Accelerating Data Generation for Neural Operators via Krylov Subspace Recycling

Hong Wang[1], Zhongkai Hao[2*], Jie Wang[1†], Zijie Geng[1], Zhen Wang[1], Bin Li[1], Feng Wu[1]*

[1] *University of Science and Technology of China*
[2] *Tsinghua University*

ICLR 2024 (Spotlight)

Paper **WeChat** **MIRA Lab**

Email: wanghong1700@mail.ustc.edu.cn

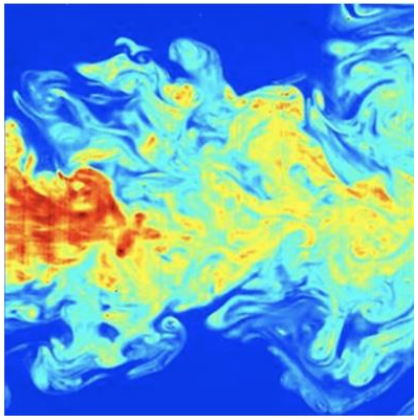
01 Introduction

02 Methodology

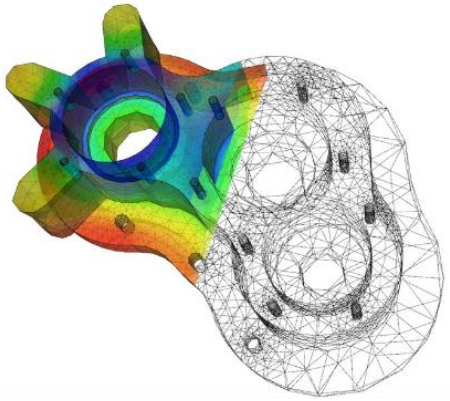
03 Experiments

04 Discussion

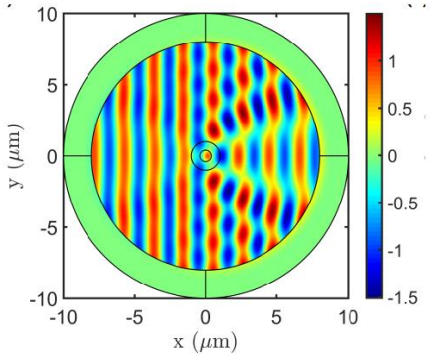
PDEs (Partial Differential Equations)



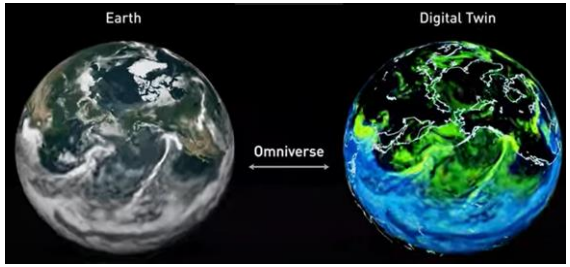
Fluids



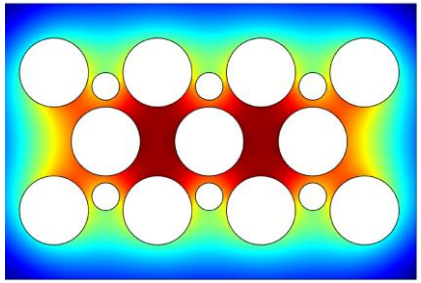
Solid Mechanics



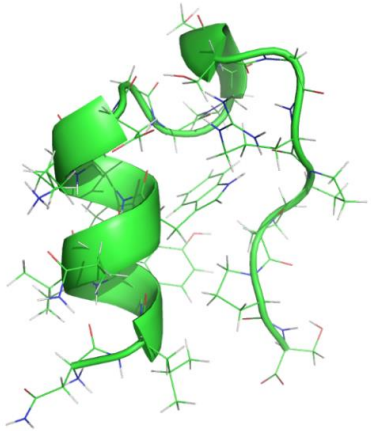
Electric/Magnetic field



Climate



Heat transfer



Protein

AI for PDE

Neural operator

Learning maps from parameters to PDE solutions using neural networks

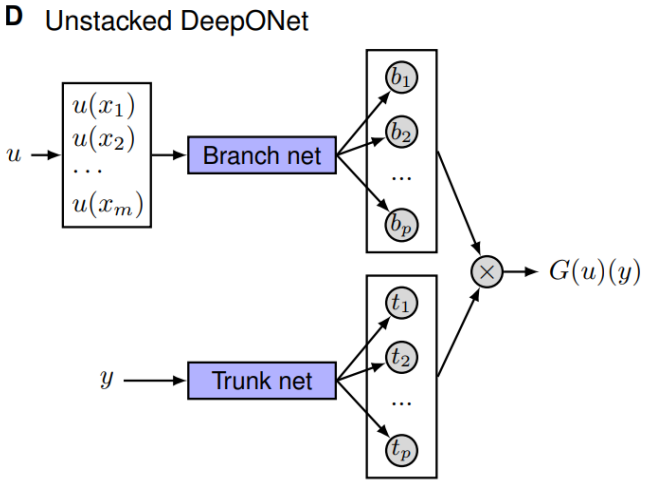
$$
\begin{aligned}
-\nabla \cdot (a(x)\nabla u(x)) &= f(x), \quad && x \in D \\
u(x) &= 0, && x \in \partial D
\end{aligned}
$$

Neural Operators

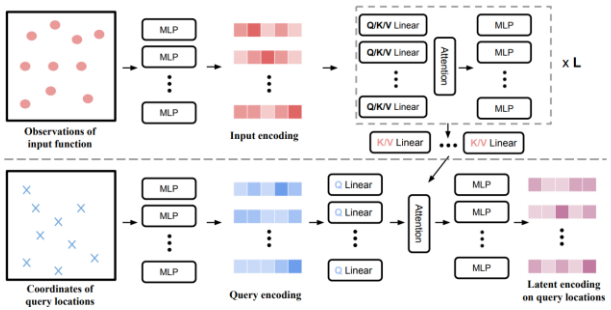
The development of neural operators

- DeepONet
  - MIONet
  - POD-DeepONet…
- FNO
  - GeoFNO
  - MWT, MG-TFNO
- Transformer
  - Galerkin Transforme
  - Oformer
  - GNOT

