Computers and Fluids 140 (2016) 406–421

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Smoothed truncation error in functional error estimation and correction using adjoint methods in an unstructured finite volume solver

Mahkame Sharbatdar[1,*], Alireza Jalali[2], Carl Ollivier-Gooch[3]

Department of Mechanical Engineering, The University of British Columbia, Vancouver, BC, V6T 1Z4, Canada

ARTICLE INFO

Article history:
Received 22 March 2016
Revised 9 September 2016
Accepted 19 October 2016
Available online 19 October 2016

Keywords:
Output functional correction
Adjoint methods
Smooth estimation of truncation error
Unstructured meshes
Finite volume methods

ABSTRACT

The goal of this paper is to develop a reliable truncation error estimator for finite-volume schemes and to use this truncation error estimate to correct the output functional of interest. The rough modes in the truncation error for unstructured mesh are dominant and if *p*-truncation error estimation is used, functional correction has a poor performance. So, we are trying to obtain a smooth estimate of the truncation error to improve the performance of output error estimation. The correction term is based on the truncation error and the adjoint solution. Both discrete and continuous adjoint solutions are used for correcting the functional. Our results for a variety of governing equations suggest that the smoothing scheme can improve the correction process significantly.

1. Introduction

In the field of computational aerodynamics and fluid dynamics, lift and drag are output functionals of interest and the desire for efficient computational methods producing reliable and accurate lift and drag values drives algorithm research in the field. The adjoint method has played an important role in this context because of the great flexibility it offers with regard to the physics model and to the definition of output functionals. The history of the use of adjoint equations in fluid dynamics design goes back to the work by Pironneau [35] and particularly in the field of computational aerodynamics design to the work by Jameson [16]. Since then, adjoint methods have been used for design applications for both internal and external flows [15,16,18,19,23,36]. The adjoint theory was first presented in the context of linear algebra by using the algebraic equations obtained from the discretization of the original problem. This is the basis for the discrete adjoint approach. The continuous adjoint approach, on the other hand, is formulated based on the adjoint PDE which is discretized and solved independently [11].

The adjoint problem also plays a key role in estimating and reducing the error in output functionals. Within the context of finite element methods, output functional correction has been outlined by Becker and Rannacher [5] and Larson and Barth [20] based on structural finite element methods [2]. The adjoint-based error correction technique developed by Pierce and Giles [33] extends the inherent super-convergence properties of finite element methods to cover numerical results obtained from finite difference and finite volume methods without natural super-convergence properties. Moreover, the technique can be used to improve the accuracy of super-convergent functionals obtained from finite element methods by constructing smoother, higher-order interpolants of the primal and adjoint solutions [33]. This method for output error estimation and correction for two-dimensional inviscid flows was applied by Venditti and Darmofal [41] for a second-order finite volume discretization.

For numerical efficiency, computing the functional value to a higher order of accuracy than the primal solution is advantageous. A super-convergent estimate of the functional may be obtained by computing the leading error term in the original functional estimate and using this as a correction. Pierce and Giles [34] showed that the error in the functional value based on the reconstructed primal solution can be expressed as a function of the truncation error, implying that truncation error estimation is required for output error estimation.

Truncation error is defined as the difference between the continuous PDE and the finite discretized equation. Historically,

* Corresponding author.
E-mail addresses: mahkame@interchange.ubc.ca (M. Sharbatdar), arjalali@interchange.ubc.ca (A. Jalali), cfog@mech.ubc.ca (C. Ollivier-Gooch).

[1] PhD Candidate.
[2] PhD Candidate.
[3] Professor.

http://dx.doi.org/10.1016/j.compfluid.2016.10.019
0045-7930/

truncation error analysis for structured mesh schemes is a routine application of Taylor series analysis, while analysis of unstructured mesh schemes has lagged behind. Recent work on analysis of unstructured mesh schemes includes accuracy of discretization schemes on irregular grids [8], error comparisons for cell-centered and vertex-centered discretizations [9], and Taylor-based accuracy assessment of cell-centered discretizations [14]. Ollivier-Gooch and Van-Altena [31] performed Taylor series truncation error analysis for the Laplacian on regular triangular meshes, and this was extended to general stencils [13,14].

