SPECTRAL PRECONDITIONING FROM PREDICTED INVARIANT SUBSPACES

Anonymous authors
Paper under double-blind review

ABSTRACT

Krylov subspace methods are indispensable for solving large sparse linear systems, but their cost can be dominated by slowly converging spectral components. Preconditioning is widely used to accelerate Krylov subspace methods, and several preconditioners have been implemented in scientific computing packages. Neural Krylov Iteration (NeurKItt) showed that a neural operator can amortize the prediction of an invariant subspace and use it to augment a Krylov process for PDE-induced linear systems. The key idea behind NeurKItt is to address the slowly converging components separately. We argue that the invariant subspace can also define a problem-specific preconditioner. Using the invariant subspace predicted by NeurKItt, we define a symmetric positive definite (SPD), low-rank structured spectral preconditioner. The construction requires only matrix–vector products and dense algebra at the dimension of the predicted subspace. It clusters the spectrum of the preconditioned system and thereby reduces the number of Krylov iterations in methods such as GMRES. Theoretical analysis and experiments demonstrate the feasibility and effectiveness of the learned preconditioner.

1 INTRODUCTION

Large sparse linear systems are a central computational bottleneck in scientific computing, with applications across science and engineering. Discretizing a partial differential equation (PDE) often requires repeated solution of

$$\boldsymbol{A}(a)\boldsymbol{x} = \boldsymbol{b}, \tag{1}$$

where a denotes the spatially varying PDE coefficient, $\boldsymbol{x}$ contains the discrete solution values, and $\boldsymbol{b}$ contains the forcing term together with boundary contributions. Krylov methods seek an approximate solution in a low-dimensional subspace, typically by minimizing a residual or an energy norm, which makes them attractive at scale (Saad, 2003; Liesen & Strakoš, 2013). They expand the subspace until the requested accuracy is reached.

However, convergence can be slow when the matrix is ill-conditioned or has unfavorable spectral clusters (Greenbaum, 1997; Chen, 2005). Two complementary approaches address this problem. Preconditioning replaces the original system with a spectrally more favorable one. Classical choices include Jacobi, incomplete factorizations, additive Schwarz, and multigrid methods (Saad, 2003; Zhang, 1992; Toselli & Widlund, 2004). Deflation and recycling (Parks et al., 2006; Burke et al., 2025) instead explicitly remove convergence-limiting components by retaining approximate invariant-subspace information across a sequence of related systems. Such sequences arise, for example, in time-dependent simulations and variational data assimilation.

With the development of deep learning, supervised methods have been used to learn both preconditioners and invariant subspaces for deflation. Learned methods have supplied initial guesses, solver parameters, approximate factorizations, multigrid components, and Krylov search information (Hsieh et al., 2019; Greenfeld et al., 2019; Luz et al., 2020; Li et al., 2023). Many learned preconditioners optimize an approximate inverse under a prescribed structure. More recently, Neural Krylov Iteration (NeurKItt) was proposed to accelerate PDE-induced linear systems. Motivated by viewing the map from a PDE coefficient field to an invariant subspace as an operator between function spaces, NeurKItt uses a neural operator (Li et al., 2020a;b; Kovachki et al., 2021b) to predict a low-dimensional invariant subspace. It orthonormalizes the prediction with a thin QR factorization and augments GMRES (Generalized Minimal Residual) and GCRODR (Generalized Conjugate

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Residual with Deflated Restarting) with the prediction (Luo et al., 2024). The convergence-limiting subspace can be a more useful and reusable prediction target than a low-accuracy neural approximation of the PDE solution.

Because preconditioning and recycling share the same acceleration target, this paper asks: *can the predicted subspace itself define a robust preconditioner?* We answer affirmatively for this problem. Our method, called NEURSP, is a neural spectral preconditioner that combines NeurKItt-style subspace prediction with a symmetric low-rank structure. Thus, the preconditioner reuses the invariant subspace already predicted by NeurKItt.

Our contributions include:

- We turn a NeurKItt invariant-subspace prediction $\boldsymbol{S} \in \mathbb{R}^{n\times k}$ into an SPD preconditioner without requiring an additional learned model.
- We characterize the corrected spectrum for an exact invariant subspace and prove positive definiteness for any full-rank predicted basis. The formulation interpolates between the identity and the exact inverse.
- We derive exact- and approximate-subspace formulations, showing how the Rayleigh–Ritz matrix maps selected spectral components toward one.
- We evaluate the resulting preconditioner on three PDE-induced families of linear systems within GMRES.

2 RELATED WORK

Krylov recycling, augmentation, and deflation. Augmented and recycled Krylov methods retain an approximate invariant subspace and remove slow spectral components from later solves (Morgan, 2002; Parks et al., 2006; Gaul, 2014). Their benefit is governed by principal angles between the retained and target subspaces (Beattie et al., 2004; Kilmer & De Sturler, 2006). Recycling is inexpensive when consecutive systems have aligned eigenspaces, but a stored basis may degrade as parameters or meshes change. NeurKItt replaces direct reuse by an amortized neural prediction (Luo et al., 2024). We preserve that prediction module but use its output to modify the spectrum through preconditioning rather than to modify the Arnoldi relation through recycling.

Classical and learned preconditioning. Sparse approximate inverses and incomplete factorizations approximate $\boldsymbol{A}^{-1}$ under structural constraints (Benzi et al., 1996; Lin & Moré, 1999). Domain decomposition and multigrid target complementary scales (Toselli & Widlund, 2004; Azulay & Treister, 2022). Learned preconditioners have optimized sparse patterns, factorizations, smoothers, and transfer operators (Götz & Anzt, 2018; Stanaityte, 2020; Greenfeld et al., 2019; Luz et al., 2020; Taghibakhshi et al., 2021). These approaches usually learn matrix entries or graph-local operations. In contrast, NEURSP learns only a compact spectral subspace; a deterministic algebraic layer turns it into a valid SPD operator.

