A proportional-derivative control strategy for varying the restart parameter in GMRES($m$)

Rolando Cuevas†
Christian E. Schaerer‡
Amit Bhaya‡

†Polytechnic, National University of Asuncion,
Campus de la UNA, P.O.Box: 2111 SL, San Lorenzo, Paraguay.
‡Department of Electrical Engineering, COPPE, Federal University of Rio de Janeiro,
Rio de Janeiro 21945-970, Brazil.
E-mail: rcuevas@pol.una.py, cschaer@pol.una.py, amit@nacad.ufrj.br

ABSTRACT

The Generalized Minimal Residual (GMRES) method is one of the most popular algorithms for the solution of large, sparse and unsymmetric linear systems of equations $Ax = b$ [6]. The idea behind the GMRES is to construct an orthogonal matrix $V_k$ such that its $k$ columns span a Krylov subspace. The time and memory costs of computing $V_k$ are proportional to $k$, so that, as $k$ grows, the computational cost of GMRES becomes unreasonably high. Thus, in practice, the restarted GMRES method (denoted as GMRES($m$)) is used, with $m$ small. In each cycle $j$, GMRES($m$) uses an initial residual as a starting guess for GMRES which constructs a Krylov subspace of dimension $m$ and computes a new residual. Until a stopping criterion is met, GMRES($m$) uses the computed new residual as the starting residual for the next cycle, i.e. the next call to a GMRES routine.

Normally, at each cycle, the restart parameter $m$ is set to a constant value (practitioners usually set $m = 30$). However, if the appropriate $m$ is not chosen, the convergence of the GMRES($m$) algorithm is not guaranteed [2], and the method may experience stagnation or slow convergence. Unfortunately, it is difficult to know how to choose $m$ a priori and several papers [1, 3, 7, 4, 5, 9] have proposed different adaptive choices of the parameter $m$. Roughly speaking, these adaptive strategies or rules can be categorized into three groups: (a) simple rules, which however contain empirically chosen parameters that are hard to guess/estimate [7, 4]; (b) rules involving non-trivial calculations of eigenvalues or zeros of polynomials [5, 9] and (c) simple empirical rules [1, 3].

In this paper, we regard the GMRES($m$) method as a control problem, in which the parameter $m$ is the controlled variable and propose a new control-inspired strategy for choosing the parameter $m$ adaptively at cycle $j$, using a discrete proportional-derivative controller which has the form:

$$m_{j+1} = m_j - \alpha \left( \frac{r_j}{r_{j-1}} \right) + \beta \left( \frac{r_j}{r_{j-1}} - \frac{r_{j-2}}{r_{j-1}} \right),$$

(1)

where $\alpha$ and $\beta$ are chosen experimentally. The advantage of this method, named PD-GMRES, is that only a few additional vectors need to be stored and the controller has the capacity to increase the dimension of the Krylov subspace if any convergence problem is detected. Note, however, that empirical parameter choices (of $\alpha$, $\beta$) still need to be made.

Numerical comparisons between the proposed PD-GMRES method and other proposals for choosing $m$ ([1, 3]) are presented. The test problems, chosen from the University of Florida’s Matrix Repository [8], are: (1) add20, (2) edde1, (3) circuit, (4) fpga_trans_01, (5) orsirr_1, (6) orsreg_1, (7) pde2961, (8) raefsky1, (9) raefsky2, (10) rdb2048, (11) sherman4, (12) steam2, (13) wang2, (14) watt_1, (15) young3c. For problems with an unspecified right hand side, $b$
was generated randomly using a uniform distribution with values between the minimum and maximum values in $A$.

Figure 1: Bar graph of relative execution times $t_{\text{algorithm}}/t_{\text{GMRES(30)}}$ (vertical axis) versus problem instance (horizontal axis). A bar height less than one indicates an execution time better than GMRES(30).

—For problem 15, only the proposed PD-GMRES algorithm converged. Alg 1 is the algorithm presented in Baker et al. [1], while Alg 2 is the one presented in Gonzalez et al. [3].

Figure 1 shows the results of a comparison between the proposed PD-GMRES and the algorithms of group (c). The bars correspond to $t_{\text{algorithm}}/t_{\text{GMRES(30)}}$ (where $t_{\text{algorithm}}$ is the algorithm running time) for each problem. Notice that values of $t_{\text{algorithm}}/t_{\text{GMRES(30)}} < 1$ corresponds to an execution time better than that of GMRES(30). It is important to mention that all algorithms converged for all test problems, except for problem 15, for which only the proposed PD-GMRES converged.

It is important to emphasize that the proposed PD-GMRES algorithm is easy to implement and has a better rate of convergence than GMRES(30) and the two other algorithms tested, for all but two (problems 7 and 10) of the 15 test problems. In addition, the PD-GMRES algorithm proposed here also avoided slow convergence and stagnation on this test set, so that these preliminary results should be considered to be very encouraging.

**Keywords**: Adaptive GMRES($m$), proportional derivative (PD) controller, control formulation

**References**


