

Review of Discretization Error Estimators in Scientific Computing

Christopher J. Roy¹

Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 24061

Discretization error occurs during the approximate numerical solution of differential equations. Of the various sources of numerical error, discretization error is generally the largest and usually the most difficult to estimate. The goal of this paper is to review the different approaches for estimating discretization error and to present a general framework for their classification. The first category of discretization error estimator is based on estimates of the exact solution to the differential equation which are higher-order accurate than the underlying numerical solution(s) and includes approaches such as Richardson extrapolation, order refinement, and recovery methods from finite elements. The second category of error estimator is based on the residual (i.e., the truncation error) and includes discretization error transport equations, finite element residual methods, and adjoint method extensions. Special attention is given to Richardson extrapolation which can be applied as a post-processing step to the solution from any discretization method (e.g., finite difference, finite volume, and finite element). Regardless of the approach chosen, the discretization error estimates are only reliable when the numerical solution, or solutions, are in the asymptotic range, the demonstration of which requires at least three systematically refined meshes. For complex scientific computing applications, the asymptotic range is often difficult to achieve. In these cases, it is appropriate to treat the numerical error estimates as an uncertainty. Issues related to mesh refinement are addressed including systematic refinement, the grid refinement factor, fractional refinement, and unidirectional refinement. Future challenges in discretization error estimation are also discussed.

I. Introduction

A *mathematical model* is defined here as a system of partial differential or integral equations and associated auxiliary equations, along with appropriate initial and boundary conditions, that are used to describe a physical system. In *scientific computing*, one is concerned with finding approximate solutions to this mathematical model, a process that involves the *discretization* of both the mathematical model and the domain. The approximation errors associated with this discretization process are called *discretization errors*, and they occur in every single scientific computing simulation. The discretization error can be formally defined as the difference between the exact solution to the discrete equations and the exact solution to the mathematical model:

$$\varepsilon_h = u_h - \tilde{u} \quad (1)$$

In Equation (1), u_h represents the solution to the discrete equations on a mesh with a representative cell length of h , and \tilde{u} is the exact solution to the mathematical model. Discretization error arises out of the interplay between the chosen discretization scheme for the mathematical model, the mesh resolution, the mesh quality, and the behavior of the solution and its derivatives. Discretization error is the most difficult type of numerical error to estimate reliably and is usually the largest of the four numerical error sources. (The three sources are round-off error, iterative convergence error, and statistical sampling error.)

The discretization error has two components: one that is *locally-generated* and the other that is *transported* from elsewhere in the domain. The transported component is called pollution error by the finite element community (Babuska et al., 1997). This can be shown mathematically by examining the error transport equations (see Section

¹ Associate Professor, Aerospace and Ocean Engineering Department, 215 Randolph Hall, Associate Fellow AIAA.

II.B.1), which can be used to relate the convergence of the numerical method (i.e., the discretization error) to the consistency of the discretization scheme (i.e., the truncation error). The truncation error is the difference between the discrete equations and the mathematical model equations. Thus the discretization error is transported in the same manner as the underlying solution properties (e.g., it can be convected and diffused) and it is locally generated according to the truncation error.

An example of error transport for the Euler equations is shown below in Figure 1, which gives the error in the density for the inviscid, Mach 8 flow over an axisymmetric sphere-cone (Roy, 2003). The flow is from left to right, and large discretization errors are generated at the bow shock wave where the shock and the grid lines are misaligned. In the subsonic (i.e., elliptic) region of the flow immediately behind the normal shock, these errors are convected along the local streamlines. In the supersonic (hyperbolic) regions these errors propagate along characteristic Mach lines and reflect off the surface. Additional error is generated at the sphere-cone tangency point, which represents a singularity due to the discontinuity in the surface curvature. Errors from this region also propagate downstream along the characteristic Mach line. An adaptation process which is driven by the global error levels would adapt to the characteristic line emanating from the sphere-cone tangency point, which is not desired. An adaptation process driven by the local contribution to the error should adapt to the sphere-cone tangency point, thus obviating the need for adaption to the characteristic line that emanates from it.

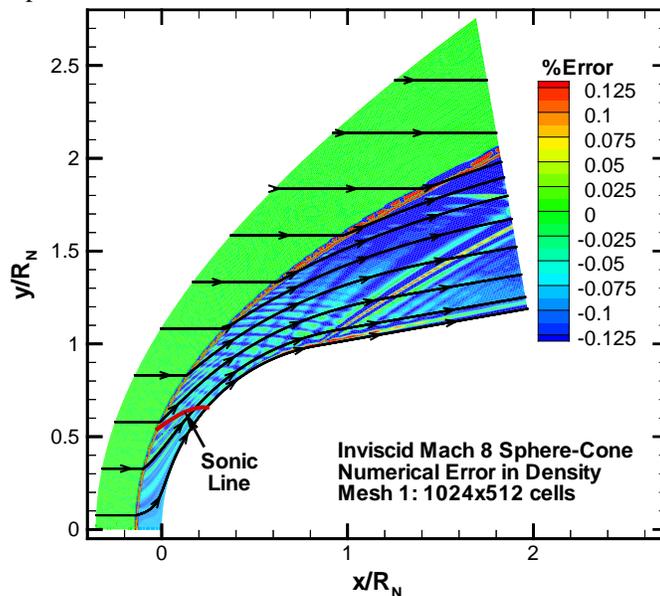


Figure 1. Contours of total estimated discretization error in density for the flow over an inviscid hypersonic sphere cone (from Roy, 2003).

Section II of this paper contains an *overview* and *classification* of several approaches for *estimating the discretization error*. Many of these approaches have arisen out of the finite element method, which due to its nature provides for a rigorous mathematical analysis (Ainsworth and Oden, 2000). Other approaches to discretization error estimation seek to find an estimate of the exact solution to the mathematical model which is of a higher formal order of accuracy than the underlying discrete solution. Once this higher-order estimate is found, it can be used to estimate the discretization error in the solution. While a wide range of approaches are discussed for estimating discretization error, particular attention is focused on *Richardson extrapolation* which relies on discrete solutions on two meshes to estimate the discretization error. The reason for this choice is straightforward: Richardson extrapolation is the only discretization error estimator that can be applied as a post-processing step to both local solutions and system response quantities from any discretization approach (finite difference, finite volume, finite element, etc.). The main drawback is the expense of generating, and then computing another discrete solution on, another systematically-refined mesh. *Systematic mesh refinement* requires the mesh to be refined both uniformly over the domain and consistently with regards to mesh quality and these two conditions are discussed further in Sections II.A.1 and V.

Regardless of the approach used for estimating the discretization error, the *reliability* of the resulting error estimate requires that the underlying numerical solution (or solutions) be in the *asymptotic range* (see Section III). Achieving this asymptotic range can be surprisingly difficult, and confirming that it has indeed been achieved using

the *observed order of accuracy* generally requires at least three discrete solutions. For complex scientific computing applications involving coupled, nonlinear, hyperbolic, multidimensional, multiphysics equations, it is unlikely that the asymptotic range will be achieved without relying on the solution adaptive procedures (e.g., see Roy, 2009; Oberkampf and Roy, 2010).

The most common situation in scientific computing is when the discretization error estimate has been computed, but the confidence in that estimate is either 1) low because the asymptotic range has not been achieved or 2) unknown because three discrete solutions are not available. In these cases, the discretization error is more appropriately characterized as an *epistemic uncertainty* due to the lack of knowledge of the true value of the error. Roache's Grid Convergence Index effectively converts the error estimate from Richardson extrapolation into an uncertainty by providing error bands, and is discussed in Section IV.

Another important topic addressed in this paper is the role of *systematic mesh refinement* (as defined in Oberkampf and Roy, 2010) for Richardson extrapolation-based discretization error estimators and also for assessing the general reliability of all discretization error estimators (Section V). The importance of systematic mesh refinement over the entire domain is discussed, along with approaches for assessing the systematic nature of the refinement. Issues with refinement in space and time, unidirectional refinement, fractional (or non-integer) refinement, and recommendations for refinement versus coarsening are also discussed. This paper concludes with a discussion of some open issues related to discretization error estimation which have not yet been adequately addressed by the research community (Section VI). For more details on methods for estimating discretization error in scientific computing, see Oberkampf and Roy (2010).

II. Approaches for Estimating Discretization Error

There are a number of approaches available for *estimating discretization error*. These methods can be broadly categorized as *a priori* methods and *a posteriori* methods. The *a priori* methods are those that allow a bound to be placed on the discretization error before any numerical solution is even computed. In general, one looks to bound the discretization error by an equation of the form

$$\varepsilon_h \leq C(u)h^p \quad (2)$$

where ε_h is the discretization error, the function $C(u)$ usually depends on various derivatives of the exact solution, h is a measure of the element size (e.g., Δx), and p is the formal order of accuracy of the method. One approach to developing an *a priori* discretization error estimator is to perform a truncation error analysis for the scheme, relate the truncation error to the discretization error (e.g., through a discretization error transport equation), then develop some approximate bounds on the solution derivatives that comprise $C(u)$. The main failing of *a priori* error estimators is that $C(u)$ is extremely difficult to bound and even when this is possible for simple problems, the resulting error estimate greatly over-estimates the true discretization error. *A priori* methods are generally only useful for assessing the formal order of accuracy of a discretization scheme. Current efforts in estimating the discretization error are focused on *a posteriori* methods. These methods provide an error estimate only after the numerical solution has been computed. They use the computed solution to the discrete equations, possibly with additional information supplied by the equations, to estimate the error relative to the exact solution to the mathematical model.

