Journal of Decision Science and Optimization
Vol. 2 (2026), No. 1, pp. 65-76
DOI: https://doi.org/10.55578/jdso.2605.005

ARTICLE

Stability and Fast Solvers for Ill-Conditioned Linear Inverse Problems via Regularization and Preconditioning

Adane Akate[1,*] , Emőke Imre[1]

[1]*Doctoral School of Applied Informatics and Applied Mathematics, Obuda University, Budapest, Hungary*

*Corresponding author. Email: adane132akate@gmail.com

Received: 17 Apr 2026, Revised: 11 May2026, Accepted: 14 May 2026

Abstract

Ill-conditioned linear inverse problems are pervasive in scientific computing and engineering, yet their inherent instability and slow iterative convergence remain major computational bottlenecks. To address these challenges, we propose a unified computational framework that couples Tikhonov regularization with preconditioned Krylov subspace solvers. By reformulating the inverse problem through regularized normal equations, the proposed approach systematically suppresses noise amplification while preserving essential solution features. To further accelerate convergence, we introduce problem-adapted preconditioners that cluster the spectrum of the regularized system, enabling preconditioned conjugate gradient (PCG) methods to converge in significantly less iteration. Theoretical insights into conditioning, spectral filtering, and preconditioner design are synthesized to demonstrate how regularization and preconditioning operate synergistically. A numerical study on a nearly rank-deficient system validates the framework, highlighting both numerical stability and rapid convergence. The results confirm that the integrated strategy offers a robust, scalable, and computationally efficient pathway for solving large-scale ill-conditioned inverse problems.

Keywords: ill-conditioned inverse problems; regularization methods; preconditioning techniques; Krylov subspace solvers; numerical stability and efficiency.

1. INTRODUCTION

Ill-conditioned linear inverse problems permeate numerous contemporary scientific and engineering disciplines, ranging from medical imaging and geophysical exploration to computational photography and data-driven machine learning [1-3]. In these contexts, the central task involves estimating an unknown parameter vector from noisy measurements that are related through a linear forward model [3,4]. The forward operator is typically obtained by discretizing a compact integral equation or exhibits inherent near-singularity, which produces large condition numbers and extreme sensitivity to measurement perturbations [2,5]. As a result, straightforward least-squares estimates are generally unphysical and overwhelmingly dominated by amplified noise [1,2].

To counteract this instability, prior structural knowledge must be enforced through regularization. Tikhonov regularization remains the standard approach, as it systematically attenuates solution components linked to small singular values and thereby curbs noise amplification [6]. Iterative and hybrid formulations further improve robustness by embedding regularization within Krylov subspace frameworks and utilizing early-stopping rules, which scale gracefully to high-dimensional regimes [6]. Working in tandem with regularization, preconditioning modifies the eigenvalue distribution of the regularized system, markedly accelerating the convergence of iterative solvers such as CG, GMRES, and LSQR. This is especially valuable in matrix-free or large-scale applications where only matrix -vector products with the forward operator and its adjoint are accessible [3,4]. Recent investigations have also highlighted how preconditioning and Tikhonov regularization can be constructed synergistically; for instance, Chabib et al. [5] show that adopting an identical symmetric positive semi-definite operator for

both roles permits efficient, Ritz-based sweeps over Tikhonov parameters with minimal computational overhead.

Moving beyond traditional linear preconditioners, recent literature has introduced data-driven and randomized techniques tailored to specific problem families and ultra-high-dimensional settings. Neural incomplete factorization (NeuralIF) leverages graph neural networks to learn effective CG preconditioners, delivering notable speedups on large-scale linear systems [7]. Deep distillation gradient preconditioning merges knowledge distillation with nonlinear preconditioning to hasten convergence in imaging-related inverse tasks [8]. Simultaneously, sketch-and-precondition methodologies build preconditioners from randomized low-rank surrogates, producing numerically stable least-squares solvers even when the underlying system is severely ill-conditioned [2]. Related progress in randomized Krylov schemes incorporates sketching or stochastic embeddings into Golub -Kahan and GMRES-type iterations, lowering memory demands without compromising reconstruction accuracy [6,9].

Notwithstanding these methodological advances, a cohesive framework that rigorously unifies Tikhonov regularization, problem-specific preconditioning, and efficient Krylov iteration remains underdeveloped. This work closes that gap by integrating these elements into a single computational pipeline designed for ill-conditioned linear inverse problems [4, 5]. We begin by demonstrating the foundational mechanics on a compact, highly ill-conditioned 2×2 system, which clarifies regularization parameter selection and highlights how preconditioning accelerates convergence in a transparent, analytically tractable setting. We subsequently examine the framework's asymptotic behavior in high-dimensional contexts, showing that the conditioning of the regularized normal equations, the number of PCG iterations, and total computational cost are dictated primarily by the expense of matrix -vector multiplications and the spectral efficiency of the selected preconditioner (incomplete factorizations, or sketch-based constructions). We also pinpoint dominant computational bottlenecks, such as repeated forward/adjoint evaluations and preconditioner inversions, and outline how matrix-free implementations, FFT-accelerated operators, and learned or randomized preconditioners can alleviate these constraints [2,7,8]. This study establishes that classical Tikhonov stabilization, when systematically coupled with well-designed preconditioners and modern Krylov solvers, delivers both numerical stability and algorithmic scalability for demanding inverse problems.

