A serial implementation of a matrix-free uniparametric LU-SGS preconditioner

Rafael S. Coelho* **Lucia Catabriga**
Universidade Federal do Espírito Santo - Departamento de Informática
29075-910, Vitória, ES
E-mail: rafaelsantoscoelho@gmail.com, luciac@inf.ufes.br

ABSTRACT

Keywords: *Jacobian-free inexact Newton-Krylov method, Matrix-free preconditioners, Portable and Extensible Toolkit for Scientific Computation library*

1 - Introduction

For many years and up until nowadays, the Jacobian-free inexact Newton-Krylov (JFNK) method has been a regular choice for tackling large-scale nonlinear problems that require the solution to sparse linear systems. It is known to be advantageous for showing reasonable efficiency, ease of parallelization and mainly for delivering the user from the need to explicity form and store the Jacobian matrix, therefore saving computer memory and avoiding programming errors. That is because, in most numerical algorithms, the Jacobian matrix is only employed within matrix-vector operations, so given $v \in \mathbb{R}^n$, some smooth nonlinear vector function $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ and its corresponding Jacobian matrix $\mathcal{J}(z)$, evaluated at $z \in \mathbb{R}^n$, the product $\mathcal{J}(z)v$ can be approximated by

$$\mathcal{J}(z)v \approx \frac{\mathcal{F}(z + \epsilon v) - \mathcal{F}(z)}{\epsilon}, \tag{1}$$

where $\epsilon > 0$ is a small perturbation. The downside to this method is that the approximations made to compute the matrix-vector products often incur precision errors which may slow down or preclude the convergence of many linear solvers. One practical approach to mitigate that aspect consists of applying preconditioners to speed up the solution process of linear systems. Matrix-free preconditioners, in such scenario, come off particularly appealing because they do not demand the actual assembling of the preconditioning matrix. Surprisingly though, there seems to be little effort on the part of the scientific community towards devising good and general-purpose matrix-free preconditioners for the JFNK method. This work aims at empirically analyzing the performance of the serial JFNK method combined with an implementation of a matrix-free uniparametric LU-SGS (Lower-Upper Symmetric Gauss-Seidel) [1] preconditioner using the PETSc (Portable and Extensible Toolkit for Scientific Computation) library [2].

2 - Methodology

The problem at hand revolves around solving a system of nonlinear equations denoted by $\mathcal{F}(x) = 0$. When using the JFNK method, in each nonlinear iteration we must solve a linear system of the type $P^{-1}\mathcal{J}(x_k)s_k = -P^{-1}\mathcal{F}(x_k)$, where x_k is the kth approximate solution, s_k is the kth Newton correction, $\mathcal{J}(x_k) = L + D + U$, $P = (D + \omega L)D^{-1}(D + \omega U)$ is the preconditioning matrix and $\omega \in \mathbb{R}^*_+$ is a relaxation parameter evaluated every nonlinear iteration by

*PIBIC/FAPES Scientific Initiation scholarship holder means of the recurrence relation

$$\omega_{j+1} = \omega_j - \left[1 + \frac{1}{2}\left(\frac{f''(\omega_j)^2}{f'''\left(\omega_j - \frac{f'(\omega j)}{3f''(\omega j)}\right) f'(\omega_j)} - 1\right)^{-1}\right] \frac{f'(\omega_j)}{f''(\omega_j)} \tag{2}$$

for $j \geq 0$, $\omega_0 = 1$ and where $f(\omega) = a_2\omega^4 + a_1(\omega - 1)\omega^2 + a_0(\omega - 1)^2$, $a_2 = \|LD^{-1}U\mathcal{F}(x_k)\|^2$, $a_1 = 2[(L+U)\mathcal{F}(x_k)]^T[LD^{-1}U\mathcal{F}(x_k)]$ and $a_0 = \|(L+U)\mathcal{F}(x_k)\|^2$. We considered an instance of the Bratu-Gelfand [3] test-problem that commonly models solid fuel combustion phenomena and is described by

$$\triangle u + 4e^u = 0, \tag{3}$$

where $\triangle$ is the Laplacian operator and $e \approx 2.718$ is the Euler's number. Furthermore, homogeneous Dirichlet boundary conditions were enforced in a 128×128 structured mesh, in the domain $\Omega = (0,1) \times (0,1)$ and the restarted GMRES (Generalized Minimal Residual) method was chosen to numerically solve the linear systems. We ran each test five times on a Intel Dual Core 1.80GHz CPU, 2MB cache, 2GB RAM, under Fedora Core 8 GNU/Linux and the results were averaged. The analytic derivatives in (3) were discretized via forward finite differences and we manipulated (1) whenever the preconditioning matrix (or parts of it) was needed.

3 - Experimental results

According to Tab. 1 and contrary to expectations, the results showed that the preconditioner did not contribute to a substantial reduction of linear iterations with respect to the unpreconditioned NK method. Nevertheless, when compared to the unpreconditioned JFNK method, there was a decrease of 71.11% in the total number of linear iterations for the JFNK/ULU-SGS (optimal ω) case and 62.02% for the JFNK/ULU-SGS ($\omega = 1$) case. Moreover, we observed that the action of the preconditioner caused the runtimes to grow which is due to the fact that in each linear iteration the application of the preconditioner requires the solution of two triangular systems. In other words, the computational overload that the preconditioner itself imposed severely outweighed the benefit of accelerating the rate of convergence. Ideas for future works include mixing the JFNK method with graph coloring heuristics and also investigating the impact of frozen preconditioning and preconditioner updates techniques on the overall runtime.

Table 1: RT_{tot} : total runtime in seconds, MF_{tot} : total megaflops ($\times 10^6$), RT_{prec} : preconditioning runtime in seconds, MF_{prec} : preconditioning megaflops ($\times 10^6$), NK : classic inexact Newton-Krylov method

Method/Preconditioner	Iterations	RT_{tot}	RT_{prec}	MF_{tot}	MF_{prec}
JFNK/ULU-SGS ($\omega = 1$)	188	281.1	271.4	587.4	168.3
JFNK/ULU-SGS (optimal ω)	143	197.3	179.3	374.9	137.3
Unpreconditioned JFNK	495	26.6	-	2130.9	-
Unpreconditioned NK	**160**	**2.6**	-	**320.4**	-

References

[1] K. Jisheng and L. Yitian, A uniparametric LU-SGS method for systems of nonlinear equations, *Applied Mathematics and Computation*, 189 (2007) 235-240.

[2] S. Balay and W. D. Gropp and L. C. McInnes and B. F. Smith, Efficient Management of Parallelism in Object Oriented Numerical Software Libraries, *Modern Software Tools in Scientific Computing*, Birkhäuser Press, (1997) 163-202.

[3] R. Buckmire, Application of a Mickens Finite-diference Scheme to the Cylindrical Bratu-Gelfand Problem, *Wiley Periodicals*, (2004) 327-337.