The mathematical formalism that underlies the finite element method makes it fertile ground for the rigorous estimation of discretization error. Beginning with the pioneering studies of Babuska and Rheinboldt (1978a,b), a tremendous amount of work has been done over the last 30 years on *a posteriori* estimation of discretization error by the finite element community (e.g., see Ainsworth and Oden, 2000). The initial developments up to the early-1990s were concentrated on linear, elliptic, scalar mathematical models and focused on the h-version of finite elements. Early extensions of the *a posteriori* methods to parabolic and hyperbolic mathematical models were made by Eriksson and Johnson (1987) and Johnson and Hansbo (1992), respectively. Up to this point, *a posteriori* error estimation in finite elements was limited to analysis of the energy norm of the discretization error, which for Poisson's equation can be written on element k as:

$$\|\varepsilon_h\|_k = \left[\int_{V_k} |\bar{\nabla} u_h - \bar{\nabla} \tilde{u}|^2 dV \right]^{1/2} \quad (3)$$

where $\bar{\nabla}$ is the vector form of the gradient operator.

The finite element method produces the numerical solution from the chosen set of basis functions which minimizes the energy norm of the discretization error (Szabo and Babuska, 1991). Taken locally, the energy norm can provide guidance on where adaptive refinement should occur. Taken globally, the energy norm provides a global measure of the overall optimality of the finite element solution. In the early 1990s, important extensions of *a posteriori* error estimators to system response quantities were found that require the solution to an adjoint, or dual, problem (e.g., Johnson and Hansbo, 1992). For additional information on *a posteriori* error estimation in finite element methods, see Babuska et al. (1986), Whiteman (1994), Ainsworth and Oden (1997, 2000), and Estep et al. (2000). A more introductory discussion of error estimation in finite element analysis is presented by Akin (2005).

In general, the level of maturity for *a posteriori* error estimation methods is strongly problem dependent. All of the discretization error estimators to be discussed here were originally developed for elliptic problems. As a result, they tend to work well for elliptic problems, but are not as well-developed for mathematical models that are parabolic or hyperbolic in nature. The level of complexity of the problem is also an important issue. The error estimators work well for smooth, linear problems with simple physics and geometries; however, strong nonlinearities, discontinuities, singularities, and physical and geometric complexity can significantly reduce the reliability and applicability of *a posteriori* discretization error estimation methods.

There are two types of discretization error estimators discussed in this section. In the first type, an estimate of the exact solution to the mathematical model (or possibly its gradient) is obtained which is of higher formal order of accuracy than the underlying solution. This higher-order estimate relies only on information from the discrete solution itself, and thus can often be applied in a post-processing manner. For mesh and order refinement methods, higher-order estimates can be easily obtained for system response quantities as well. Residual-based methods, by contrast, also incorporate information on the specific problem being solved into the error estimate. While their implementation in a scientific computing code is generally more difficult and code-intrusive, they have the potential to provide more detailed information on the discretization error and its various sources. The extension of residual-based methods to provide discretization error estimates in system response quantities generally requires the solution to an adjoint (or dual) problem.

A. Higher-Order Estimates

One approach to error estimation is to compare the discrete solution to a *higher-order estimate* of the exact solution to the mathematical model. While this approach uses only information from the discrete solution itself, in some cases, more than one discrete solution is needed, with the additional solutions being obtained either on systematically-refined/coarsened meshes or with different formal orders of accuracy.

1. Mesh Refinement Methods and Richardson Extrapolation

Mesh refinement methods are based on the general concept of *Richardson extrapolation* (Richardson 1911, 1927). The basic concept behind Richardson extrapolation is as follows. If one knows the formal rate of convergence of a discretization method with mesh refinement, and if discrete solutions on two systematically-refined meshes are available, then one can use this information to obtain an estimate of the exact solution to the mathematical model. Depending on the level of confidence one has in this estimate, it can be used to either correct the fine mesh solution or to provide a discretization error estimate for it. While Richardson's original work applied the approach locally over the domain to the dependent variables in the mathematical model, it can be readily applied to any system response quantity. There is, however, the additional requirement that the numerical approximations (integration, differentiation, etc.) used to obtain the system response quantity be at least of the same order of accuracy as the underlying discrete solutions.

Recall the definition of the discretization error given by Equation (1). Consider now a more general local *or global* solution variable f on a mesh with spacing h

$$\varepsilon_h = f_h - \tilde{f} \quad (4)$$

where f_h is the exact solution to the discrete equations and \tilde{f} is the exact solution to the mathematical model. Recall that we can expand the numerical solution f_h in either a Taylor series about the exact solution

$$f_h = \tilde{f} + \frac{\partial \tilde{f}}{\partial h} h + \frac{\partial^2 \tilde{f}}{\partial h^2} \frac{h^2}{2} + \frac{\partial^3 \tilde{f}}{\partial h^3} \frac{h^3}{6} + \dots \quad (5)$$

or simply a power series in h

$$f_h = \tilde{f} + g_1 h + g_2 h^2 + g_3 h^3 + \dots \quad (6)$$

Moving \tilde{f} to the left-hand side allows us to write the discretization error for a mesh with spacing h as

$$\varepsilon_h = f_h - \tilde{f} = g_1 h + g_2 h^2 + g_3 h^3 + \dots \quad (7)$$

where the g coefficients can take the form of functions of derivatives of the exact solution to the mathematical model \tilde{f} with respect to either the mesh size h (as shown in Equation (5)) or to the independent variables through the relationship with the truncation error (see Section II.B.1). In general, we require numerical methods which are higher than first-order accurate, and thus discretization methods are chosen which cancel out selected lower-order terms. For example, if a second-order accurate numerical scheme is chosen, then the general discretization error expansion becomes:

$$\varepsilon_h = f_h - \tilde{f} = g_2 h^2 + g_3 h^3 + g_4 h^4 + \dots \quad (8)$$

Equation (8) forms the basis for generalized Richardson extrapolation which is described next.

Richardson extrapolation can be generalized to p^{th} -order accurate schemes and for two meshes systematically refined by an arbitrary factor. First consider the discretization error expansion for a p^{th} -order scheme:

$$\varepsilon_h = f_h - \tilde{f} = g_p h^p + g_{p+1} h^{p+1} + g_{p+2} h^{p+2} + \dots \quad (9)$$

Introducing the grid refinement factor as the ratio of the coarse to fine grid spacing we have

$$r = \frac{h_{\text{coarse}}}{h_{\text{fine}}} > 1 \quad (10)$$

and the coarse grid spacing can thus be written as $h_{\text{coarse}} = r h_{\text{fine}}$. Choosing $h_{\text{fine}} = h$, the discretization error equations on the two meshes can be written as

$$\begin{aligned} f_h &= \tilde{f} + g_p h^p + g_{p+1} h^{p+1} + O(h^{p+2}) \\ f_{rh} &= \tilde{f} + g_p (rh)^p + g_{p+1} (rh)^{p+1} + O(h^{p+2}) \end{aligned} \quad (11)$$

These equations can be used to eliminate the g_p coefficient and solve for \tilde{f} to give

$$\tilde{f} = f_h + \frac{f_h - f_{rh}}{r^p - 1} + g_{p+1} h^{p+1} \frac{r^p (r-1)}{r^p - 1} + O(h^{p+2}). \quad (12)$$

Combining terms of order h^{p+1} and higher with the exact solution \tilde{f} gives

$$\bar{f} = \tilde{f} - g_{p+1} \frac{r^p (r-1)}{r^p - 1} h^{p+1} + O(h^{p+2}) \quad (13)$$

which is a higher order accurate estimate of the exact solution. Substituting this expression into Equation (12) results in the generalized Richardson extrapolation estimate \bar{f} :

$$\bar{f} = f_h + \frac{f_h - f_{rh}}{r^p - 1} \quad (14)$$

As is shown clearly by Equation (13), this estimate of the exact solution is generally a $(p+1)$ -order accurate estimate of the exact solution to the mathematical model unless additional error cancellation occurs in the higher-order terms.

There is often confusion as to the order of accuracy of the Richardson extrapolation estimate. In Richardson's original work (Richardson, 1911), he used this extrapolation procedure to obtain a higher-order accurate solution for the stresses in a masonry dam based on two second-order accurate numerical solutions. The original partial differential equation was Poisson's equation and he employed central differences which cancelled out the odd

powers in the truncation error. His estimate for the exact solution was thus fourth-order accurate. However, unless this type of cancellation occurs, the estimates from generalized Richardson extrapolation are only $(p+1)$ -order accurate as is clearly shown in Equation (13).

Assumptions for Richardson Extrapolation

There are five basic assumptions required for Richardson extrapolation to provide reliable estimates of the exact solution to the mathematical model: 1) that *both* discrete solutions are in the asymptotic range, 2) that the meshes have a uniform (Cartesian) spacing over the domain, 3) that the coarse and fine meshes are related through systematic refinement, 4) that the solutions are smooth, and 5) that the other sources of numerical error are small. These five assumptions are now discussed in detail.

Asymptotic Range

The *formal order of accuracy* of a discretization scheme is the theoretical rate at which the discretization error is reduced as the mesh is refined. Oberkampf and Roy (2010) use a continuous error transport equation to relate the formal order of accuracy to the lowest order term in the truncation error. This lowest order term will necessarily dominate the higher-order terms in the limit as the mesh spacing parameter h goes to zero. The dependent solution variables generally converge at the formal order of accuracy in the *asymptotic range*, as do any system response quantities of interest (unless of course lower-order numerical approximations are used in their evaluation). One should keep in mind that this asymptotic requirement applies not just to the fine mesh solution but to the coarse mesh solution as well. Procedures for confirming that the asymptotic range has been reached will be given in Section III.