Neural operators for numerical acceleration. Fourier neural operators (FNOs) parameterize global convolution kernels in the Fourier domain and have been effective for parametric PDEs (Li et al., 2020a; Kovachki et al., 2021a). Yet a neural surrogate generally does not satisfy a stringent linear-solver tolerance. Hybrid methods instead use learning to accelerate a conventional solver, which continues until the requested residual is achieved (Luna et al., 2021; Zhang et al., 2022). Our method follows this hybrid design: the network proposes spectral information, while the iterative solver certifies the final algebraic accuracy.

3 BACKGROUND

Krylov methods exhibit different convergence behavior for different matrix classes. We summarize standard polynomial convergence bounds for symmetric systems. GMRES is a typical choice in this setting.

The mth Krylov space generated from $\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$ is

$$\mathcal{K}_m(\boldsymbol{A}, \boldsymbol{r}_0) = \operatorname{span}\{\boldsymbol{r}_0, \boldsymbol{A}\boldsymbol{r}_0, \ldots, \boldsymbol{A}^{m-1}\boldsymbol{r}_0\}. \tag{2}$$

The residual satisfies

$$\boldsymbol{r}_m = p_m(\boldsymbol{A})\boldsymbol{r}_0, p_m(0) = 1, \tag{3}$$

hence we have the convergence upper bound:

$$\|\boldsymbol{r}_m\|_2 \leq \min_{p\in\Pi_m, p(0)=1} \max_{\lambda\in\sigma(\boldsymbol{A})} |p(\lambda)|\|\boldsymbol{r}_0\|_2, \tag{4}$$

where Π_m denotes the set of polynomials of degree at most m and $\sigma(\boldsymbol{A})$ denotes the spectrum of $\boldsymbol{A}$.

Positive definite matrix. When $\boldsymbol{A}$ is SPD, the upper bound can be expressed in terms of its condition number. In this case,

$$\|\boldsymbol{r}_m\|_2 \leq \min_{p\in\Pi_m, p(0)=1} \max_{\lambda\in[a,b]} |p(\lambda)|\|\boldsymbol{r}_0\|_2, \tag{5}$$

where $a = \lambda_{\min}(\boldsymbol{A})$ and $b = \lambda_{\max}(\boldsymbol{A})$. Mapping the interval $[a, b]$ to $[-1, 1]$ and choosing

$$p_m(\lambda) = \frac{T_m(\frac{a+b-2\lambda}{b-a})}{T_m(\frac{a+b}{b-a})} \tag{6}$$

with $T_m(t)$ being the Chebyshev polynomial of the first kind. Therefore, we have

$$\min_{p\in\Pi_m, p(0)=1} \max_{\lambda\in[a,b]} |p(\lambda)| \leq \frac{1}{|T_m(\frac{a+b}{b-a})|} \leq 2\left(\frac{\sqrt{\kappa(\boldsymbol{A})}-1}{\sqrt{\kappa(\boldsymbol{A})}+1}\right)^m. \tag{7}$$

Symmetric-indefinite matrix. Assume that $\boldsymbol{A} = \boldsymbol{A}^T$ and its spectrum is contained in two symmetric intervals, $\sigma(\boldsymbol{A}) \subset [-b, -a] \cup [a, b]$, where $0 < a < b$. The general bound gives

$$\|\boldsymbol{r}_m\|_2 \leq \min_{p\in\Pi_m, p(0)=1} \max_{\lambda\in[-b,-a]\cup[a,b]} |p(\lambda)|\|\boldsymbol{r}_0\|_2. \tag{8}$$

Since the spectral set is symmetric, we construct a polynomial satisfying $p(-\lambda) = p(\lambda)$. An even polynomial of degree $2j$ can be written as $p_{2j}(\lambda) = q_j(\lambda^2)$. Since $a \leq |\lambda| \leq b$, we have

$$\max_{\lambda\in[-b,-a]\cup[a,b]} |p_{2j}(\lambda)| = \max_{\mu\in[a^2,b^2]} |q_j(\mu)|. \tag{9}$$

Similar to the SPD case, we have

$$\frac{\|\boldsymbol{r}_{2j}\|_2}{\|\boldsymbol{r}_0\|_2} \leq \frac{1}{T_j(\frac{a^2+b^2}{b^2-a^2})} \leq 2(\frac{b-a}{b+a})^j. \tag{10}$$

These bounds show that moving eigenvalues away from zero and clustering the positive and negative spectra can accelerate Krylov methods.

Remark 1. The convergence of a Krylov subspace method is governed by polynomial approximation over the spectrum, so the condition number is not a universal predictor. For SPD matrices, the condition number $\kappa(\boldsymbol{A})$ in (7) controls a worst-case convergence bound. For symmetric indefinite matrices, spectral gaps and clustering also matter. Consequently, the condition number alone is not a reliable measure of solver performance. However, it can indicate convergence behavior to some extent, especially when the polynomial approximation is complicated.

4 Neural Spectral Preconditioning

We now develop a spectral preconditioner for SPD systems. This assumption is essential to the analysis below, rather than being without loss of generality. For symmetric indefinite matrices, the heuristic preconditioner can also be applied.