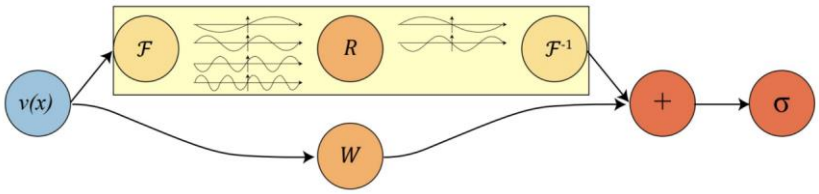
$$G(u)(y) \approx \sum_{k=1}^{p} \underbrace{b_k(u)}_{\text{branch}} \underbrace{t_k(y)}_{\text{trunk}}$$



DeepONet



OFormer



FNO

Problem Background

The primary limitation of neural operators

In recent years, with the reascendance of deep learning, it has become popular to learn PDE solvers circumventing the lengthy and often tedious process of solver design. But we are left with a proverbial 'chicken-and-egg problem'. **From where do we obtain the abundant data needed to train said neural solvers? It has to be generated with a classical solver, after all.**

---"Lie Point Symmetry Data Augmentation for Neural PDE Solvers"

Max Welling

We close this section on forward modeling with a discussion of challenges faced by neural solvers that have largely been unaddressed by current works.

A primary limitation of learned solvers is the requirement of an adequate number of training data generated by costly numerical solvers, which is particularly problematic at the industry scale.

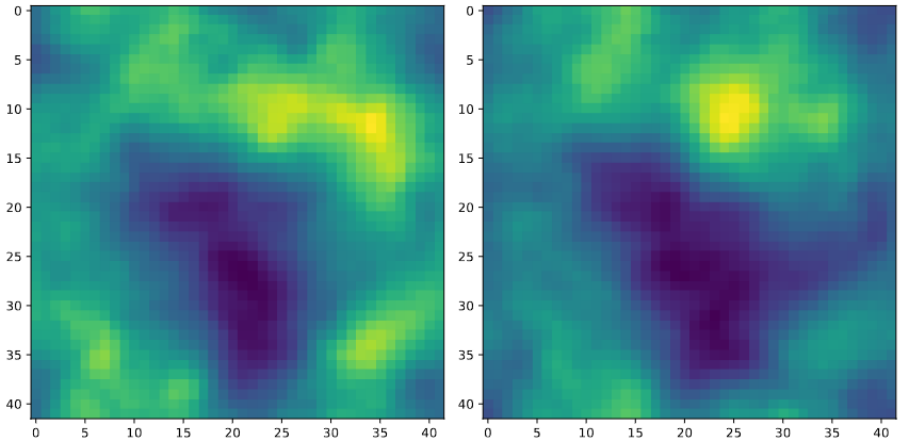
---"Artificial Intelligence for Science in Quantum, Atomistic, and Continuum Systems"

Shuiwang Ji

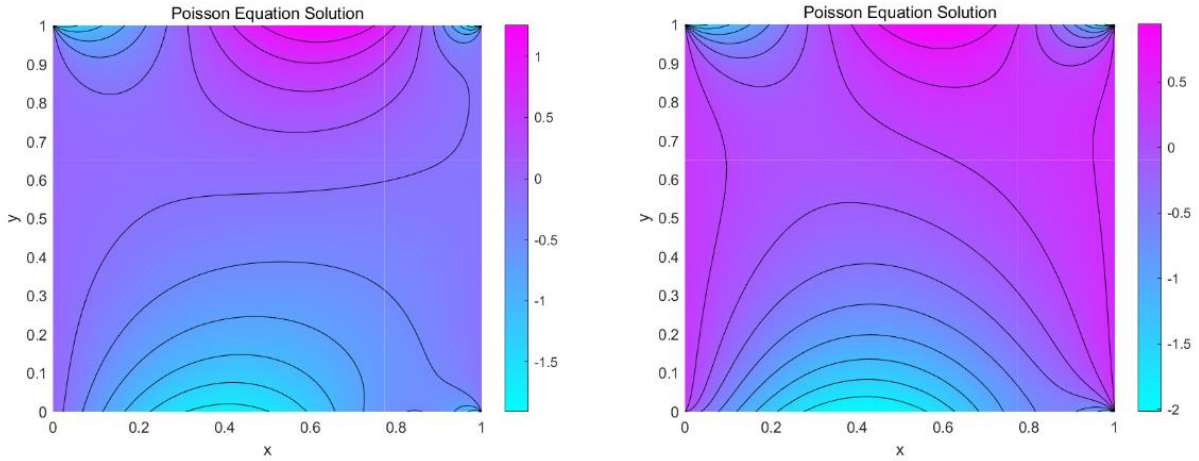
1. PDE datasets are not general; each PDE type needs its own dataset.