Uniform Mesh Spacing

The discretization error expansion is in terms of a single mesh spacing parameter h . This parameter is a measure of the size of the discretization, and thus has units of length for spatial discretizations and time for temporal discretizations. This could be strictly interpreted as allowing only Cartesian meshes with spacing h in each of the spatial coordinate directions. While this restriction seemingly prohibits the use of Richardson extrapolation for practical scientific computing applications, this is in fact not the case. For non-Cartesian meshes (including unstructured grids), local or global transformations are employed. If analytic transformations are used, then the mesh quality can affect the formal order of accuracy of the method, thus the use of extrapolation procedures requires sufficient mesh regularity (stretching, aspect ratio, skewness, etc.). Discrete transformations are additionally required to be the same order of accuracy as the discretization scheme or higher. Thompson et al. (1985) and Mastin (1999) note that there may be accuracy advantages to evaluating the discrete transformation metrics with the same underlying discretization used for the solution gradients, at least for the case of central differences.

Systematic Mesh Refinement

An often overlooked requirement for the use of Richardson extrapolation is that the two mesh levels be *systematically refined*. Systematic mesh refinement (Oberkampf and Roy, 2010) requires that the mesh refinement be both uniform and consistent. *Uniform refinement* (e.g., Roy, 2005) requires that the mesh be refined by the same factor over the entire domain, which precludes the use of local refinement or adaptation during the Richardson extrapolation procedure. *Consistent refinement* (Oberkampf and Roy, 2010) requires that the mesh quality must either remain constant or improve with mesh refinement. Examples of mesh quality metrics include cell aspect ratio, cell skewness, and cell-to-cell stretching factor. Techniques for evaluating the uniformity and consistency of the mesh refinement are given in Section V. For an example of how Richardson extrapolation can fail in the presence of nonuniform mesh refinement even in the asymptotic range, see Eca and Hoekstra (2009a).

Smooth Solutions

As discussed earlier, the coefficients g in the discretization expansion given by Equation (9) are generally functions of the solution derivatives. As such, the Richardson extrapolation procedure will tend to break down in the presence of discontinuities in any of the dependent variables or their derivatives. This is further complicated by the fact that the observed order of accuracy often reduces to first order or lower in the presence of certain discontinuities and singularities, regardless of the formal order of accuracy of the method (Banks et al., 2008). See Section VI.A for more details.

Other Numerical Errors Sources

Recall that the discretization error is defined as the difference between the exact solution to the discrete equations and the exact solution to the mathematical model. The exact solution to the discrete equations is never

known due to round-off error, iterative error, and statistical sampling errors (when present). In practice, the available numerical solutions are used as surrogates for the exact solution to the discretized equations. If these other numerical error sources are too large, then they will adversely impact the Richardson extrapolation procedure since any extrapolation procedure will tend to amplify “noise” (Roache, 1998). A good rule of thumb is to ensure that all other sources of numerical error are at least two orders of magnitude smaller than the discretization error in the fine grid numerical solution (Roy, 2005; Eca and Hoekstra, 2009b).

Extensions

This section describes three extensions of Richardson extrapolation when it is applied locally throughout the domain. The first addresses a method for obtaining the Richardson extrapolation estimate on all fine grid spatial points, while the second extends it to all fine mesh points in space and time. The third approach combines a general extrapolation procedure with a discrete residual minimization, and thus could be considered a hybrid extrapolation/residual method (see Section II.B).

Completed Richardson Extrapolation in Space

If the Richardson extrapolation procedure is to be applied to the solution point-wise in the domain, then it requires that one obtain fine mesh solution values at the coarse grid points. For systematic mesh refinement with integer values for the refinement factor on structured grids, this is automatically the case. However, applying Richardson extrapolation to cases with integer refinement will result in estimates of the exact solution only at the coarse grid points. In order to obtain exact solution estimates at the fine grid points, Roache and Knupp (1993) developed the *completed Richardson extrapolation* procedure. Their approach requires interpolation of the fine grid correction (rather than the Richardson extrapolation estimate) from the coarse grid to the fine grid. This interpolation should be performed with an order of accuracy at least as high as the underlying discretization schemes. When this fine grid correction is combined with the discrete solution on the fine grid, an estimate of the exact solution to the mathematical model is obtained that has the same order of accuracy as the Richardson extrapolation estimates on the coarse grid points.

Completed Richardson Extrapolation in Space and Time

Richards (1997) further extended the complete Richardson extrapolation procedure of Roache and Knupp (1993). The first modification provides the higher-order estimate of the exact solution to be obtained on all the fine grid spatial points for integer refinement factors other than two (three, four, five, etc.). The second, more significant modification provides higher-order accurate estimates of the exact solution after a chosen number of coarse grid time steps. The approach allows for different formal orders of accuracy in space and time by choosing the temporal refinement factor in such a way as to obtain the same order of error reduction found in the spatial discretization. For a discretization that is formally p^{th} -order accurate in space and q^{th} -order accurate in time, the temporal refinement factor r_t should be chosen such that

$$r_t = (r_x)^{p/q} \quad (15)$$

where r_x is the refinement factor in space. This procedure is analogous to the combined order verification procedure discussed in Oberkampf and Roy (2010).

Least Squares Extrapolation

Garbey and Shyy (2003) have developed a hybrid extrapolation/residual method for estimating the exact solution to the mathematical model. Their approach involves forming a more accurate solution by taking linear combinations of discrete solutions on multiple mesh levels using a set of spatially-varying coefficients. Spline interpolation is employed to obtain a smooth representation of this solution on a yet finer grid. The coefficients are then determined by a least squares minimization of the discrete residual formulated on this finer mesh. Their approach thus requires only residual evaluations on this finer mesh, which are expected to be significantly less expensive than computing a discrete solution on this mesh. This least squares extrapolated solution is demonstrated to be order $(p+1)$, where p is the formal order of accuracy of the method. The higher-order estimate of the exact solution to the mathematical model can be used as a local error estimator or to provide solution initialization within a multigrid-type procedure.

Discretization Error Estimation

While it may be tempting to use the Richardson extrapolated value as a more accurate solution than the fine grid numerical solution, this should only be done when there is a high degree of confidence that the five assumptions

underlying Richardson extrapolation have indeed been met. In particular, the *observed order of accuracy* (which requires discrete solutions on three systematically-refined meshes as discussed in Section III.B) must first be shown to match the *formal order of accuracy* of the discretization scheme. In any case, when only two mesh levels are available, the asymptotic nature of the solutions cannot be confirmed, thus one is limited to simply using the Richardson extrapolated value to estimate the discretization error in the fine grid numerical solution.

Substituting the estimated exact solution from the generalized Richardson extrapolation expression (Equation (14)) into the definition of the discretization error on the fine mesh (Equation (4)) gives the estimated discretization error for the fine mesh (with spacing h)

$$\bar{\varepsilon}_h = f_h - \bar{f} = f_h - \left(f_h + \frac{f_h - f_{rh}}{r^p - 1} \right)$$

or simply

$$\bar{\varepsilon}_h = -\frac{f_h - f_{rh}}{r^p - 1} \quad (16)$$

While this does provide for a consistent discretization error estimate as the mesh is systematically refined, there is no guarantee that the estimate will be reliable for any given fine mesh (h) and coarse mesh (rh) discrete solutions. Therefore, if only two discrete solutions are available then this error estimate should be converted into an uncertainty as discussed in Section IV.

An example of using the Richardson extrapolation procedure as an error estimator was presented by Roy and Blottner (2003). They examined the hypersonic, transitional flow over a sharp cone. The system response quantity was the heat flux distribution along the surface. The surface heat flux is shown versus the axial coordinate in Figure 2a for three systematically-refined mesh levels: fine (160×160 cells), medium (80×80 cells), and coarse (40×40 cells). Also shown are Richardson extrapolation results found from the fine and medium mesh solutions. The sharp rise in heat flux at $x = 0.5$ m is due to the specification of the location for transition from laminar to turbulent flow. In Figure 2b, the Richardson extrapolation results are used to estimate the discretization error in each of the numerical solutions. Neglecting the immediate vicinity of the transition location, the maximum estimated discretization errors are approximately 8%, 2%, and 0.5% for the coarse, medium, and fine meshes, respectively. The solutions thus appear to be converging as $h \rightarrow 0$. Furthermore, these estimated errors display the expected h^p reduction for these formally second-order accurate computations. In the turbulent region, the maximum errors are also converging at the expected rate giving error estimates of approximately 4%, 1% and 0.25%. More rigorous methods for assessing the reliability of discretization error estimates are addressed in Section III.