The behavior of truncation error on unstructured meshes is completely different from structured meshes. Several researchers, including Diskin and Thomas [7,9], and Jalali and Ollivier-Gooch [14], have demonstrated that the truncation error for unstructured finite volume schemes is asymptotically larger than the discretization error, which in turn is typically of the same order as the solution approximation error in the scheme. This behavior is in contrast with the structured mesh case for which the truncation error has the same asymptotic order of accuracy as the discretization error. Another feature of the truncation error for unstructured mesh schemes is its noisy appearance, caused by the discontinuous jump of the coefficients of the terms in the Taylor series expansion of the error from one control volume to another; this is in contrast with structured mesh schemes where the truncation error is smooth. These two features of the truncation error for unstructured mesh finite volume schemes, non-smoothness and large magnitude, are related to each other by the eigensystem of the discrete problem as shown by Ollivier-Gooch and Roy [30]. Sharbatdar and Ollivier-Gooch [39] have shown by eigenanalysis of the truncation error that the rough modes dominate the unstructured mesh truncation error. The dominance of the rough modes causes difficulties in output error estimation. We use a higher-order flux integral as an estimation of the truncation error, the p−TE method, and the rough modes are responsible for the difficulty in accurately estimating discrete truncation error. We try to develop a smooth estimation of the truncation error which can be exploited in output functional correction. Venditti and Darmofal presented an error estimation strategy based on the adjoint formulation for estimating errors in functional outputs for one-dimensional problems and their error estimation procedure was applied to a standard, second-order finite volume discretization [40].

The goal of this paper is to investigate continuous truncation error estimates by producing a continuous approximation to the finite volume discrete solution and estimating truncation error as the higher-order residual of the PDE for this continuous function. The truncation error can be used to find the leading error term in the functional and we compare the performance of continuous and discrete adjoint approaches in the correction procedure. We begin by briefly describing our higher order finite volume flow solver followed by adjoint problem definition and implementation in Section 3. The output functional calculation process and error correction calculation is described in Section 4. We will show that the correction term is a function of the truncation error and our method for estimating and improving the truncation error is illustrated in Section 5. Several test cases with different governing equations and physical behaviors are shown in Section 6. We show the capability of the correction scheme we developed for unstructured mesh finite volume scheme for both scalar equations and system of equations; advection and Poisson as examples of scalar equations and Euler and Navier–Stokes for system of equations.

2. Higher order finite volume flow solver

To discretize the flow equations using the finite volume method, the governing equations should be recast in fully conservative form as:

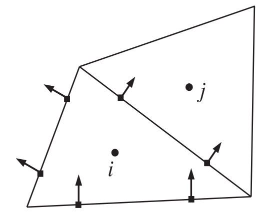

Fig. 1. Schematic illustration of Gauss quadrature points for third and fourth-order .

$$\frac{\partial \vec{U}}{\partial t} + \nabla \cdot \vec{F} = 0 \tag{1}$$

where U denotes the solution vector and F is the flux vector. Integrating Eq. (1) over an arbitrary control volume and using the divergence theorem gives the 2D finite volume formulation of the governing equations in the form of a volume and a surface integral

$$\iint_{CV} \frac{\partial \vec{U}}{\partial t} dA + \oint\!\!\!\oint_{CS} \vec{F} \cdot \hat{n}\, ds = 0 \tag{2}$$

where $\hat{n}$ represents the outward unit normal vector, CV and CS are the control volume and control surface (the boundary of the control volume), respectively, and ds is the infinitesimal. Assuming that the discretized physical domain does not change in time, the time derivative can be brought out from the integral in Eq. (2), giving an evolution equation for the vector $\overline{U}_i$, the control volume average solution,

$$\frac{d\overline{U}_i}{dt} = -\frac{1}{A_{CV_i}} \oint_{CS_i} \vec{F} \cdot \hat{n}\, ds \tag{3}$$

The right hand side is called the flux integral or residual and represents the spatial discretization of the same control volume. To compute the flux integral for each control volume, the numerical flux is computed at Gauss quadrature points based on the reconstructed solution. For second-order, one quadrature point is required and for third and fourth order flux integration, two quadrature points per edge are necessary. Fig. 1 shows the higher-order quadrature points for the flux integration schematically.

To obtain a single flux from two different values at each Gauss point, we use Roe's scheme [38] for inviscid fluxes; for viscous fluxes, we average the gradients and add a jump term [27]. The control volume flux integral in Eq. (3) is approximated as the summation of flux integrals over the edges and can be re-written as:

$$\frac{d\overline{U}_i}{dt} = -R_i\left(\overline{U}_i\right) \tag{4}$$

where $R_i\left(\overline{U}_i\right)$ is called the residual.

For steady-state problems, $R\left(\bar{U}_i\right) = 0$. One could solve the non-linear system of equations by the direct application of Newton's methods for steady-state problems. However, Newton's method will diverge if the initial guess is too far from the real solution. As a result, the non-linear system is augmented by a damping term which mimics the time derivative in the original time-dependent equations and prevents the evolution of non-physical solution at each iteration. This is called the implicit pseudo time-stepping method [22]. For implicit Euler integration of the discretized equation in time, the next time level solution for both the space and the time discretizations are used. Assuming that the solution average vector at the current time level n is denoted by $\overline{U}_i^n$, both sides of Eq. (4) should be evaluated at the next time level $n+1$. After linearization, we get

$$\left(\frac{I}{\Delta t} + \frac{\partial R}{\partial \overline{U}}\right)\delta U_i = -R_i(\overline{U}_i^n), \qquad \overline{U}_i^{n+1} = \overline{U}_i^n + \delta U_i \tag{5}$$