2. Training requires a lot of data, generated by expensive traditional algorithms.

Motivation



Solutions of Helmholtz equations with close parameters



Solutions of Possion equations with close parameters

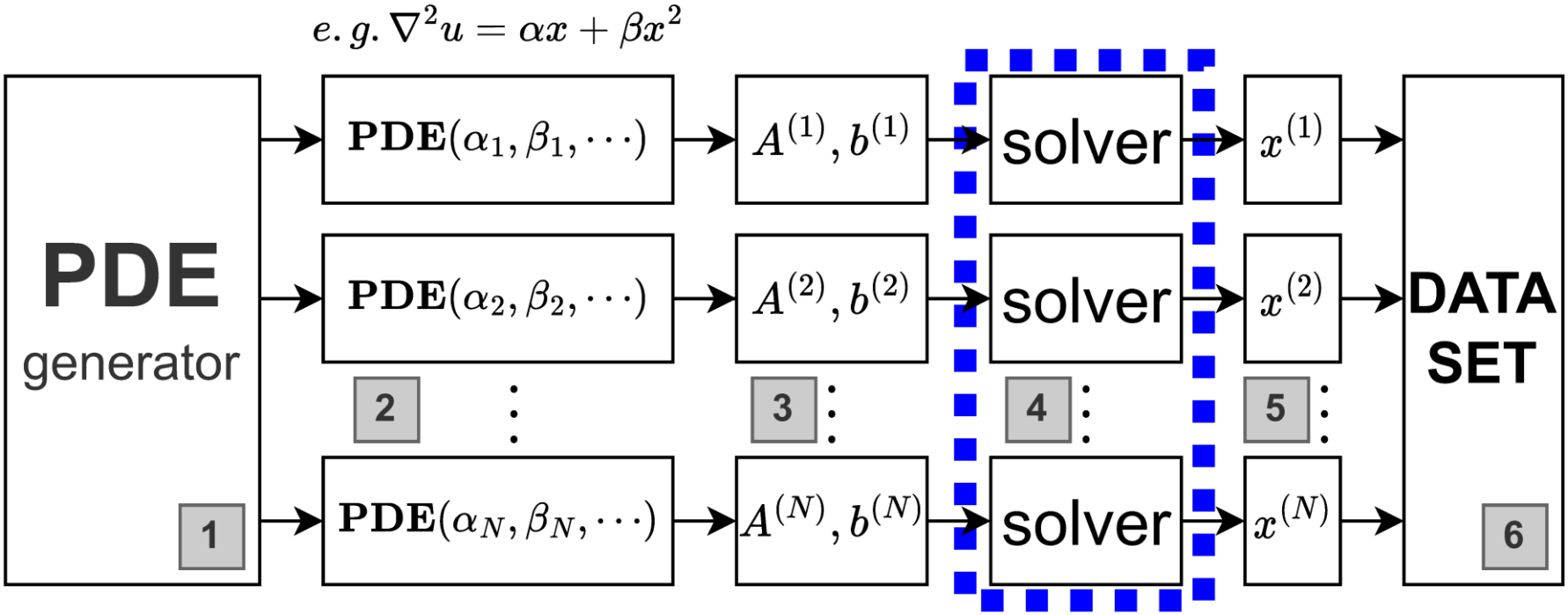
01 Introduction

02 Methodology

03 Experiments

04 Discussion

PDE data set generation process

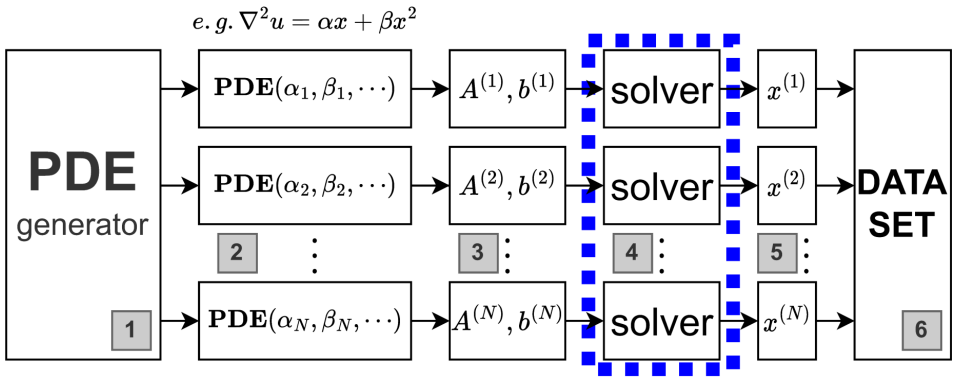


Example: 2D Possion equation

$$\nabla^2 u(x,y) = f(x,y),$$

using a 2×2 internal grid (i.e., $N_x = N_y = 2$ and $\Delta x = \Delta y$)

$$\begin{bmatrix} -4 & 1 & 1 & 0 \\ 1 & -4 & 0 & 1 \\ 1 & 0 & -4 & 1 \\ 0 & 1 & 1 & -4 \end{bmatrix} \begin{bmatrix} u_{1,1} \\ u_{1,2} \\ u_{2,1} \\ u_{2,2} \end{bmatrix} = \begin{bmatrix} f(x_1,y_1) \\ f(x_1,y_2) \\ f(x_2,y_1) \\ f(x_2,y_2) \end{bmatrix}$$



Krylov Subspace Method

linear equation system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$

where the matrix $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ and the vector $\boldsymbol{b} \in \mathbb{C}^{n}$