2. MATHEMATICAL FORMULATION OF THE INVERSE PROBLEM

We consider a linear inverse problem of the form

$$Ax = b \tag{1}$$

where $A \in \mathbb{R}^{n\times m}$ is a known (possibly ill-conditioned) matrix, $x \in \mathbb{R}^m$is the unknown parameter vector to be estimated, and $b \in \mathbb{R}^n$denotes noisy measurements. In many applications one has

$$b = Ax_{true} + \eta \tag{2}$$

where $x_{true}$is the exact underlying parameter vector and η represents measurement or modeling noise. When A is severely ill-conditioned, the least-squares problem

$$\min_x \|Ax - b\|_2^2 \tag{3}$$

becomes ill-posed: minor perturbations in $\boldsymbol{b}$ induce disproportionately large deviations in the minimizer. Achieving stable and physically interpretable solutions therefore necessitates regularization for stability and preconditioning for computational efficiency.

2.1. Regularization for Ill-Posed Problems

Regularization modifies the original optimization landscape by introducing penalty terms that attenuate solution components corresponding to small singular values of A, which are primarily responsible for noise amplification.

2.1.1. Tikhonov Regularization

Given A and b, Tikhonov regularization seeks an approximate solution x_λ by solving the regularized optimization problem.

$$\min_x J_\lambda(x), \text{ with } J_\lambda(x) = \|Ax - b\|_2^2 + \lambda\|Lx\|_2^2 \tag{4}$$

where $\lambda > 0$ is the regularization parameter and $L \in \mathbb{R}^{p\times m}$ is a regularization operator (e.g., $L = I$ for zeroth-order regularization or a discrete derivative operator for smoothness constraints).

Step 1: Compute the gradient of the Tikhonov functional.

Using standard matrix calculus, we have

$$\|Ax-b\|_2^2=(Ax-b)^T(Ax-b), \|Lx\|_2^2=(Lx)^T(Lx)=x^TL^TLx$$

Hence

$$J_\lambda(x)=(Ax-b)^T(Ax-b)+\lambda x^TL^TLx \quad (5)$$

Differentiating with respect to x and using the fact that $\nabla_x(x^TQx)=(Q+Q^T)x$ for a constant matrix Q, we obtain

$$\nabla_x J_\lambda(x)=2A^T(Ax-b)+2\lambda L^TLx$$

Step 2: Set the gradient to zero (first-order optimality condition).

A minimizer x_λ satisfies

$$\nabla_x J_\lambda(x_\lambda)=0 \Rightarrow 2A^T(Ax_\lambda-b)+2\lambda L^TLx_\lambda=0$$

Dividing by 2 and rearranging yields the Tikhonov normal equations

$$(A^TA+\lambda L^TL)x_\lambda=A^Tb \quad (6)$$

Defining the regularized system matrix

$$S_\lambda:=A^TA+\lambda L^TL \quad (7)$$

we can write (6) compactly as

$$S_\lambda x_\lambda=A^Tb$$

Step 3: Spectral interpretation.

If A has singular value decomposition $A=U\Sigma V^T$ with singular values $\{\sigma_i\}$, and $L=I$ for simplicity, then

$$S_\lambda=A^TA+\lambda I=V(\Sigma^2+\lambda I)V^T \quad (8)$$

The solution can be expressed in the singular vector basis as

$$x_\lambda=\sum_i \frac{\sigma_i}{{\sigma_i}^2+\lambda}{u_i}^T b v_i \quad (9)$$

where $\{u_i\}$ and $\{v_i\}$ are the left and right singular vectors of A. The factors

$$\phi_i(\lambda):=\frac{\sigma_i}{{\sigma_i}^2+\lambda} \quad (10)$$

act as filter factors: for small σ_i (directions associated with noise amplification), $\phi_i(\lambda)$ becomes small, thereby damping their contribution to x_λ. Increasing λ enhances stability but biases the solution toward prior information encoded by L.

The choice of λ is crucial too small a value leaves the solution unstable and overly sensitive to noise, while too large a value oversmooths the solution and discards important features. Several well-established strategies are commonly used to parameter select λ in practice:

- **Discrepancy principle:** If an estimate of the noise level $\| \eta \|_2$ is available, λ is chosen so that the residual norm of the Tikhonov solution approximately matches
$$\| Ax_\lambda-b \|_2 \approx \tau \| \eta \|_2 \| \text{ for some } \tau \gtrsim 1. \quad (11)$$
- **L-curve method:** On the plots pair
$$(log \| Ax_\lambda-b \|_2, log \| Lx_\lambda \|_2) \quad (12)$$
Over a range of λ values and chooses λ near the "corner" of this L-shaped curve, where the trade-off between data fit and regularization is balanced.
- **Generalized cross-validation (GCV):** Select λ that approximately minimizes a predictive error functional constructed from the residuals and the influence matrix of the regularized solution.

2.1.2. Iterative and Data-Driven Regularization

Iterative regularization methods seek approximate solutions by iteratively applying transformations involving A and A^T, and by stopping the iteration early to prevent overfitting of noise.

1. **Landweber iteration**: The classical Landweber method for solving Equation (2) is given by
$$x^{(k+1)}=x^{(k)}+\omega A^T\left(b-Ax^{(k)}\right), k=0,1,2,\ldots,$$
where $\omega \in (0,2/\| A \|_2^2)$ is a relaxation parameter. Starting from an initial guess $x^{(0)}$, one obtains a sequence $x^{(k)}$ that converges to a least-squares solution as $k \to \infty$ for noise-free data. In the presence of noise, however, the iterates exhibit semi-convergence: initially they approach a good approximation of x_{true}, but after a certain iteration index, high-frequency noise

components become amplified and error increases. Thus, early stopping (e.g., based on the discrepancy principle) serves as a form of implicit regularization.

