Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 10, n. 1, 2023.

Trabalho apresentado no XLII CNMAC, Universidade Federal de Mato Grosso do Sul - Bonito - MS, 2023

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

On adaptative GMRES(m) in the PETSc package

Eduardo Gómez[1],
Lucas Vega[2]
Verónica Domiguez[3]
Juan Carlos Cabral[4]
Christian E. Schaerer[5]
Polytechnic School, National University of Asuncion, San Lorenzo, Paraguay

The Restarted Generalized Minimal Residual method (GMRES(m)) is a standard method for solving non-symmetric indefinite large linear systems of equations of the form $Ax = b$ [1, 2]. It has the limitation that if the restating parameter m is not adequately chosen can present either a slow convergence or stagnation [3]. This problem has been faced in several previous works [4–6]. However, to be useful for solving practical engineering and simulation problems, it is necessary to have the method in a computationally appropriate platform, allowing the simultaneous implementation in parallel and distributed architectures, and the implementation of several preconditioners. This work introduces a PETSc (Portable, Extensible Toolkit for Scientific Computation) routine that enforces the adaptation in the restarted parameter of the restarted-GMRES method of the solver in the PETSc.

The routine starts from a default value for $m = 30$. This value is systematically decreased, evaluating at each cycle of the GMRES(m) the rate of convergence. This process continues until either a stagnation or a critical slowing down in the convergence is detected. A possible strategy to overcome this problem consists in modifying the restart parameter. A control updating law is presented at Algorithm 1 below [3, 6], where the input are: m_{j-1} last value of m, $m_{initial}$: last initial value of m, m_{min} the minimum value of m, m_{max} the maximum value of m, m_{step}: value by which $m_{initial}$ is incremented when $m < m_{min}$ or m_j is decremented when $m > m_{max}$, r_q is the last q residuals, r_{j-1}, r_{j-2}, $\cdots$, r_{j-q}, and the output is m_j *the restart parameter used in GMRES(m),* $m_{initial}$ *for iteration* j.

[1]eduardocadenas97@fpuna.edu.py
[2]lukissvegap@fpuna.edu.py
[3]veronicanicoll97@fpuna.edu.py
[4]jccabral@pol.una.py
[5]cschaer@pol.una.py

Algorithm 1 PD Control Rule for Restart Parameter m

1. *if* $j > 3$, *then*
2. $m_j = m_{j-1} + \left\lfloor (\alpha)\frac{\|r_{j-1}\|}{\|r_{j-2}\|} + (\gamma)\frac{\|r_{j-1}\|-\|r_{j-3}\|}{2\|r_{j-2}\|} \right\rfloor$
3. *else if* $j > 2$, *then* $m_j = m_{j-1} + \left\lfloor (\alpha)\frac{\|r_{j-1}\|}{\|r_{j-2}\|} \right\rfloor$
4. *else* $m_j = m_{initial}$ *end if*
5. *if* $m_j < m_{initial}$, *then* $m_{initial} = m_{initial} + m_{step}$, $m_j = m_{initial}$ *end if*
6. *if* $m_j > m_{max}$, *then* $m_j = m_j - m_{step}$ *end if*
7. *return* m_j, $m_{initial}$ *for the next iteration.*

This technique allows having a dynamic m, saving computational time and memory requirements when the dimension of the Krylov subspace can be decreased or increased but in a controlled way. The package (with the implemented algorithm) was evaluated using numerical examples and compared with the existing Matlab implementation. The results showed that the proposed implementation in PETSc is able to emulate the results up to the compared dimensions, in terms of the behavior of the residual as a function of the number of iterations, with better computational performance.

References

[1] Yousef Saad. **Iterative Methods for Sparse Linear Systems**. SIAM, 2003. ISBN: 978-0-89871-534-7.

[2] Christian E. Schaerer, Eugenius Kaszkurewicz, and Norberto Mangiavacchi. "A multilevel Schwarz shooting method for the solution of the Poisson equation in two dimensional incompressible flow simulations". In: **Applied Mathematics and Computation** 153.3 (2004), pp. 803–831. DOI: https://doi.org/10.1016/S0096-3003(03)00679-9.

[3] Rolando Cuevas Nuñez, Christian E. Schaerer, and Amit Bhaya. "A proportional-derivative control strategy for restarting the GMRES(m) algorithm". In: **Journal of Computational and Applied Mathematics** 337 (2018), pp. 209–224. DOI: https://doi.org/10.1016/j.cam.2018.01.009.

[4] Ed Bueler. **PETSc for Partial Differential Equations: Numerical Solutions in C and Python**. Software, environments and tools. SIAM, 2020. ISBN: 978-1-611976-30-4.

[5] "A simple strategy for varying the restart parameter in GMRES(m)". In: **Journal of Computational and Applied Mathematics** 230.2 (2009), pp. 751–761. ISSN: 0377-0427. DOI: https://doi.org/10.1016/j.cam.2009.01.009. URL: https://www.sciencedirect.com/science/article/pii/S0377042709000132.

[6] Juan C. Cabral, Christian E. Schaerer, and Amit Bhaya. "Improving GMRES(m) using an adaptive switching controller". In: **Numerical Linear Algebra with Applications** (2020). DOI: https://doi.org/10.1002/nla.2305.