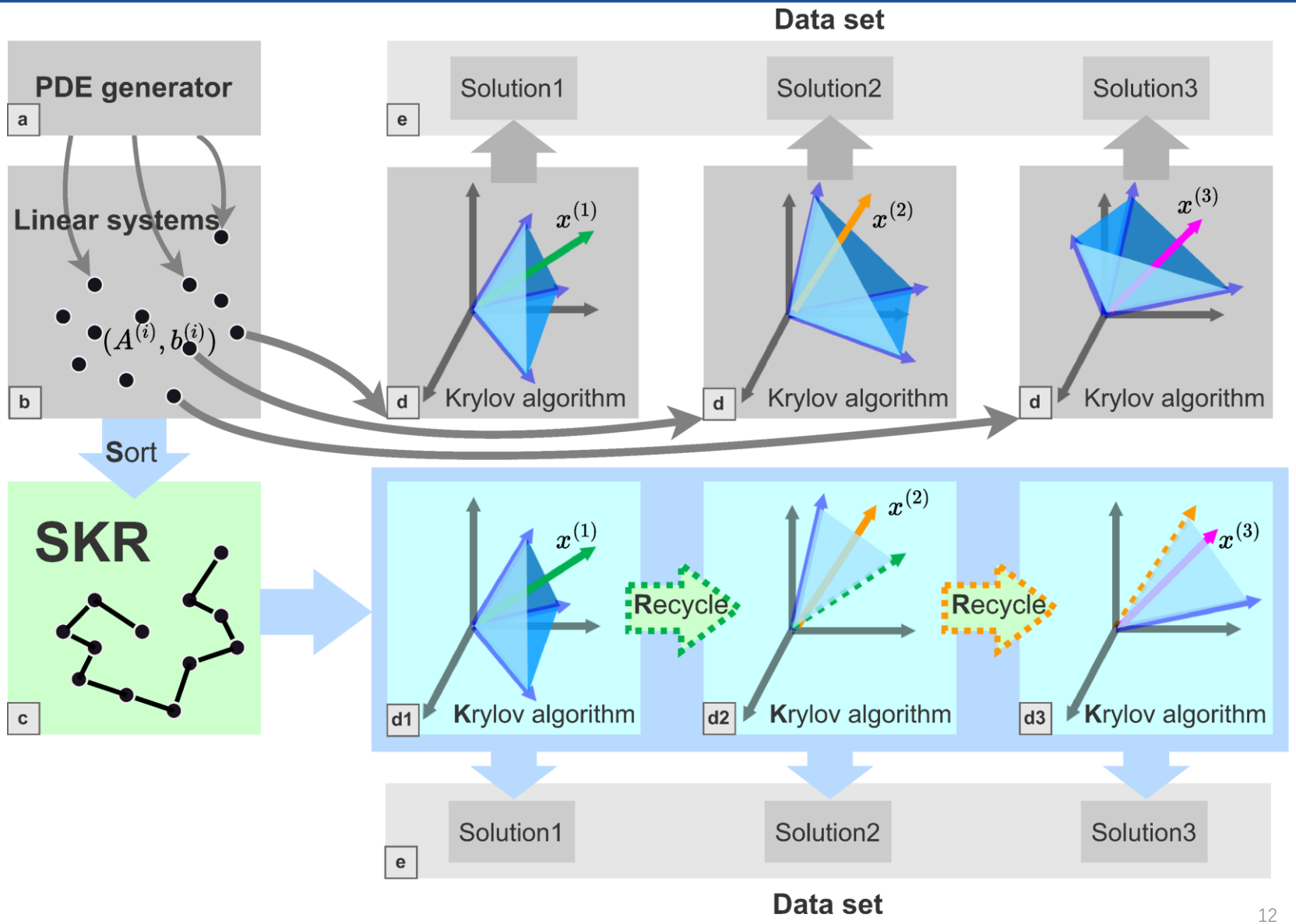
$$\mathcal{K}_m(\boldsymbol{A}, \boldsymbol{r}) = \text{span}\{\boldsymbol{r}, \boldsymbol{A}\boldsymbol{r}, \boldsymbol{A}^2\boldsymbol{r}, \cdots, \boldsymbol{A}^{m-1}\boldsymbol{r}\}.$$

$$\boldsymbol{A}\boldsymbol{V}_m = \boldsymbol{V}_{m+1}\underline{\boldsymbol{H}}_m,$$

Arnoldi relation

where $\boldsymbol{V}_m \in \mathbb{C}^{n\times m}$ and $\underline{\boldsymbol{H}}_m \in \mathbb{C}^{(m+1)\times m}$ is upper Hessenberg.

Our SKR (Sorting Krylov Recycling)



SKR: The Sorting Algorithm

$$\boldsymbol{A}^{(i)}\boldsymbol{x}^{(i)} = \boldsymbol{b}^{(i)} \quad i = 1, 2, \cdots,$$

where the matrix $\boldsymbol{A}^{(i)} \in \mathbb{C}^{n\times n}$ and the vector $\boldsymbol{b}^{(i)} \in \mathbb{C}^{n}$ vary based on different PDEs.

Algorithm 1: The Sorting Algorithm

Input: Sequence of linear systems to be solved $\boldsymbol{A}^{(i)} \in \mathbb{C}^{n\times n}, \boldsymbol{b}^{(i)} \in \mathbb{C}^{n}$, corresponding parameter matrix $\boldsymbol{P}^{(i)} \in \mathbb{C}^{p\times p}$ and $i = 1, 2, \cdots, N$

Output: Sequence for solving systems of linear equations seq_{mat}

Initialize the list with sequence $seq_0 = \{1, 2, \cdots, N\}$, seq_{mat} is an empty list;
Set $i_0 = 1$ as the starting point. And remove 1 from seq_0 and append 1 to seq_{mat};
for $i = 1, \cdots, N-1$ **do**
    Refresh dis and set it to a large number, e.g., 1000;
    for *each* j *in* seq_0 **do**
        dis_j = the Frobenius norm of the difference between $\boldsymbol{P}^{(i_0)}$ and $\boldsymbol{P}^{(j)}$;
        if $dis_j < dis$ **then**
            $dis = dis_j$ and $j_{min} = j$;
    Remove j_{min} from seq_0 and append j_{min} to seq_{mat} and set $i_0 = j_{min}$;
Get the sequence for solving linear systems seq_{mat}.