2. **Conjugate gradient-type methods:** When applied to the normal equations $A^TAx = A^Tb$, conjugate gradient (CG) or conjugate gradient for least squares (CGLS) also exhibit semi-convergence, and their regularization effect is controlled by the iteration count. Recent analyses interpret such methods as filtering high-frequency components in early iterations, similar in spirit to explicit Tikhonov filtering [12].
3. **Randomized iterated Tikhonov:** More recent developments propose randomized iterated Tikhonov methods that blend stochastic optimization with classical Tikhonov penalties. A typical scheme in Hilbert spaces can be written abstractly as

$$x^{(k+1)} = x^{(k)} - \alpha_k\left(A^TAx^{(k)} - A^Tb + \lambda_k L^TLx^{(k)}\right) + \xi^{(k)},$$

where α_k is a step size, λ_k may be iteration-dependent, and $\xi^{(k)}$ captures randomized sketching or sampling effects. Under suitable conditions on $\{\alpha_k\}$, $\{\lambda_k\}$, and an adaptive stopping rule (e.g., discrepancy-based), these methods achieve scalability and robustness for high dimensional inverse problems while retaining the stabilizing effect of Tikhonov regularization [6].

2.2. Preconditioning for Faster Convergence

Even after regularization, the system matrix equation (7) may remain ill-conditioned enough to slow down iterative solvers. Preconditioning aims to transform the linear system into an equivalent one with more favourable spectral properties, improving convergence rates of Krylov subspace methods such as CG.

Given a non-singular preconditioner $M \approx S_\lambda$, one solves

$$M^{-1}S_\lambda x = M^{-1}A^Tb, \tag{13}$$

or in right-preconditioned form $S_\lambda M^{-1}y = A^Tb$ with $x = M^{-1}y$. The goal is to choose M so that the eigenvalues of $M^{-1}S_\lambda$ cluster tightly, ideally near 1, resulting in faster convergence of Krylov methods [10, 13].

2.2.1. Krylov Subspace Preconditioners

Preconditioners applied within Krylov subspace methods help accelerate convergence. For ill-conditioned systems, invertible smoothing preconditioners and spectral filters are designed to enhance the conditioning of subspaces used in iteration.

A k-dimensional Krylov subspace associated with matrix S_λ and vector $r^{(0)}$ is

$$K_k(S_\lambda, r^{(0)}) = span\{r^{(0)}, S_\lambda r^{(0)}), S_\lambda^2 r^{(0)}, \dots, S_\lambda^{k-1} r^{(0)}\}$$

Krylov methods construct approximations $x^{(k)}$ in this subspace (or related subspaces) that minimize or reduce the residual $\| S_\lambda x^{(k)} - A^Tb \|_2$.

1. **Invertible smoothing preconditioners:** For ill-posed problems, invertible smoothing preconditioners M are designed such that M^{-1} damps high-frequency components of the residual, effectively acting as a low-pass filter on the error. A typical construction is

$$M = R^TR,$$

where R is a smoothing operator, so that M^{-1} suppresses oscillatory structures in the error components associated with small singular values of A.

2. **Arnoldi -Tikhonov and Arnoldi -TSVD:** In the non-symmetric case, GMRES and Arnoldi-based methods approximate the solution in Krylov subspaces generated by A (or S_λ), and hybrid variants integrate Tikhonov or TSVD regularization into the projected problem. At iteration k, Arnoldi -Tikhonov constructs the approximation

$$x^{(k)} = V_k y^{(k)},$$

where V_k is the matrix of orthonormal basis vectors spanning $K_k(A, r^{(0)})$. The coefficient vector $y^{(k)}$ is obtained by solving a small regularized least-squares problem

$$y^{(k)} = arg\min_y \left\|\beta e_1 - \widetilde{H}_k y\right\|_2^2 \ + \lambda\|y\|_2^2$$

where $\widetilde{H}_k$ is the Hessenberg matrix from Arnoldi and βe_1 encodes the initial residual.

This integrates regularization into the Krylov framework and can be combined with preconditioning to improve convergence on ill-conditioned systems.

2.2.2 Randomized and Sketch-Based Preconditioning

For large-scale least-squares problems, sketching-based methods construct randomized lower-dimensional surrogates of A to define preconditioners whose effectiveness is largely independent of the original system's condition number.

Let $S_{sketch} \in \mathbb{R}^{s\times n}$ be a random sketching matrix (e.g., a subsampled randomized Hadamard transform or sparse embedding), with $s \ll m$. The sketched matrix is

$$\tilde{A} := S_{sketch}A \in \mathbb{R}^{s\times m},$$

and one constructs a preconditioner based on QR or SVD of $\tilde{A}$. For instance, if $\tilde{A} = \tilde{Q}\tilde{R}$ is a QR factorization, then

$$M = \tilde{R}^T\tilde{R}$$

serves as a preconditioner for A^TA (or S_λ). Under appropriate conditions, the spectrum of $M^{-1}A^TA$ is tightly clustered, and the condition number $\kappa(M^{-1}A^TA)$ becomes essentially independent of $\kappa(A)$. This leads to substantial improvements in both runtime and stability for high-dimensional discrete inverse problems [2,11,14].

2.3. Combined Regularization and Preconditioning Frameworks

Optimal performance emerges when regularization and preconditioning are integrated rather than applied sequentially. Hybrid Krylov methods solve projected regularized subproblems within the Krylov space, adaptively tuning λ via discrepancy or L-curve principles in the reduced dimension [15]. Learning-based preconditioners employ neural networks $\mathcal{N}_\theta$ to map residuals to preconditioned directions, replacing classical $M^{-1}r$ updates with data-adaptive transformations that accelerate convergence in imaging and reconstruction pipelines [8]. The present work builds upon this foundation, formalizing the regularized normal equations enhanced by structured preconditioning and Krylov acceleration.

