Journal of Computational Physics 321 (2016) 55–75

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A monolithic homotopy continuation algorithm with application to computational fluid dynamics

David A. Brown *, David W. Zingg

University of Toronto Institute for Aerospace Studies, Toronto, Ontario, M3H 5T6, Canada

ARTICLE INFO

Article history:
Received 24 September 2015
Received in revised form 5 April 2016
Accepted 17 May 2016
Available online 25 May 2016

Keywords:
Homotopy
Continuation
Globalization
Newton–Krylov
Computational fluid dynamics

ABSTRACT

A new class of homotopy continuation methods is developed suitable for globalizing quasi-Newton methods for large sparse nonlinear systems of equations. The new continuation methods, described as monolithic homotopy continuation, differ from the classical predictor–corrector algorithm in that the predictor and corrector phases are replaced with a single phase which includes both a predictor and corrector component. Conditional convergence and stability are proved analytically. Using a Laplacian-like operator to construct the homotopy, the new algorithm is shown to be more efficient than the predictor–corrector homotopy continuation algorithm as well as an implementation of the widely-used pseudo-transient continuation algorithm for some inviscid and turbulent, subsonic and transonic external aerodynamic flows over the ONERA M6 wing and the NACA 0012 airfoil using a parallel implicit Newton–Krylov finite-difference flow solver.

1. Introduction

Newton's method is a popular fixed-point iterative method for solving systems of nonlinear equations due to its quadratic convergence rate. When combined with an iterative Krylov solver such as the Generalized Minimal Residual method (GMRES) for solving the linear systems of equations that arise, the method becomes an inexact Newton method, but the convergence rate can still be super-linear or even quadratic [10]. However, it is well-known that Newton's method will usually not converge without a suitable initial guess. Obtaining the initial guess is referred to as globalization and is performed using a continuation method. Thus, solving a nonlinear system of equations with an inexact Newton method generally consists of two phases: a globalization phase and an inexact Newton phase.

When solving a large sparse nonlinear algebraic system of equations, such as those arising in computational fluid dynamics (CFD) when the steady flow equations are discretized in space, the globalization phase can be computationally very expensive, often more expensive than the inexact Newton phase. At the current state of CFD development and computer technology, many CFD problems of interest remain computationally demanding so the efficiency of the nonlinear equation solver is of engineering importance for CFD practitioners. In addition, the algorithm may not always converge to the solution; some algorithms may stall or become unstable. The reliability of the algorithm to converge to the solution is known as robustness and is also of importance. Many algorithms contain tunable parameters to allow for a user to balance speed and robustness. However, regular tuning of parameters is undesirable and is also a consideration for algorithm design: parameters should be few, intuitive, and should not need regular retuning.

* Corresponding author.
E-mail address: utiasdavid.brown@mail.utoronto.ca (D.A. Brown).

http://dx.doi.org/10.1016/j.jcp.2016.05.031
0021-9991/

The most common continuation method for globalizing steady CFD problems is pseudo-transient continuation (PTC) [4,8,23,24]. This is a method which imitates physical time marching, though time-accuracy is not required. A viable alternative to PTC is homotopy continuation. Homotopy continuation methods are discrete algorithms based on continuous deformations known as homotopies. The most common application of homotopy continuation in the literature has been for specialized study of systems with certain mathematical properties. For example, homotopy continuation can be used to study systems where multiple solutions exist [31,35] or where solutions may be unstable [34,22] in a more methodical and reliable way than can be done with a method such as PTC. Homotopy continuation has also been applied to facilitate the solution to CFD problems at high Reynolds numbers by solving the same problem at a lower Reynolds number and gradually increasing the Reynolds number [5].

Hicken and Zingg [20] and Hicken et al. [18] designed a homotopy continuation algorithm which they found could be used as a general globalization method for a Newton–Krylov external aerodynamic flow solver developed by Hicken and Zingg [19] and Osusky and Zingg [29]. They found that the continuation method, which they termed *dissipation-based continuation* since the homotopy was constructed based on a dissipation operator, was more efficient than PTC in some cases, especially inviscid flows, but was not as competitive for turbulent flows. To model the turbulent flows, they used the Reynolds-averaged Navier–Stokes equations with Spalart–Allmaras [33] turbulence model (RANS-SA). We subsequently made several improvements to their formulation, including the implementation of a classical predictor–corrector approach adapted to the Newton–Krylov algorithm [3]. For the cases that we investigated, we found that the homotopy continuation method performed better than PTC for some three-dimensional inviscid and laminar cases, as well as some two-dimensional RANS cases, but did not perform as well for some three-dimensional RANS cases.

Homotopy continuation algorithms based on predictor–corrector methods are prevalent in the homotopy literature; we refer to Allgower and Georg [1] and the references therein. In a context where the homotopy continuation algorithm is needed to be as efficient as possible, predictor–corrector algorithms can be wasteful because, in order to ensure that the algorithm is convergent, parameters are chosen such that the corrector phase is often over-solved, by which we mean that much of the work done in the corrector phase leads to only marginal improvement to the quality of the predictor update. In addition, depending on the predictor method and corrector method employed, both of these phases may require the solution to linear systems of equations and the algorithm can be made more efficient if these linear solves are combined. These potential gains in computational efficiency have been the motivation for developing the monolithic homotopy (MH) continuation algorithm. The MH algorithm was developed based on a principle known as dynamic inversion [14–16], which is the mathematically analogous problem of inexactly predicting an implicitly-defined trajectory traced by a dynamic system. In addition to the gains in efficiency, combining the predictor and corrector phases into a single phase reduces the number of user parameters, simplifying user control. The objective of this paper is to present the monolithic homotopy continuation algorithm, to characterize its performance on some compressible flow problems, and to compare its performance to both a typical PTC algorithm and a homotopy continuation algorithm based on a predictor–corrector method.

2. Flow solver

The flow solver to which the monolithic homotopy continuation algorithm is applied is a Newton–Krylov–Schur parallel implicit flow solver based on a finite-difference [26] discretization applicable to multi-block structured grids. The finite-difference discretization is based on the SBP-SAT [6,9,12,25] approach, which uses Summation-By-Parts (SBP) operators to represent the discrete derivatives and Simultaneous Approximation Terms (SATs) to enforce the boundary conditions and couple the flow equations at block interfaces. The flow solver originated as an inviscid flow solver due to Hicken and Zingg [19] and was extended to the RANS-SA equations by Osusky and Zingg [29]. The flow solver can be used for both subsonic and transonic operating conditions. For transonic cases, a first-order dissipation operator is included with a pressure sensor [21] for shock capturing. To parallelize the flow solver, the domain is decomposed into blocks. Parallel preconditioning of the Krylov solver is performed using the Schur complement method [32] with block incomplete lower-upper (ILU) preconditioning applied to the domain blocks. The specific type of ILU factorization used in the current study is known as ILU(p) [32], where p is the fill level. The ILU(p) factorization is constructed based on an approximate Jacobian matrix using nearest neighbour nodes only. Since the Schur complement preconditioner can vary slightly throughout the Krylov solution process, a flexible variant of the Krylov solver GMRES is used, which is termed Flexible Generalized Minimal Residual, or FGMRES [32].

3. Jacobian-free Newton–Krylov method

Consider a nonlinear algebraic system of equations, represented by

$$\mathcal{F}(\mathbf{q}) = \mathbf{0}, \qquad (1)$$

$$\mathcal{F}: \mathbb{R}^N \to \mathbb{R}^N, \ \mathbf{q} \in \mathbb{R}^N.$$

The update due to Newton's method, when applied to this system of equations, is calculated by solving the linear system of equations