Journal of Computational Physics 352 (2018) 265–284

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A stopping criterion for the iterative solution of partial differential equations

Kaustubh Rao [a], Paul Malan [a], J. Blair Perot [b,*]

[a] Computational Fluid Dynamics Group, Simulia Inc., Johnston, RI, 02919, USA
[b] Theoretical and Computational Fluid Dynamics Laboratory, University of Massachusetts, Amherst, MA, 01003, USA

ARTICLE INFO

Article history:
Received 12 July 2017
Accepted 19 September 2017
Available online 28 September 2017

Keywords:
Convergence estimate
Error estimate
Fluid dynamics
Stopping criteria

ABSTRACT

A stopping criterion for iterative solution methods is presented that accurately estimates the solution error using low computational overhead. The proposed criterion uses information from prior solution changes to estimate the error. When the solution changes are noisy or stagnating it reverts to a less accurate but more robust, low-cost singular value estimate to approximate the error given the residual. This estimator can also be applied to iterative linear matrix solvers such as Krylov subspace or multigrid methods. Examples of the stopping criterion's ability to accurately estimate the non-linear and linear solution error are provided for a number of different test cases in incompressible fluid dynamics.

1. Introduction

The stopping criterion for iterative nonlinear equation solvers is an implementation detail that gets less attention than it warrants. This small aspect of the iterative solver can have an outsized impact on the overall performance of the implementation because performing additional iterations due to overly conservative stopping estimates wastes computational resources. In most situations iteration of a partial differential equation (PDE) problem until errors or residuals are even close to machine precision is computational overkill because the solution already has some level of error due to the discretization process. An efficient solver implementation will stop the iterations once a level of error specified by the user is obtained. Therefore, an efficient implementation of an iterative solution method requires an estimate of the solution error.

Fig. 1 juxtaposes the stopping criterion proposed in this work with the classic 3 decade reduction in residual stopping criterion, in order to highlight the importance of a good stopping criterion. A well-chosen stopping criterion can result in either computational savings or improved solution quality. In this example the user considers a relative solution error below 1% to be sufficiently converged for their needs. The figure depicts the progress of the relative solution error (error norm/solution norm) and the normalized residual (residual/initial residual) as a function of the iteration number for a turbulent diffuser flow simulation (described in section 3.2).

Fig. 1(a) shows the evolution of the x-momentum solution. In this case the initial guess (potential flow) is sufficiently good that a 3-order reduction in the residual (thick black line) results in excessive iterations. In this particular case, the savings produced by stopping at the right time (circle on the thin red line) is around 20%. But it can often be much larger. Fig. 1(b) shows the evolution of turbulent kinetic energy for the same problem. In this case the initial guess for the

* Corresponding author.
E-mail address: perot@umass.edu (J.B. Perot).

https://doi.org/10.1016/j.jcp.2017.09.033

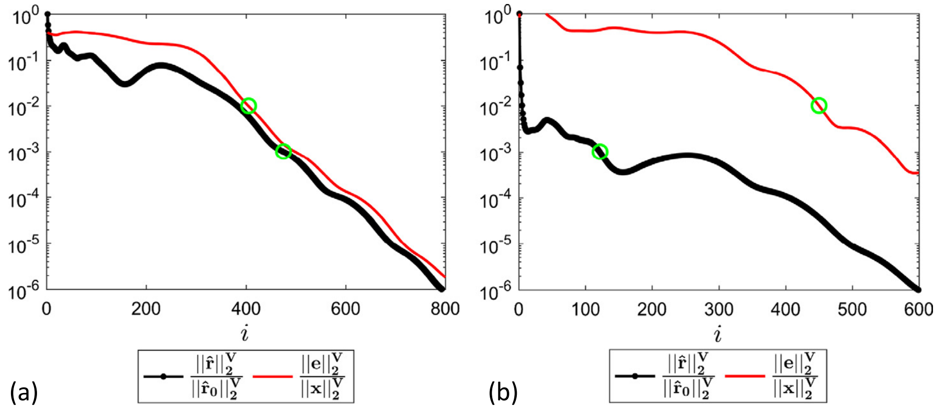

Fig. 1. Normalized residual (thick black lines) and relative error (thin red lines) for the diffusor problem described in section 3.2 (a) x-momentum convergence showing the situation when waiting for a 3 order drop in residuals leads to excessive iterations, (b) turbulent kinetic energy convergence showing the situation when the residuals drops by 3 orders but the error in the solution is still very high.

turbulence solution is bad (it is initialized to a constant value) and a 3-order reduction is easily achieved resulting in an early exit but a final turbulence solution with an excessively large error.