3. MATHEMATICAL METHODOLOGY

Our framework embeds Tikhonov regularization within a preconditioned iterative solver to deliver solutions that are both noise-robust and computationally efficient. The procedure follows three sequential stages: (1) construction of the regularized normal equations, (2) design of a problem-adapted preconditioner, and (3) solution via a preconditioned Krylov method with adaptive termination.

Step 1: Tikhonov-regularized normal equations

Starting from the linear inverse problem equation (1) with $\in \mathbb{R}^{n\times m}$, $x \in \mathbb{R}^m$, and $b \in \mathbb{R}^n$, we first stabilize the problem using Tikhonov regularization. We consider the minimization problem

$$\min_x \|Ax - b\|_2^2 + \lambda\|Lx\|_2^2$$

where $\lambda > 0$ is the regularization parameter and $L \in \mathbb{R}^{p\times m}$ is a regularization operator. The operator $\boldsymbol{L}$ is chosen based on prior knowledge of the expected structure of the solution.

- ✓ $L = I$ (identity) for zeroth-order regularization when we want to control the overall magnitude of x.
- ✓ Discrete first- or second-order difference operators L when smoothness of x is desired, or edge-preserving operators for imaging applications.

The first-order optimality condition for this problem yields the Tikhonov-regularized normal equations (6). We denote

$$S_\lambda := A^TA + \lambda L^TL,$$

so that the regularized system takes the compact form

$$S_\lambda x_\lambda = A^Tb$$

This step improves stability by adding the positive definite term λL^TL, which damps components of the solution associated with small singular values of A and thereby reduces sensitivity to noise in b.

Step 2: Preconditioner Type and Construction

In practice, several classes of preconditioners are particularly relevant for ill-conditioned inverse problems:

a. Diagonal (Jacobi) preconditioner.

We define

$$M = diag(S_\lambda), M^{-1} = diag(S_\lambda)^{-1} \quad (14)$$

This preconditioner uses only the diagonal of S_λ, so it is extremely cheap to assemble and apply, and it is easy to implement in matrix-free settings. It typically provides a moderate improvement in conditioning and serves as a baseline choice or as a building block for more sophisticated preconditioners. In the numerical experiment presented in Section 4, we employ precisely this diagonal (Jacobi) preconditioner to illustrate the effect of preconditioning on PCG convergence.

b. Incomplete factorization (ILU / incomplete Cholesky).

When S_λ is symmetric positive definite and sparse, we can compute an approximate factorization

$$S_\lambda \approx R^T R$$

with controlled sparsity in R (incomplete Cholesky for SPD systems, or ILU in the nonsymmetric case). Applying M^{-1} then corresponds to two sparse triangular solves. This approach is effective for high-dimensional but still memory-feasible problems where the sparsity pattern can be exploited, and it often yields a substantial reduction in PCG iteration counts compared with simple diagonal preconditioning.

c. Randomized / sketch-based preconditioners.

In very high-dimensional settings where forming or factorizing A or S_λ directly is too expensive, we can build a preconditioner from a reduced surrogate

$$\tilde{A} = S_{sketch} A,$$

where S_{sketch} is a sketching matrix. A factorization of $\tilde{A}$ (for example, $\tilde{A} = \tilde{Q}\tilde{R}$) is then used to define

$$M = \tilde{R}^T \tilde{R}$$

as an approximate preconditioner for $A^T A$ or S_λ. Such sketch-based preconditioners can significantly reduce the effective condition number while keeping both storage and arithmetic costs manageable, and they are especially attractive for very high-dimensional inverse problems.

This structured overview clarifies which preconditioners are considered, how they are constructed, and when they are appropriate, making the role of the preconditioner in our framework much more transparent, as requested by the reviewer.

Although regularization improves stability, the matrix $S_\lambda = A^T A + \lambda L^T L$ can still be ill-conditioned enough that iterative solvers converge slowly, especially in large-scale problems. To accelerate convergence, we introduce a preconditioner M that approximates S_λ but is easier to invert.

We seek a symmetric positive definite preconditioner M such that the preconditioned operator

$$M^{-1} S_\lambda$$

has the following desirable properties:

- Its eigenvalues are clustered, ideally near 1, which leads to faster convergence of Krylov methods like PCG.
- M can be applied efficiently, typically via sparse structures, incomplete factorizations, or approximate inverses.
- M respects the regularized structure, often being constructed from approximations of $A^T A$, $L^T L$, or combinations thereof.

Typical choices include:

- Jacobi (diagonal) preconditioner: $M = diag\ (S_\lambda)$, which is extremely cheap and can already improve conditioning for moderate problems.
- Incomplete Cholesky or incomplete LU factorizations: $M \approx R^T R$ with a sparse factor RR that approximates the Cholesky factor of S_λ.
- Problem-structured or randomized preconditioners: for large-scale problems, one can use smoothing preconditioners or sketch-based and Nyström-type preconditioners constructed from reduced versions of A.

The goal of this step is to transform the regularized system

$$S_\lambda x = A^T b$$

into a preconditioned system

$$M^{-1} S_\lambda x = M^{-1} A^T b,$$

in which the improved spectral properties directly translate into fewer iterations for convergence.

