A numerical investigation on iterative methods for the two-dimensional Poisson equation discretized with high-order mimetic operators

Gustavo Espínola[1], Juan Cabral[2], Jhabriel Varela[3], and Christian Schaerer[4]
[1,2,3,4] Polytechnic School, National University of Asuncion, Paraguay
Grupo de Investigación en Computación Científica y Matemática Aplicada - CCyMA
gustavoespinola@fpuna.edu.py[1], jccabral@pol.una.py[2], jhabriel@pol.una.py[3], cschaer@pol.una.py[4]

Abstract

Mimetic operators are of increasing interest to the scientific computing community due to their ability to preserve many important properties of the continuous problem (e.g. conservation laws) while maintaining the same order of accuracy on the boundary as in the interior points. However, this results in locally dense and potentially ill-conditioned linear systems that are challenging to solve. This issue can partially be addressed using adequate iterative solvers, which is the focus of this work. Using the two-dimensional Poisson equation and mimetic operators of order $k = 2$ and $k = 4$, we compare the computational times obtained with different Krylov-subspace-based iterative methods used for the resolution of the linear systems.

Keywords: mimetic operators, Krylov subspace methods, sparse linear systems.

Introduction

Mimetic operators for the gradient ($G = \nabla$), divergence ($D = \nabla\bullet$) and Laplacian ($L = \nabla^2 = \Delta$) are discrete analogs of the continuum counterparts that still fulfill the classical identities from vector calculus. For example, while providing uniform order of accuracy, the operators satisfy the discrete version of the Gauss extended divergence theorem:

$$h\langle Dv, f\rangle_Q + h\langle v, Gf\rangle_P = \langle Bv, f\rangle \tag{1}$$

The discretization scheme is the staggered grid: scalar variables are stored at the centers of the cells, while vector components are placed at the edges.

When using mimetic operators, we are not discretizing the equations as it is done with standard finite-difference methods, but instead we construct matricial analogs to the differential operator:

$$\Delta\phi = f \text{ or } \nabla\bullet\nabla\phi = f \Rightarrow L\phi = f \text{ or } DG\phi = f \tag{2}$$

In this work, the studied operators are those developed by Corbino and Castillo [1].

Methodology

The numerical experiments, a follow-up from the previous work at [3], were run on a computer with Intel Core i5-12450 with 16GB RAM, using the software MATLAB R2022a for Windows 11. The chosen benchmark problem is the 2D minimal Poisson equation, subject to Robin boundary conditions:

$$-\Delta u = f \text{ in } \Omega \tag{3}$$
$$\alpha u + \beta\nabla u \bullet n = \gamma \text{ on } \Gamma. \tag{4}$$

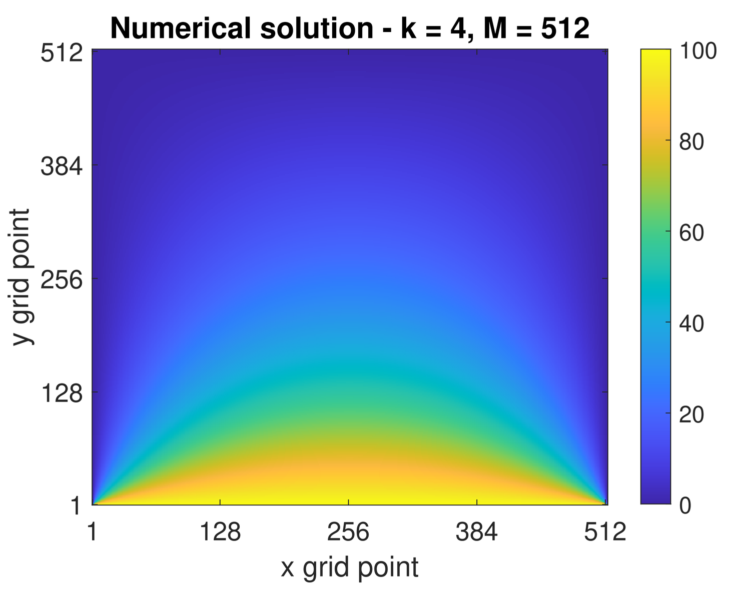

Figure 1: Numerical solution of the 2D Poisson equation.

