [54]. It consists in choosing the regularization parameter such that the solution \(x_{\text{reg}}\) satisfies

\[
\varepsilon := \tau \|e\| = \|Ax_{\text{reg}} - b\|.
\]

(21)

where \(\tau\) is a (safety) parameter, usually chosen slightly bigger than 1. When \(e\) is Gaussian white noise with covariance matrix \(\eta^2 I\), one typically takes \(\|e\| = \eta \sqrt{n}\) (ie, the expected value of \(\|e\|\)). When considering Tikhonov method (4), one applies a zero finder, for example, [55], to solve with respect to \(\alpha\) the nonlinear equation \(\varepsilon = \|Ax_{\text{reg}} - b\|\). Note that this is an equivalent formulation to the (unconventional) one reported in Table 2; more details will be given in Section 4.2. When considering purely iterative methods satisfying (13), this corresponds to stopping the iterations as soon as the discrepancy norm \(\|Ax_{\text{reg}} - b\|\) lies below \(\varepsilon\) for an iteration \(k\) (the subsequent iterations are guaranteed to have discrepancy norms lying below \(\varepsilon\) because of the optimality property of the residual in (13) that, in this case, coincides with the discrepancy). When
considering hybrid methods (20), at the $k$th iteration, one typically solves (with respect to $\alpha$) the projected version of (21), that is,

$$\varepsilon = \|A\mathbf{x}_k(\alpha) - b\| = \|A\mathbf{x}_k(\alpha_k) - b\| = \|\widetilde{R}_k\mathbf{y}_k(\alpha_k) - Z_{k+1}^Tb\|.$$ (22)

The last equality in (22) holds thanks to the generic decomposition (12) and the properties of the matrices appearing therein. In other words, one typically determines the Tikhonov parameter $\alpha_k$ to be employed at the $k$th hybrid iteration by imposing (22), and sets an (unrelated) stopping criterion for the iterations. For instance, the authors of [57] first determine a suitable iteration $k$ assuring that equation (22) has a unique solution, apply a zero finder to solve (22), and then compute $\mathbf{x}_k(\alpha_k)$. However the value of $k$ so determined may be too small to guarantee a good approximation, and usually a few extra iterations are performed to improve the quality of the solution by using a richer approximation subspace. The authors of [56] derive the so-called “secant update” method, which solves (22) by performing one step of a secant-like zero finder at each hybrid iteration (to update the regularization parameter for the projected problem (20)), stopping when (22) is satisfied. As an upside, when a method (13) and (20) is adopted with the discrepancy principle, one can typically prove that the resulting strategy is a regularization method according to the definition given in (9). As downsides, the discrepancy principle relies on having an accurate estimate of $\|\varepsilon\|$, and tends to oversmooth the regularized solution [21, Chapter 7].

Another parameter choice rule that can be employed when the standard deviation $\eta$ of the noise $\varepsilon$ is known is the unbiased predictive risk estimator (UPRE), which is based on a statistical estimator of the mean squared norm of the predictive error, and consists in computing

$$\min U_{\text{reg}}(\mathbf{x}_{\text{reg}}), \quad \text{where} \quad U_{\text{reg}}(\mathbf{x}_{\text{reg}}) = \frac{1}{m} \|A\mathbf{x}_{\text{reg}} - b\|^2 + \frac{2\eta^2}{m} \text{trace}(AA_{\text{reg}}^*) - \eta^2,$$ (23)

where the minimization happens over the set of admissible regularization parameters. When considering Tikhonov method (4) one typically applies a local optimization method (e.g., Newton’s method) to find a local minimizer of $U_{\text{reg}}(\mathbf{x}_{\text{reg}}) = U(\mathbf{x}(\alpha))$ with respect to $\alpha$; some strategies based on Krylov subspace methods, or randomized trace estimators, may be employed to efficiently estimate the trace term in (23) when adopting a first-regularize-then-project approach (19); see [49, 65]. When considering purely iterative regularization methods (13) one should monitor the value of the functional $U_{\text{reg}}(\mathbf{x}_{\text{reg}}) = U_k(\mathbf{x}_k)$ at the $k$th iteration, and stop as soon as a significant growth happens. UPRE has been adapted in [58] to work in the hybrid framework (20), where a projected version of the optimization problem (23) is solved with respect to the Tikhonov parameter at each iteration $k$, to determine a suitable $\alpha = \alpha_k$ to be employed in (20).

Parameter choice rules that do not use $\|\varepsilon\|$ nor any other information about $\varepsilon$ are sometimes called “heuristic methods” because it has been shown that no regularization method equipped with any of these rules can be proven to satisfy the

| Parameter choice rule | Objective function $P(\mathbf{x}_{\text{reg}}) = P(\mathbf{x}(\alpha))$ | References |
|------------------------|----------------------------------|------------|
| Discrepancy principle (DP) | $b^T((\alpha I - \alpha A^T b + 2AA^T \log((\alpha A^T + \alpha I)^{-1})) b - b^T(\alpha A^T b)^2 \mathbf{x}(\alpha)$ (where $T \approx \|\varepsilon\|/\|b\|$) | [54,55] [40] [56,57] |
| Unbiased Predictive (UPRE) | $\frac{1}{\text{trace}(I - A(\alpha A^T - \alpha I)^{-1} A^T)} \text{trace}(A(\alpha A^T - \alpha I)^{-1} A^T) - \eta^2$ | [2, Chapter 7] [2, Chapter 7] [58] |
| Generalized Cross Validation (GCV) | $\frac{\|\mathbf{y}(\alpha)\|^2}{\text{trace}(I - A(\alpha A^T - \alpha I)^{-1} A^T)} \text{trace}(I - A(\alpha A^T - \alpha I)^{-1} A^T) - \eta^2$ | [59,60] [21, Chapter 7] [30,58,61] |
| L-curve | $\frac{\log(\|\mathbf{y}(\alpha)\|)}{\log(\|\mathbf{y}(\alpha)\|)} - (\log(\|\mathbf{y}(\alpha)\|) \log(\|\mathbf{y}(\alpha)\|))^{\frac{1}{2}}$ | [62] [62] [31,63] |
| Regińska criterion | $\|\mathbf{b} - A\mathbf{x}(\alpha)\| \|\mathbf{x}(\alpha)\|^2, \quad \xi > 0$ | [64] [52] [31] |
A popular strategy belonging to this class of rules is the so-called generalized cross validation (GCV), which is based on a statistical tool (cross validation) used to predict possible missing data values (so that the regularization parameter is chosen to obtain the best prediction with respect to any supposedly missing entry \( b_i, i = 1, \ldots, m \), of the vector \( b \)). In practice, GCV requires the computation of

\[
\min G_{\text{reg}}(x_{\text{reg}}), \quad \text{where} \quad G_{\text{reg}}(x_{\text{reg}}) = \frac{\|Ax_{\text{reg}} - b\|^2}{\text{trace}(I - AA_{\text{reg}}^\dagger))^2}
\]  

(24)

and where the minimization happens over the set of admissible regularization parameters. This strategy was first proposed and analyzed in [59] for Tikhonov method (4). Application of GCV to purely iterative methods (13) is not straightforward, because of the different approaches that can be adopted to compute the trace at the denominator of \( G_{\text{reg}} \) in (24); see [21, Chapter 7]. When adopting a first-regularize-then-project approach (19) in connection with large-scale (and possibly matrix-free) problems, and similarly to UPRE (23), the main difficulty in applying GCV is the computation of the denominator of the GCV functional \( G_{\text{reg}} \). This issue is commonly alleviated either by using a Monte Carlo approach [66] or by using stochastic trace estimators, for example, Hutchinson’s stochastic trace estimator [65]. More precisely, [60] uses the latter strategy, by also exploiting links between GKB and Gaussian quadrature rules to approximate the GCV functional \( G_{\text{reg}} \) as a function of the Tikhonov regularization parameter \( \alpha \); once these approximations give bounds that are tight enough, they are used to determine a suitable value for \( \alpha \), and method (19) can be employed to find a regularized solution; see also [51, Chapter 15] and references therein for more details. GCV and variations thereof have also been used in connection with hybrid methods (20), based either on GKB or other decompositions of the kind (12); for instance, [61] adopts a weighted GCV approach, and [30,58] derive a modified GCV approach, where also insight about the relationship between a projected functional approximating the GCV functional \( G_{\text{reg}}(x_{\text{reg}}) \) and \( G_{\text{reg}}(x_{\text{reg}}) \) itself is given. All these hybrid strategies have in common that an approximated version of the optimization problem (24) is solved with respect to the Tikhonov parameter \( \alpha \) at each iteration \( k \), to determine a suitable \( \alpha = \alpha_k \) to be employed in (20); in doing so, these strategies often overcome another known drawback of the GCV rule (24), namely, the fact that the GCV function \( G_{\text{reg}}(x_{\text{reg}}) \) is usually very flat around its minimum [67].

Another very popular scheme that does not require information about the noise \( e \) in (1) is the so-called L-curve criterion [62]: it consists in plotting the curve \( L(x_{\text{reg}}) = (\log(\|x_{\text{reg}}\|), \log(\|Ax_{\text{reg}} - b\|)) \) with respect to different values of the regularization parameter (so that each point on the L-curve corresponds to one regularization parameter), and then finding the corner of such curve (which is hopefully shaped as an “L”). The rationale behind the L-curve criterion is that the corner of the L-curve is supposed to represent an good balance between the value of \( \|x_{\text{reg}}\| \) (which should not be neither too small nor too big, to avoid over- or under-regularization, respectively) and the value of \( \|Ax_{\text{reg}} - b\| \) (which should not be neither too small nor too big, to avoid under- or over-regularization, respectively). In practice, the L-curve criterion requires the computation of

\[
\min (-\chi(L(x_{\text{reg}}))), \quad \text{where} \quad \chi \quad \text{denotes the curvature of the L-curve},
\]  

(25)

and where the minimization happens over the set of admissible regularization parameters. When considering Tikhonov method (4), any optimization routine can be run to solve (25) with \( x_{\text{reg}} = x(\alpha) \) with respect to the Tikhonov regularization parameter \( \alpha \); see [62]. Although the L-curve criterion works well in many cases, when paired with Tikhonov method it might not be consistent with the classical definition of regularization method (9); see, for example, [68,69] and [21, Chapter 7]. When considering a purely iterative method (13), the L-curve criterion can be meaningful only if both \( \|Ax_k - b\| \) and \( \|x_k\| \) are monotonic; while the former is guaranteed by the optimality property of the residual (11), the latter may not hold for all solvers (eg, among the ones listed in Table 1, GMRES and RR-GMRES). When dealing with large-scale problems by adopting a first-regularize-then-project approach, once one should typically solve one regularized problem for each of the points lying on the L-curve, a possible computationally feasible fix is to compute only a few points of the curve using the strategy (19), and then find the corner using interpolation techniques; see [62]. When adopting a hybrid method (20), a projected L-curve is computed at the kth iteration (ie, \( L_k(x_k(\alpha_k)) = (\log(\|x_k(\alpha_k)\|), \log(\|Ax_k(\alpha_k) - b\|)) \) and an optimization problem analogous to (25) is solved with respect to the Tikhonov parameter \( \alpha_k \) at each iteration \( k \); computing every point on the projected L-curve is feasible (either because \( k \ll \min \{m, n\} \) [31] or because computations for a given set of predetermined values of the Tikhonov parameter are simultaneously updated from one iteration to the next one [63]). When hybrid methods based on GKB are employed, the links between GKB and Gaussian quadrature rules have
been exploited in [70,71] to derive upper and lower bounds (ie, boxes) for the points on the L-curve (this is the so-called “L-ribbon” method) or to compute curvature bounds [72] at each iteration.