4.1 SPECTRAL PRECONDITIONING

4.1.1 POSITIVE DEFINITE CASE

Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ be symmetric positive definite. The worst-case convergence bound motivates an invertible preconditioner $\boldsymbol{P}$ satisfying $\kappa(\boldsymbol{PA}) < \kappa(\boldsymbol{A})$. Left preconditioning gives the transformed system

$$\boldsymbol{PAx} = \boldsymbol{Pb}. \tag{11}$$

Suppose that $\boldsymbol{V} \in \mathbb{R}^{n\times k}$, $1 \le k < n$, has orthonormal columns and approximates an invariant subspace of $\boldsymbol{A}$. We consider the preconditioner

$$\boldsymbol{P} = \alpha\boldsymbol{I} + \boldsymbol{V}\left((\boldsymbol{V}^T\boldsymbol{AV})^{-1} - \alpha\boldsymbol{I}\right)\boldsymbol{V}^T, \qquad \alpha > 0. \tag{12}$$

Here, $\boldsymbol{V}^T\boldsymbol{AV}$ is the Rayleigh–Ritz matrix associated with the same approximate subspace $\boldsymbol{V}$. Because $\boldsymbol{A} \succ 0$ and $\boldsymbol{V}$ has orthonormal columns, $\boldsymbol{P} \succ 0$. Moreover, $\boldsymbol{P} = \boldsymbol{A}^{-1}$ when $k = n$.

The preconditioner $\boldsymbol{P}$ is defined for any matrix $\boldsymbol{V}$ with orthonormal columns. To show its potential to improve the condition number, we first suppose that $\boldsymbol{V} = [\boldsymbol{v}_1, \dots, \boldsymbol{v}_k]$ contains exact eigenvectors associated with the k smallest eigenvalues. Then $\boldsymbol{V}^T\boldsymbol{AV} = \boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \dots, \lambda_k)$, where $0 < \lambda_1 < \lambda_2 < \cdots < \lambda_n$, and $\kappa(\boldsymbol{A}) = \lambda_n/\lambda_1$.

The following theorem characterizes the spectrum of the preconditioned system and gives a sufficient choice of the scaling parameter. The proof is deferred to the appendix.

Theorem 2 (Exact eigenvectors)**.** *Let $\boldsymbol{A} = \boldsymbol{A}^T \succ 0$ have eigenvalues $0 < \lambda_1 < \cdots < \lambda_n$, and let $\boldsymbol{P}$ be defined by*

$$\boldsymbol{P} = \alpha\boldsymbol{I} + \sum_{i=1}^{k}\left(\frac{1}{\lambda_i} - \alpha\right)\boldsymbol{v}_i\boldsymbol{v}_i^T. \tag{13}$$

Then the set of eigenvalues is

$$\sigma(\boldsymbol{PA}) = \left\{\underbrace{1, \dots, 1}_{k \text{ times}}, \alpha\lambda_{k+1}, \dots, \alpha\lambda_n\right\}. \tag{14}$$

Consequently,

$$\kappa(\boldsymbol{PA}) = \frac{\max\{1, \alpha\lambda_n\}}{\min\{1, \alpha\lambda_{k+1}\}}. \tag{15}$$

In particular, if

$$\frac{1}{\lambda_n} \le \alpha \le \frac{1}{\lambda_{k+1}}, \tag{16}$$

then

$$\kappa(\boldsymbol{PA}) = \frac{\lambda_n}{\lambda_{k+1}} < \frac{\lambda_n}{\lambda_1} = \kappa(\boldsymbol{A}). \tag{17}$$

Moreover,

$$\min_{\alpha>0}\kappa(\boldsymbol{PA}) = \frac{\lambda_n}{\lambda_{k+1}}, \tag{18}$$

and every α satisfying (16) is a minimizer.

Remark 3 (Preconditioning with eigenvectors of the largest eigenvalues). The analysis above uses eigenvectors associated with the k smallest eigenvalues. A spectral preconditioner can instead be constructed from the eigenvectors associated with the largest eigenvalues and can likewise improve the condition number with an appropriate scaling. Following the NeurKItt analysis (Luo et al., 2024), we focus on the invariant subspace associated with the smallest eigenvalues because it is minimally affected when changes in a matrix occur in the subspace corresponding to the larger eigenvalues.

The scaling parameter α determines the location of the uncorrected eigenvalues

$$\alpha\lambda_{k+1}, \dots, \alpha\lambda_n \tag{19}$$

relative to the corrected eigenvalue 1. Several choices are possible depending on the desired criterion. Although the optimal scaling differs for a subspace spanned by exact eigenvectors, the difference is negligible for an inexact subspace.

Minimax centering of the uncorrected spectrum. Although (16) contains infinitely many condition-number optimal choices, a unique scaling can be obtained by minimizing the maximum distance of the uncorrected spectrum from one:

$$\min_{\alpha>0} \max_{\lambda\in[\lambda_{k+1},\lambda_n]} |1-\alpha\lambda|. \tag{20}$$

The solution is

$$\alpha_{\mathrm{mm}} = \frac{2}{\lambda_{k+1}+\lambda_n}. \tag{21}$$

When only the extreme eigenvalues of $\boldsymbol{A}$ are available, the related heuristic $2/(\lambda_1+\lambda_n)$ may be used instead.