SKR: Krylov recycling

$$\mathcal{K}_m(\boldsymbol{A}, \boldsymbol{r}) = \text{span}\{\boldsymbol{r}, \boldsymbol{A}\boldsymbol{r}, \boldsymbol{A}^2\boldsymbol{r}, \cdots, \boldsymbol{A}^{m-1}\boldsymbol{r}\}.$$

$$\boldsymbol{A}\boldsymbol{V}_m = \boldsymbol{V}_{m+1}\underline{\boldsymbol{H}}_m,$$

Arnoldi relation

$\widetilde{\boldsymbol{Y}}_k = [\widetilde{\boldsymbol{y}}_1, \widetilde{\boldsymbol{y}}_1, \cdots, \widetilde{\boldsymbol{y}}_k]$ we retain k approximate eigenvectors

Let $[\boldsymbol{Q}, \boldsymbol{R}]$ be the reduced QR-factorization of $\boldsymbol{A}\widetilde{\boldsymbol{Y}}_k$.

$$\boldsymbol{C}_k = \boldsymbol{Q}$$
$$\boldsymbol{U}_k = \widetilde{\boldsymbol{Y}}_k\boldsymbol{R}^{-1}$$
$$\boldsymbol{x}_1 = \boldsymbol{x}_0 + \boldsymbol{U}_k{\boldsymbol{C}_k}^H\boldsymbol{r}_0$$
$$\boldsymbol{r}_1 = \boldsymbol{r}_0 - \boldsymbol{C}_k{\boldsymbol{C}_k}^H\boldsymbol{r}_0$$

SKR Arnoldi relation

$$(\boldsymbol{I} - \boldsymbol{C}_k\boldsymbol{C}_k^H)\boldsymbol{A}\boldsymbol{V}_{m-k} = \boldsymbol{V}_{m-k+1}\underline{\boldsymbol{H}}_{m-k}.$$

Theoretical Analysis

Convergence Analysis

We define the $one-sided\ distance$ from the subspace $\mathcal{Q}$ to the subspace $\mathcal{C}$ as

$$\delta(\mathcal{Q},\mathcal{C}) = \|(\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{C}})\boldsymbol{\Pi}_{\mathcal{Q}}\|_2,$$

where $\boldsymbol{\Pi}$ represents the projection operator for the associated space.

Theorem 1. *Given a space $\mathcal{C} = \text{range}(\boldsymbol{C}_k)$, let $\mathcal{V} = \text{range}(\boldsymbol{V}_{m-k+1}\underline{\boldsymbol{H}}_{m-k})$ be the $(m-k)$ dimensional Krylov subspace generated by SKR. Let $\boldsymbol{r}_0 \in \mathbb{C}^n$, and let $\boldsymbol{r}_1 = (\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{C}})\boldsymbol{r}_0$. Then, for each $\mathcal{Q}$ such that $\delta(\mathcal{Q},\mathcal{C}) < 1$,*

$$\min_{\boldsymbol{d}_1 \in \mathcal{V}\oplus\mathcal{C}} \|\boldsymbol{r}_0 - \boldsymbol{d}_1\|_2 \leq \min_{\boldsymbol{d}_2 \in (\boldsymbol{I}-\boldsymbol{P}_{\mathcal{Q}})\mathcal{V}} \|(\boldsymbol{I} - \boldsymbol{P}_{\mathcal{Q}})\boldsymbol{r}_1 - \boldsymbol{d}_2\|_2 + \frac{\gamma}{1-\delta}\|\boldsymbol{P}_{\mathcal{Q}}\|_2 \cdot \|(\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{V}})\boldsymbol{r}_1\|_2,$$

where $\gamma = \|(\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{C}})\boldsymbol{P}_{\mathcal{Q}}\|_2$.

Theoretical Analysis

Selection of Recycle Subspaces

$$\boldsymbol{A} = [\boldsymbol{Q}_1\boldsymbol{Q}_2\boldsymbol{Q}_3]\mathrm{diag}(\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \boldsymbol{\Lambda}_3)[\boldsymbol{Q}_1\boldsymbol{Q}_2\boldsymbol{Q}_3]^H,$$

Theorem 5.2. *Let $\boldsymbol{A}$ be Hermitian positive definite and have the eigendecomposition given in (14), and let $\boldsymbol{E}$, ϵ, η, μ, and $\tilde{\mu}$ be defined as above. Then there exists a matrix $\tilde{\boldsymbol{Q}}_1$ conforming to $\boldsymbol{Q}_1$ such that range($\tilde{\boldsymbol{Q}}_1$) is a simple invariant subspace of $\boldsymbol{A} + \boldsymbol{E}$, and*

$$\tan\theta_1(range(\boldsymbol{Q}_1), range(\tilde{\boldsymbol{Q}}_1)) \leq \frac{\epsilon}{\tilde{\mu}}.$$

Theoretical Analysis

The Rationality of Sorting Algorithms

We define the $one-sided\ distance$ from the subspace $\mathcal{Q}$ to the subspace $\mathcal{C}$ as

$$\delta(\mathcal{Q},\mathcal{C}) = \|(\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{C}})\boldsymbol{\Pi}_{\mathcal{Q}}\|_2,$$

where $\boldsymbol{\Pi}$ represents the projection operator for the associated space.

Theorem 1. *Given a space $\mathcal{C} = \text{range}(C_k)$, let $\mathcal{V} = \text{range}(V_{m-k+1}\underline{H}_{m-k})$ be the $(m-k)$ dimensional Krylov subspace generated by SKR. Let $r_0 \in \mathbb{C}^n$, and let $r_1 = (\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{C}})r_0$. Then, for each $\mathcal{Q}$ such that $\delta(\mathcal{Q},\mathcal{C}) < 1$,*

$$\min_{\boldsymbol{d}_1 \in \mathcal{V}\oplus\mathcal{C}} \|\boldsymbol{r}_0 - \boldsymbol{d}_1\|_2 \leq \min_{\boldsymbol{d}_2 \in (\boldsymbol{I}-\boldsymbol{P}_{\mathcal{Q}})\mathcal{V}} \|(\boldsymbol{I} - \boldsymbol{P}_{\mathcal{Q}})\boldsymbol{r}_1 - \boldsymbol{d}_2\|_2 + \frac{\gamma}{1-\delta}\|\boldsymbol{P}_{\mathcal{Q}}\|_2 \cdot \|(\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{V}})\boldsymbol{r}_1\|_2,$$

where $\gamma = \|(\boldsymbol{I} - \boldsymbol{\Pi}_{\mathcal{C}})\boldsymbol{P}_{\mathcal{Q}}\|_2$.