Closely related to the L-curve criterion is the Regińska criterion [64], which consists in solving

\[
\min R_{\text{reg}}(x_{\text{reg}}), \quad \text{where} \quad R_{\text{reg}}(x_{\text{reg}}) = \|Ax_{\text{reg}} - b\|\|x_{\text{reg}}\|^\xi, \quad \text{where} \quad \xi > 0 \quad \text{is fixed},
\]

and where the minimization happens over the set of admissible regularization parameters. This strategy was first proposed and analyzed in [64] for Tikhonov method (4); an algorithm to find a local minimum of \( R_{\text{reg}}(x_{\text{reg}}) = R_{\text{reg}}(x(\alpha)) \) with respect to \( \alpha \) is derived in [73]. Application in the setting of purely iterative methods (13) was described in [52].

The Regińska criterion (26) has then been adapted in [31] to work within the framework of hybrid methods (20); essentially, this involves computing a projected Regińska functional at the \( k \)th iteration (ie, \( R_{\text{reg}}(x_{\text{reg}}) = R_k(x_k(\alpha_k)) = \|Ax_k(\alpha_k) - b\|\|x_k(\alpha_k)\|^\xi \)) and minimizing it with respect to the Tikhonov parameter \( \alpha_k \).

### 4 A NEW BI-LEVEL OPTIMIZATION FRAMEWORK FOR REGULARIZATION

All the rules presented in Section 3 for hybrid regularization (ie, first-project-then-regularize) methods (20) have in common that, together with the regularization method itself, can be reformulated as a sequence of problems of the form

\[
\min_{\alpha_k \geq 0} P_k(x_k(\alpha_k)) \quad \text{subject to} \quad x_k(\alpha_k) = \arg\min_{x \in \mathbb{K}_k} F(x, \alpha_k), \quad k = 1, 2, \ldots
\]

where \( F(x, \alpha_k) \) is the objective function appearing in the first optimization problem in (20) and \( P_k(x_k(\alpha_k)) \) is a condition that allows a suitable choice of \( \alpha_k \geq 0 \) (typically a projected version of the strategies listed in Section 3 and also listed in Table 2). For instance, when using Regińska criterion,

\[
P_k(x_k(\alpha_k)) = R_k(x_k(\alpha_k)) = \|Ax_k(\alpha_k) - b\|\|x_k(\alpha_k)\|^\xi = \|\overline{R}_k y_k(\alpha_k) - Z_{k+1}^T b\|\|y_k(\alpha_k)\|^\xi, \quad \xi > 0,
\]

where the functional \( R_k(x_k(\alpha_k)) \) can be regarded as a projected version of the functional \( R_{\text{reg}} \) appearing in (26), specific of hybrid methods (obtained by exploiting decomposition (12) and the properties of the matrices appearing therein). Problem (27) is formally a sequence of bi-level optimization problems, each consisting of a lower-level optimization problem whose solution \( x_k(\alpha_k) \) is an argument of the higher-level minimization problem; see [74]. Thanks to the particular form of \( F(x, \alpha_k) \) and assuming \( k \ll \min\{m, n\} \), one can derive a closed-form solution for \( x_k(\alpha_k) \), and substitute its expression in \( P_k(x_k(\alpha_k)) \), so that each problem (27) can be essentially regarded as a single-level optimization problem. Even in this favourable case, nothing can be concluded in general regarding the convexity of problem (27), as it will depend on properties of the higher-level functional \( P_k(x_k(\alpha_k)) \).

Note that, as explained in Section 3, when solving (27), one fully runs (till convergence) a parameter choice strategy, with the outcome of selecting a suitable regularization parameter for iteration \( k \) (ie, this is in principle a local choice, good for the \( k \)th projected Tikhonov problem only): therefore, solving (27) to high precision for every \( k \) may be worthless. Moreover, even if only one stopping criterion for the higher-level problem should be set, one needs an additional (and often heuristic) stopping criterion to set \( k \), that is, to guarantee that problem (27) is a good approximation to its full dimensional counterpart, so that one is effectively dealing with two stopping criteria.

Problem (27) can also be regarded as a projected version of the bi-level optimization problem

\[
\min_{\alpha \geq 0} P(x(\alpha)) \quad \text{subject to} \quad x(\alpha) = \arg\min_{x \in \mathbb{R}^n} F(x, \alpha),
\]

arising when applying Tikhonov method (4) (lower-level problem) equipped with a given parameter choice rule (higher-level problem) to compute a regularized solution to (1). As for problem (27), nothing can be stated in general about the convexity of problem (29): a couple of instances (one convex and the other nonconvex) will be considered in Sections 4.2 and 4.3 (respectively); see also [74] and the references therein. Again, thanks to the particular form of \( F(x, \alpha) \), one can derive a closed-form solution for \( x(\alpha) \), and substitute its expression in \( P(x(\alpha)) \), so that problem (29) is essentially a single-level optimization problem. However, in practice and as explained in Section 1, one can obtain \( x(\alpha) \) directly.
only when some factorizations of $A$ (such as the SVD) can be computed: this is not the case for large-scale unstructured problems (1). In these situations, one should resort to an iterative linear solver to approximate the solution $x(\alpha)$ of the lower-level problem in (29), while a nonlinear solver is used to compute an approximation to the higher-level problem in (29). Because of this, problem (29) should still be treated as a bi-level optimization problem. In particular, an inner-outer iterative scheme may be naturally established when solving (29). As partially explained in Section 3, this may be the case for first-regularize-then-project methods (19), which iteratively solve a lower level problem of the form (4) for every value of the Tikhonov regularization parameter $\alpha$ computed within the iterations of $\min_{\alpha} \geq 0 \| P(x(\alpha)) \|$. This approach is equivalent to applying the well-known variable projection method [75] for minimizing problems over two sets of dependent variables, that is, in general, one set of unknowns is explicitly eliminated, and the minimization is computed for the resulting variable projection functional that depends only on the remaining variables. In the case of (29), for the unknown variables $x$ and $\alpha$, $x = x(\alpha)$ can be computed for a given $\alpha$ by iteratively solving a problem of the form (4), so that the minimization is performed over the variable $\alpha$ for the variable projection functional $P(x(\alpha))$. Note that the intermediate values of $\alpha$ so determined are subsequently computed for the full-dimensional problem (4), and two stopping criteria are involved: one for the inner iterations (to be repeatedly applied), and one for the outer iterations.

This section introduces a new efficient class of parameter choice strategies for large-scale problems (4), which can be viewed in the framework of (29), but leverage ideas typical of the hybrid approach to (27). The new strategies simultaneously compute a value for $\alpha_k$, $x_k$ and $x_k(\alpha_k)$, thereby computing a good approximation of the solution of the original problem (29).

The core idea behind the new strategies is to “interlace” the iterations needed to solve the lower-level problem and the higher-level problem in (29). These strategies result in only one iteration cycle, bypassing both the first-regularize-then-project (19) and first-project-then-regularize approaches (20). Namely, each iteration of the new methods consists in performing one step of a Krylov projection method for solving the linear lower-level problem in (29), and one step of a nonlinear (e.g., Newton-like) iterative scheme for solving the nonlinear higher-level problem in (29) (as sketched in Algorithm 1).

**Algorithm 1.** New adaptive bi-level optimization algorithm for problem (29)

1. Choose an initial guess $\alpha_1$.
2. for $k = 1, 2, \ldots$ until a stopping criterion is satisfied do
3.     Compute the Krylov subspace $\mathcal{K}_k$ and project problem (4).
4.     Apply a step of a nonlinear solver to compute $\alpha_{k+1}$ (given $\mathcal{K}_k$ and $\alpha_k$).
5. end for
6. Take $x_k(\alpha_{k+1}) \in \mathcal{K}_k$ as an approximation of the solution of (4).

In this way, the approximation subspace for the solution of (29) is enlarged while a suitable value for the Tikhonov regularization parameter is set. Also, by avoiding nested iteration cycles, only one stopping criterion should be set (this is typically a standard stopping criterion applied to the higher-level problem in (29)). The new strategy, in addition to being conceptually simpler, potentially allows for great computational savings, even when compared with hybrid methods (27). Indeed the latter, for each $k$, still requires the repeated solution of the lower-level problem (27), which may become expensive when $k$ increases; the following sections report more detailed estimates of the involved computational costs. Although a similar approach was proposed in [56] for the discrepancy principle, the present strategies are new in that they are casted in the framework of bi-level optimization problems (29), and can work in connection with every parameter choice rule of the form (29).

In the following, details of Algorithm 1 will be only developed for projection methods that are built upon the Golub-Kahan bidiagonalization (GKB) algorithm. This is because approximations of the functionals (listed in Table 2) associated to the higher-level problem can be obtained by exploiting the connections between GKB and Gaussian quadrature rules (see Section 4.1). However, the same approach could be used in connection with other projection methods, provided that approximations of the higher-level functional $P(x(\alpha))$ are available, can be computed cheaply at each iteration $k$, and converge to $P(x(\alpha))$ as $k$ increases. When using GKB, and when the functional $P(x(\alpha))$ in (29) is associated with the discrepancy principle, it can be proved that the regularized solution computed by the new strategy converges to the solution $x(\alpha)$ of (29) (see Section 4.2); when considering other higher-level functionals, no conclusions about convergence can be drawn at this stage (see Section 4.3).
4.1 Golub-Kahan bidiagonalization (GKB) and its connections to Gaussian Quadrature rules

A pivotal remark that will be exploited in this section is that the symmetric Lanczos algorithm [22, Chapter 6] and the GKB algorithm are closely related. Indeed, multiplying the second equation in (17) from the left by \(A\), and using the first equation in (17), one obtains

\[
AA^TU_k = AV_kB_k = U_kB_kB_k^T + \zeta_{k+1}u_{k+1}^Te_k^TB_k^T = U_kT_k + \zeta_{k+1}\rho_ku_{k+1}^Te_k^T. \tag{30}
\]

Here, the lower bidiagonal matrix \(B_k\) defined in (18) can also be regarded as the Cholesky factor of the symmetric positive definite tridiagonal matrix \(T_k = B_kB_k^T\) obtained after \(k\) iterations of the symmetric Lanczos algorithm applied to \(AA^T\) with initial vector \(b\). Moreover, multiplying the first expression in (17) from the left with \(A^T\), and using again the second equation in (17), one obtains

\[
A^TAV_k = A^TU_kB_k + \zeta_{k+1}A^Tu_{k+1}e_k^T = A^TU_{k+1}B_k = V_{k+1}B_{k+1}^T\bar{B}_k
\]

\[
= V_k\underbrace{B_k\bar{B}_k}_{:=T_k} + \rho_{k+1}\zeta_{k+1}v_{k+1}e_k^T, \tag{31}
\]

so that \(V_k\) can be regarded as the matrix generated by performing \(k\) steps of the symmetric Lanczos algorithm applied to \(A^TA\), with initial vector \(A^Tb\). After computing the QR-factorization \(B_k = Q_k\bar{B}_k\), where \(\bar{B}_k \in \mathbb{R}^{k \times k}\) is lower bidiagonal, one can see that \(\bar{B}_k^T\) is the Cholesky factor of the symmetric positive definite tridiagonal matrix \(\tilde{T}_k = \bar{B}_k^T\bar{B}_k = \bar{B}_k\bar{B}_k^T\).