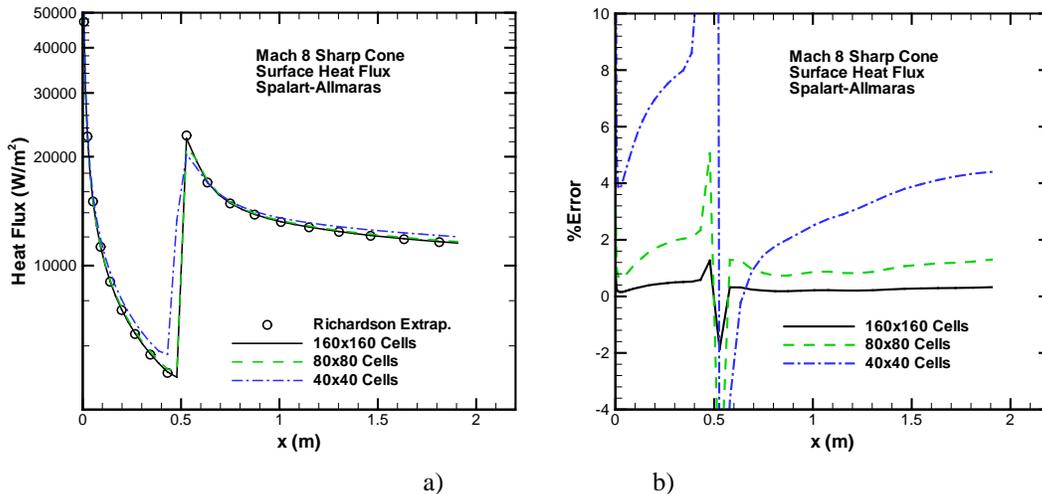


Figure 2. a) Surface heat flux and b) relative discretization error for the transitional flow over a sharp cone (Roy and Blottner, 2003).

Advantages and Disadvantages of Richardson Extrapolation

The primary advantage that Richardson extrapolation holds over other discretization error estimation methods is that it can be used as a post-processing technique applied to any discretization scheme (finite difference, finite volume, finite element, etc.). In addition, it gives estimates in the total error, which includes both locally generated errors and those transported from other regions of the domain. Finally, it can be used for any quantity of interest be it a local solution quantity or a derived system response quantity (assuming that any numerical approximations have been made with sufficient accuracy).

There are, however, some disadvantages to using discretization error estimators based on Richardson extrapolation. First and foremost, they rely on having multiple numerical solutions in the asymptotic grid convergence range. This can place significant additional burdens on the grid generation process, which is already a bottleneck in many scientific computing applications. Furthermore, these additional solutions can be extremely expensive to compute. Consider the case where one starts with a 3D mesh consisting of 1 million elements. Performing mesh refinement with a refinement factor of two thus requires a solution on a mesh with 8 million elements. When one also accounts for the additional time steps or iterations required for this finer mesh, the solution cost easily increases by an order of magnitude with each refinement (note that integer refinement is generally not required, see Section V.C).

The underlying theory of Richardson extrapolation requires smooth solutions, thus reducing the effectiveness of these error estimators for problems with discontinuities or singularities. In addition, the extrapolation procedure tends to amplify other sources of error such as round-off and iterative convergence error (Roache, 1998). Finally, the extrapolated quantities will not satisfy the same governing and auxiliary equations as either the numerical solutions or the exact solution. For example, if an equation of state is used to relate the density, pressure, and temperature in a gas, there is no guarantee that extrapolated values for density, pressure, and temperature will also satisfy this equation.

2. Order Refinement Methods

Order refinement methods are those which employ two or more discretizations on the same mesh but with differing formal orders of accuracy. The results from the two numerical solutions are then combined to produce a discretization error estimate. An early example of order refinement methods for error estimation is the Runge–Kutta–Fehlberg method (Fehlberg, 1969) for adaptive step size control in the solution of ordinary differential equations. This approach combines a basic 4th-order Runge-Kutta integration of the differential equations with an inexpensive 5th-order estimate of the error. Order refinement can be difficult to implement in finite difference and finite volume discretizations due to difficulties formulating higher-order accurate gradients and boundary conditions. Order refinement methods have been implemented within the context of finite elements under the name hierarchical bases (e.g., see Bank, 1996).

3. Finite Element Recovery Methods

Recovery methods for estimating the discretization error were developed by the finite element community (e.g., Zienkiewicz and Zhu, 1987). For the standard h-version of finite elements with linear basis functions the solution is piece-wise linear; therefore, the gradients are only piece-wise constant and are discontinuous across the element faces. The finite element analyst is often more interested in gradient quantities such as stresses rather than the solution itself, so most finite element codes provide for post-processing of these discontinuous gradients into piece-wise linear gradients using existing finite element infrastructure. In some cases (see the discussion of superconvergence below), this reconstructed gradient is of a higher order of accuracy than the gradient found in the underlying finite element solution. Recall the definition of the energy norm of the discretization error given in Equation (3). If the true gradient from the mathematical model is available, then this important error measure can be computed exactly. For the case where a reconstructed gradient is higher-order accurate than the finite element gradient, then it can be used to approximate the true gradient in the energy norm. In addition to providing estimates of the discretization error in the solution gradients, due to their local nature, recovery methods are also often used as indicators of where solution refinement is needed in adaptive solutions.

In order to justify the use of the recovered gradient in the energy norm, it must in fact be higher-order accurate than the gradient from the finite element solution. This so-called superconvergence property can occur when certain regularity conditions on the mesh and the solution are met (Wahlbin, 1995) and results in gradients that are up to one order higher in accuracy than the underlying finite element gradients. For linear finite elements, the superconvergence points occur at the element centroids, whereas for quadratic finite elements, the location of the superconvergence points depends on the element topology. If the reconstructed gradient is superconvergent, and if certain consistency conditions are met by the gradient reconstruction operator itself, then error estimators based on

this recovered gradient can be shown to be asymptotically exact (Ainsworth and Oden, 2000). While the superconvergence property appears to be a difficult one to attain for complex scientific computing applications, the discretization error estimates from some recovery methods tend to be “astonishingly good” for reasons that are not well-understood (Ainsworth and Oden, 2000).

Recovery methods have been shown to be most effective when the reconstruction step employs solution gradients rather than solution values. The superconvergent patch recovery (SPR) method (Zienkiewicz and Zhu, 1992) is the most widely-used recovery method in finite element analysis. Assuming the underlying finite element method is of order p , the SPR approach is based on a local least squares fitting of the solution gradient values at the superconvergence points using polynomials of degree p . The SPR recovery method was found to perform extremely well in an extensive comparison of *a posteriori* finite element error estimators (Babuska et al., 1994). A more recent approach called polynomial preserving recovery (PPR) was proposed by Zhang and Naga (2005). In their approach, they use polynomials of degree $p + 1$ to fit the solution values at the superconvergence points, then take derivatives of this fit to recover the gradient. Both the SPR and PPR gradient reconstruction methods can be used to obtain error estimates in the global energy norm and in the local solution gradients.

B. Residual-Based Methods

Residual-based methods use the discrete solution along with additional information from the problem being solved such as the mathematical model, the discrete equations, or the sources of discretization error. Examples of residual-based methods are error transport equations (both continuous and discrete) and finite element residual methods. As is shown in the next section, all of these residual-based methods are related through the truncation error. The truncation error can be approximated either by inserting the exact solution to the mathematical model (or an approximation thereof) into the discrete equation or by inserting the discrete solution into the continuous mathematical model. The former is the discrete residual which is used in most discrete discretization error transport equations. The latter is simply the definition of the finite element residual. The use of adjoint methods to extend discretization error estimation methods to provide error estimates in system response quantities is also discussed

1. Discretization Error Transport Equations

Discretization error is transported through the domain in a similar fashion as the solution in the underlying mathematical model (Ferziger and Peric, 2002). For example, if a mathematical model governs the convection and diffusion of a scalar variable, then a discrete solution to the mathematical model will contain discretization error that is also convected and diffused. Babuska and Rheinboldt (1978a) appear to be the first to develop such a discretization error transport equation within the context of the finite element method. However, rather than solve this transport equation directly for the discretization error, the typical approach used in finite elements is to use this equation to either indirectly bound the error (explicit residual methods) or approximate its solution (implicit residual methods). The solution of error transport equations with finite volume schemes can be found in Zhang et al. (2000) and Shih and Williams (2009).

Continuous Discretization Error Transport Equation

The following development is applicable to any discretization approach and is based on the *generalized truncation error expression* developed in Roy (2009). Consider a (possibly nonlinear) governing equation operator from the mathematical model $L(\cdot)$ and a discrete equation operator $L_h(\cdot)$. These continuous and discrete operators are solved exactly by \tilde{u} (the exact solution to the mathematical model) and u_h (the exact solution to the discrete equations), respectively. Thus we can write:

$$L(\tilde{u}) = 0 \tag{17}$$

and

$$L_h(u_h) = 0 \tag{18}$$

Furthermore, the partial differential equation and the discretized equation are related through the *generalized truncation error expression* (Roy, 2009) as

$$L_h(u) = L(u) + TE_h(u) \tag{19}$$

which assumes some suitable mapping of the operators onto either a continuous or discrete space. Substituting u_h into Equation (19) and then subtracting Equation (17) gives:

$$L(u_h) - L(\tilde{u}) + TE_h(u_h) = 0 \quad (20)$$

If the equations are linear, or if they are linearized, then we have $L(u_h) - L(\tilde{u}) = L(u_h - \tilde{u})$. With the definition of the discretization error

$$\varepsilon_h = u_h - \tilde{u} \quad (21)$$

we can thus rewrite Equation (20) as

$$L(\varepsilon_h) = -TE_h(u_h) \quad (22)$$

Equation (22) is the (continuous) mathematical model that governs the transport of the discretization error ε_h through the domain. Furthermore, the truncation error acting upon the discrete solution serves as a source term which governs the local generation or removal of discretization error due to the local discretization parameters (Δx , Δy , etc.). Equation (22) is called the *continuous discretization error transport equation*. This equation can be solved for the discretization error in the solution variables assuming that the truncation error is known or can be estimated.