Trace-based scaling. An alternative choice that does not explicitly require the extreme eigenvalues is

$$\alpha_{\mathrm{tr}} = \frac{n}{\mathrm{tr}(\boldsymbol{A})} = \frac{1}{\bar{\lambda}}, \tag{22}$$

where $\bar{\lambda} = \frac{1}{n}\sum_{i=1}^{n}\lambda_i$ is the arithmetic mean of the eigenvalues. This scaling normalizes the mean eigenvalue of $\alpha\boldsymbol{A}$ to one:

$$\frac{1}{n}\,\mathrm{tr}(\alpha_{\mathrm{tr}}\boldsymbol{A}) = 1. \tag{23}$$

With exact eigenvectors,

$$\sigma(\boldsymbol{P}\boldsymbol{A}) = \left\{1,\ldots,1,\frac{\lambda_{k+1}}{\bar{\lambda}},\ldots,\frac{\lambda_n}{\bar{\lambda}}\right\}. \tag{24}$$

Since $\bar{\lambda}\le\lambda_n$,

$$\kappa(\boldsymbol{P}\boldsymbol{A}) = \frac{\lambda_n}{\min\{\lambda_{k+1},\bar{\lambda}\}} < \kappa(\boldsymbol{A}). \tag{25}$$

Nevertheless, α_{tr} can be attractive for large-scale problems because $\mathrm{tr}(\boldsymbol{A})$ may be easier to compute or estimate than λ_{k+1} and λ_n.

4.1.2 Symmetric-indefinite case

We next characterize the preconditioner's behavior for an approximate invariant subspace. Define the Ritz residual

$$\boldsymbol{R} := \boldsymbol{A}\boldsymbol{V} - \boldsymbol{V}\boldsymbol{H}, \boldsymbol{H} = \boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V}. \tag{26}$$

For an inexact subspace, $\boldsymbol{H}$ need not be diagonal and $\boldsymbol{A}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{H}$ need not hold. By the definition of $\boldsymbol{P}$,

$$\boldsymbol{P}\boldsymbol{V} = \alpha\boldsymbol{V} + \boldsymbol{V}(\boldsymbol{H}^{-1}-\alpha\boldsymbol{I}) = \boldsymbol{V}\boldsymbol{H}^{-1}. \tag{27}$$

Moreover, $\boldsymbol{V}^T\boldsymbol{R} = \boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V} - \boldsymbol{H} = 0$. Hence,

$$\boldsymbol{P}\boldsymbol{A}\boldsymbol{V} = \boldsymbol{P}(\boldsymbol{V}\boldsymbol{H}+\boldsymbol{R}) = \boldsymbol{P}\boldsymbol{V}\boldsymbol{H} + \boldsymbol{P}\boldsymbol{R} = \boldsymbol{V} + \alpha\boldsymbol{R}. \tag{28}$$

Although $\boldsymbol{P}\boldsymbol{A}$ is generally nonsymmetric, it is similar to the symmetric matrix $\boldsymbol{P}^{1/2}\boldsymbol{A}\boldsymbol{P}^{1/2}$. The following result represents this matrix as a simple block-diagonal operator plus a perturbation.

Theorem 4 (Approximate-subspace spectral preconditioning)**.** *Let $\boldsymbol{W}\in\mathbb{R}^{n\times(n-k)}$ be such that $\boldsymbol{Q} = [\boldsymbol{V},\boldsymbol{W}]$ is orthogonal, and set*

$$\boldsymbol{C} := \boldsymbol{W}^T\boldsymbol{A}\boldsymbol{V}, \qquad \boldsymbol{G} := \boldsymbol{W}^T\boldsymbol{A}\boldsymbol{W}. \tag{29}$$

Then $\boldsymbol{P}$ is symmetric positive definite. Moreover, $\boldsymbol{P}\boldsymbol{A}$ is similar to a symmetric positive definite matrix whose spectrum coincides with that of

$$\boldsymbol{\mathcal{B}} = \begin{pmatrix} \boldsymbol{I}_k & \sqrt{\alpha}\,\boldsymbol{H}^{-1/2}\boldsymbol{C}^T \\ \sqrt{\alpha}\,\boldsymbol{C}\boldsymbol{H}^{-1/2} & \alpha\boldsymbol{G}\end{pmatrix}. \tag{30}$$

Define

$$m_0 := \min\{1,\alpha\lambda_{\min}(\boldsymbol{G})\}, \qquad M_0 := \max\{1,\alpha\lambda_{\max}(\boldsymbol{G})\}, \tag{31}$$

and

$$\eta := \sqrt{\alpha}\left\|\boldsymbol{C}\boldsymbol{H}^{-1/2}\right\|_2. \tag{32}$$

If $\eta < m_0$*, then*

$$\lambda_{\min}(\boldsymbol{P}\boldsymbol{A}) \geq m_0 - \eta, \qquad \lambda_{\max}(\boldsymbol{P}\boldsymbol{A}) \leq M_0 + \eta, \tag{33}$$

and consequently

$$\kappa(\boldsymbol{P}\boldsymbol{A}) := \frac{\lambda_{\max}(\boldsymbol{P}\boldsymbol{A})}{\lambda_{\min}(\boldsymbol{P}\boldsymbol{A})} \leq \frac{M_0 + \eta}{m_0 - \eta}. \tag{34}$$

In particular, writing

$$0 < \lambda_1 = \lambda_{\min}(\boldsymbol{A}) \leq \lambda_{\max}(\boldsymbol{A}) = \lambda_n, \tag{35}$$

a sufficient condition for strict improvement over the original condition number is

$$\eta < \frac{\lambda_n m_0 - \lambda_1 M_0}{\lambda_1 + \lambda_n} \tag{36}$$

provided that the right-hand side is positive. Under (36),

$$\kappa(\boldsymbol{P}\boldsymbol{A}) < \kappa(\boldsymbol{A}) = \frac{\lambda_n}{\lambda_1}. \tag{37}$$

Corollary 5. *Under the assumptions of Theorem 4, the perturbation parameter* η *can be bounded directly by the invariant-subspace residual*

$$\boldsymbol{R} := (\boldsymbol{I} - \boldsymbol{V}\boldsymbol{V}^T)\boldsymbol{A}\boldsymbol{V} = \boldsymbol{A}\boldsymbol{V} - \boldsymbol{V}(\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V}). \tag{38}$$

Specifically,

$$\eta \leq \sqrt{\frac{\alpha}{\lambda_{\min}(\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V})}}\left\|(\boldsymbol{I} - \boldsymbol{V}\boldsymbol{V}^T)\boldsymbol{A}\boldsymbol{V}\right\|_2. \tag{39}$$

Hence a sufficiently accurate approximate invariant subspace guarantees preservation of the condition-number improvement achieved by the ideal spectral preconditioner.