Now, denote by \(C \in \mathbb{R}^{p \times p}\) a generic symmetric positive semidefinite matrix, having spectral decomposition \(C = W\Lambda W^T\), where \(\Lambda\) is a diagonal matrix whose diagonal elements are the eigenvalues \(0 \leq \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_p\) of \(C\), and \(W\) is the orthonormal matrix whose columns are the normalized eigenvectors of \(C\). The remaining part of this section presents a strategy to compute bounds for general quadratic forms

\[
f(\phi, C, u) = u^T\phi(C)u, \tag{32}
\]

where \(u \in \mathbb{R}^p\) is a given vector and \(\phi\) is a given smooth function on the interval \([0, +\infty)\) of the real line. Using standard definitions and derivations, (32) can be expressed as

\[
f(\phi, C, u) = u^T\phi(C)u = u^TW\phi(\Lambda)W^Tu = \sum_{i=1}^p \phi(\lambda_i)(W^Tu)_i^2 = \int_0^{+\infty} \phi(\lambda)d\omega(\lambda) = : I(\phi). \tag{33}
\]

The last equality comes from considering the sum as a Riemann-Stieltjes integral, where the distribution function \(\omega\) is a non-decreasing step function with jump discontinuities at the eigenvalues \(\lambda_i\). The chain of equalities (33) makes it natural to consider quadrature rules to approximate the quadratic form in (32). Gaussian quadrature rules will be employed for this purpose, and they will be computed applying the symmetric Lanczos algorithm to \(C\) with initial vector \(u\). In Section 4, only quadratic forms of the kind \(b^T\phi(AA^T)b\) and \((A^Tb)^T\phi(A^TA)(A^Tb)\) have to be approximated, so that only the symmetric Lanczos algorithm applied to \(AA^T \in \mathbb{R}^{m \times m}\) with initial vector \(b \in \mathbb{R}^m\), or applied to \(A^TA \in \mathbb{R}^{n \times n}\) with initial vector \(A^Tb \in \mathbb{R}^n\), have to be considered: this is done implicitly by applying the breakdown-free GKB algorithm to \(A \in \mathbb{R}^{m \times n}\) and \(b \in \mathbb{R}^m\) (see assumption (16) and equations (30) and (31)).

Let \(T_k = B_kB_k^T \in \mathbb{R}^{k \times k}\) be the symmetric positive definite tridiagonal matrix appearing in (30), produced after performing \(k \leq \min\{n, m\}\) steps of the Lanczos algorithm applied to the matrix \(AA^T\) with initial vector \(b\). Let \(\{q_i(\lambda)\}_{i=0}^\infty\) be the family of orthonormal polynomials with respect to the inner product induced by the measure \(\omega(\lambda)\) (associated with \(AA^T\) and \(b\)), and let \(T_k = Y_k\Theta_kY_k^T\) be the spectral decomposition of \(T_k\), where \(Y_k\) is the orthonormal matrix whose columns are the normalized eigenvectors of \(T_k\), and \(\Theta_k\) is the diagonal matrix of eigenvalues \(0 < \theta_1 \leq \cdots \leq \theta_k\). It is well-known that
the \(k\)-point Gauss quadrature rule with respect to the measure \(\omega(\lambda)\), defined as

\[
G_k(\phi, AA^T, b) := \sum_{j=1}^{k} \phi(\theta_j) \left| b \right|^2 (e_1^T Y_k e_j)^2 = \left| b \right|^2 e_1^T Y_k \phi(\Theta_k) Y_k^T e_1 = \left| b \right|^2 e_1^T \phi(T_k) e_1 , \tag{34}
\]

approximates (33) with \(C = AA^T\) and \(u = b\). More specifically, the eigenvalues of \(T_k\) are the zeros of the polynomial \(q_k(\lambda)\), as well as the quadrature nodes, and the quadrature weights \(\mu_j\) are given by the rescaled and squared first components of the eigenvectors of \(T_k\) [51]. Analogously, the \(k\)-point Gauss-Radau quadrature rule with one assigned node at the origin and with respect to the measure \(\omega(\lambda)\), approximating (33) with \(C = AA^T\) and \(u = b\), can be obtained by suitably modifying the symmetric Lanczos process to compute a symmetric positive semidefinite matrix \(\bar{T}_k\) of order \(k\) with one prescribed eigenvalue at the origin. This amounts to taking \(\bar{T}_k = \bar{B}_{k-1} \bar{B}_{k-1}^T \in \mathbb{R}^{k \times k}\), where \(\bar{B}_{k-1}\) is the \((k-1) \times k\) version of the matrix \(B_k\) in (18) (or, alternatively, is the matrix obtained by selecting the first \((k-1)\) columns of the Cholesky factor \(B_k\) of \(T_k\)); see [60] for a proof. Eventually, such a quadrature rule reads

\[
R_k(\phi, AA^T, b) := \sum_{j=1}^{k} \phi(\bar{\theta}_j) \left| b \right|^2 (e_1^T Y_k e_j)^2 = \left| b \right|^2 e_1^T Y_k \phi(\bar{\Theta}_k) Y_k^T e_1 = \left| b \right|^2 e_1^T \phi(\bar{T}_k) e_1 , \tag{35}
\]

where \(\bar{T}_k = Y_k \bar{\Theta}_k Y_k^T\) is the spectral decomposition of \(\bar{T}_k\), with \(Y_k\) orthonormal and \(\bar{\Theta}_k = \text{diag}(\bar{\theta}_1, \ldots, \bar{\theta}_k), \ 0 = \bar{\theta}_1 < \bar{\theta}_2 \leq \ldots \leq \bar{\theta}_k\).

Now let \(\bar{T}_k = \bar{B}_k \bar{B}_k^T \in \mathbb{R}^{k \times k}\) be the symmetric positive definite tridiagonal matrix appearing in (31), produced after performing \(k \leq \min \{ n, m \}\) steps of the Lanczos algorithm applied to \(A^TA\) with initial vector \(A^Tb\). Similarly to the previous derivations, the \(k\)-point Gauss quadrature rule with respect to the measure \(\omega(\lambda)\) (associated with \(A^TA\) and \(A^Tb\)), defined as

\[
G_k(\phi, A^TA, A^Tb) := \sum_{j=1}^{k} \phi(\bar{\theta}_j) ||A^Tb||^2 (e_1^T Y_k e_j)^2 = ||A^Tb||^2 e_1^T \phi(\bar{T}_k) e_1 , \tag{36}
\]

approximates (33) with \(C = A^TA\) and \(u = A^Tb\). Here, using notations analogous to the previous ones, \(\bar{T}_k = \bar{Y}_k \bar{\Theta}_k \bar{Y}_k^T = Y_k \text{diag}(\bar{\theta}_1, \ldots, \bar{\theta}_k) Y_k^T\) is the spectral decomposition of the matrix \(\bar{T}_k\). Finally, the matrix \(\tilde{T}_k := \bar{B}_{k-1} \bar{B}_{k-1}^T \in \mathbb{R}^{k \times k}\), where \(\bar{B}_{k-1}\) is the matrix constructed by selecting the first \(k-1\) columns of the Cholesky factor \(B_k\) of \(\bar{T}_k\), is symmetric positive semidefinite with one prescribed eigenvalue at the origin. Therefore, the \(k\)-point Gauss-Radau quadrature rule with respect to the measure \(\omega(\lambda)\) (associated with \(A^TA\) and \(A^Tb\)), with one node assigned at the origin, can be expressed as

\[
R_k(\phi, A^TA, A^Tb) := \sum_{j=1}^{k} \phi(\tilde{\theta}_j) ||A^Tb||^2 (e_1^T \bar{Y}_k e_j)^2 = ||A^Tb||^2 e_1^T \phi(\tilde{T}_k) e_1 , \tag{37}
\]

where \(\tilde{T}_k = Y_k \tilde{\Theta}_k \bar{Y}_k = Y_k \text{diag}(\tilde{\theta}_1, \ldots, \tilde{\theta}_k) \bar{Y}_k^T\) is the spectral decomposition of the matrix \(\tilde{T}_k\).

Assuming that \(\phi\) is a \(2k\)-times differentiable function, the quadrature errors \(E_{G_k}(\phi) = I(\phi) - Q_k(\phi)\) associated with \(k\)-point Gauss and Gauss-Radau quadrature rules (with \(Q_k(\phi) = G_k(\phi)\) and \(Q_k(\phi) = R_k(\phi)\), respectively, and where the dependence on the matrices \(AA^T, A^TA\), and the vectors \(b, A^Tb\), has been removed in the interest of generality), are given by

\[
E_{G_k}(\phi) = \frac{\phi^{(2k)}(C_{\mathcal{G}})}{(2k)!} \int_0^{+\infty} \prod_{i=1}^{k} (t - \xi_i)^2 d\omega(t) , \tag{38}
\]
A rule based on the discrepancy principle and a new modified Newton method

Some preliminary derivations should be performed to see how Tikhonov regularization (4) equipped with the discrepancy principle (21) fits into the framework of problem (29). Consider the quadratic form

$$P(x(a)) = b^T \psi_{DP}(AA^T, \alpha) b,$$

where

$$\psi_{DP}(t, \alpha) = \alpha - \alpha \overline{\varepsilon}^2 + 2t \log((t + \alpha)^{-1}) - t^2(\alpha + t)^{-1},$$

is a function defined for \( t \geq 0, \alpha > 0 \) (see also (33)), and \( \overline{\varepsilon} = \varepsilon/\|b\| = \tau\|e\|/\|b\| \) in (1) (see also (21) and Table 2). The first and second derivatives with respect to \( \alpha \) of the function \( \psi_{DP}(t, \alpha) \) in (40) read

$$\phi_{DP}(t, \alpha) := \partial_\alpha \psi_{DP}(t, \alpha) = \alpha^2(t + \alpha)^{-2} - \overline{\varepsilon}^2$$

and

$$\partial^2_\alpha \psi_{DP}(t, \alpha) = 2\alpha(t + \alpha)^{-3},$$

respectively. Since \( \partial^2_\alpha \psi_{DP}(t, \alpha) \geq 0 \) for \( \alpha > 0 \), then \( \partial^2_\alpha b^T \psi_{DP}(AA^T, \alpha) b \geq 0 \) for \( \alpha > 0 \) (i.e., \( b^T \psi_{DP}(AA^T, \alpha) b \) is convex as a function of \( \alpha \) for \( \alpha > 0 \)). Therefore, solving (29) amounts to solving the nonlinear equation (see [72] for complete derivations)

$$0 = b^T \phi_{DP}(AA^T, \alpha) b = a^2 b^T (AA^T + aI)^{-2} b - \varepsilon^2 = \|b - Ax(a)\|^2 - \varepsilon^2$$

with respect to \( \alpha \) (which is basically (21) specified for Tikhonov method). Since the continuous function \( b^T \phi_{DP}(AA^T, \alpha) b \) is increasing in \( \alpha \), there exists a unique zero \( \alpha^* \) in \( (0, \infty) \) provided that

$$\lim_{\alpha \to 0} b^T \phi_{DP}(AA^T, \alpha) b = -\varepsilon^2 < 0 \quad \text{and} \quad \lim_{\alpha \to \infty} b^T \phi_{DP}(AA^T, \alpha) b = \|b\|^2 - \varepsilon^2 > 0,$$

where the last inequality obviously holds if \( \varepsilon^2 < \|b\|^2 \) (this is a reasonable bound for the amount of noise in the data, which will be assumed in the following). Equation (42) agrees with the standard discrepancy principle formulation (21) and, in theory, one can easily apply a zero-finder (e.g., Newton method) to compute \( \alpha^* \). Since \( \phi_{DP}(t, \alpha) \) is not convex for \( \alpha > 0 \), Newton method is not guaranteed to globally converge. As proposed in [55], the simple change of variable \( \beta = 1/\alpha \) is performed in (42), so that

$$\phi_{DP}(t, \beta) := (\beta t + 1)^{-2} - \overline{\varepsilon}^2$$

and

$$\hat{\phi}_{DP}(t, AA^T, \beta) := b^T \phi_{DP}(AA^T, \beta) b - \varepsilon^2$$

are decreasing and convex for \( \beta > 0 \), and a unique zero \( \beta^* \) exists if conditions analogous to (43) are satisfied. Newton method applied to solve the nonlinear equation (with respect to \( \beta \))

$$0 = \hat{\phi}_{DP}(\beta^*, AA^T, \beta) = b^T (\beta AA^T + I)^{-2} b - \varepsilon^2,$$
globally converges, and can be easily implemented if the SVD of $A$ is available. Since this is not the case in general for large-scale problems (as remarked earlier in this section and in Section 1), an alternative new solution approach for (45) is derived.