$$\frac{\gamma}{1-\delta}$$

01 Introduction

02 Methodology

03 Experiments

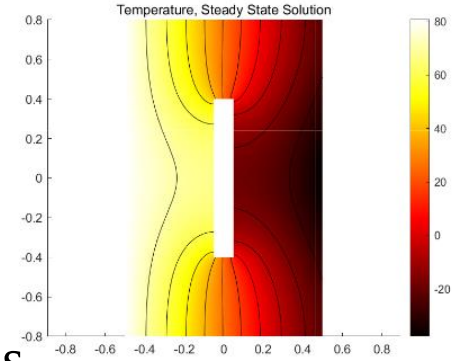
04 Discussion

Experiments

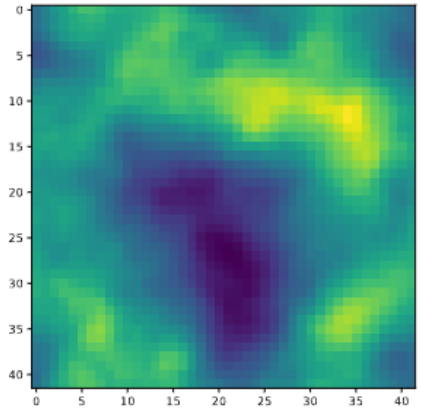
- Baseline
  - GMRES
- Datasets
  - Darcy flow
  - Thermal
  - Possion
  - Helmholtz
- Preconditon
  - None
  - Jacobi
  - BJacobi
  - SOR
  - ASM
  - ICC
  - ILU

- Matrix sizes
  - 5-6 variations
- Accuracy
  - 5-8 tolerances
- Performance Metrics
  - Computational Duration
  - Iteration Count

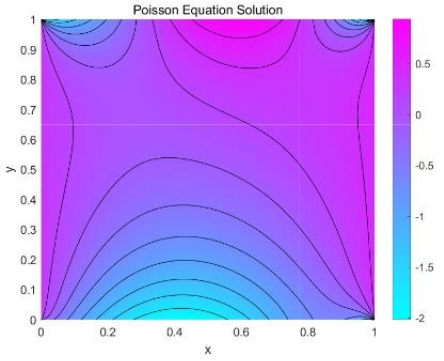
- Total
  - Nearly **3000** experiments



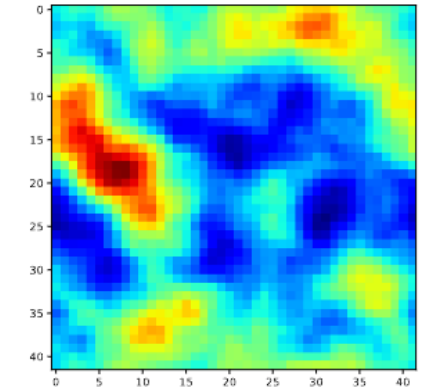
Thermal



Helmholtz



Possion



Darcy flow

Experiments

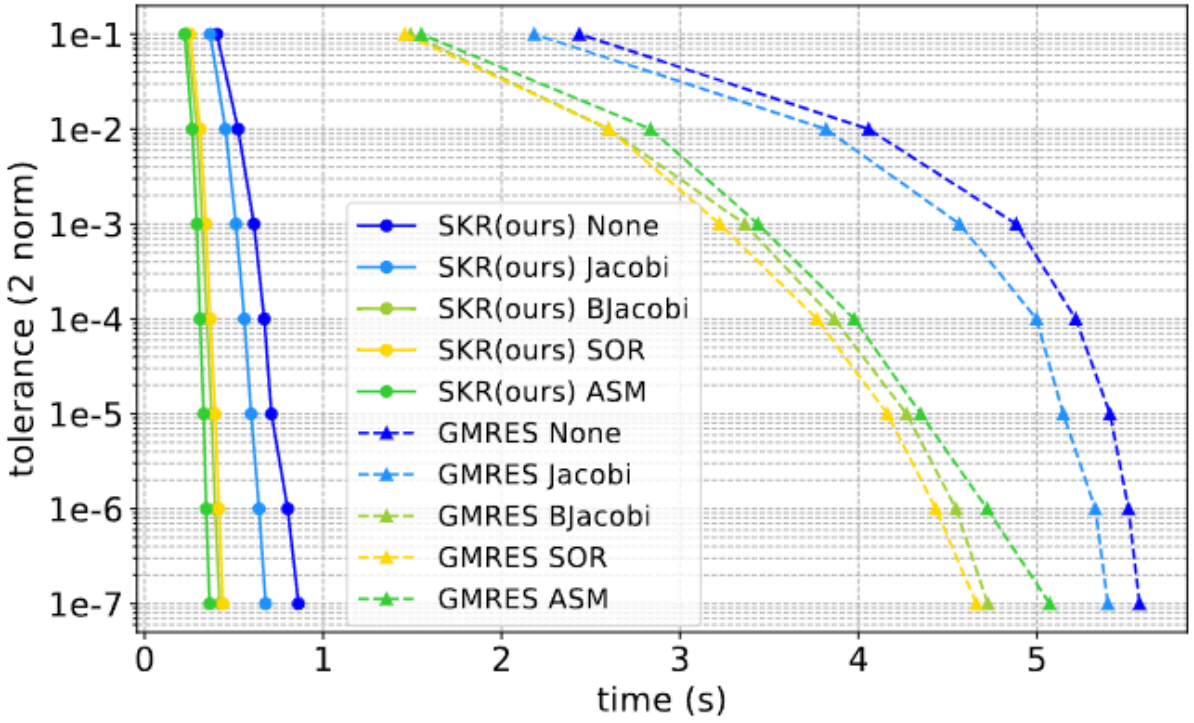
Main results

Table 1: Comparison of our SKR and GMRES computation time and iterations across datasets, preconditioning, and tolerances. The first column lists datasets with matrix side lengths, the next details tolerances. The data is displayed as 'computation time speed-up ratio/iteration count speed-up ratio'. A GMRES/SKR ratio over 1 denotes better SKR performance.