Remark 6 (Exact invariant subspace). If $\mathrm{range}(\boldsymbol{V})$ is an exact invariant subspace of $\boldsymbol{A}$, then

$$(\boldsymbol{I} - \boldsymbol{V}\boldsymbol{V}^T)\boldsymbol{A}\boldsymbol{V} = 0, \tag{40}$$

so $\boldsymbol{C} = 0$ and $\eta = 0$. In this case,

$$\boldsymbol{\mathcal{B}} = \begin{pmatrix} \boldsymbol{I}_k & 0 \\ 0 & \alpha\boldsymbol{G} \end{pmatrix}. \tag{41}$$

Thus the selected invariant subspace is mapped exactly to the eigenvalue 1, recovering the standard exact spectral-preconditioning result.

Remark 7 (Interpretation of $\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V}$). The matrix $\boldsymbol{H} = \boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V}$ should not be viewed as introducing an independent eigenvalue error. It is the Rayleigh–Ritz representation of $\boldsymbol{A}$ on the approximate subspace $\mathrm{range}(\boldsymbol{V})$ and is therefore coupled to the subspace approximation. The quantity

$$\boldsymbol{R} = (\boldsymbol{I} - \boldsymbol{V}\boldsymbol{V}^T)\boldsymbol{A}\boldsymbol{V} \tag{42}$$

measures precisely the failure of $\mathrm{range}(\boldsymbol{V})$ to be invariant under $\boldsymbol{A}$.

Heuristic scaling choice For exact eigenvectors, trace-based scaling is attractive because it is easy to compute or estimate. For an approximate subspace, however, the choice of α is more subtle: the sufficient condition above requires

$$0 < \sqrt{\frac{\alpha}{\lambda_{\min}(\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V})}}\left\|(\boldsymbol{I} - \boldsymbol{V}\boldsymbol{V}^T)\boldsymbol{A}\boldsymbol{V}\right\|_2 < \frac{\lambda_n m_0 - \lambda_1 M_0}{\lambda_1 + \lambda_n}.$$

For the SPD case, we can adopt trace-based scaling for the preconditioner. Extending this scaling rule to indefinite systems may be problematic. However, we observe that this heuristic scaling choice performs well for indefinite systems.

5 LEARNING AN INVARIANT SUBSPACE

The preconditioner in Section 4 requires spectral information that would be too expensive to recompute for every member of a parametric PDE family. We therefore learn this information once and amortize the prediction cost over unseen parameter instances. The construction follows the subspace-learning principle of NeurKItt (Luo et al., 2024), but the predicted space is subsequently used to form a spectral preconditioner rather than to augment GMRES with recycling.

5.1 NEURAL-OPERATOR PREDICTOR

Let $a(\mu) \in \mathbb{R}^{d_a}$ denote the coefficient field or other descriptor of a PDE instance, and let $\boldsymbol{A}(\mu) \in \mathbb{R}^{n\times n}$ be the matrix obtained after discretization. The desired map

$$G^\star : a(\mu) \longmapsto \mathcal{V}_k(\boldsymbol{A}(\mu)) \tag{43}$$

takes a function-valued input to the k-dimensional invariant subspace associated with the smallest eigenvalues. Because both its input and output are discretizations of functions, we parameterize this map with a Fourier neural operator (FNO) (Li et al., 2020a). A lifting layer first embeds $a(\mu)$ into c channels; a stack of Fourier layers mixes local and global information; and a final projection produces $\boldsymbol{Z}_\theta(\mu) \in \mathbb{R}^{n\times k}$. Each column of $\boldsymbol{Z}_\theta$ represents one candidate direction, but the columns need not be orthogonal.

A thin QR factorization converts this unconstrained output into an orthonormal basis,

$$\boldsymbol{S}_\theta(\mu) = \mathrm{qr}(\boldsymbol{Z}_\theta(\mu)), \qquad \boldsymbol{S}_\theta^T\boldsymbol{S}_\theta = \boldsymbol{I}_k. \tag{44}$$

This layer also removes irrelevant column scaling. More importantly, it makes the representation depend on the predicted subspace rather than on a particular, non-unique choice of basis.

5.2 A BASIS-INVARIANT TRAINING OBJECTIVE

Training targets are computed offline. For each sampled parameter μ_j, we extract an orthonormal basis $\boldsymbol{V}_k(\mu_j)$ for the eigenspace corresponding to the k smallest eigenvalues of $\boldsymbol{A}(\mu_j)$. Directly regressing the entries of $\boldsymbol{V}_k$ is inappropriate: sign changes and rotations within a repeated or clustered eigenspace describe the same object. We instead compare the orthogonal projectors,

$$\mathcal{L}_{\mathrm{proj}}(\boldsymbol{S}_\theta, \boldsymbol{V}_k) = \frac{1}{k}\|\boldsymbol{S}_\theta\boldsymbol{S}_\theta^T - \boldsymbol{V}_k\boldsymbol{V}_k^T\|_F^2. \tag{45}$$

This loss is unchanged under $\boldsymbol{S}_\theta \mapsto \boldsymbol{S}_\theta\boldsymbol{R}$ or $\boldsymbol{V}_k \mapsto \boldsymbol{V}_k\widehat{\boldsymbol{R}}$ for orthogonal $\boldsymbol{R}$ and $\widehat{\boldsymbol{R}}$. It therefore optimizes the geometric object used by the preconditioner. Its connection to solver quality can be monitored through the largest principal angle

$$\sin\phi_{\max} = \|(\boldsymbol{I} - \boldsymbol{S}_\theta\boldsymbol{S}_\theta^T)\boldsymbol{V}_k\|_2, \tag{46}$$

which vanishes exactly when the predicted and target spaces agree.