The following result proves the convergence of a specific modification of the classical Newton zero finder, which can be used in a general setting whenever dealing with a sequence of functions $\{\ell_k\}_{k \geq 1}$ satisfying certain assumptions. This method will be later applied to solve (45).

**Theorem 1.** Let $f : (0, +\infty) \to \mathbb{R}$ be a strictly decreasing, convex, differentiable function such that $\lim_{x \to +\infty} f(x) < 0$. Let $\{\ell_k\}_{k \geq 1} : (0, +\infty) \to \mathbb{R}$, be a sequence of strictly decreasing, convex, differentiable, increasing lower bounds for $f$, that is,

$$\ell_k(x) \leq f(x) \quad \text{for all } k \geq 1, x \in (0, +\infty),$$

such that $\lim_{x \to 0} \ell_k(x) > 0$ for all $k \geq 1$, and $\lim_{x \to +\infty} \ell_k(x) = f(x)$ for all $x \in (0, +\infty)$. Then, given $x_1$ such that $\ell_1(x_1) \geq 0$, the sequence $\{x_k\}_{k \geq 1}$ obtained from the recursion

$$x_{k+1} = x_k - \frac{\ell_k(x_k)}{\ell'_k(x_k)},$$

monotonically converges to the root $x^*$ of $f$ from the left.

**Proof.** The assumptions assure that the functions $f$ and $\ell_k$, $k \geq 1$, have exactly one zero in $(0, +\infty)$. Given $x_k$, $k \geq 1$, define the function

$$t_k(x) = \ell_k(x_k) + \ell'_k(x_k)(x - x_k),$$

that is, the tangent line to the graph of $\ell_k$ at $(x_k, \ell_k(x_k))$. Relation (47) is established by imposing $t_k(x_{k+1}) = 0$ and, together with the convexity of $\ell_k$ and (46), leads to

$$0 = t_k(x_{k+1}) \leq \ell_k(x_{k+1}) \leq \ell_k(x_k) \leq f(x_{k+1}).$$

Replacing $k$ by $k - 1$ in the above relation implies $\ell_k(x_k) \geq 0$ which, together with $\ell'_k(x_k) < 0$ and (47), leads to $x_k \leq x_{k+1}$. Moreover, since $f$ is decreasing and $f(x_{k+1}) \geq 0$, $x_{k+1} \leq x^*$. Therefore the sequence $\{x_k\}_{k \geq 1}$ is monotonically increasing and bounded above by $x^*$. Taking the limit for $k \to \infty$ in (48) implies $\ell_{k+1}(x_{k+1}) \to f(x_{k+1})$, so that $x_{k+1}$ converges to $x^*$ thanks to the convergence of Newton method.

**Remark 1.** Given a sequence of functions $\{\ell_k\}_{k \geq 1}$, the $k$th iteration of the modified Newton method (47) consists in performing only one (standard) Newton iteration on the $k$th function $\ell_k$. Figure 1 gives a geometrical illustration of recursion (47).
Remark 2. Theorem 1 still holds if assumption $\mathcal{L}_1(x_1) \geq 0$ is removed and $f, \{\mathcal{L}_k\}_{k \geq 1} : \mathbb{R} \to \mathbb{R}$ (ie, considering functions defined on the whole real line). Indeed, in this setting

$$0 = t_1(x_2) \leq \mathcal{L}_1(x_2),$$

so that the reasoning in the proof of Theorem 1 can be applied starting from $x_2$.

Turning now to the discrepancy principle (45), since the matrix functional $\hat{f}_{DP}$ is strictly decreasing, convex, and differentiable with respect to $\beta$, and $\frac{\partial^2}{\partial \beta^2} \hat{f}_{DP}(t, \beta) < 0$ for all $k \geq 1$, $t \geq 0$, $\beta > 0$, lower bounds for $\hat{f}_{DP}$ are obtained by applying the Gauss quadrature rule, leading to

$$G_k(\hat{f}_{DP}, A A^T, b, \beta) = \|b\|^2 e_1^T (\beta B_k B_k^T + I)^{-2} e_1 - \epsilon^2, \quad k \geq 1;$$

(49)

see Section 4.1 and equation (38). These bounds are increasing (see [76], Theorem 2.1), so that, under assumption (16), and using the shorthand notations $G_k(\beta)$ and $\hat{f}_{DP}$ for $G_k(\hat{f}_{DP}, A A^T, b, \beta)$ and $\hat{f}_{DP}(\hat{f}_{DP}, A A^T, b, \beta)$, respectively,

$$G_1(\beta) \leq G_2(\beta) \leq \cdots \leq G_{\min\{n,m\}−1}(\beta) \leq G_{\min\{n,m\}}(\beta) = \hat{f}_{DP}(\beta).$$

The functions $G_k(\beta)$, $1 \leq k \leq p$, are strictly decreasing, convex, and differentiable with respect to $\beta$, $\lim_{\beta \to 0} G_k(\beta) = \|b\|^2 - \epsilon^2 > 0$ (reasonable bound for the amount of noise, see (43)), and $\lim_{\beta \to +\infty} G_k(\beta) = -\epsilon^2 < 0$. The same limits hold for $\hat{f}_{DP}$. The above derivations assure that the assumptions of Theorem 1 are satisfied, so that the following result holds.

Corollary 1. Let $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ be as in (1), and let $\hat{f}_{DP}$ be defined as in (44); consider $\hat{f}_{DP}$ in (44) and $\{\beta_k\}$ in (49) as functions of $\beta > 0$. Let $\beta_1 > 0$ be such that $G_1(\beta_1) = 0$. Then the sequence $\{\beta_k\}_{k \geq 1}$ obtained from the recursion

$$\beta_{k+1} = \beta_k - \frac{G_k(\beta_k)}{G_k'(\beta_k)}$$

(50)

monotonically converges to the root $\beta^*$ of (45) from the left.

Remark 3. Relation (50) is formally similar to (47). Notably, since $G_k(\beta) = \hat{f}_{DP}(\beta)$ for $k \geq \min\{n, m\}$, relation (50) reduces to (standard) Newton method when $k \geq \min\{n, m\}$. However, this never happens in practice, because the bounds $G_k(\beta)$ are observed to quickly approach $\hat{f}_{DP}(\beta)$ and the convergence of (standard) Newton method is quadratic; see also Section 5.

The computational cost of the new algorithm depends on the update rule (50) for the value of the regularization parameter for problem (4). At iteration $k$, in (49) is needed to compute $a_{k+1}$ (recall that $a_{k+1} = 1/\beta_{k+1}$): to achieve this, $k$ iterations of the GKB algorithm should be performed (see Section 4.1 and equation (34)) to build the required Krylov subspace $\mathcal{K}_k(A A^T, b)$. Assuming that $A$ is full, the computational cost of this task is dominated by $O(2knm)$ floating point operations, since two matrix vector products (one with $A$ and one with $A^T$) are computed at each GKB iteration. Computing $G_k(\beta_k)$ and $G_k'(\beta_k)$ in (50) requires the solution of two linear systems with coefficient matrix $(\beta_k B_k B_k^T + I)$ (see also [55]): exploiting the tridiagonal structure of the involved matrices, this amounts to $O(k)$ floating point operations and is therefore negligible. To compute an approximate solution $\hat{x}_k(a_{k+1})$ for problem (4), one needs to consider the space $\mathcal{K}_k(A^T A, A^T b)$, and project problem (4) onto it, that is, solve problem (20) (with the Tikhonov regularization parameter equal to $1/\beta_{k+1}$). This can be done inexpensively once the bound $k$ is computed, since $k$ iterations of the GKB algorithm generate both spaces $\mathcal{K}_k(A^T A, A^T b)$ and $\mathcal{K}_k(A A^T, b)$ (see Section 4.1), and only the order-$k$ least squares problem in (20) needs to be solved: to compute $\hat{y}_k(\beta_{k+1}) := y_k(1/a_{k+1}) \in \mathbb{R}^k$ ($O(k)$ floating point operations) and to then form $\hat{x}_k(\beta_{k+1}) = x_k(1/a_{k+1})$ ($O(kn)$ floating point operations). The cost of these computations is negligible if $k \ll \min\{m, n\}$; moreover this can be performed only once a stopping criterion for the iterations is satisfied. However, $\hat{y}_k(\beta_{k+1}) \in \mathbb{R}^k$ may be needed to devise a suitable stopping criterion, and therefore should be computed at a negligible additional cost at each iteration. Indeed, it should be remarked that the discrepancy functional $\hat{f}_{DP}$ (44) associated with the approximate solution $\hat{x}_k(\beta_{k+1})$ satisfies

$$\|b\|^2 e_1^T (\beta_{k+1} B_k B_k^T + I)^{-2} e_1 - \epsilon^2 = \mathcal{R}_{k+1}(\hat{f}_{DP}, A A^T, b, \beta_{k+1}), \quad k \geq 1;$$

(51)
see (36) (full derivations are provided in [72]). Since \( \psi_{(k+1)}^{(t)}(t, \beta) > 0 \) for all \( k \geq 1, t \geq 0, \beta > 0 \), \( R_{k+1} \) is an upper bound for \( \hat{f}_{DP} \) (both considered as functions of \( \beta \)); see (39). Note that (51) is equivalent to (22), that is, the projected version of the discrepancy principle (21) for (direct) Tikhonov regularization.

If, for solving the \( k \)th instance of problem (27), one had to adopt a well-established hybrid regularization method based on GKB (for the lower-level problem) and Newton method (for the higher-level problem), \( O(2kmn) \) floating points operations should still be needed to compute the partial GKB factorizations (17) (for a full \( A \)) and, at each iteration of the (standard) Newton method, two linear systems with tridiagonal coefficient matrix \((B_{k+1} + I)\) of dimension \( k + 1 \) should be inverted (see again [55] for more details). If \( k_N \) Newton iterations have to be performed, the latter amounts to \( O(k_N(k + 1)) \) floating point operations: although this cost is negligible when compared to the cost associated to GKB, it can become significant (and much higher than the \( O(k) \) operations required by the new strategy (50)) when \( k \) increases and if many Newton iterations \( k_N \) have to be performed.