The Mimetic Operators Library Enhanced (MOLE), available at `https://github.com/csrc-sdsu/mole`, was employed for the discretization process, which results in large sparse linear systems. To solve them, the following iterative methods were applied: GMRES(m), in which case the built-in MATLAB implementation is executed, LGMRES(m, l), GMRES-E(m, d), and PD-GMRES(m_j) [2], which is equipped with a discrete Proportional-Derivative controller for the restart parameter m:

$$m_j = m_{j-1} + \left\lfloor \alpha_P \frac{\|r_s^{(j)}\|}{\|r_s^{(j-1)}\|} + \alpha_D \frac{\|r_s^{(j)}\| - \|r_s^{(j-2)}\|}{2\|r_s^{(j-1)}\|} \right\rfloor. \tag{5}$$

where $\alpha_P, \alpha_D \in \mathbb{R}$ and $\lfloor(.)\rfloor$ denotes the integer part.

All these solvers are available in the Krylov Subspace-Based Adaptive Solvers (KrySBAS) library (`https://github.com/nidtec-una/krysbas-dev`).

Numerical results

Table 1: List of test problems in the Poisson problem with different precision orders (k) and discretization sizes (M).

k	M	size(A)	nnz(A)	condPreJacobi(A)	condPostJacobi(A)
2	32	1 156	5 252	753.8693	603.0954
2	64	4 356	20 740	3.0170E+093	2.413E+03
2	128	16 900	82 436	1.2008E+04	9.655E+03
2	256	66 564	328 708	4.8281E+04	3.863E+04
2	512	264 196	1 312 772	1.9312E+05	1.545E+05
4	32	1 156	13 316	1.0089E+03	843.0009
4	64	4 536	53 252	4.0391E+03	3.3748E+03
4	128	16 900	212 996	1.6160E+04	1.3502E+04
4	256	66 564	851 972	6.4642E+04	5.4011E+04
4	512	264 196	3 407 876	2.5857E+05	2.1605E+05

Table 2: Metrics for different problems and iterative methods: average computational time in seconds and number of inner iterations required for $\|r_j\|/\|r_0\| < 10^{-9}$. For the cases where it does not reach convergence before 100 restarts (outer iterations) the method is stopped and the time is denoted as NC. Bold face indicate smaller computational time.

k, M	**GMRES(m) Time (Iterations)**	**PD-GMRES(m_j) Time (Iterations)**	**LGMRES Time (Iterations)**	**GMRES-E Time (Iterations)**
2, 32	**0.0202** (170)	0.051 (183)	0.0618 (150)	0.1434 (120)
2, 64	0.1491 (561)	**0.1369** (391)	0.1370 (270)	0.1498 (240)
2, 128	1.9149 (1665)	0.5187 (710)	**0.4962** (510)	0.5649 (540)
2, 256	NC	3.0038 (1536)	**2.4703** (960)	4.2089 (1440)
2, 512	NC	NC	**46.488** (1740)	NC
4, 32	**0.0078** (253)	0.013 (250)	0.011 (210)	0.016 (150)
4, 64	0.233 (700)	0.138 (471)	0.126 (330)	**0.125** (300)
4, 128	2.698 (2154)	0.673 (769)	**0.639** (570)	0.806 (660)
4, 256	NC	4.593 (1787)	**3.643** (1140)	6.638 (1770)
4, 512	NC	NC	**61.96** (2010)	NC