Following the alternative implementation in (Luo et al., 2024), we consider the columnwise leakage loss

$$\mathcal{L} = \sum_{i=1}^{k}\|\boldsymbol{v}_i - \boldsymbol{S}_\theta\boldsymbol{S}_\theta^T\boldsymbol{v}_i\|_2. \tag{47}$$

This loss is basis-dependent because it sums the leakage of individual target vectors. It is not equal to the largest-principal-angle loss. Both losses vanish when the predicted and target subspaces coincide, and we monitor the basis-invariant principal angle during evaluation.

Effect of subspace error Let $\boldsymbol{V}_k$ denote the exact target invariant subspace and $\boldsymbol{S}$ the prediction. Their discrepancy is measured by the largest principal angle

$$\sin\phi = \|(\boldsymbol{I} - \boldsymbol{S}\boldsymbol{S}^T)\boldsymbol{V}_k\|_2. \tag{48}$$

When $\phi = 0$, the spectral preconditioner applies exact eigenvectors. For an inexact prediction,

$$\boldsymbol{A}\boldsymbol{S} = \boldsymbol{S}(\boldsymbol{S}^T\boldsymbol{A}\boldsymbol{S}) + (\boldsymbol{I} - \boldsymbol{S}\boldsymbol{S}^T)\boldsymbol{A}\boldsymbol{S}. \tag{49}$$

The second term is the invariant-subspace residual (or leakage) analyzed in (39). Standard invariant-subspace perturbation results imply that the angle remains small when the target eigenvalue cluster is separated and the operator perturbation projected onto that cluster is small (Kilmer & De Sturler, 2006; Stewart & Sun, 1990). Thus the preconditioner continuously approaches ideal correction as the learned subspace improves. Exact individual eigenvectors are unnecessary when the clustered invariant subspace is accurate.

Accuracy and safeguards Because $\boldsymbol{P}$ is SPD for every full-rank $\boldsymbol{S}$, a poor prediction may reduce acceleration but does not change the solution of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. The residual stopping criterion remains authoritative.

Deflation strategy Because deflation and preconditioning are complementary, we also consider combining the proposed spectral preconditioner with deflation. However, the recycling method GCRODR applies the preconditioner to deflated linear systems, whose spectra differ from those of the original systems. Extending our preconditioner to GCRODR remains future work.

6 Numerical Experiments

We evaluate whether the resulting NeurSP preconditioner reduces the GMRES iteration count relative to standard alternatives.

6.1 Data, implementation, and metrics

The benchmark contains three two-dimensional parametric PDE families: Helmholtz, Darcy flow, and heat conduction in a heterogeneous medium (Zhang et al., 2022; Li et al., 2020a; Rahman et al., 2022; Beck et al., 1985; Güttel & Tisseur, 2017). Details of the PDE discretizations and dataset construction are provided in Appendix B. Since our method builds on the invariant-subspace predictor introduced by NeurKItt, we retain its neural-network architecture and hyperparameter settings. The PDE coefficient fields are sampled from Gaussian random fields and discretized to form sparse linear systems. The matrix dimensions are $62{,}500$ for Helmholtz, $90{,}000$ for heat conduction, and $160{,}000$ for Darcy flow. The Darcy and heat datasets use 8,000 training and 1,600 test instances; Helmholtz uses 1,000 and 200, respectively. The FNO has five Fourier layers; the number of retained modes is selected from $\{20, 32, 40\}$, the channel width from $\{32, 50, 64\}$, and the learning rate is 10^{-3}.

We use GMRES without a preconditioner in PETSc 3.19.4 as the baseline method. We report the iteration count required to reach $\|\boldsymbol{r}_m\|_2/\|\boldsymbol{r}_0\|_2 \le \epsilon$. The baseline preconditioners include Jacobi, block Jacobi (BJacobi), SOR, additive Schwarz (ASM), incomplete Cholesky (ICC), and incomplete LU (ILU). Because BJacobi, ICC, and ILU perform similarly on these systems, the main tables report ICC as their representative. We implement the proposed spectral preconditioner using PETSc's `PCShellSetApply` interface. We evaluate predicted invariant subspaces $\boldsymbol{S} \in \mathbb{R}^{n\times k}$ with $k \in \{5, 10, 20\}$ to study the effect of subspace dimension.

Ablation study of the choice of α. We first investigate the effect of α on the iteration count at fixed relative tolerances. On ten Helmholtz systems of dimension $62{,}500$, we compare unpreconditioned GMRES, ICC(0), and our preconditioner constructed from nearly exact eigenvectors, and report the mean iteration count. We denote a configuration by (k, α_i), where $\alpha_1 = 2/(\lambda_1 + \lambda_n)$, $\alpha_2 = 2/(\lambda_{k+1} + \lambda_n)$, and $\alpha_3 = n/\operatorname{tr}(\boldsymbol{A})$. Table 1 shows that all three scaling choices perform similarly and that increasing k further reduces the iteration count.