Summarizing, the \( k \)th iteration of the new adaptive strategy to solve problem (29) when \( P(x(\alpha)) \) is associated with the discrepancy principle consists in applying the new modified Newton zero finder (50) to (45), which involves computing lower bounds \( \hat{f}_{DP} \). The discrepancy functional \( ||b - AX_k(\beta_{k+1})||^2 - \epsilon^2 = ||B_k\hat{x}_k(\beta_{k+1}) - ||b||e||^2 - \epsilon^2 \) for the intermediate approximate solutions \( \hat{x}_k(\beta_{k+1}) = x_k(1/\alpha_{k+1}) \) of problem (4) lies on upper bounds \( R_{k+1} \) for \( \hat{f}_{DP} \). See also Algorithm 2. An illustration of the behavior of the bounds for the function \( \hat{f}_{DP} \) (44) is given in Section 5 for an image deblurring test problem.

Since the modified Newton method (50) can essentially be regarded as a Newton-like update formula applied to a sequence of iteration-dependent converging functions, standard stopping criteria for Newton method can be adapted to this setting to determine both a value of the regularization parameter \( \beta_{k+1} \) and the dimension of the approximation subspace for \( \hat{x}_k(\beta_{k+1}) \). Typically, the updates (50) should stop when the space \( K_k(A^TA, A^Tb) \) is large enough to contain a suitable approximation to the solution of (4) and when a value of the regularization parameter suitable for the full dimensional problem (4) has been computed: these requirements are interrelated and, as mentioned in Section 3, they are also desirable for hybrid methods; see [31,61,72].

It is natural to stop the updates (50) as soon as

\[
\hat{f}_{DP}(\beta_{k+1}) = ||b - AX_k(\beta_{k+1})||^2 - \epsilon^2 \leq \theta \epsilon^2, \quad \text{for a given tolerance } \theta > 0.
\]  \( \tag{52} \)

However, computing \( \hat{x}(\beta_{k+1}) \) would require solving the full-dimensional problem (4) with \( a = 1/\beta_{k+1} \), which may be prohibitively expensive for large-scale problems. Therefore, estimates for the function on the left of (52) should be considered. By taking the upper bound (51), one can replace (52) by

\[
R_{k+1}(\hat{f}_{DP}, AA^T, b, \beta_{k+1}) \leq \theta \epsilon^2. \tag{53} \]

Satisfying (53) implies satisfying (52). Alternatively (and using reduced notations), an estimate of \( ||b - AX_k(\beta_{k+1})||^2 \) is obtained by averaging its upper (51) and lower (49) bounds evaluated at \( \beta_{k+1} \), and (52) can be replaced by

\[
\frac{1}{2}(R_{k+1}(\beta_{k+1}) + G_k(\beta_{k+1})) \leq \theta \epsilon^2. \tag{54} \]

**Algorithm 2.** Algorithm 1 specified for GKB and the discrepancy principle

1: Choose an initial guess \( \beta_1 \) such that \( G_1(\beta_1) \geq 0, G_k \) as defined in (49).
2: \( k = 1, 2, \ldots \) until (53) or (54) are satisfied do
3: Apply GKB to update the \( k \)th factorizations of the form (17)
4: Compute \( \beta_{k+1} \) by (50)
5: Compute \( \hat{y}_k(\beta_{k+1}) = y(1/\alpha_{k+1}) \) by (20)
6: end for
7: Take \( x_k(\alpha_{k+1}) = V_ky_k(\alpha_{k+1}) \in K_k(A^TA, A^Tb) \) as solution of (27), approximating the solution of (29).
4.3 Extensions to other parameter choice rules

As already mentioned, the new class of algorithms can be potentially employed to approximate a solution to regularization methods that can be formulated as bilevel optimization problems of the form (29), which includes all the parameter choice rules described in Section 3 used together with Tikhonov regularization (4) (see also Table 2). This section unfolds, in a quite heuristic way, specific derivations for the case of Regińska criterion (26), (28).

As for the discrepancy principle in Section 4.2, the kth iteration of the new approach (starting, in general, from an iteration $k \geq k^*$) consists in performing one minimization step by applying a modified Newton method to the higher level functional $P(x(a))$ in (29), while expanding the approximation subspace for the solution of the lower level functional $F(x, a)$ in (29). Since the evaluation of $P(x(a))$ in (29) at each iteration $k$ is computationally prohibitive, one should employ a sequence of functionals $P_k(a)$, which have a local minimum converging to a local minimum of $P(x(a))$ but, in general, may not explicitly depend on the current approximate solution $x_k(a)$. In the case of the Regińska criterion, the functionals $P_k(a)$ are obtained by approximating $P(x(a))$ via Gaussian quadrature rules (see Section 4.1). For this reason, at the kth iteration of the new solver, the Krylov subspaces $\mathcal{K}_k(A^TA, A^Tb)$ and $\mathcal{K}_k(AA^T, b)$ are built, and the new update formula for the Tikhonov regularization parameter reads

$$a_{k+1} = a_k - \frac{\partial_y P_k(a_k)}{\partial^2_y P_k(a_k)}, \quad \text{for} \quad k \geq k^* .$$

(55)

Although the above relation is formally similar to (47), where $L_k = \partial_y P_k$ and a zero finder is applied to $\partial_y P(x(a)) = 0$, its application may not be as straightforward as in the case considered in Section 4.2.

The functional associated with the Regińska criterion (26) can be expressed in terms of quadratic forms as

$$P(x(a)) = (a^2 b^T \phi_R(AA^T, a)b)^{1/2}((A^Tb)^T \phi_R(A^TA, a)A^Tb)^{\xi/2}, \quad \text{where} \quad \phi_R(t, a) = (a + t)^{-2}, \quad \text{and} \quad \xi > 0 \quad \text{is fixed.}$$

(56)

Since $\partial^2_y (2k-1) \phi_R(t, a) < 0$ for $k \geq 1, t \geq 0, a > 0$, Gauss-Radau quadrature rules can be used to compute a sequence of increasingly sharper upper bounds for $a^2 b^T \phi_R(AA^T, a)b$ and $(A^Tb)^T \phi_R(A^TA, a)A^Tb$. Namely, taking

$$\tilde{R}_k(a) := \langle R_{k+1}(\phi_R, AA^T, b, a) \rangle^{1/2}(R_k(\phi_R, A^TA, A^Tb, a))^{\xi/2} = \|A^Tb\|^2 \|b\|(\epsilon_1^T \phi_R(B_k, a) e_1)^{1/2}(\epsilon_1^T \phi_R(\hat{B}_k, a) e_1)^{\xi/2}$$

(57)

and knowing that both $R_{k+1}(\phi_R, AA^T, b, a)$ and $R_k(\phi_R, A^TA, A^Tb, a)$ decrease with increasing $k \geq 2$, one gets $\tilde{R}_{k+1}(a) \leq \tilde{R}_k(a)$ (see Section 4.2 and equations (35), (37)). Similarly, since $\epsilon_1^{2k} \phi_R(t, a) > 0$ for all $k \geq 1, t \geq 0, a > 0$, Gauss quadrature rules can be used to compute lower bounds for $P(x(a))$ in (56). Namely, taking

$$\tilde{C}_k(a) := \langle C_{k+1}(\phi_R, AA^T, b, a) \rangle^{1/2}(C_k(\phi_R, A^TA, A^Tb, a))^{\xi/2} = \|A^Tb\|^2 \|b\|(\epsilon_1^T \phi_R(B_k, a) e_1)^{1/2}(\epsilon_1^T \phi_R(\hat{B}_k, a) e_1)^{\xi/2}$$

(58)

and knowing that

$$C_{k+1}(\phi_R, AA^T, b, a) \geq C_k(\phi_R, AA^T, b, a) \quad \text{and} \quad C_{k+1}(\phi_R, A^TA, A^Tb, a) \geq C_k(\phi_R, A^TA, A^Tb, a) ,$$

one gets $\tilde{C}_{k+1}(a) \geq \tilde{C}_k(a)$ (see equations (34) and (36)). An illustration of the behavior of the upper (57) and lower (58) bounds for Regińska’s functional (56) is given in Section 5 for an image deblurring test problem; this example is quite representative of the typical behavior found in other test problems. Because of these features, the choice $P_k(a) = \tilde{R}_k(a)$ seems more convenient and will be considered from now on, while $\tilde{C}_k(a)$ will be used to devise suitable stopping criteria for the new solver. Moreover, since $P_k(a)$ is defined for $k \geq 2$, it is natural to select $k^* = 2$ in (55). Note also that both the kth upper (57) and lower (58) bounds for Regińska’s functional (56) are different from its projection $R_k(x_k(a_k))$ expressed in (28): indeed, using the notation of Gaussian quadrature rules, $R_k(x_k(a_k)) = (R_{k+1}(\phi_R, AA^T, b, a))^{1/2}(C_k(\phi_R, A^TA, A^Tb, a))^{\xi/2}$; see [72] for analogous derivations.

As in Section 4.2, the kth step of the update rule (55) requires that $k$ iterations of the GKB algorithm should be performed, so that, if $A$ is full, this task costs $O(2k^3n)$ floating point operations. Updating the regularization parameter $a_{k+1}$ using equation (55), with $P_k(a_k) = \tilde{R}_k(a_k)$ in (57), requires the solution of four linear systems with tridiagonal coefficient
matrices (two with \( \overline{B}_k \overline{B}_k^T + \alpha_k I \) and two with \( \overline{B}_{k-1} \overline{B}_{k-1}^T + \alpha_k I \)), amounting to \( O(k) \) floating point operations. The same reasoning applies to the estimation of the cost for the computation of \( \tilde{R}_k(\alpha_{k+1}) \) in (57) and in \( \tilde{G}_k(\alpha_{k+1}) \) in (58), which is still \( O(k) \) floating point operations: this quantities can be useful to devise a stopping criterion for the new strategy. Computing the approximate solution \( x_k(\alpha_{k+1}) = V_k y_k(\alpha_{k+1}) \) amounts to \( O(kn) \) floating point operations, although this can be done once the stopping criterion is satisfied. If one had to solve the \( k \)th instance of problem (27) adopting a well-established hybrid regularization method based on GKB, the computational cost would depend on the optimization method adopted for the higher-level problem: for \( k \ll \min \{ m, n \} \), the \( O(2kmn) \) floating points operations needed to compute the partial GKB factorizations (17) would dominate the computational cost; however, \( k_M \) iterations of an optimization routine to minimize \( P_k(x_k(\alpha_i)) \) based on golden section search would require \( k_M + 1 \) evaluations of the function (28), amounting to \( O((k_M + 1)k) \) floating point operations. Although this cost is negligible when compared to the cost associated to GKB, it can become significant (and much higher than the \( O(k) \) operations required by the new strategy (55)) when \( k \) and \( k_M \) increase.