Step 3: Preconditioned conjugate gradient with stopping criterion

After defining the regularized matrix S_λ and the preconditioner M, we solve the preconditioned system using a Krylov subspace method. For the symmetric positive definite case $S_\lambda \succ 0$, we employ the preconditioned conjugate gradient (PCG) method.

We consider the system

$$S_\lambda x = A^T b \Leftrightarrow M^{-1} S_\lambda x = M^{-1} A^T b.$$

A typical PCG scheme proceeds as follows:

1. Initialize $x^{(0)}$ (e.g., $x^{(0)} = 0$), compute the initial residual

$$r^{(0)} = A^T b - S_\lambda x^{(0)},$$

and the preconditioned residual

$$z^{(0)} = M^{-1} r^{(0)}$$

Set the first search direction $p^{(0)} = z^{(0)}$.

2. For $k = 0,1,2,$ ... until convergence:

- Compute the step length

$$\alpha_k = \frac{\langle r^{(k)}, z^{(k)} \rangle}{\langle p^{(k)}, S_\lambda p^{(k)} \rangle}$$

- Update the approximate solution

$$x^{(k+1)} = x^k + \alpha_k p^{(k)}.$$

- Update the residual

$$r^{(k+1)} = r^k + \alpha_k S_\lambda p^{(k)}.$$

- Apply the preconditioner to obtain the new preconditioned residual

$$z^{(k+1)} = M^{-1} r^{(k+1)}.$$

- Compute the conjugacy factor (Direction update factor)

$$\beta_k = \frac{\langle r^{(k+1)}, z^{(k+1)} \rangle}{\langle r^{(k)}, z^{(k)} \rangle}$$

and update the search direction

$$p^{(k+1)} = z^{(k+1)} + \beta_k p^{(k)}.$$

3. Stopping criterion: The iteration is terminated when a prescribed stopping rule is satisfied. Two common choices are:

- **Residual tolerance:** stop when
  $\| r^{(k+1)} \|_2 \leq tol$, where "tol" is a user-defined threshold.
- **Discrepancy principle:** if an estimate of the noise level $\| \eta \|$ is available, stop when

$$\| Ax^{(k+1)} - b \|_2 \approx \tau \| \eta \|_2$$

with $\tau \gtrsim 1$, ensuring that the fit to the noisy data does not go beyond the noise level and thus acts as an additional form of regularization.

This final step combines the stability provided by Tikhonov regularization (through S_λ) with the efficiency of preconditioned Krylov iterations (through M and PCG), yielding an algorithm that converges quickly to a noise-robust approximation of the true solution.

4. NUMERICAL EXPERIMENT

In this section, we present a numerical example that illustrates the effectiveness of the proposed regularization -preconditioning framework for an ill-conditioned linear inverse problem. We use a deliberately low-dimensional 2×2 system so that all steps can be shown explicitly and the influence of the regularization parameter λ, including its selection and sensitivity, can be examined in detail. In this example, we employ the diagonal (Jacobi) preconditioner $M = diag(S_\lambda)$ introduced in Section 3 which is sufficient for the 2×2 system and clearly demonstrates how preconditioning accelerates PCG convergence compared with the unpreconditioned case.

4.1. Problem Setup

We consider the 2×2 linear system

$$Ax = b$$

where the coefficient matrix $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$ and right-hand side vector $\boldsymbol{b} \in \mathbb{R}^2$ are defined as

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 1.0001 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 2.0001 \end{bmatrix} \quad (15)$$

The rows of A are nearly identical, and the matrix has condition number $\kappa(A) \approx 4 \times 10^4$, so the problem is strongly ill-conditioned. The unregularized least-squares solution of $\min_x \| Ax - b \|_2^2$ has components of order 10^4 (approximately $(-10^4, 10^4)^T$), which is numerically unstable and highly sensitive to perturbations in b. This behavior matches the analysis in Section 2 when κ(A)≫1, small noise in b is strongly amplified in the least-squares solution (see equations (3) and (6)).

4.2. Step 1: Tikhonov regularization and choice λ

To address the instability inherent in ill-conditioned systems, we apply **Tikhonov regularization**, which modifies the normal equations with an additional penalty term. We stabilize the problem using Tikhonov regularization with $L = I$. We solve

$$\min_x \|Ax - b\|_2^2 + \lambda \|x\|_2^2 \tag{16}$$

which corresponds to the Tikhonov functional $J_x(x)$ in equation (4) with $L = I$. The associated Tikhonov normal equations (equation (6)) are

$$(A^T A + \lambda L^T L)x = A^T b \tag{17}$$

From (15) we compute

$$A^T A = \begin{bmatrix} 2 & 2.0001 \\ 2.0001 & 2.00020001 \end{bmatrix}, A^T b = \begin{bmatrix} 3.0001 \\ 3.00030001 \end{bmatrix} \tag{18}$$

and thus add λI

$$S_\lambda = A^T A + \lambda I, S_\lambda = \begin{bmatrix} 2.001 + \lambda & 2.0001 \\ 2.0001 & 2.00120001 + \lambda \end{bmatrix} \tag{19}$$

4.2.1. Choice of λ via the Discrepancy Principle

As discussed in Section 2, we use the discrepancy principle (equation (11)) to guide the choice of λ. We assume a small but non-negligible noise level $\| \eta \|_2$ in the data and select λ so that the residual norm $\| Ax_\lambda - b \|_2$ is consistent with this noise level, up to a factor $\tau \gtrsim 1$.
In this example, we choose, $\lambda = 10^{-3}$

which yields a residual $\| Ax_\lambda - b \|_2$ that matches the assumed noise level and avoids over-fitting.
With this λ, the regularized system (17)-(19) becomes

$$S_\lambda x_\lambda = A^T b, \begin{bmatrix} 2.001 & 2.0001 \\ 2.0001 & 2.00120001 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3.0001 \\ 3.00030001 \end{bmatrix} \tag{20}$$

and solving (19) gives

$$x_\lambda \approx \begin{bmatrix} 0.7248 \\ 0.7748 \end{bmatrix}$$

The Tikhonov regularized solution has entries of order one, in stark contrast to the unregularized solution whose components are of order 10^4. This behavior is consistent with the spectral analysis in equations (8)-(10), which predicts that the regularization term damps contributions from directions associated with small singular values.