It is important to note that errors in the solution of PDEs arise from different sources. Classical error estimation (sometimes called discretization error, or local error estimation) is used to implement grid refinement and coarsening algorithms. This sort of error estimation is concerned with the error between a discrete approximation and its continuous counterpart (the original PDE). Classical error estimation is a broad and well developed area of research (see references [1–3] for some examples) but is not the area of discussion in this paper. This work focuses on the iterative error (sometimes called linearization error) that exists when an iterative method is converging to its discrete target solution. In the context of the present work, the PDE problem is already considered to be discretized, and the exact solution (when computing and discussing errors in this work) will be considered to be the exact solution to the *discrete* PDE problem. We are not concerned in this work, with how well that discrete solution approximates the continuous PDE solution (the realm of mesh adaptation). We are instead concerned with how close our current iterative solution is to the exact solution to the given discrete system.

Many stopping criteria are based on a norm of the residual vector [4–6]. But stopping iteration based solely on the residual is neither a safe nor a robust solution as shown in Fig. 1. The magnitude of any residual is totally arbitrary (see section 2.4 for details). Normalizing the residual can remove the magnitude problem, but (as shown in Fig. 1) is still problematic. If the initial guess is a good one, iteration may be incapable of achieving the prescribed relative reduction in the residual (due to reaching round-off). Or the iterations may just waste resources (as in Fig. 1(a)). If the initial guess is very bad, the iterative procedure will exit prematurely, when the solution error is still large (as in Fig. 1(b)).

Stopping criteria based on the residual and additional information about the problem perform better but still have issues. For example, a classic stopping criterion is to use the condition number of the Jacobian times the normalized residual to guarantee a certain reduction in the relative error ($\frac{\mathbf{e}}{\mathbf{x}} \leq \kappa \frac{\mathbf{r}}{\mathbf{r}(0)}$). There are two problems with this approach. First it requires a condition number estimate. Second, and much more importantly, the bound being used in this approach is excessively conservative so this stopping criterion can cause excessive iteration. Note that the problem of excessive iteration is more and more likely as mesh sizes (and therefore condition numbers) get larger. It is therefore only more recently, with the advent of large 3D meshes (and large condition numbers) that the inadequacy of this classic stopping criterion has become particularly pressing. The ultimate cost saving for high performance computing problems such as the direct numerical simulations in references [7] and [8] can be as large as 50,000 CPU hours.

This paper takes an unconventional approach to developing the stopping criterion, and abandons the residual (or its norm) as a useful starting point. Instead, the proposed stopping criterion looks at the size of the solution changes between each iteration. Error estimation based on progress (prior solution changes) is a non-trivial task because small changes in the solution do not necessarily mean the solution is converged, it may simply indicate that this is a difficult problem to solve and the method is converging slowly. Nevertheless, this approach is tenable and is not entirely previously unknown. In the past, classic linear iterative solvers such as Jacobi iteration and Gauss–Seidel iteration sometimes used prior solution changes and an extrapolation hypothesis to estimate progress [9,10]. Most modern matrix solution methods (such as Krylov subspace methods) typically have a far more erratic convergence behavior in both the residuals and the solution increments than Jacobi or Gauss–Seidel iteration. This has essentially led to the total abandonment of error estimation via progress extrapolation (though an exception is ref. [11]). However, in this work we will rejuvenate the extrapolation approach by using a robust and parameter-free smoothing approach. We will also focus primarily on the outer nonlinear iterations.

Section 2 of this paper presents the mathematical background for this work. It is shown that for PDE problems, certain vector and matrix norms are particularly attractive for performing the error estimation. Section 3 describes the test cases