Dataset	Time/Iter	None	Jacobi	BJacobi	SOR	ASM	ICC	ILU
Darcy 6400	1e-2	2.62/19.2	2.88/22.6	3.21/23.6	2.69/23.3	2.23/13.4	1.97/9.55	1.93/9.44
	1e-5	2.92/21.1	3.42/24.5	4.07/28.6	3.45/27.9	3.66/22.2	3.18/14.9	3.08/14.4
	1e-8	2.70/19.1	3.09/22.9	4.00/27.5	3.54/27.5	4.53/25.8	4.11/18.9	3.70/17.2
Thermal 11063	1e-5	4.53/20.8	3.32/15.0	2.38/10.3	1.96/8.76	2.46/10.3	2.40/10.3	2.35/10.3
	1e-8	5.35/23.6	3.06/13.5	2.73/11.1	2.34/9.30	2.83/11.1	2.77/11.1	2.69/11.1
	1e-11	5.47/24.9	2.93/12.8	3.05/11.7	2.62/10.2	3.14/11.7	3.08/11.7	3.01/11.7
Poisson 71313	1e-5	1.27/4.69	1.28/4.68	1.13/3.87	1.19/4.05	1.46/3.93	0.99/3.92	0.98/3.92
	1e-8	1.74/6.29	1.75/6.30	1.19/3.97	1.35/4.45	1.94/4.83	1.32/4.83	1.30/4.87
	1e-11	1.90/6.83	1.91/6.82	1.19/3.95	1.33/4.35	2.12/5.11	1.42/5.13	1.38/5.13
Helmholtz 10000	1e-2	7.74/17.3	8.44/20.3	8.83/21.5	8.37/22.3	10.6/25.4	4.77/21.4	3.88/17.6
	1e-5	7.61/16.5	8.62/20.0	11.4/26.5	10.5/26.2	13.1/29.3	6.38/28.3	6.13/27.2
	1e-7	6.47/13.68	7.96/18.1	11.3/26.2	10.6/25.9	13.9/30.0	6.72/29.3	6.34/28.0

Experiments

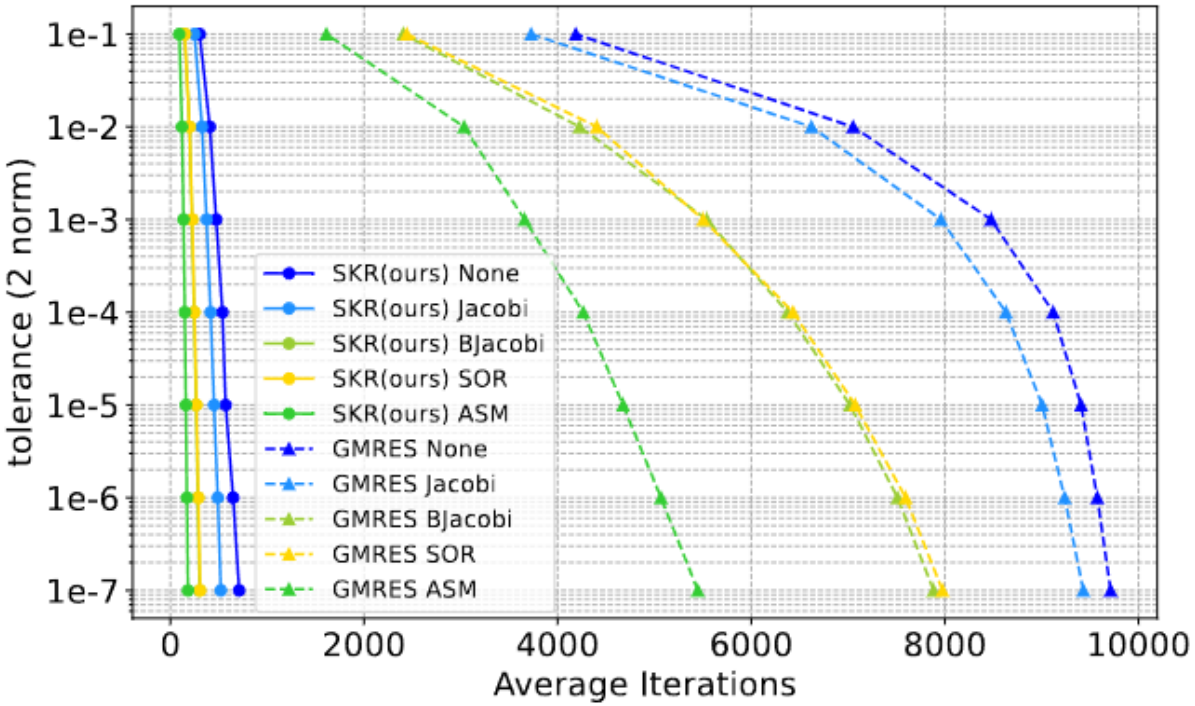
Convergence speed analysis (time)



Solver	Precondition	Slope
SKR(oures)	None	-6.80×10^{-5}
SKR(oures)	Jacobi	-1.25×10^{-4}
SKR(oures)	BJacobi	-2.13×10^{-4}
SKR(oures)	SOR	-2.22×10^{-4}
SKR(oures)	ASM	-2.85×10^{-4}
GMRES	None	-6.27×10^{-5}
GMRES	Jacobi	-4.17×10^{-5}
GMRES	BJacobi	-2.27×10^{-5}
GMRES	SOR	-2.03×10^{-5}
GMRES	ASM	-1.38×10^{-5}