Similarly to Section 4.2, traditional stopping criteria for (standard) Newton method applied to \( \partial_\alpha P(x(\alpha)) = 0 \) can be adapted to the modified Newton method (55), in such a way that expensive computation of \( x(\alpha_{k+1}) \) is avoided. Moreover, when \( \{ P_k(\alpha) \}_k \) are approximations of \( P(x(\alpha)) \) of improving quality (ie, when \( P_k(\alpha) \) becomes closer to \( P(x(\alpha)) \) as \( k \) increases), one can also jointly monitor the convergence of \( P_k(x_{k+1}(\alpha)) \) to \( P(x(\alpha_{k+1})) \) (ie, the convergence of the approximate functionals to the full-dimensional one at the current approximation of \( \alpha \)) together with the convergence of \( \alpha_{k+1} \) to a zero of \( \partial_\alpha P_k(\alpha) \). This is done by stopping, for instance, as soon as

\[
\frac{|P_k(x_{k+1}(\alpha)) - P(x(\alpha_{k+1}))|}{|P(x(\alpha_{k+1}))|} + \frac{|\partial_\alpha P_k(x_{k+1})(\alpha)|}{|P_k(x_{k+1})|} \lesssim \left| \frac{\tilde{R}_k(x_{k+1})(\alpha) - \tilde{G}_k(x_{k+1})(\alpha)}{|\tilde{R}_k(x_{k+1}) + \tilde{G}_k(x_{k+1})|} \right| < \theta, \quad \text{for a given tolerance } \theta > 0.
\]

(59)

The above approximation is specific for the case of the new modified Newton method applied to the upper bounds \( P_k(\alpha) = \tilde{R}_k(\alpha) \) for the Regińska functional (57), with \( P(x(\alpha_{k+1})) \approx 1/2(\tilde{R}_k(x_{k+1}) + \tilde{G}_k(x_{k+1})) \), that is, the average of the upper and lower approximations of \( P(x(\alpha)) \) in \( \alpha = \alpha_{k+1} \).

5 | NUMERICAL EXPERIMENTS

This section investigates the performance of the new class of bi-level optimization algorithms on two large-scale test problems modeling imaging applications. All the experiments are performed running MATLAB R2017a and using some of the functionalities available within the MATLAB toolbox IR Tools [77]. No so-called inverse crime will be committed in that, as detailed in each example below, no perfect agreement between the model and the data in the discretized inverse problem will be assumed. The behavior of different regularized solutions to problem (1) is mainly assessed by evaluating the quality of the reconstructed solutions. If \( x_{\text{true}} \) is available, the relative restoration error (RRE) (8) is computed, as a function of the Tikhonov regularization parameter \( \alpha \) and the dimension \( k \) of the Krylov subspaces for the approximation of the solution (to highlight this, the notation RRE(\( \alpha, k \)) will be used).

5.1 | Example 1

This example models an image deblurring test problem, involving the exact image \( X_{\text{true}} \), of size 512×512 pixels, appearing in false-color in Figure 2, frame (a); \( X_{\text{true}} \) represents a galaxy and it is publicly available.2 The image \( X_{\text{true}} \) is corrupted by applying a spatially invariant Gaussian blur and Gaussian white noise of level \( \| e \| / \| b \| = 0.05 \); the resulting blurred and noisy image is shown in Figure 2, frame (b). In this setting, given a point-spread function (PSF) analytically defined as a 2D Gaussian function (which describes how a single pixel of the test image is deformed), a blurring process is modeled as a 2D convolution of the PSF and the exact discrete finite image \( X_{\text{true}} \). Reflexive boundary conditions, which prescribe pixel values outside of \( X_{\text{true}} \) for the convolution operation to be well-defined, are assigned. To avoid inverse crime, no perfect agreement between the boundary conditions in the forward model and the operator used to generate the example.

---

1 MATLAB implementations of our methods, which should be used jointly with IR Tools, are available at https://github.com/silviagazzola.

2 It appears among the ones in the NRAO repository, obtained by processing data acquired by a ground-based radio-telescope; see https://public.nrao.edu/gallery/topic/galaxies/
A 2D image restoration problem can be rewritten as a linear system (1), where the square matrix $A$ incorporates the convolution process together with the boundary conditions, the unknown $x$ should approximate the sharp image $X_{true}$ (or, equivalently, the vector $x_{true}$ obtained by stacking the columns of $X_{true}$), and the vector $b$ is obtained by stacking the columns of the 2D blurred and noisy image; see [3] for more details on image deblurring problems. For this test problem, the coefficient matrix $A$ has order $m = n = 262144$ and it is not explicitly stored; the action of $A$ on any vector is computed efficiently as described in [77].

Looking at Figure 3 it is immediate to see that both the upper and the lower bounds for the discrepancy functional tend to overlap quite quickly and, in particular, the upper and lower bounds at iteration $k = 100$ essentially coincide for all the considered values of the regularization parameter $\beta = 1/\alpha$ (implying that these bounds already converged to the full dimensional functional $\partial_\alpha P(x(\alpha))$). Looking at Figure 4, it can be observed that both the upper and the lower bounds for the Regińska functional tend to approach one another more slowly than the bounds for the discrepancy functional for every considered value of the regularization parameter; however, they quickly overlap in the rightmost section of the flat region, where the minimum is located. It is important to note that the Regińska functional displays its characteristic flatness around the minimum, which make the minimization problem hard in the full dimensional case (see, eg, [73] for additional examples). This drawback is partially alleviated when considering the upper
bounds, which are used in this example to compute the regularization parameter at each iteration via the new adaptive algorithms.

Figure 5 displays: the so-called discrepancy surface, that is, a surface whose points lie on \((a, k, \|b - Ax_k(a)\|)\); the so-called Regińska surface, that is, a surface whose points lie on \((a, k, \|b - Ax_k(a)\|)\); the so-called error surface, that is, a surface whose points lie on \((a, k, RRE(a, k))\). Note that, for every fixed \(k\), \(\|b - Ax_k(a)\|^2\) corresponds to the upper bound \(R_k(a) + \epsilon^2\) for the discrepancy \(\|b - Ax(a)\|^2\) (see Section 4.2); however, there is not such correspondence between the projected Regińska functional \(\|b - Ax_k(a)\|\) and the computed bounds for the Regińska functional displayed in Figure 4 (recall the discussion in Section 4.3). Special markers are used to highlight the points on the different surfaces that are selected by the hybrid and new algorithms used together with the discrepancy principle or Regińska criterion, and the points that deliver a minimal error (ie, an ideally optimal parameter choice rule). Figure 5 gives a good overview on how the discrepancy, the Regińska, and the error functionals evolve across different values of the Tikhonov regularization parameter and across different iterations, and how they relate to each other. For instance, looking at frames (c) and (f), one can see that the optimal value of \(a\) (ie, the \(a\) minimizing \(RRE(a, k)\)) stabilizes for increasing \(k\); correspondingly, looking at frames (a),(b),(d) and (e), one can see that the value of \(a\) satisfying the discrepancy principle and the value of \(a\) minimizing the Regińska functional stabilize for increasing \(k\). Moreover, as \(k\) increases, the parameters selected using the discrepancy principle are essentially optimal (see frames (a) and (d)). It is important to remark that the computation of the discrepancy, the Regińska and the error surfaces merely has illustrative purposes: in practice, the \(k\)th iteration of the new algorithm requires evaluating the approximated higher-level functionals for only one value of \(a\) once (or twice if a stopping criterion of the form (53) or (54) is implemented, and to compute the approximate solution at the final iteration).

Figure 6 evaluates the performance of the new adaptive bi-level algorithm used in connection with the discrepancy principle (frames (a) and (b)) and Regińska criterion (frames (c) and (d)), by displaying comparisons with other methods. More precisely, the value of the relative restoration error \(RRE(\hat{a}_{k+1}, k)\) and the value of the regularization parameter \(a_{k+1}\) computed by the new method at each iteration \(k\) are compared against the values obtained by running (i) a hybrid method with optimal parameter choice at each iteration (so that \(RRE(a, k)\) is minimized with respect to \(a\) for each \(k\)); (ii) a hybrid method selecting the parameter according to the discrepancy principle at each iteration (in frames (a) and (b)) and the Regińska criterion at each iteration (in frames (c) and (d)); (iii) LSQR used as purely iterative regularization method (ie, without imposing additional Tikhonov regularization within the iterations).

For this test problem, the initial value \(\beta_1 = 1/a_1 = 10^{-10}\) is set for the new algorithm used together with the discrepancy principle: this value is small enough to satisfy \(G_1(\beta_1) \geq 0\), so that the assumptions of Corollary 1 hold and convergence to the solution of (29) is guaranteed. On the other hand, the initial value \(a_1 = 10^{-3}\) is set for the new algorithm used together with Regińska criterion. Looking at Figure 6 it is evident that the new algorithm used together with either the discrepancy principle or Regińska criterion delivers results that are very similar to the ones obtained when applying their hybrid counterparts (20) (note that the discrepancy principle can be applied in the hybrid framework starting from iteration 7, because equation (22) does not have any zero till that iteration; see, eg, [55]). For this test problem, the performances of both the new algorithm and the hybrid method, used in combination with the discrepancy principle, are very similar to the ones obtained by adopting an optimal regularization parameter at each iteration of a traditional hybrid method. Also, in this example, the combination of a projection method and Tikhonov regularization successfully maintains the
FIGURE 5  Example 1. Two perspectives of different surfaces in logarithmic scale: (a),(d): discrepancy surface (ie, the points lying on the discrepancy surface have coordinates $(a,k,\|b - Ax_k(a)\|^2)$); (b),(e): Regińska surface (ie, the points lying on the Regińska surface have coordinates $(a,k,\|b - Ax_k(a)\|\|Ax_k(a)\|)$); (c),(f): error surface (ie, the points lying on the error surface have coordinates $(a,k,\text{RRE}(a,k))$). In all frames, the blue square markers correspond to the values of the parameters $a_k$ computed at each iteration with the new method (using the discrepancy principle in (a),(d) and Regińska criterion in (b),(e)); the red circular markers correspond to the parameters $a_k$ computed at each hybrid iteration (using the discrepancy principle in (a),(d) and Regińska criterion in (b),(e)), and the green asterisk markers correspond to the values of $a_k$ that deliver a minimal error (optimal) solution (ie, $\min_k \text{RRE}(a,k)$ at each sampled iteration $k$).

error of the norm at the level of semiconvergence (that affects the purely iterative regularization method LSQR). For both parameter choice strategies, when using the new algorithm, as well as the traditional hybrid algorithms, the values of the regularization parameter (Figure 6 frames (b) and (d)) and the relative restoration error (Figure 6 frames (a) and (c)) stabilize as the iterations proceed. There is perfect agreement between the behavior observed in Figures 5 and 6. Another desirable feature that can be observed in Figure 6 is that the stopping criteria listed in Section 4.2 and 4.3 for the new algorithm succeed in stopping the iterations when a good regularization parameter is computed (ie, at a point that is quite close to the optimal one), and when the relative restoration error is quite low: for this test problem, the threshold $\theta = 10^{-3}$ is set for the discrepancy principle (stopping criteria (53) and (54)), and $\theta = 10^{-1}$ is set for Regińska criterion (stopping criterion (59)). Some stabilization in the values of both the relative restoration errors and the Tikhonov regularization parameters happens when the new algorithm stops according to these stopping criteria; in general, $\theta$ can be tuned to find a trade-off between speed and stabilization (to avoid performing additional iterations once some stabilization in the Tikhonov regularization parameters is detected).

Finally, Figure 7 displays some relevant reconstructions computed using different strategies; frame (a) displays the best achievable reconstruction (corresponding to the iteration minimizing the RRE for the optimal parameter choice strategy) and frames (b)-(e) display different reconstructions obtained using the new solvers. Frames (b) and (c) show to the reconstructions at the iterations minimizing the relative reconstruction error for the new solver using the discrepancy principle and the Regińska criterion, respectively. Note that frame (b) also corresponds to the reconstruction for the new algorithm using the discrepancy principle at the iteration given by the stopping criterion (53). At last, frame (d) shows the reconstruction computed by the new solver using the discrepancy principle at the iteration selected by the stopping criteria (54) and frame (e) shows the reconstruction computed by the new solver using Regińska criterion at the iteration selected by the stopping criterion (59). It can be observed that the quality of the reconstruction computed by the new algorithm approaches the optimal achievable quality, and that the stopping criteria successfully pick an early iteration that is close to the best possible iteration.
Example 1. (a) RRE versus number of iterations $k$ using the discrepancy principle; (b) regularization parameter values versus number of iterations $k$ using the discrepancy principle; (c) RRE versus number of iterations $k$ using Regińska criterion; (d) regularization parameter values versus number of iterations $k$ using Regińska criterion. Circular and triangular markers highlight the iterations satisfying the stopping criterion (53) and (54), respectively, while diamond-shaped markers highlight the iteration satisfying the stopping criterion (59).