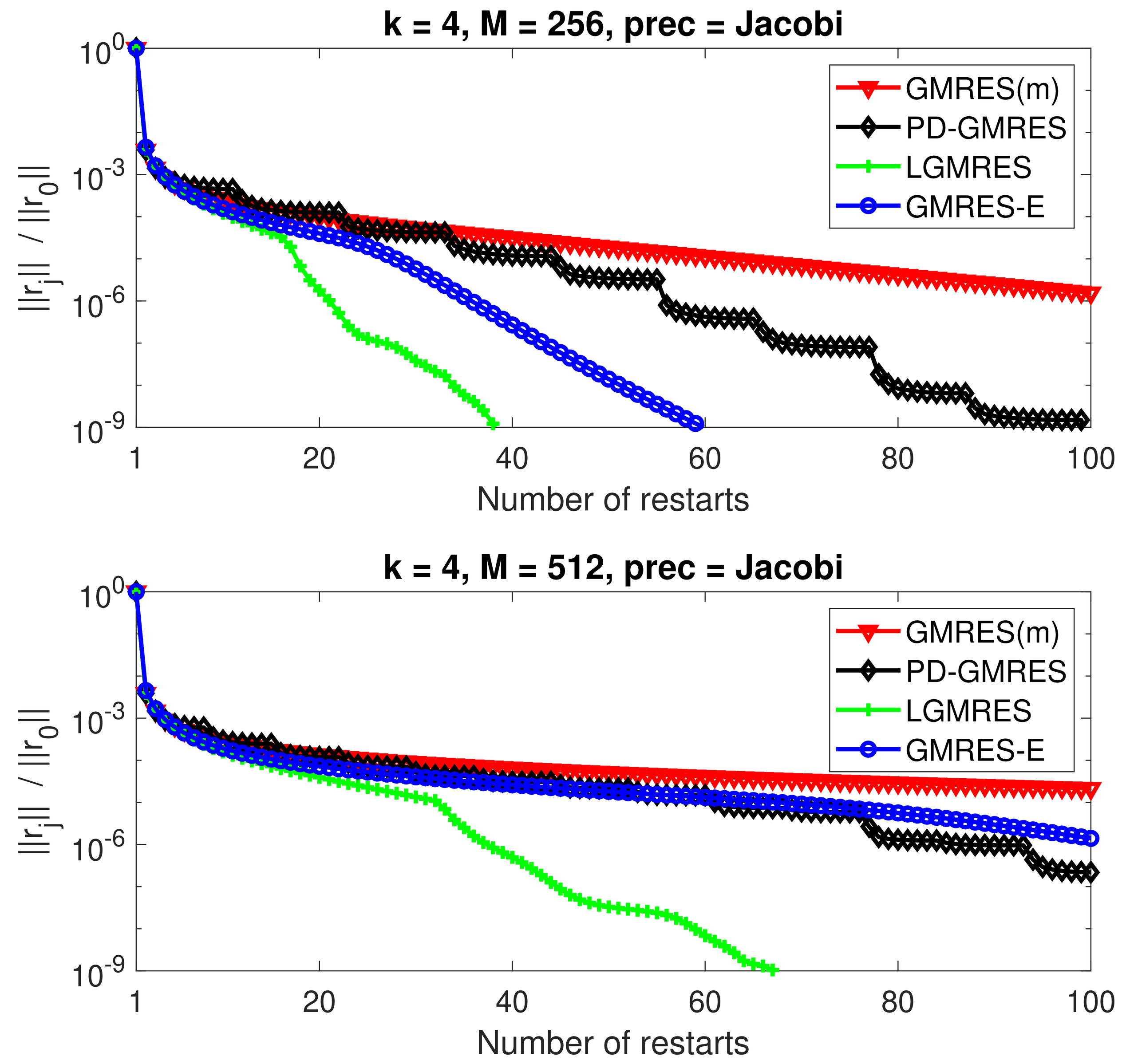

Figure 2: Relative residual norms v. number of GMRES restarts, for two examples.

Conclusion

Numerical experiments for different discretization sizes M (of an $(M+2)$-by-$(M+2)$ grid), precision orders of the mimetic operators (k) and iterative methods for the resulting linear systems are compared. According to the preliminary results, shown in Table 2, a strategy for modifying the size of the search subspace adaptively is good enough to accelerate convergence up to a certain number of discretization size: in the cases when the 1D step size is smaller, even the adjustment of the restart parameter m from GMRES is not sufficient to reach convergence under the iteration constraints. Hence, an enrichment of the search subspace is of substantial help. For the selected problems, the LGMRES(m, k) has performed better in the majority of the cases.

Acknowledgements

The autors are funded by the Polytechnic Faculty (FPUNA) through its Research Department. GE is funded by the Paraguayan National Council of Science and Technology (CONACYT) through the Incentive for the Training of Researchers in National Postgraduate Programs. JV is funded by CONACYT through the Advanced Human Capital Insertion Program in Academia PRIA01-8.

References

[1] J. Corbino and J. Castillo. High-order mimetic finite-difference operators satisfying the extended Gauss divergence theorem. *Journal of Computational and Applied Mathematics*, 364:112326, 2020.

[2] R. Cuevas, C. Schaerer, and A. Bhaya. A proportional-derivative control strategy for restarting the GMRES (m) algorithm. *Journal of Computational and Applied Mathematics*, 337:209–224, 2018.

[3] G. Espínola, J. Cabral, and C. Schaerer. Mimetic operator discretization 1D: Exploration of classical iterative methods and preconditioners. *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, 10(1), 2023.