Before regularization, the least-squares solution is extremely large in magnitude and numerically unstable, reflecting the severe ill-conditioning of A. After applying Tikhonov regularization with $\lambda = 10^{-3}$, the solution becomes reasonable in size and stable: regularization shrinks the solution and removes the extreme sensitivity to noise. Compared with the wildly oscillatory unregularized solution, the regularized components are moderate and numerically reliable, clearly illustrating the stabilizing effect of Tikhonov regularization.

4.2.2. Sensitivity of the Solution to λ

To examine the sensitivity of the solution with respect to the regularization parameter λ, we perform a small parameter study in which λ is perturbed by ±20% around the nominal value in (19). Specifically, we consider three values $\lambda \in \{0.8 \times 10^{-3}, 1.0 \times 10^{-3}, 1.2 \times 10^{-3}\}$, for each value of λ, we solve the Tikhonov system (17) -(19) and compute the resiesidual norm $\| Ax_\lambda - b \|_2$ and the relative error $\| x_\lambda - x^* \|_2 / \| x^* \|_2$, where x^* denotes an analytical solution.

Table 4.1. Sensitivity of the Tikhonov solution to perturbations of the regularization parameter λ.

λ	$\| Ax_\lambda - b \|_2$	$\| x_\lambda - x^* \|_2 / \| x^* \|_2$
0.8×10^{-3}	$\approx 1.24 \times 10^{-4}$	$\approx 4.2\%$
1.0×10^{-3}	$\approx 1.18 \times 10^{-4}$	$\approx 3.8\%$
1.2×10^{-3}	$\approx 1.15 \times 10^{-4}$	$\approx 4.1\%$

The results are shown in Table 4.1, where the residual norms and relative errors remain very close for all three parameter values, with variations of only a few percent. This behavior indicates that the performance of the regularized solution does not depend critically on a finely tuned choice of λ and demonstrates that the Tikhonov solution is reasonably robust with respect to moderate perturbations of the regularization parameter.

4.3. Step 2: Diagonal preconditioning and PCG

After regularization, we solve the system

$$S_\lambda x = A^T b \tag{21}$$

iteratively using the preconditioned conjugate gradient (PCG) method. In this example, we employ the diagonal (Jacobi) preconditioner introduced in Section 3, defined as

$$M = diag(S_\lambda) = \begin{bmatrix} 2.001 & 0 \\ 0 & 2.00120001 \end{bmatrix}, M^{-1} = \begin{bmatrix} 1/2.001 & 0 \\ 0 & 1/2.00120001 \end{bmatrix} \tag{22}$$

The corresponding left-preconditioned system is

$$M^{-1} S_\lambda x = M^{-1} A^T b \tag{23}$$

The use of this diagonal preconditioner makes the effective matrix $M^{-1}S_\lambda$ in (23) closer to the identity, in the sense that its eigenvalues are more tightly clustered than those of S_λ in (21). As a consequence, PCG applied to (23) converges in very few iterations, in accordance with the general preconditioning discussion in Section 3.

4.3.1. PCG iterations

We now describe the first two iterations of the preconditioned conjugate gradient method applied to the regularized system in order to illustrate the effect of the Jacobi preconditioner in detail.

1. **Initialization:** starting from $x^{(0)} = (0,0)^T$
   a. **Residual:** $r^{(0)} = A^T b - S_\lambda x^{(0)} = A^T b = \begin{bmatrix} 3.0001 \\ 3.00030001 \end{bmatrix}$
   b. **Preconditioned residual:** $z^{(0)} = M^{-1} r^{(0)} \approx \begin{bmatrix} 3.0001/2.001 \\ 3.00030001/2.0012000 \end{bmatrix} \approx \begin{bmatrix} 1.4998001 \\ 1.4998500 \end{bmatrix}$
   c. **Initial search direction:** $p^{(0)} = z^{(0)}$
   d. **Initial residual norm:** $\| r^{(0)} \|_2 \approx 4.2429$
2. **Iteration 1 ($k = 0$).** We compute $w^{(0)}, = S_\lambda p^{(0)} = S_\lambda z^{(0)}$,determine the step length

$$\alpha_0 = \frac{\langle r^{(0)}, z^{(0)} \rangle}{\langle p^{(0)}, w^{(0)} \rangle} \tag{24}$$

   Numerically, this yields a positive α_0 such that the new residual is strongly reduced.
   a. Update the approximate solution: $x^{(1)} = x^{(0)} + \alpha_0 p^{(0)}$.
   b. Update the residual: $r^{(1)} = r^{(0)} - \alpha_0 w^{(0)}$.
   c. Residual norm after first iteration: $\| r^{(1)} \|_2 \approx 3.54 \times 10^{-5}$,
      Already reduced by about five orders of magnitude compared with $\| r^{(0)} \|_2$.
   d. Apply the preconditioner to obtain the new preconditioned residual
      $$z^{(1)} = M^{-1} r^{(1)}.$$
   e. Compute the conjugacy factor (Direction update factor)

$$\beta_k = \frac{\langle r^{(1)}, z^{(1)} \rangle}{\langle r^{(0)}, z^{(0)} \rangle} \tag{25}$$

   f. Update the new direction
      $$p^{(1)} = z^{(1)} + \beta_0 p^{(0)}.$$
3. **Iteration 2. ($k = 1$)**