Experiments

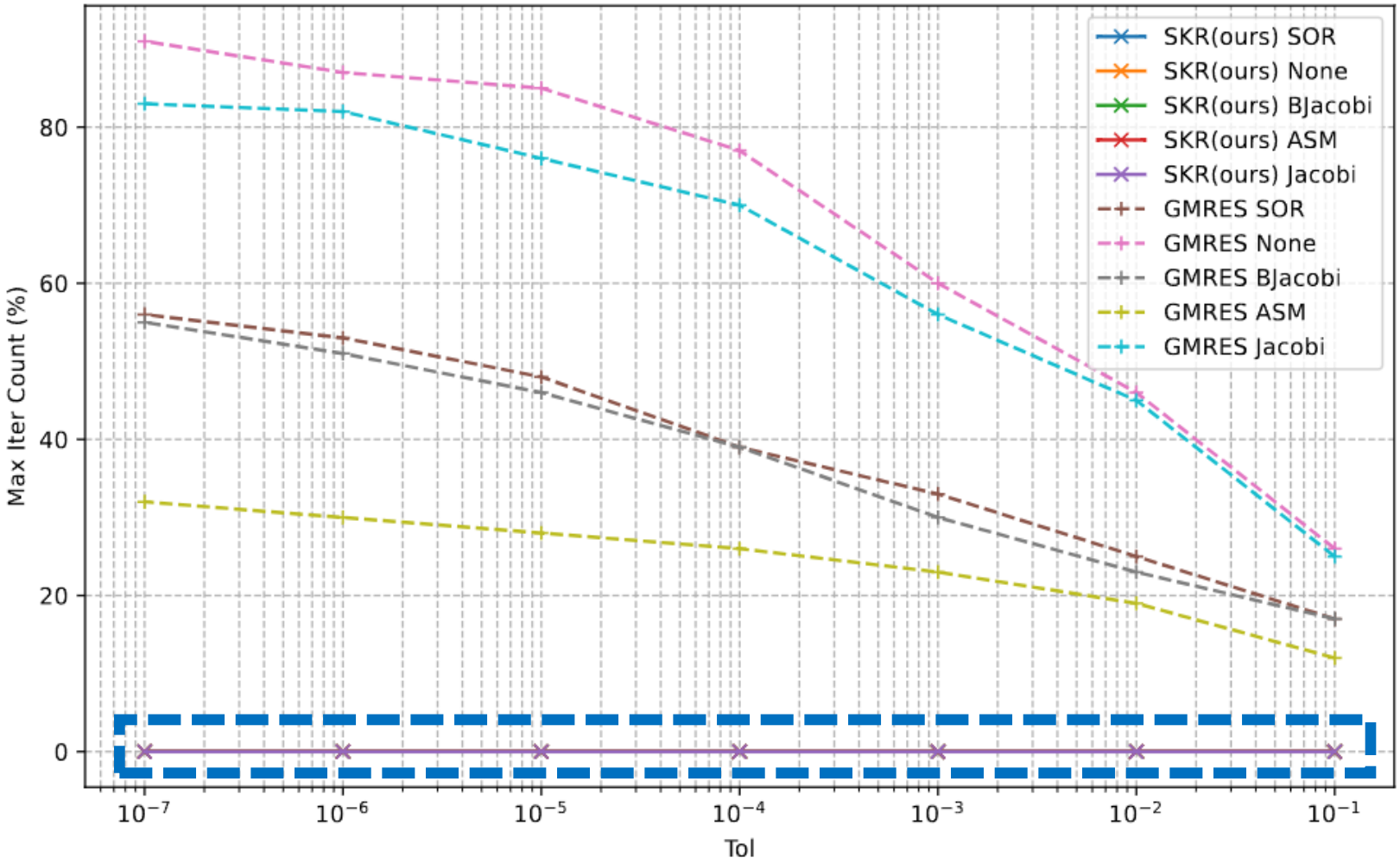
Convergence speed analysis (iteration)



Solver	Precondition	Slope
SKR(oures)	None	-7.27×10^{-8}
SKR(oures)	Jacobi	-1.44×10^{-7}
SKR(oures)	BJacobi	-2.68×10^{-7}
SKR(oures)	SOR	-2.60×10^{-7}
SKR(oures)	ASM	-4.49×10^{-7}
GMRES	None	-3.31×10^{-8}
GMRES	Jacobi	-2.37×10^{-8}
GMRES	BJacobi	-1.18×10^{-8}
GMRES	SOR	-1.13×10^{-8}
GMRES	ASM	-1.29×10^{-8}

Experiments

Stability analysis



Experiments

Ablation Experiment

	Time(s)	Iter	δ
SKR(sort)	0.101	183.9	0.90
SKR(nosort)	0.114	202.5	0.95

Comparison of algorithm performance with and without sort in Darcy flow problem, using SOR preconditioning, matrix size 10^4, and computational tolerance $1e-8$.

1. The δ metric declined by 5% after employing the 'sort' algorithm. This effectively demonstrates that the 'sort' algorithm can enhance the correlation between consecutive linear equation sets, achieving the initial design goal of 'sort'.

2. Using 'sort' enhances the SKR algorithm's computational speed by roughly 13% and decreases its iterations by 9.2%. This implies that by increasing the coherence among sequential linear equation sets, the number of iterations needed is minimized, thus hastening the system's solution process.

01 Introduction

02 Methodology

03 Experiments

04 Discussion

Discussion

Limitations and Future Prospects

- While this paper predominantly addresses linear PDEs, for other types of PDEs, there's a need for designing SKR algorithms specifically tailored to these PDE types to achieve optimized computational speeds.

- In the context of the sorting algorithm within SKR, there's potential to identify superior distance metrics based on the specific PDE, aiming to bolster the correlation of the sorted linear systems.

- Strategies to broaden the application of the recycling concept to other analogous data generation domains remain an open question.

MIRA