Discrete Discretization Error Transport Equation

A discrete version of the discretization error transport equation can be derived as follows. First the exact solution to the mathematical model \tilde{u} is substituted into Equation (19) and then Equation (18) is subtracted to get:

$$L_h(u_h) - L_h(\tilde{u}) + TE_h(\tilde{u}) = 0 \quad (23)$$

If the equations are again linear (or linearized), then this equation can be rewritten as

$$L_h(\varepsilon_h) = -TE_h(\tilde{u}) \quad (24)$$

Equation (24) is the discrete equation that governs the transport of the discretization error ε_h through the domain and is therefore called the *discrete discretization error transport equation*. This equation can be solved for the discretization error if the truncation error and the exact solution to the original partial differential equation (or an approximation of it) are known.

Approximating the Truncation Error

While the development of error transport equations is relatively straightforward, questions remain as to the treatment of the truncation error which acts as the source term. The truncation error can be difficult to derive for complex, nonlinear numerical flux schemes such as those used for the solution to the compressible Euler equations in fluid dynamics. However, if the truncation error can be reliably approximated, then this approximation can be used as the source term for the error transport equation.

Here we present three approaches for approximating the truncation error, with the first two approaches beginning with the *generalized truncation error expression* given by Equation (19) (Roy, 2009). In the first approach, the exact solution to the mathematical model \tilde{u} is inserted into Equation (19). Since this exact solution will exactly solve the mathematical model, the term $L(\tilde{u}) = 0$, thus allowing the truncation error to be approximated as:

$$TE_h(\tilde{u}) = L_h(\tilde{u}) \quad (25)$$

Since this exact solution is generally not known, it could be approximated by plugging an estimate of the exact solution, for example from Richardson extrapolation or any other local error estimator, into the discrete operator:

$$TE_h(u_{RE}) \approx L_h(u_{RE}).$$

Alternatively, the solution from a fine grid solution u_h could be inserted into the discrete operator for a coarse grid $L_{rh}(\cdot)$:

$$TE_h(\tilde{u}) = \frac{1}{r^p} TE_{rh}(\tilde{u}) \approx \frac{1}{r^p} TE_{rh}(u_h) = \frac{1}{r^p} L_{rh}(u_h).$$

Note that the subscript rh denotes the discrete operator on a grid that is a factor of r coarser in each direction than the fine grid. For example, $r = 2$ when the coarse mesh is formed by eliminating every other point in each direction of a structured mesh. This approach was used by Shih and Qin (2007) to estimate the truncation error for use with a discrete discretization error transport equation.

A second approach for estimating the truncation error is to insert the exact solution to the discrete equations u_h into Equation (19). Since this solution exactly solves the discrete equations $L_h(u_h) = \mathbf{0}$, we have:

$$TE_h(u_h) = -L(u_h). \quad (26)$$

If a continuous representation of the solution is available then this evaluation is straightforward. In fact, the right-hand side of Equation (26) is the definition of the finite element residual that is given in the next section. For other numerical methods (e.g., finite difference and finite volume), a continuous projection of the numerical solution must be made in order to estimate the truncation error. For example, Sonar (1993) formed this residual by projecting a finite volume solution onto a finite element subspace with piece-wise linear shape functions.

A third approach that is popular for hyperbolic problems (e.g., compressible flows) is based on the fact that central-type differencing schemes often require additional numerical (artificial) dissipation to maintain stability and to prevent numerical oscillations. This numerical dissipation can either be explicitly added to a central differencing scheme (e.g., see Jameson et al., 1981) or incorporated as part of an upwind differencing scheme. In fact, it can be shown that any upwind scheme can be written as a central scheme plus a numerical dissipation term (e.g., Hirsch, 1990). These two approaches can thus be viewed in the context of central schemes with the numerical dissipation contributions serving as the leading terms in the truncation error. While this approach may only be a loose approximation of the true truncation error, it merits discussion due to the fact that it can be readily computed with little additional effort.

System Response Quantities

A drawback to the error transport equation approach is that it provides for discretization error estimates in the local solution variables, but not in system response quantities. While adjoint methods can be used to provide error estimates in the system response quantities (see Section II.B.3), Cavallo and Sinha (2007) have developed a simpler approach which uses an analogy with experimental uncertainty propagation to relate the local solution errors to the error in the system response quantity. However, their approach appears to provide extremely conservative error bounds for integrated quantities since it does not allow for the cancellation of competing errors. A less conservative approach would be to use the local error estimates to correct the local quantities, then compute the integrated quantity with these corrected values. This “corrected” integrated quantity could then be used to provide the desired discretization error estimate (Oberkampf and Roy, 2010).

2. Finite Element Residual Methods

In a broad mathematical sense, a *residual* refers to what is left over when an approximate solution is inserted into an equation. For linear systems, iterative convergence is often assessed in terms of the iterative residuals which are found by substituting an approximate iterative solution into the discrete equations. Consider now the general mathematical operator $L(\tilde{u}) = \mathbf{0}$ which is solved exactly by \tilde{u} . Because the finite element method provides for a continuous representation of the numerical solution u_h , it is natural to define the finite element residual in a continuous sense over the domain as

$$\mathfrak{R}(u_h) = L(u_h). \quad (27)$$

In a manner analogous to the development of the previous section, a continuous discretization error transport equation can be derived within the finite element framework (Babuska and Rheinboldt, 1978a). This so-called residual equation has three different types of terms: 1) interior residuals that determine how well the finite element solution satisfies the mathematical model in the domain, 2) terms associated with any discretized boundary conditions on the domain boundary (e.g., Neumann boundary conditions), and 3) interelement residuals which are functions of the discontinuities in normal fluxes across element-element boundaries (Ainsworth and Oden, 2000). It is the treatment of these three terms that differentiates between explicit and implicit residual methods.

Explicit Residual Methods

Explicit residual methods are those which employ information available from the finite element solution along with the finite element residuals to directly compute the error estimate. First developed by Babuska and Rheinboldt (1978b), explicit residual methods lump all three types of residual terms under a single, unknown constant. The analysis requires the use of the triangle inequality, which does not allow for cancellation between the different residual types. Due both to the use of the triangle inequality and the methods for estimating the unknown constant, explicit residual methods are conservative estimates of the discretization error. They provide an element-wise estimate of the *local contribution* to the bound for the global energy norm of the error, but not a local estimate of the true error, which would include both local and transported components. Since explicit residual methods deal only with local contributions to the error, they can also be used for solution adaptation procedures. Stewart and Hughes (1998) have provided a tutorial on explicit residual methods and their relationship to *a priori* error estimation.

Implicit Residual Methods

Implicit residual methods avoid the approximations required in explicit residual methods by seeking solutions to the residual equation which governs the transport and generation of the discretization error. In order to achieve non-trivial solutions to the global residual equation, the mesh would either have to be refined or the order of the finite element basis functions increased. Both of these approaches would be significantly more expensive than obtaining the original finite element solution and therefore are not considered practical. Instead, the global residual equation is decomposed into a series of uncoupled, local boundary value problems which will approximate the global equation. These local problems can be solved over a single element using the element residual method (Demkowicz, 1984; Bank and Weiser, 1985) or over a small patch of elements using the subdomain residual method (Babuska and Rheinboldt, 1978a,b). The solution to the local boundary value problems provides the local discretization error estimate, while the global error estimate is simply summed over the domain. By directly treating all three types of terms that show up in the residual equation, implicit residual methods retain more of the structure of the residual equation than do the explicit methods, and thus should in theory provide tighter error bounds.

3. Adjoint Methods for System Response Quantities

Both error transport equations and finite element residual methods give localized estimates of the discretization error, which can then be combined through an appropriate norm to provide quantitative measures of the overall “goodness” of the discrete solutions. However, the scientific computing practitioner is often instead interested in system response quantities that can be post-processed from the solution. These system response quantities can take the form of integrated quantities (e.g., net flux through or force acting on a boundary), local solution quantities (e.g., maximum stress or maximum temperature), or even an average of the solution over some region.

Adjoint methods in scientific computing were initially used for design optimization problems (e.g., Jameson, 1988). In the optimization setting, the adjoint (or dual) problem can be solved for sensitivities of a solution functional (e.g., a system response quantity) that one wishes to optimize relative to some chosen design parameters. The strength of the adjoint method is that it is efficient even when a large number of design parameters are involved. In the context of optimization in scientific computing, adjoint methods can be thought of as constrained optimization problems where a chosen solution functional is to be optimized subject to the constraint that the solution must also satisfy the mathematical model (or possibly the discrete equations).