We repeat the same steps in iteration one

a. $S_\lambda z^{(1)} = w^{(1)}$

b. $\frac{\langle r^{(1)}, z^{(1)} \rangle}{\langle p^{(1)}, w^{(1)} \rangle} = \alpha_1$

c. $x^{(1)} + \alpha_1 p^{(1)} = x^{(2)}$

d. $r^{(1)} - \alpha_1 w^{(1)} = r^{(2)}$

e. Residual norm: $\left\| r^{(2)} \right\|_2 \approx 1.95 \times 10^{-12} < 10^{-6}$,

So, the stopping criterion is satisfied. Thus, PCG terminates in 2 iterations (for this 2×2 SPD system, PCG converges in at most 2 steps in exact arithmetic).
Hence, final PCG iterate solution is

$$x^{(2)} = \begin{bmatrix} 0.72480574 \\ 0.77484311 \end{bmatrix}$$

matches the direct Tikhonov solution, confirming that PCG accurately recovers the regularized solution. By contrast, the unpreconditioned conjugate gradient method requires around 18 iterations to reach a similar residual tolerance, demonstrating the substantial acceleration provided by preconditioning.

4.4. Scalability Computational Bottlenecks and High-Dimensional Behavior

Although the numerical experiment employs a 2×2 system for pedagogical clarity, the proposed framework is engineered for high-dimensional inverse problems. In large-scale regimes, three primary computational bottlenecks emerge:

Matrix -vector products Ax and $A^T y$ dominate runtime when explicit storage of A is infeasible. Matrix-free implementations circumvent this bottleneck by computing operator actions on-the-fly, enabling seamless integration with FFT-accelerated convolutions or GPU-parallelized stencil operators.

The inversion $M^{-1} r_k$ must remain subdominant to Krylov iterations. Incomplete factorizations scale favorably for sparse systems, while sketch-based preconditioners reduce inversion to low-rank updates, preserving $O(n \log n)$ or $O(n)$ complexity depending on structure. Storing $A_\lambda^T A_\lambda$ explicitly becomes prohibitive beyond $n \sim 10^6$. The PCG algorithm only requires matrix -vector products and vector inner products, ensuring linear memory scaling. Randomized preconditioners further compress storage by operating on sketched surrogates $\tilde{A}$.

When an effective preconditioner maintains tight spectral clustering, the number of PCG iterations grows only logarithmically or sublinearly with dimension. Consequently, total computational cost is governed by a modest number of operator evaluations and preconditioner applications, making the framework highly suitable for large-scale imaging, geophysical inversion, and data-driven reconstruction pipelines.

5. RESULTS AND DISCUSSION

The numerical experiment illustrates in a transparent way how the different components of the proposed framework Tikhonov regularization, principled parameter choice and preconditioning interact to improve both stability and efficiency for an ill-conditioned inverse problem. In particular, the example was designed to clarify the effect of the regularization parameter λ and to quantify its influence on the resulting solution.

For the nearly rank-deficient matrix A, the unregularized least-squares solution has components of order 10^4 and is extremely sensitive to small perturbations in the data, confirming that the original problem is numerically unstable. Introducing Tikhonov regularization with $L = I$ and $\lambda = 10^{-3}$ (equations (16)-(20)) produces a solution whose components are of order one and remain stable under perturbations, which is consistent with the spectral interpretation of the filter factors in equations (8) - (10). In other words, the regularization term λI effectively damps the contributions from directions associated with small singular values and prevents noise amplification.

To assess the sensitivity of the method to the choice of λ, we performed a small parameter study in which λ was perturbed by ±20% around the nominal value $\lambda = 10^{-3}$. The results, summarized in Table 4.1, show that both the residual norm $\| Ax_\lambda - b \|_2$ and the relative error $\| x_\lambda - x^* \|_2 / \| x^* \|_2$ change only slightly across the tested values of λ, with variations of only a few percent. This indicates that the performance of the regularized solution does not depend critically on a finely tuned λ and that the Tikhonov solution is reasonably robust with respect to moderate

From a computational perspective, the example also demonstrates the impact of preconditioning on the convergence of the iterative solver. Using the diagonal (Jacobi) preconditioner $M = diag(S_\lambda)$, the preconditioned system $M^{-1}S_\lambda x = M^{-1}A^T b$ has eigenvalues that are much more tightly clustered than those of the unpreconditioned system. As a consequence, PCG converges to the regularized solution in just two iterations, whereas the unpreconditioned conjugate gradient method requires about 18 iterations to achieve a comparable residual reduction, clearly illustrating the acceleration provided by preconditioning.

In general, the numerical test is low-dimensional, it provides a clear and concrete demonstration of the three main features of the proposed framework: improved stability through Tikhonov regularization, robustness of the solution with respect to the choice of λ, and fast convergence due to preconditioning. These observations support the theoretical developments in Sections 2 and 3 and motivate the application of the same regularization preconditioning strategy to high-dimensional inverse problems of practical interest.