Table 1: Average GMRES iteration counts with different preconditioners.

tol	none	icc0	$(10, \alpha_1)$	$(10, \alpha_2)$	$(10, \alpha_3)$	$(20, \alpha_1)$	$(20, \alpha_2)$	$(20, \alpha_3)$
10^{-5}	3556.2	378.9	44.8	44.7	44.8	26.0	26.0	26.0
10^{-7}	5008.9	534.1	230.9	230.9	230.9	143.0	143.0	143.0
10^{-10}	7189.4	769.4	543.0	543.0	543.0	338.0	338.0	338.0

6.2 PERFORMANCE ON THE THREE PDE PROBLEMS

Table 2 compares the proposed spectral preconditioner with standard alternatives on all three PDE families. Here, *ours*, *ours10*, and *ours20* use predicted invariant subspaces of dimension $k = 5$, 10, and 20, respectively. Across all nine problem–tolerance combinations, one of our $k = 10$ or $k = 20$ variants attains the lowest iteration count. Compared with the strongest classical baseline, ICC(0), the best variant reduces the iteration count by 86.3%, 68.2%, and 52.0% on Helmholtz; 86.1%, 61.4%, and 38.4% on Heat; and 89.2%, 64.8%, and 42.4% on Darcy, for tolerances 10^{-5}, 10^{-7}, and 10^{-10}, respectively. The gains over unpreconditioned GMRES and Jacobi are substantially larger, particularly for the Heat and Darcy systems.

Increasing the predicted subspace dimension generally strengthens the preconditioner. At the two stricter tolerances, $k = 20$ consistently gives the best result, requiring only 169.7–929.4 iterations despite the larger and more difficult systems. At tolerance 10^{-5}, however, $k = 10$ is slightly better than $k = 20$ for Heat (71.9 versus 77.2) and Darcy (95.2 versus 97.4), indicating that the benefit is not strictly monotone once a sufficiently accurate residual has been reached. The $k = 5$ variant is less robust at stringent tolerances and can underperform ICC(0), whereas $k \geq 10$ preserves a clear advantage in every setting.

Table 2: Average number of GMRES iterations under different preconditioners and tolerances. Best result is shown in **bold**.

Problem	Tol.	none	jacobi	icc0	sor	ours	ours10	ours20
Helmholtz (62500)	10^{-5}	3558.5	3558.5	379.1	519.6	507.3	72.2	**52.0**
	10^{-7}	5012.1	5012.1	534.3	730.8	1138.5	259.0	**169.7**
	10^{-10}	7193.8	7193.8	769.7	1038.7	2076.3	569.4	**369.5**
Heat (90000)	10^{-5}	7352.8	3757.2	519.0	686.0	521.8	**71.9**	77.2
	10^{-7}	10356.0	4768.8	706.4	923.7	1557.2	379.3	**272.6**
	10^{-10}	14863.4	6615.7	964.8	1250.4	3091.0	929.6	**594.6**
Darcy (160000)	10^{-5}	13771.7	6060.9	882.5	1118.5	794.5	**95.2**	97.4
	10^{-7}	19389.6	8435.0	1192.0	1452.5	2782.9	598.8	**419.0**
	10^{-10}	27833.2	11875.0	1613.0	1907.4	5724.8	1446.1	**929.4**

Computational cost comparison. Although the proposed preconditioner reduces the number of Krylov iterations, its computational cost may exceed that of other preconditioners, see the comparison in the Appendix. This difference arises from the floating-point operations (FLOPs) required in the preconditioning step. Let $\boldsymbol{V} \in \mathbb{R}^{n\times k}$ denote the subspace, where $k \ll n$. The projection $\boldsymbol{V}\boldsymbol{V}^T r$ can be evaluated without explicitly forming the dense matrix $\boldsymbol{V}\boldsymbol{V}^T$ by computing $\boldsymbol{V}(\boldsymbol{V}^T r)$. This requires approximately $4nk$ FLOPs for each vector r. In contrast, applying an incomplete Cholesky (ICC) preconditioner requires two sparse triangular solves with the factors $\boldsymbol{L}$ and $\boldsymbol{L}^T$. If $s_L = \mathrm{nnz}(\boldsymbol{L})$, the corresponding computational cost is approximately $4s_L$ FLOPs per iteration. Therefore, the relative computational costs of NEURSP and ICC depend on the recycling dimension k and the fill-in of the incomplete factorization.

7 CONCLUSION

We introduced NEURSP, a learned spectral preconditioner that converts the invariant subspace predicted by NeurKItt into an explicit low-rank correction. The construction reuses an existing prediction rather than learning a dense approximation to the inverse, requires only matrix–vector products and dense algebra of size k, and is SPD for any full-rank predicted basis. Our analysis characterizes the corrected spectrum for exact eigenvectors and establishes perturbation bounds for approximate invariant subspaces, thereby relating the quality of the learned subspace to the quality of the preconditioner. Experiments on Helmholtz, heat-conduction, and Darcy-flow systems demonstrate that this spectral correction substantially reduces GMRES iterations. With a predicted subspace of dimension 10 or 20, NEURSP achieves the lowest iteration count in every tested problem–tolerance combination. Relative to ICC(0), NEURSP reduces the iteration count by 38.4%–89.2%, with larger subspaces providing particularly consistent benefits at the stricter tolerances.

AI USE STATEMENT

Generative AI tools were used for language editing and consistency checks. The authors reviewed all AI-assisted changes and take responsibility for the final content, including all claims and results.

ETHICS STATEMENT

This work concerns numerical algorithms for PDE-induced linear systems and does not involve human subjects or personal data. We do not anticipate direct ethical risks beyond the general environmental cost and potential misuse of large-scale computation.

REPRODUCIBILITY STATEMENT

Section 6 reports the datasets, solver settings, stopping criteria, and principal hyperparameters. Appendices A and B describe the PDE discretizations and dataset construction, and the appendix provides complete proofs of the theoretical results. The complete code, including data-generation and evaluation scripts, will be made available on GitHub upon acceptance.