5.2 Example 2

This example uses publicly available data, produced by performing a real tomographic scan of an object (in this case, a carved piece of cheese); see [78]. The data (sometimes referred to as sinogram) consists of measurements of the damping of fan-beam straight X-rays that penetrate the object. More precisely, 180 projections spanning the full 360 degree circle around the object are collected; the inverse problem involves reconstructing an image of the spatially varying attenuation coefficients of a 2D cross-section of the object. This problem leads to a linear system (1), where $A$ is a sparse coefficient matrix of size $178020 \times 65536$ (where only approximately 0.8% of its entries are nonzero), $x$ is the vectorialized 2D image (with resolution $256 \times 256$ pixels), and $b$ is the vectorialized sinogram. Note that an estimate of the magnitude of the (allegedly Gaussian white) noise $e$ that affects the measurements is not provided.

The following illustrates the performance of the GKB-based Krylov method LSQR (used as a purely iterative regularization method), the GKB-based hybrid methods, and the GKB-based new approach. The maximum number of iterations (i.e., the maximum dimension of the Krylov subspace for the approximation of the solution) is 250. Both the discrepancy principle (after a noise estimate has been recovered) and Regińska criterion are used to adaptively set the Tikhonov regularization parameter. Since the solution is expected to be piecewise-constant (as, essentially, only two different attenuation coefficients for the cheese material and for the air in the carvings have to be recovered), a finite difference approximation of the gradient operator is considered as regularization matrix $L$ (whose null space is spanned by constant vectors). More
FIGURE 7  Example 1. (a) best achievable reconstruction, corresponding to the iteration minimizing the relative reconstruction error (RRE) for the optimal parameter choice strategy; (b) best reconstruction obtained by the new method used in connection with the discrepancy principle (DP); (c) best reconstruction obtained by the new method used in connection with Regińska criterion (R); (d) reconstruction obtained by the new method used in connection with DP when the stopping criterion (54) is satisfied; (e) reconstruction obtained by the new method used in connection with R when the stopping criterion (59) is satisfied.

precisely, one should take

$$L = \begin{bmatrix} D_1 \otimes I \\ I \otimes D_1 \end{bmatrix} \in \mathbb{R}^{2\sqrt{n}(\sqrt{n}-1)\times n},$$  \hspace{1cm} (60)$$

where $D_1 \in \mathbb{R}^{(\sqrt{n}-1)\times \sqrt{n}}$ is a finite difference approximation of the one-dimensional first derivative and $I \in \mathbb{R}^{\sqrt{n}\times \sqrt{n}}$, so that, by exploiting the properties of the Kronecker product $\otimes$, the first and the second blocks of $L$ represent derivatives in the horizontal and vertical directions of a vectorialized 2D image, respectively. As remarked in Section 2, when considering regularizing Krylov methods $L^*_A$ may be formally regarded as right preconditioner that affects the approximation subspace for the solution; in the specific case of the GKB-based methods considered for this example, $x_k \in x_0 + \mathcal{K}_A(L^*_A(L^*_A)^T A^T A, L^*_A(L^*_A)^T A^T b)$, where $x_0 \in \mathcal{N}(L)$. Computationally convenient ways of expressing $L^*_A$, which exploit the structure of (60), are derived in [44]. The following also compares results obtained taking $L = I$ and $L$ as in (60).

The leftmost frame of Figure 8 plots the relative residual norm history (ie, $||b - Ax_k||/||b||$ versus $k$) associated to both LSQR (corresponding to regularization in standard form) and preconditioned LSQR (PLSQR, corresponding to regularization in general form, with $L$ as in (60)). One can clearly see that both residuals, despite being non-increasing because of the optimality property (13), somewhat stabilize: this happens very quickly (ie, approximately after 10 iterations) when LSQR is used. Also, the PLSQR residual norm always lays above the LSQR one: this fact reveals that, for this example (and in agreement with the theory hinted in Section 2), PLSQR is actually converging more slowly to the (unregularized) solution of (1). Indeed, looking at the top frames of Figure 9, one can clearly see that LSQR exhibits a quite severe semiconvergence (ie, the reconstruction obtained after 16 iterations is good but, as the iterations advance, a sudden deterioration in the quality of the reconstructions happens and the approximation obtained after 250 iterations is completely meaningless); on the contrary, within the performed 250 iterations, PLSQR seems to somewhat slowly but steadily progress toward...
a solution of good quality (ie, the reconstruction obtained after 16 iterations is oversmoothed, but semiconvergence does not happen and the reconstructions are less sensitive to any adopted stopping criterion).

Once LSQR has run, and the norm of the LSQR residual is observed to stabilize, its value can be taken as an estimate of the the magnitude $\|e\|$ of the noise affecting the data; see [79] for more details. Considering now the methods that combine GKB and Tikhonov regularization, the estimate $\|b - Ax_k\| \approx \tau \|e\|$ can therefore be used (with a safety factor $\tau = 1.05$) to apply the discrepancy principle; note that, equivalently, the noise level $\|e\|/\|b\|$ for this problem is approximately $8.08 \cdot 10^{-2}$. Alternatively, also Regińska criterion (which does not rely on an estimate of $\|e\|$) can be used, with exponent $\xi = 1$.

The middle frame of Figure 8 displays the values of the Tikhonov regularization parameter selected at each GKB iteration when $L = I$ and when both the new strategy and the classical hybrid approach are employed; both the discrepancy principle and Regińska criterion are used. One can see quite a good agreement between different solvers and regularization parameter choice rules. When the discrepancy principle is used within the hybrid method, equation (22) does not have any zero until iteration 9; the Tikhonov regularization parameter soon stabilizes during the following iterations. When the new strategy is used, a very large Tikhonov regularization parameter is selected trying to satisfy the discrepancy principle during the very early iterations (so that the problem is initially over-regularized), but then, around the 10th iteration, some stabilization happens to the same value of the Tikhonov regularization parameter computed by the hybrid method. Both stopping criteria (53) and (54) with $\theta = 10^{-5}$ are satisfied at the 23rd iteration (but the maximum number of allowed iterations is performed anyway to illustrate the behavior of the new method even after the stopping criteria are satisfied). The (2D versions of the) reconstructions $x_k$ computed by the new method for $k = 23$ and $k = 250$ are displayed in the frames in position (2,1) and (2,2) of Figure 9: the quality of both reconstructions is quite good, meaning that the discrepancy principle (used within the new framework, but also within the classical hybrid framework) selects a suitable regularization parameter, which also allows to overcome semiconvergence. Similarly, when Regińska criterion is used together with the hybrid method and as the iterations proceed, the value of the Tikhonov regularization parameter quickly stabilizes to a value that is slightly lower than the one selected by the discrepancy principle. When the new strategy is used, the parameter selected by Regińska criterion undergoes an initial increase but then, toward the 20th iteration, it decreases and stabilizes on the same value selected by the hybrid method; the stopping criterion (59) with $\theta = 10^{-1}$ is satisfied at the 24th iteration (when, in particular, the value of the parameter is stabilized). The (2D versions of the) reconstructions $x_k$ computed by the new method for $k = 24$ and $k = 250$ are displayed in the frames in position (3,1) and (3,2) of Figure 9: these are almost identical to the ones computed when employing the discrepancy principle (shown in the frames in position (2,1) and (2,2)).

The rightmost frame of Figure 8 displays the values of the Tikhonov regularization parameter selected at each GKB iteration when $L$ is the matrix in (60) and when both the new strategy and the classical hybrid approach are employed; both the discrepancy principle and Regińska criterion are used. Contrarily to the $L = I$ case, one can see several differences between distinct solvers and regularization parameter choice rules. When the discrepancy principle is used within the hybrid method, equation (22) does not have any zero until iteration 86 (which is to be expected because, as displayed
in the leftmost frame of Figure 8, the PLSQR residual norm decreases much more slowly than the LSQR one); starting from approximately the 200th iteration, the value of the Tikhonov regularization parameter stabilizes to a value much larger than the one selected for the $L = I$ case: this has a desirable effect on the ideally piecewise-constant reconstruction, as the regularization matrix (60) enforces constant reconstructions. When the new strategy is used, after approximately 150 iterations, some stabilization happens to the same value of the Tikhonov regularization parameter computed by the hybrid method. Taking $\theta = 10^{-5}$, the stopping criterion (53) is satisfied at the 70th iteration, while stopping criterion (54) is satisfied at the 88th iteration (also in this case, the maximum number of allowed iterations is performed anyway to illustrate the behavior of the new method even after the stopping criteria are satisfied); one can clearly see that both stopping criteria prescribe to stop before some stabilization in the values of the Tikhonov regularization parameter happens, so that the reconstructions may be slightly over-regularized. The (2D versions of the) reconstructions $x_k$ computed by the new method for $k = 88$ and $k = 250$ are displayed in the frames in position (2,3) and (2,4) of Figure 9: despite the different values $\alpha_{89} = 114.57$ and $\alpha_{251} = 94.52$, visually the two reconstructions are almost indistinguishable. Regińska criterion is not very effective for this instance of this experiment: when used together with the hybrid method, the value of the regularization parameter basically stagnates from the beginning of the iterations; when used together with the new strategy, the value of the regularization parameter steeply increases as the iterations proceed (so that a good stopping criterion is crucial in this situation). This behavior can be explained considering the shape of the functionals $P_k(x_k(\alpha_k))$ (28) to be minimized at each iteration of the hybrid method, and the function $P_k(\alpha) = \tilde{R}_k(\alpha)$ to be used in (55). Even if not shown here, $P_k(x_k(\alpha))$ is pathologically flat, making any minimization routine (eg, MATLAB’s \texttt{fminbnd} that is based on the golden section search) massively reliant on a good estimate of the search interval; indeed, for every iteration $k$, \texttt{fminbnd} performs 500 steps (ie, the default maximum number of steps) and the value of the minimizer barely moves from the starting one. The behavior of the upper bounds $\tilde{R}_k(\alpha)$ is similar to the one shown in Figure 4 for the image deblurring example, even

**FIGURE 9** Example 2. Reconstructions achieved by the purely iterative LSQR and PLSQR methods (first row) and by the new algorithm used together with the discrepancy principle (DP) (second row) or Regińska criterion (R) (third row). Frames in columns 1 and 2 consider $L = I$, while frames in columns 3 and 4 take $L$ as in (60). Frames in columns 1 and 3 display early iterations or iterations satisfying the stopping criterion ((54) for DP, (59) for R), while frames in columns 2 and 4 display the last iteration.
if the increase (when \(a\) gets large and small) is much steeper, with an an almost constant \(\frac{\partial^2 P_k(a)}{\partial a} = 0\). The stopping criterion (59) with \(\theta = 10^{-1}\) for the new method is satisfied at the 22nd iteration, where the selected regularization parameter is much higher than the suitable one selected when the new method based on the discrepancy principle is stopped. For these reasons, one can conclude that, when \(L\) is chosen as in (60), Regińska criterion seems to fail for this test problem. However, on the upside, the new method proves to be much more computationally convenient than the hybrid method: adapting the estimates derived at the end of Section 4.3 (exploiting the fact that \(A\) is sparse and exploiting the structure of \(L^*_A\) when deriving the computational cost of matrix-vector products), one can conclude that performing 250 iterations of the hybrid method requires \(O(4.8 \cdot 10^{10})\) flops, while performing 22 iterations of the new method requires \(O(4.2 \cdot 10^9)\) flops (i.e., approximately 8% of the computations required by the hybrid approach). The (2D versions of the) reconstructions \(x_k\) computed by the new method for \(k = 22\) (with \(\alpha_{23} = 664.27\)) and \(k = 250\) (with \(\alpha_{251} \sim 10^{23}\)) are displayed in the frames in position (3,3) and (3,4) of Figure 9: these are both over-smoothed and look quite different from the ones computed when the discrepancy principle is adopted (frames in position (3,1) and (3,2) of the same figure).