Adjoint methods can also be used for estimating the discretization error in a system response quantity in scientific computing applications. Consider a scalar solution functional $f_h(u_h)$ evaluated on mesh h . An approximation of the discretization error in this functional is given by

$$\varepsilon_h = f_h(u_h) - f_h(\tilde{u}) \quad (28)$$

Performing a Taylor series expansion of $f_h(\tilde{u})$ about the discrete solution gives

$$f_h(\tilde{u}) \cong f_h(u_h) + \left. \frac{\partial f_h}{\partial u} \right|_{u_h} (\tilde{u} - u_h) \quad (29)$$

where higher order terms have been neglected. Next, an expansion of the discrete operator $L_h(\cdot)$ is performed at \tilde{u} about u_h :

$$L_h(\tilde{u}) \cong L_h(u_h) + \left. \frac{\partial L_h}{\partial u} \right|_{u_h} (\tilde{u} - u_h) \quad (30)$$

where $L_h(\tilde{u})$ is the discrete residual, an approximation of the truncation error from Equation (25), and $\left. \frac{\partial L_h}{\partial u} \right|_{u_h}$ is

the Jacobian which linearizes the discrete equations with respect to the solution. This Jacobian may already be computed since it can also be used to formulate implicit solutions to the discrete equations and for design optimization. Since $L_h(u_h) = 0$, Equation (30) can be rearranged to obtain:

$$(\tilde{u} - u_h) = \left[\left. \frac{\partial L_h}{\partial u} \right|_{u_h} \right]^{-1} L_h(\tilde{u}) \quad (31)$$

Substituting this equation into Equation (29) gives

$$f_h(\tilde{u}) \cong f_h(u_h) + \left. \frac{\partial f_h}{\partial u} \right|_{u_h} \left[\left. \frac{\partial L_h}{\partial u} \right|_{u_h} \right]^{-1} L_h(\tilde{u}) \quad (32)$$

or

$$f_h(\tilde{u}) \cong f_h(u_h) + \Psi^T L_h(\tilde{u}) \quad (33)$$

where Ψ^T is the row vector of discrete adjoint sensitivities. The adjoint sensitivities are found by solving

$$\Psi^T = \left. \frac{\partial f_h}{\partial u} \right|_{u_h} \left[\left. \frac{\partial L_h}{\partial u} \right|_{u_h} \right]^{-1} \quad (34)$$

which can be put into the standard linear equation form by transposing both sides of Equation (34)

$$\left[\left. \frac{\partial L_h}{\partial u} \right|_{u_h} \right]^T \Psi = \left[\left. \frac{\partial f_h}{\partial u} \right|_{u_h} \right]^T. \quad (35)$$

The adjoint solution provides the linearized sensitivities of the solution functional f_h to perturbations in the discrete operator $L_h(\cdot)$. As such, the adjoint solution vector components are often referred to as the adjoint sensitivities. Equation (33) shows that the adjoint solution provides the sensitivity of the discretization error in the solution functional $f(\cdot)$ to the local sources of discretization error (i.e., the truncation error) in the domain. This observation can be used as the basis for providing solution adaptation targeted for solution functionals. Because the discrete operator $L_h(\cdot)$ is used above, this approach is called the *discrete adjoint method*. A similar analysis using expansions of the continuous mathematical operator $L(\cdot)$ and functional $f(\cdot)$ can be performed to obtain discretization error estimates using the *continuous adjoint method*. Both continuous and discrete adjoint methods also require appropriate formulations of initial and boundary conditions.

Adjoint Methods in the Finite Element Method

While the use of explicit and implicit residual methods for finite elements has reached a certain level of maturity for elliptic problems (Ainsworth and Oden, 2000), the drawback to these methods is that they only provide error estimates in the energy norm of the discretization error. While the energy norm is a natural quantity by which to judge the overall goodness of a finite element solution, in many cases scientific computing is used to make an engineering decision with regards to a specific system response quantity (called “quantities of interest” by the finite element community). Extension of both the explicit and implicit residual methods to provide error estimates in a system response quantity generally requires the solution to the adjoint system (i.e., the dual problem).

In one approach (Ainsworth and Oden, 2000), the discretization error in system response quantity is bounded by the product of the energy norm of the adjoint solution and the energy norm of the error in the original solution. Assuming the solutions are asymptotic, the use of the Cauchy-Schwarz inequality produces a conservative bound. In this case, the discretization error in the system response quantity will be reduced at twice the rate of the solution error. In another approach (Estep et al., 2000), the error estimate in the system response quantity is found as an inner product between the adjoint solution and the residual. This approach results in a more accurate (i.e., less conservative) error estimate at the expense of losing the rigorous error bound. For more information on error estimation using adjoint methods in finite elements, see Johnson and Hansbo (1992), Paraschivoiu et al. (1997), Rannacher and Suttmeier (1997), Estep et al. (2000), and Cheng and Paraschivoiu (2004).

Adjoint Methods in the Finite Volume Method

Pierce and Giles (2000) have proposed a continuous adjoint approach that focuses on system response quantities (e.g., lift and drag in external aerodynamics) and is not tied to a specific discretization scheme. They use the adjoint solution to relate the residual error in the mathematical model to the resulting error in the integral quantity of interest. Their approach also includes a defect correct step that increases the order of accuracy of the integral quantity. For example, if the original solution and the adjoint solution are both second-order accurate, then the integral quantity will have an order of accuracy equal to the product of the orders of the original and adjoint solutions, or fourth order. Their approach effectively extends the superconvergence property of finite elements to other discretization schemes, and can also be used to further increase the order of accuracy of the integral quantities for the finite element method.

Venditti and Darmofal (2000) have extended the adjoint approach of Pierce and Giles (2000) to allow for the estimation of local mesh size contributions to the integral quantity of interest. Their approach is similar to that described above, but expands the functional and discrete operator on a fine grid solution u_h about a coarse grid solution $u_{\tilde{h}}$. The solution is not required on the fine grid, only residual evaluations. Their approach is thus a discrete adjoint method rather than continuous adjoint. In addition, their focus is on developing techniques for driving a mesh adaptation process. Their initial formulation was applied to 1D inviscid flow problems, but they have also extended their approach to 2D inviscid and viscous flows (Venditti and Darmofal, 2002, 2003). While adjoint methods hold significant promise as discretization error estimators for solution functionals, they currently require significant code modifications to compute the Jacobian and other sensitivity derivatives and have not yet seen widespread use in commercial scientific computing software.

III. Reliability of Discretization Error Estimators

One of the key requirements for *reliability of any of the discretization error estimators* discussed in this paper is that the solution(s) must be in the asymptotic range. This section provides a discussion of just what this *asymptotic range* means for discretization approaches involving both mesh (h) refinement and order (p) refinement. Regardless of whether h - or p -refinement is used, the *demonstration* that the asymptotic range has been achieved generally requires that at least three discrete solutions be computed. Demonstrating that the asymptotic range has been reached can be surprisingly difficult for complex scientific computing applications involving nonlinear, hyperbolic, coupled systems of equations. It is unlikely that the asymptotic range will be reached without the use of solution adaptation (e.g., see Roy, 2009 and Oberkampf and Roy, 2010).

A. Asymptotic Range

The *asymptotic range* is defined differently depending on whether one is varying the mesh resolution or the formal order of accuracy of the discretization scheme. When *mesh refinement* is employed, then the asymptotic range is defined as the sequence of systematically-refined meshes over which the discretization error reduces at the formal order of accuracy of the discretization scheme. Examining the discretization error expansion for a p^{th} -order accurate scheme given by Equation (9), the asymptotic range is achieved when h is sufficiently small that the h^p term is much larger than all of the higher-order terms combined. Due to possible differences in the signs for the higher-order terms, the behavior of the discretization error outside of the asymptotic range can be extremely unpredictable. Confirming that the asymptotic range has been reached using systematic mesh refinement is achieved by evaluating the *observed order of accuracy*. The observed order assesses the behavior of the discrete solutions over a range of meshes and its evaluation is discussed in the next section.

For discretization methods involving *order refinement*, the asymptotic range is determined by examining the behavior of the numerical solutions with successively-refined basis functions, all on the same domain mesh. As the basis functions are refined and the physical phenomena in the problem are resolved, the discrete solutions will

eventually become better approximations of the exact solution to the mathematical model. Convergence is best monitored with error norms since convergence can be oscillatory with increased basis order. An example of hierarchical basis functions used within the finite element method is given by Bank (1996).

B. Observed Order of Accuracy

The *observed order of accuracy* is the measure that is used to assess the confidence in a discretization error estimate. When the observed order of accuracy is shown to match the *formal order*, then one can have a high degree of confidence in the error estimate. When the exact solution is not known, three numerical solutions on systematically-refined meshes are required to calculate the observed order of accuracy. For the observed order of accuracy to match the formal order of the discretization scheme, the requirements are the same as those given in Section II.A.1 for Richardson extrapolation. When any of these requirements fail to be met, unrealistic values for the observed order of accuracy can be obtained (e.g., see Roy, 2003; Salas, 2006; and Eca and Hoekstra, 2009).

1. Constant Grid Refinement Factor

Consider a p^{th} -order accurate scheme with numerical solutions on a fine mesh (h_1), a medium mesh (h_2), and a coarse mesh (h_3). For the case of a constant grid refinement factor, i.e.,

$$r = \frac{h_2}{h_1} = \frac{h_3}{h_2} > 1$$

we can thus write

$$h_1 = h, \quad h_2 = rh, \quad h_3 = r^2h.$$

Using the discretization error expansion from Equation (9), we can now write for the three discrete solutions:

$$\begin{aligned} f_1 &= \tilde{f} + g_p h^p + g_{p+1} h^{p+1} + O(h^{p+2}) \\ f_2 &= \tilde{f} + g_p (rh)^p + g_{p+1} (rh)^{p+1} + O(h^{p+2}) \\ f_3 &= \tilde{f} + g_p (r^2h)^p + g_{p+1} (r^2h)^{p+1} + O(h^{p+2}) \end{aligned} \quad (36)$$

Neglecting terms of order h^{p+1} and higher allows us to recast these three equations in terms of a locally-observed order of accuracy \hat{p} :

$$\begin{aligned} f_1 &= \tilde{f} + g_p h^{\hat{p}} \\ f_2 &= \tilde{f} + g_p (rh)^{\hat{p}} \\ f_3 &= \tilde{f} + g_p (r^2h)^{\hat{p}} \end{aligned} \quad (37)$$

which will only match the formal order of accuracy if the higher order terms are indeed small. Subtracting f_2 from f_3 and f_1 from f_2 yields:

$$f_3 - f_2 = g_p (r^2h)^{\hat{p}} - g_p (rh)^{\hat{p}} = g_p r^{\hat{p}} h^{\hat{p}} (r^{\hat{p}} - 1) \quad (38)$$

and

$$f_2 - f_1 = g_p (rh)^{\hat{p}} - g_p h^{\hat{p}} = g_p h^{\hat{p}} (r^{\hat{p}} - 1). \quad (39)$$