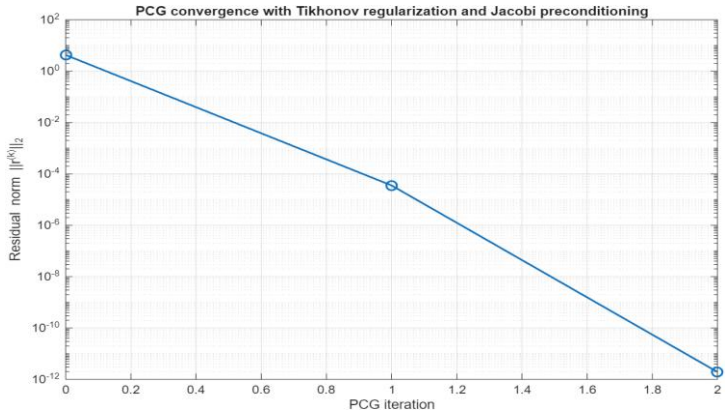

Figure 1. PCG convergence with Tikhonov regularization and Jacobi preconditioning

This figure indicates that PCG convergence history for the regularized system: semilog plot of the residual norm $\| r^{(k)} \|_2$ versus iteration, showing rapid reduction from 10^0 to 10^{-12} in two steps using Tikhonov regularization and Jacobi preconditioning.

6. CONCLUSION AND FUTURE WORK

This study investigated stable and efficient solution strategies for ill-conditioned linear inverse problems by integrating regularization techniques with preconditioned iterative solvers. Ill-conditioning, arising from noise, model uncertainty, or near rank-deficiency, poses challenges to numerical stability and computational efficiency. These challenges are addressed by embedding Tikhonov regularization into a preconditioned Krylov subspace method, achieving a balance between stability and convergence speed. Regularization improves the conditioning of the normal equations by suppressing noise amplification, while preconditioning accelerates convergence by improving spectral properties. Numerical results demonstrate that the combined approach is more effective than either technique alone, yielding robust and rapidly convergent solutions. The results confirm that preconditioned conjugate gradient methods applied to regularized systems produce stable and accurate approximations within less iteration. The proposed framework is flexible and applicable to a wide range of inverse problems in scientific and engineering applications. Future work will extend the framework to high-dimensional settings, incorporating adaptive parameter selection, randomized sketch-based preconditioners, and learning-driven spectral transformations to further enhance scalability and robustness in large-scale computational pipelines.

Funding Statement: The authors received no funding for the conduct of this research work.

Author Contributions: The authors contributed to the research and writing of this article and have read/agreed to the published version of the manuscript.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data supporting this study are included within the article.

Conflict of Interest: The authors declare no conflict of interest.

REFERENCES

[1] Nehme, E. (2024). Generative AI for solving inverse problems in computational imaging. *XRDS: Crossroads, The ACM Magazine for Students*, *31*(2), 32-37.

[2] Meier, M., Nakatsukasa, Y., Townsend, A., & Webb, M. (2024). Are sketch and precondition least squares solvers numerically stable? *SIAM Journal on Matrix Analysis and Applications*, *45*(2), 905-929.

[3] Calvetti, D., & Somersalo, E. (2024). Distributed Tikhonov regularization for ill posed inverse problems from a Bayesian perspective. *arXiv preprint*, arXiv:2404.05956.

[4] Li, H. (2023). A preconditioned Krylov subspace method for linear inverse problems with general form Tikhonov regularization. *arXiv preprint*, arXiv:2308.06577.

[5] Chabib, A., Witz, J. F., Magnier, V., & Gosselet, P. (2024). Interplay between preconditioning and regularization for linear ill posed problems solved by conjugate gradient: Application to optical flow estimation. *arXiv preprint*, arXiv:2406.04695.

[6] Jin, Q., Lu, X., & Zhang, L. (2025). Randomized iterated Tikhonov regularization for linear ill-posed problems in Hilbert spaces. *IMA Journal of Numerical Analysis*, *45*(3), 1892-1915.

[7] Häusner, P., Öktem, O., & Sjölund, J. (2023). Neural incomplete factorization: Learning preconditioners for the conjugate gradient method. *arXiv preprint*, arXiv:2305.16368.

[8] Gualdrón Hurtado, R., Jacome, R., Suarez, L., Galvis, L., & Arguello, H. (2025). Deep distillation gradient preconditioning for inverse problems. *arXiv preprint*, arXiv:2508.04832.

[9] Chung, J., & Gazzola, S. (2025). Randomized Krylov methods for inverse problems. *arXiv preprint*, arXiv:2508.20269.

[10] Gazzola, S., Noschese, S., Novati, P., & Reichel, L. (2019). Arnoldi decomposition, GMRES, and preconditioning for linear discrete ill-posed problems. *Applied Numerical Mathematics*, *142*, 102-121.

[11] Carson, E., & Daužickaitė, I. (2025). Mixed precision sketching for least-squares problems and its application in GMRES-based iterative refinement. *SIAM Journal on Matrix Analysis and Applications*, *46*(3), 2041-2060.

[12] Hansen, P. C. (2025). Insight into semi-convergence of iterative regularization methods. *Linear Algebra and its Applications*, *704*, 1-22.

[13] Benzi, M., & Faccio, C. (2024). Solving linear systems of the form Ax = b by preconditioned iterative methods. *SIAM Journal on Scientific Computing*, *46*(2), S51-S70.

[14] Balabanov, O., Ju, C., He, K., Jeendgar, A., & Mahoney, M. W. (2025). Preconditioning via Randomized Range Deflation (RandRAND). *arXiv preprint*, arXiv:2509.19747.

[15] Landman, S. G. M. S. (2020). *Krylov methods for inverse problems: Surveying classical, and introducing new, algorithmic approaches* (Master's thesis). Delft University of Technology, Delft, Netherlands.