6 | CONCLUSIONS AND OUTLOOK

This paper surveyed a variety of iterative regularization methods that are commonly used when obtaining a Tikhonov-regularized solution through direct factorizations would be too computationally demanding, e.g., when solving large-scale unstructured linear inverse problems. The main focus of the paper was on regularizing projection methods that are based on Krylov subspace methods, and on the so-called hybrid methods, which combine iterations of a Krylov subspace method and Tikhonov regularization. Hybrid methods are usually preferred to purely iterative regularization methods because, as the iterations progress, if the regularization parameters are properly tuned, the quality of the regularized solution does not deteriorate (and may even improve) at the point of semiconvergence. Well-established but often empirical ways of setting the regularization parameters have been derived in the past decades, and some of them are reviewed in this paper.

This paper also introduced and analyzed a new class of algorithms for the solution of bilevel optimization problems of the form (29) arising when simultaneously computing a Tikhonov-regularized solution and a regularization parameter according to a given rule, and when still considering large-scale unstructured linear inverse problems. By a novel use of Krylov projection methods based on the GKB algorithm, its connections with Gaussian quadrature rules, and a new modified Newton method, the proposed new algorithms “interlace” the iterations performed to apply a given (nonlinear) parameter choice rule and the iterations performed to iteratively solve the (linear) Tikhonov-regularized problem, giving rise to an efficient and principled strategy that delivers results comparable to the ones obtained with traditional hybrid methods.

The introduction of this new class of algorithms carries a number of open questions and possible extensions that may be addressed in future research. For instance, future work can include the natural extension of the new algorithms to work with Krylov projection methods that are based on decompositions other than GKB (e.g., the Arnoldi decomposition or flexible Krylov methods; see [80]); also extensions to projection strategies that handle a generic regularization matrix \(L\) by computing joint decompositions of the coefficient matrix \(A\) and \(L\) (see, for instance, [63, 81]) may be considered as an alternative to the strategy, employed within this paper, of transforming the Tikhonov problem into standard form and preconditioning the approximation subspace for the solution. Although the algorithmic details of the new approach can be easily adapted to these situations, the theoretical analysis of the resulting strategies needs to be carefully rethought. Also, other parameter choice strategies that can be expressed in the framework of bilevel optimization problems (e.g., the UPRE, GCV, and L-curve criteria reported in Section 3) can be considered. Moreover, while a solid converge proof is available for the new algorithm applied in connection with the discrepancy principle, the considered extension to other parameter choice rules is still quite heuristic and more theoretical developments are needed to prove convergence to possible local minima.

Finally, since the new class of algorithms is very general, it may be potentially extended to a variety of bi-level optimization methods that involve the solution of a nonlinear higher-level problem and a linear lower-level problem, even beyond the regularization task. However, as far as regularization is concerned, it should be mentioned that simple constraints such as non-negativity or box constraints, which sometimes have a dramatic effect in enhancing the quality of the reconstructions, cannot be straightforwardly incorporated when adopting the basic Krylov subspace methods discussed in this paper (although some approaches based on flexible Krylov methods are available; see [82]). Similarly, the methods
discussed in this paper cannot be straightforwardly applied to nonlinear ill-posed problems; see, eg, [83,84]. Extending the new class of algorithms to handle these situations can be the focus of future research.

REFERENCES

[1] P. C. Hansen, *Discrete inverse problems: Insight and algorithms*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2010.
[2] C. R. Vogel, *Computational methods for inverse problems*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2002.
[3] P. C. Hansen, J. G. Nagy, and D. P. O’Leary, *Deblurring images: Matrices, spectra, and filtering*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2006.
[4] M. Hanke and P. C. Hansen, Regularization methods for large-scale problems, Surveys Math. Indus. 3 (1993), 253–315.
[5] H. W. Engl, M. Hanke, and A. Neubauer, Regularization of inverse problems, in *Mathematics and Its Applications*, Springer, Netherlands, 1996.
[6] J. M. Bardsley and J. G. Nagy, Covariance-preconditioned iterative methods for nonnegatively constrained astronomical imaging, SIAM J. Matrix Anal. Appl. 27 (2006), 1184–1197.
[7] R. A. Renaut, I. Hnětynková, and J. Mead, Regularization parameter estimation for large-scale Tikhonov regularization using a priori information, Comput. Statist. Data Anal. 54 (2010), 3430–3445.
[8] G. H. Golub and C. F. Van Loan, *Matrix computations*, The Johns Hopkins University Press, Baltimore, MD, 1996.
[9] B. Hofman, *regularization for applied inverse and ill-posed problems*, Teubner, Stuttgart, 1986.
[10] P. C. Hansen, The discrete picard condition for discrete ill-posed problems, BIT 30 (1990), 658–672.
[11] J. Chung, G. Easley, and D. P. O’Leary, Windowed spectral regularization of inverse problems, SIAM J. Sci. Comput. 33 (2011), 3175–3200.
[12] L. Dykes and L. Reichel, On the reduction of Tikhonov minimization problems and the construction of regularization matrices, Numer. Algorithms 60 (2012), 683–696.
[13] L. Reichel and Q. Ye, Simple square smoothing regularization operators, Electron. Trans. Numer. Anal. 33 (2009), 63–83.
[14] S. Morigi, L. Reichel, and F. Sgallari, Orthogonal projection regularization operators, Numer. Algorithms 44 (2007), 99–114.
[15] L. Eldén, A weighted pseudoinverse, generalized singular values, and constrained least squares problems, BIT 22 (1982), 487–502.
[16] J. Kamm and J. G. Nagy, Kronecker product and SVD approximations in image restoration, Linear Algebra Appl. 284 (1998), 177–192.
[17] A. Aričo et al., *the anti-reflective transform and regularization by filtering*, in *Lecture Notes in Electrical Engineering*, Vol 80, Springer-Verlag, Berlin, 2011, 1–21.
[18] S. Gazzola and Landman M. Sabaté, Flexible GMRES for total variation regularization, BIT Numer. Math. 59 (2019), 721–746.
[19] F. Natterer, *The mathematics of computerized tomography*, John Wiley, New York, 1986.
[20] Jia Zhongxiao, Approximation accuracy of the Krylov subspaces for linear discrete ill-posed problems, J. Comput. Appl. Math. 374 (2020), 112786.
[21] P. C. Hansen, *Rank-deficient and discrete ill-posed problems*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1998.
[22] Y. Saad, *Iterative methods for sparse linear systems*, 2nd ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, 2003.
[23] M. Hanke, Regularization with differential operators: An iterative approach, Numer. Funct. Anal. Optim. 13 (1992), 523–540.
[24] M. Hanke, J. G. Nagy, and R. J. Plemmons, Preconditioned iterative regularization, Numer. Linear Algebra 9 (1992), 141–163.
[25] P. C. Hansen, Regularization, GSVD and truncated GSVD, BIT 29 (1989), 491–504.
[26] D. Calvetti, B. Lewis, and L. Reichel, GMRES-type methods for inconsistent systems, Linear Algebra Appl. 316 (2000), 157–169.
[27] Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual method for solving nonsymmetric linear systems, SIAM J. Sci. Comput. 7 (1986), 856–869.
[28] C. C. Paige and M. A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, Association for Computing Machinery. Transactions on Mathematical Software 8 (1982), 43–71.
[29] C. C. Paige and M. A. Saunders, Algorithm 583: LSQR: Sparse linear equations and least squares problems, Association for Computing Machinery. Transactions on Mathematical Software 8 (1982), 195–209.
[30] P. Novati and M. R. Russo, A GCV-based Arnoldi-Tikhonov regularization method, BIT 54 (2014), 501–521.
[31] S. Gazzola, P. Novati, and M. R. Russo, On Krylov projection methods and Tikhonov regularization, Electron. Trans. Numer. Anal. 44 (2015), 83–123.
[32] Y. Huang and Z. Jia, On regularizing effects of MINRES and MR-II for large scale symmetric discrete ill-posed problems, J. Comput. Appl. Math. 320 (2017), 145–163.
[33] S. Gazzola and P. Novati, Inheritance of the discrete Picard condition in Krylov subspace methods, BIT 56 (2016), 893–918.
[34] S. Gazzola, P. Novati, and M. R. Russo, Embedded techniques for choosing the parameter in Tikhonov regularization, Numer. Linear Algebra Appl. 21 (2014), 796–812.
[35] I. Hnětynková, M. Plešinger, and Z. Strakoš, The regularizing effect of the Golub-Kahan iterative bidiagonalization and revealing the noise level in the data, BIT 49 (2008), 204–220.
[36] I. Hnětynková, M. Kubínová, and M. Plešinger, Noise representation in residuals of CRAIG, LSQR and LSMR regularization, Linear Algebra Appl. 533 (2017), 357–379.
[37] P. Novati, Some properties of the Arnoldi based methods for linear ill-posed problems, SIAM J. Numer. Anal. 55 (2017), 1437–1455.
[38] T. K. Jensen and P. C. Hansen, Iterative regularization with minimum-residual methods, BIT 47 (2007), 103–120.
[39] C. Paige and M. Saunders, Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. 12 (1975), 617–629.
[40] M. Hanke, *Conjugate gradient type methods for ill-posed problems*, CRC Press, Harlow, UK, 1995.
[79] I. Hnětynková, M. Plešinger, and Z. Strakoš, The regularizing effect of the Golub-Kahan iterative bidiagonalization and revealing the noise level in the data, BIT 49 (2009), 669–696.

[80] J. Chung and S. Gazzola, Flexible Krylov Methods for $\ell_p$ Regularization, SIAM J. Sci. Comput. 41 (2019), S149–S171.

[81] M. E. Hochstenbach, L. Reichel, and X. Yu, A Golub-Kahan-type reduction method for matrix pairs, J. Sci. Comput. 65 (2015), 767–789.

[82] S. Gazzola and Y. Wiaux, Fast nonnegative least squares through Flexible Krylov subspaces, SIAM J. Sci. Comput. 39 (2017), A655–A679.

[83] A. B. Bakushinsky, M. Y. Kokurin, and A. Smirnova, *Iterative methods for ill-posed problems: An introduction*, Walter de Gruyter, Berlin, New York, 2011.

[84] B. Kaltenbacher, A. Neubauer, and O. Scherzer, *Iterative regularization methods for nonlinear ill-posed problems*, Walter de Gruyter, Berlin, New York, 2008.

**How to cite this article:** Gazzola S, Sabaté Landman M. Krylov methods for inverse problems: Surveying classical, and introducing new, algorithmic approaches. *GAMM-Mitteilungen*. 2020;43:e202000017. [https://doi.org/10.1002/gamm.202000017](https://doi.org/10.1002/gamm.202000017)