Dividing Equation (38) by Equation (39) gives

$$\frac{f_3 - f_2}{f_2 - f_1} = r^{\hat{p}}. \quad (40)$$

Taking the natural log of both sides and solving for the observed order of accuracy \hat{p} gives:

$$\hat{p} = \frac{\ln\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{\ln(r)}. \quad (41)$$

Consistent with the development of generalized Richardson extrapolation in Section II.A.1, the Richardson extrapolated estimate of the exact solution \bar{f} and the leading error term coefficient g_p are given by:

$$\bar{f} = f_1 + \frac{f_1 - f_2}{r^{\hat{p}} - 1} \quad (42)$$

and

$$g_p = \frac{f_1 - \bar{f}}{h^{\hat{p}}}. \quad (43)$$

Note that it is only when the observed order of accuracy matches the formal order of the numerical scheme that we can expect the discretization error estimate given by Equation (42) to be accurate. This is equivalent to saying that the solutions on all three meshes are in the asymptotic range and the higher-order terms in Equations (36) are small. In practice, when this locally observed order of accuracy is used for the extrapolation estimate, it is often limited to be in the range

$$1 \leq \hat{p} \leq p_f$$

where p_f is the formal order of accuracy of the discretization scheme. Allowing the observed order of accuracy to increase above the formal order can result in discretization error estimates that are not conservative (i.e., they underestimate the error).

2. Non-Constant Grid Refinement Factor

For the case of non-constant grid refinement factors

$$r_{12} = \frac{h_2}{h_1} > 1, \quad r_{23} = \frac{h_3}{h_2} > 1$$

where $r_{12} \neq r_{23}$, the determination of the observed order of accuracy \hat{p} is more complicated. For this case, the following transcendental equation (Roache, 1998) must be solved for \hat{p} :

$$\frac{f_3 - f_2}{r_{23}^{\hat{p}} - 1} = r_{12}^{\hat{p}} \left(\frac{f_2 - f_1}{r_{12}^{\hat{p}} - 1} \right). \quad (44)$$

This equation can be easily solved with a simple direct substitution iterative procedure to give

$$\hat{p}^{k+1} = \frac{\ln\left[\left(r_{12}^{\hat{p}^k} - 1\right)\left(\frac{f_3 - f_2}{f_2 - f_1}\right) + r_{12}^{\hat{p}^k}\right]}{\ln(r_{12}r_{23})}. \quad (45)$$

where an initial guess of $\hat{p}^k = p_f$ (the formal order of the scheme) can be used (Oberkampf and Roy, 2010). Once the observed order of accuracy is found, the estimate of the exact solution \bar{f} and the leading error term coefficient g_p are given by Equations (42) and (43) but replacing the constant grid refinement factor with $r = r_{12}$.

3. Application to System Response Quantities

Recall that system response quantities are defined as any solution property derived from the solution to the mathematical model its discrete approximation. Examples of common system response quantities in scientific computing are lift and drag in aerodynamics, heat flux through a surface in heat transfer analysis, and maximum stress in a structural mechanics problem. The observed order of accuracy for system response quantities can be

evaluated by the approaches described earlier in this section. For this observed order of accuracy to match the formal order of the discretization scheme, there is an additional requirement beyond those described in Section II.A.1 for Richardson extrapolation. This additional requirement pertains to the order of accuracy of any numerical approximations used to compute the system response quantity. When the system response quantity is an integral, a derivative, or an average, then the numerical approximations used in its evaluation must be of at least the same order of accuracy as the underlying discretization scheme. In most cases, the behavior of integrated quantities and averages are better behaved and converge more rapidly with mesh refinement than local quantities. However, in some cases the errors due to the numerical quadrature can interact with the numerical errors in the discrete solution and adversely impact the observed order of accuracy computation. An example of these interactions for the computation of drag on an inviscid supersonic blunt body problem (mixed elliptic/hyperbolic) is given by Salas and Atkins (2009).

4. Application to Local Quantities

Problems often arise when the observed order of accuracy is evaluated on a point-by-point basis in the domain. A simple example of a case where this local evaluation of the order of accuracy will fail is given in Figure 3 which shows the discretization error on three different meshes along a line. If the meshes are refined by a factor of two and the formal order of the scheme is first-order accurate, then we expect the discretization error to drop by a factor of two for each refinement. However, as can commonly occur in practical applications, part of the domain approaches the exact solution from above and part of it from below. Even if we neglect any other sources of numerical error (such as round-off error), the observed order of accuracy at this crossover point will be undefined, even though the discretization error on all three meshes is exactly zero.

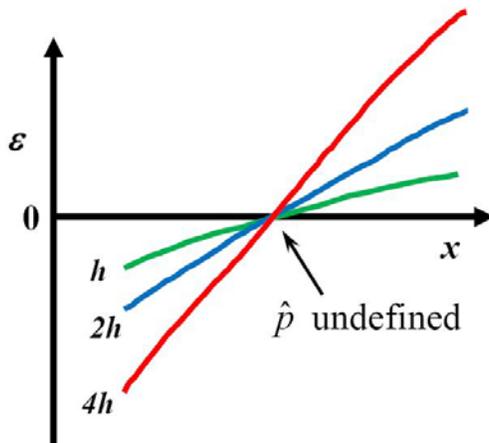


Figure 3. Simple example of how the observed order of accuracy computation will fail when applied locally over the domain: the observed order will be undefined at the crossover point.

Another example of the problems that can occur when examining the observed order of accuracy on a point-by-point basis through the domain was given by Roy (2003) and is shown in Figure 4. The problem of interest is inviscid, hypersonic flow over a sphere-cone geometry. The mathematical character of this problem is elliptic immediately behind the normal shock wave that forms upstream of the sphere, but hyperbolic over the rest of the solution domain. The observed order of accuracy for the surface pressure is plotted versus the normalized axial distance based on three uniformly refined meshes. The finest mesh is 1024×512 cells and a refinement factor of two is used to create the coarse meshes. While a formally second-order accurate finite volume discretization was used, flux limiters were employed in the region of the shock wave discontinuity to capture the shock in a monotone fashion by locally reducing the formal order of accuracy to one. The observed order of accuracy is indeed first order and well-behaved in the elliptic region (up to x/R_N of 0.2). However, in the hyperbolic region, the observed order of accuracy is found to undergo large oscillations between -4 and $+8$, with a few locations being undefined. Farther downstream, the observed order again becomes well-behaved with values near unity. The source of these oscillations is likely the local characteristic waves generated when the shock moves from one grid line to another. (This is the same example that is given in Figure 1 showing the local and transported components of the discretization error.) Clearly extrapolation-based error estimates using the local order of accuracy would not be reliable in this region.

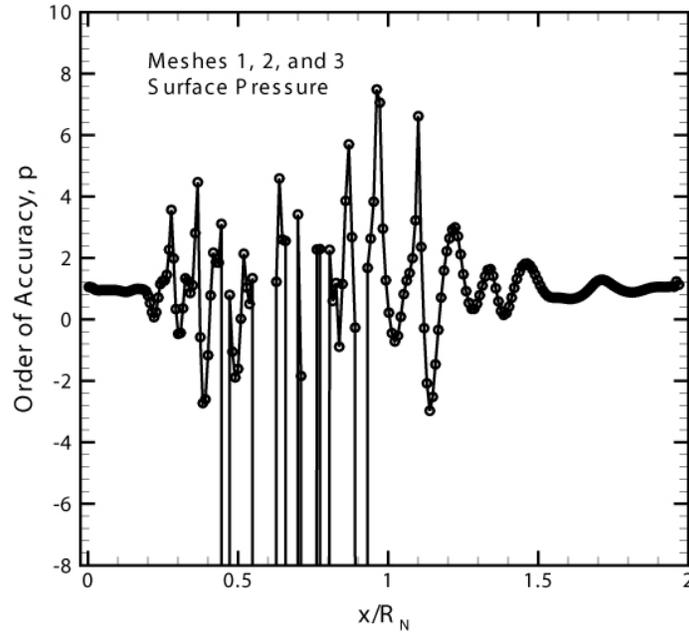


Figure 4. Observed order of accuracy in surface pressure for inviscid hypersonic flow over a sphere-cone geometry (from Roy, 2003).

IV. Discretization Error and Uncertainty

As discussed previously, when the observed order of accuracy matches the formal order, then one can have high confidence that the error estimate is accurate and therefore use the error estimate to correct the solution. However, the much more common case is when the formal order does not match the observed order. In this case, the error estimate is much less reliable and should generally be converted into a numerical uncertainty. While the difference between the discrete solution and the (unknown) exact solution to the mathematical model is still truly an error, our lack of knowledge of the true value of this error forces us to represent it as an epistemic uncertainty. Epistemic uncertainties are distinct from aleatory (or random) uncertainties in that they are due to a lack of knowledge. They can be reduced by providing more information, in this case, additional computations on more refined meshes. The treatment of these numerical uncertainties and their effects on the predictive capability of scientific computing simulations is discussed in Oberkampf and Roy (2010).