REFERENCES

Yael Azulay and Eran Treister. Multigrid-augmented deep learning preconditioners for the Helmholtz equation. *SIAM Journal on Scientific Computing*, 44(5):S127–S151, 2022.

Christopher Beattie, Mark Embree, and John Rossi. Convergence of restarted krylov subspaces to invariant subspaces. *SIAM Journal on Matrix Analysis and Applications*, 25(4):1074–1109, 2004.

James V. Beck, Ben Blackwell, and Charles R. St. Clair. *Inverse heat conduction: Ill-posed problems*. John Wiley & Sons, 1985.

Michele Benzi, Carl D. Meyer, and Miroslav Tůma. A sparse approximate inverse preconditioner for the conjugate gradient method. *SIAM Journal on Scientific Computing*, 17(5):1135–1149, 1996.

Liam Burke, Stefan Güttel, and Kirk M Soodhalter. Gmres with randomized sketching and deflated restarting. *SIAM Journal on Matrix Analysis and Applications*, 46(1):702–725, 2025.

Ke Chen. *Matrix preconditioning techniques and applications*, volume 19. Cambridge University Press, 2005.

André Gaul. *Recycling Krylov subspace methods for sequences of linear systems*. PhD thesis, Technische Universität Berlin, 2014.

Markus Götz and Hartwig Anzt. Machine learning-aided numerical linear algebra: Convolutional neural networks for the efficient preconditioner generation. In *2018 IEEE/ACM 9th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (scalA)*, pp. 49–56. IEEE, 2018.

Anne Greenbaum. *Iterative methods for solving linear systems*. SIAM, 1997.

Daniel Greenfeld, Meirav Galun, Ronen Basri, Irad Yavneh, and Ron Kimmel. Learning to optimize multigrid PDE solvers. In *International Conference on Machine Learning*, pp. 2415–2423. PMLR, 2019.

Stefan Güttel and Françoise Tisseur. The nonlinear eigenvalue problem. *Acta Numerica*, 26:1–94, 2017.

Jun-Ting Hsieh, Shengjia Zhao, Stephan Eismann, Lucia Mirabella, and Stefano Ermon. Learning neural PDE solvers with convergence guarantees. *arXiv preprint arXiv:1906.01200*, 2019.

Misha E. Kilmer and Eric De Sturler. Recycling subspace information for diffuse optical tomography. *SIAM Journal on Scientific Computing*, 27(6):2140–2166, 2006.

Nikola Kovachki, Samuel Lanthaler, and Siddhartha Mishra. On universal approximation and error bounds for Fourier neural operators. *The Journal of Machine Learning Research*, 22(1):13237–13312, 2021a.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. *arXiv preprint arXiv:2108.08481*, 2021b.

Yichen Li, Peter Yichen Chen, Wojciech Matusik, et al. Learning preconditioner for conjugate gradient PDE solvers. *arXiv preprint arXiv:2305.16432*, 2023.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020a.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020b.

Jörg Liesen and Zdeněk Strakoš. *Krylov subspace methods: principles and analysis*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2013.

Chih-Jen Lin and Jorge J. Moré. Incomplete Cholesky factorizations with limited memory. *SIAM Journal on Scientific Computing*, 21(1):24–45, 1999.

Kevin Luna, Katherine Klymko, and Johannes P. Blaschke. Accelerating GMRES with deep learning in real-time. *arXiv preprint arXiv:2103.10975*, 2021.

Jian Luo, Jie Wang, Hong Wang, Huanshuo Dong, Zijie Geng, Hanzhu Chen, and Yufei Kuang. Neural krylov iteration for accelerating linear system solving. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 128636–128667. Curran Associates, Inc., 2024. doi: 10.52202/079017-4086.

Ilay Luz, Meirav Galun, Haggai Maron, Ronen Basri, and Irad Yavneh. Learning algebraic multigrid using graph neural networks. In *International Conference on Machine Learning*, pp. 6489–6499. PMLR, 2020.

Ronald B. Morgan. GMRES with deflated restarting. *SIAM Journal on Scientific Computing*, 24(1): 20–37, 2002.

Michael L. Parks, Eric De Sturler, Greg Mackey, Duane D. Johnson, and Spandan Maiti. Recycling Krylov subspaces for sequences of linear systems. *SIAM Journal on Scientific Computing*, 28(5): 1651–1674, 2006.

Md Ashiqur Rahman, Zachary E. Ross, and Kamyar Azizzadenesheli. U-NO: U-shaped neural operators. *arXiv preprint arXiv:2204.11127*, 2022.

Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

Rita Stanaityte. *ILU and machine learning based preconditioning for the discretized incompressible Navier-Stokes equations*. PhD thesis, University of Houston, 2020.

G. W. Stewart and Jiguang Sun. *Matrix perturbation theory*. Academic Press, 1990.

Ali Taghibakhshi, Scott MacLachlan, Luke Olson, and Matthew West. Optimization-based algebraic multigrid coarsening using reinforcement learning. *Advances in Neural Information Processing Systems*, 34:12129–12140, 2021.

Andrea Toselli and Olof Widlund. *Domain decomposition methods—algorithms and theory*, volume 34. Springer Science & Business Media, 2004.

Enrui Zhang, Adar Kahana, Eli Turkel, Rishikesh Ranade, Jay Pathak, and George Em Karniadakis. A hybrid iterative numerical transferable solver (HINTS) for PDEs based on deep operator network and relaxation methods. *arXiv preprint arXiv:2208.13273*, 2022.

Xuejun Zhang. Multilevel Schwarz methods. *Numerische Mathematik*, 63(1):521–539, 1992.