A. Roache's Grid Convergence Index (GCI)

Roache (1994) proposed the Grid Convergence Index, or GCI, as a method for uniform reporting of grid refinement studies. The GCI combines the often reported relative difference between two discrete solutions with the $(r^p - 1)$ factor required in the denominator. The GCI takes the further step of converting the error estimate into an error or uncertainty band, which is again appropriate when one does not have a high degree of confidence in the error estimate.

1. Definition

The GCI for the fine grid numerical solution is defined as (Roache, 1998)

$$GCI = \frac{F_s}{r^p - 1} \left| \frac{f_2 - f_1}{f_1} \right| \quad (46)$$

where F_s is a factor of safety (see the discussion below and in the next section). The key features of the GCI are the use of available discrete solution values f_1 and f_2 , proper accounting of the $r^p - 1$ factor in the denominator, absolute values to convert the error estimate to an error band, and the addition of a factor of safety F_s . When only

two discrete solutions are available, then the formal order of accuracy of the discretization scheme should be used in the definition of the GCI along with a factor of safety of $F_s = 3$. However, when three discrete solutions are available then the observed order of accuracy should be computed to assess the reliability of the error estimates. While this observed order of accuracy should be computed locally, a global averaged value should be used in the GCI computations along with a less conservative factor of safety of $F_s = 1.25$ (Roache, 2003; Cadafalch et al., 2002). As a final note, when the discrete solution values are expected to be near zero, then the normalization by f_1 in the denominator should be omitted. For a discussion of the relationship between the GCI and an uncertainty based directly on Richardson extrapolation, see Roy (2001).

2. Factor of Safety

It is important to include the factor of safety in the GCI and the RDE_{band} . Both of these error bands are based on Richardson extrapolation, and we don't know *a priori* whether the estimated exact solution is above or below the true exact solution to the mathematical model. Consider Figure 5, which shows two numerical solutions (f_1 and f_2), the estimated exact solution from Richardson extrapolation \bar{f} , and the true exact solution \tilde{f} . In general, there is an equal chance that the true exact solution is above or below the estimated value. Thus a factor of safety of $F_s = 1$ centered about the fine grid numerical solution f_1 will only provide 50% chance that the true error \tilde{f} is within the error band. Increasing the factor of safety should increase the confidence that the true error is within the error band. This argument is made simply to justify the use of a factor of safety in converting the discretization error estimate into an epistemic uncertainty. It in no way implies that the resulting uncertainty is a randomly distributed variable (i.e., an aleatory uncertainty).

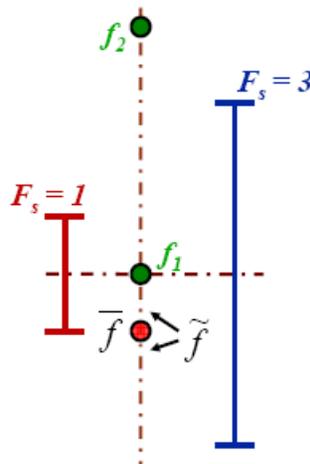


Figure 5. Factor of safety for extrapolation-based error bands (from Roy, 2005).

There is currently a debate going on in the extrapolation-based error estimation community regarding the factor of safety in the GCI and related uncertainty estimates. In the current formulation of the GCI, Roache (1998, 2003) recommends that the factor of safety be decreased from $F_s = 3$ to $F_s = 1.25$ when three discrete solutions are available on systematically-refined meshes. Roache's stated goal in formulating the GCI is to achieve a 95% certainty (i.e., that it provide conservative uncertainty estimates in 19 out of 20 cases) over a wide range of applications. Other authors have proposed different formulations for the factor of safety which allow its value to vary depending on how well the observed order of accuracy matches the formal order (e.g., Stern et al., 2001). Our own view is that the reliability of any discretization error or uncertainty estimate can only be determined by assessing the asymptotic nature of the discrete solutions. When the solutions are found to be far outside the asymptotic range, the reliability of the error/uncertainty estimate will be poor and its behavior erratic. In this case, no value for the factor of safety is guaranteed to be conservative.

3. Implementation

In order to avoid problems when the solution values are near zero, most recent implementations of the GCI omit the normalization by the fine grid solution in the denominator, i.e., the GCI is redefined as

$$GCI = \frac{F_s}{r^p - 1} |f_2 - f_1|, \quad (47)$$

This implementation thus provides an uncertainty estimate for the fine grid solution in the same units as the solution itself. Roache (1998) provides clear guidelines for choosing the factor of safety when:

1. solutions on only two grids are available ($F_s = 3$), or
2. solutions on three grids are available and the observed order of accuracy is calculated and shown to match the formal order of the scheme ($F_s = 1.25$).

In the first case, when only two grid levels are available, GCI estimates should be used with caution since one has no information on whether the solutions are in (or even near) the asymptotic range. Far from the asymptotic range, all approaches for estimating discretization error or uncertainty will be unreliable (see Section III). For the second case, when three solutions are available, if the observed order of accuracy matches the formal order, then $F_s = 1.25$ is used and either the formal or observed order of accuracy can be employed (since they match, this choice will have little effect on the estimate). Difficulties arise when the observed and formal orders of accuracy do not agree. Furthermore, exactly how does one define agreement? Ideally, one would simply continue to systematically refine the mesh until the solutions are demonstrably asymptotic, possibly incorporating local mesh refinement to achieve the asymptotic range faster. When no additional solutions are possible due to resource limitations, then Roache (1998) provides anecdotal examples of how to apply the GCI for a wide range of situations.

We propose the following procedure for calculating the GCI when solutions on three or more systematically-refined meshes are available. In all cases, the non-normalized GCI given by Equation (47) is used. When the observed order of accuracy \hat{p} agrees with the formal order p_f within 10%, then the formal order of accuracy along with a factor of safety of 1.25 is used in the GCI calculation. When the observed order of accuracy does not agree within 10%, then a factor of safety of three ($F_s = 3.0$) is used. Furthermore, the order of accuracy is limited between 0.1 and the formal order. Allowing the order of accuracy to be much larger than the formal order causes the uncertainty estimates to be unreasonably small since the GCI goes to zero as $p \rightarrow \infty$, while allowing the order of accuracy to go to zero causes the uncertainty estimate to approach infinity. These recommendations are summarized in Table 1. While these recommendations are “reasonable” (Roache, 2009), they require testing to see if they produce the desired 95% uncertainty bands on a wide range of problems.

Table 1. Proposed implementation of the GCI for solutions on three or more systematically-refined grids using Equation (47).

$\frac{ \hat{p} - p_f }{p_f}$	F_s	p
≤ 0.1	1.25	p_f
> 0.1	3.0	$\min(\max(0.1, \hat{p}), p_f)$

4. Reliability of the GCI

Three recent studies have examined the reliability of the GCI. Eca and Hoekstra (2002) examined a wide range of fluid flow problems using a large number of non-integer refined meshes (at least sixteen different grids for each case). They employed a locally-evaluated observed order of accuracy and found that the GCI with a factor of safety of $F_s = 1.25$ worked well on the finer meshes. Cadafalch et al. (2002) examined five test cases for fluid flow (some with heat transfer) using four to seven meshes for each case with a grid refinement factor of two. In their case, they employed a globally-determined order of accuracy found by averaging locally-computed values. They also found good results for the GCI with a factor of safety of 1.25. Finally, Eca et al. (2004) looked at potential flow solutions around 2D airfoils where exact solutions were available. They achieved good uncertainty estimates using the GCI

when the locally-evaluated observed order of accuracy wasn't "significantly larger" than the formal order of the method. See Pelletier and Roache (2006) for additional discussion of the effectiveness of the GCI.

V. Mesh Refinement Issues

Second only to the inability to achieve the asymptotic grid convergence range, the failure of extrapolation-based methods for estimating discretization error can often be tied to problems achieving uniform mesh refinement on the meshes in question (e.g., Baker, 2005; Salas, 2006). The use of locally-refined meshes, meshes refined in only one coordinate direction, and refinement where the refinement factor varies locally over the domain are all examples of improper approaches to mesh refinement. When combined with extrapolation-based methods for error or uncertainty estimation, these approaches are virtually guaranteed to fail.

A. Measuring Systematic Mesh Refinement

Recall the definition of systematic mesh refinement given in Section II.A.1, which requires that the mesh refinement be both uniform and consistent (Oberkampf and Roy, 2010). Uniform refinement requires that the mesh be refined by the same factor over the entire domain. This does not mean that the mesh itself must be uniform over the domain, only that the ratio of grid refinement from one mesh to another must not vary over the domain. Consistent refinement requires that the meshes should maintain the same grid quality (skewness, aspect ratio, stretching factor, etc.) or possibly provide for improved grid quality with mesh refinement.

A simple technique to ensure that two meshes retain the same volume ratio over the domain was developed by Roy et al. (2007) and applied to 3D unstructured Cartesian meshes used for aerodynamic force and moment computations. The level of uniformity of the mesh refinement was evaluated by comparing cell volume ratios between the two successive mesh levels as follows. First, the local cell volumes were calculated and stored at the nodes. Next, the fine grid volume distribution was interpolated onto the coarse grid with using an inverse distance function. Now that the coarse grid contains both fine and coarse grid cell volume information, the coarse to fine grid volume ratios can be calculated and examined. The volume ratios between two mesh levels are shown in Figure 6 and are fairly constant in the domain around a value of 1.6, with the exception of the regions where the Cartesian grid layers transition from one cell size to another. This approach could be easily extended to other mesh quality parameters such as skewness, aspect ratio, stretching factor, etc. (note: mesh quality was not an issue for the Cartesian meshes in this example).

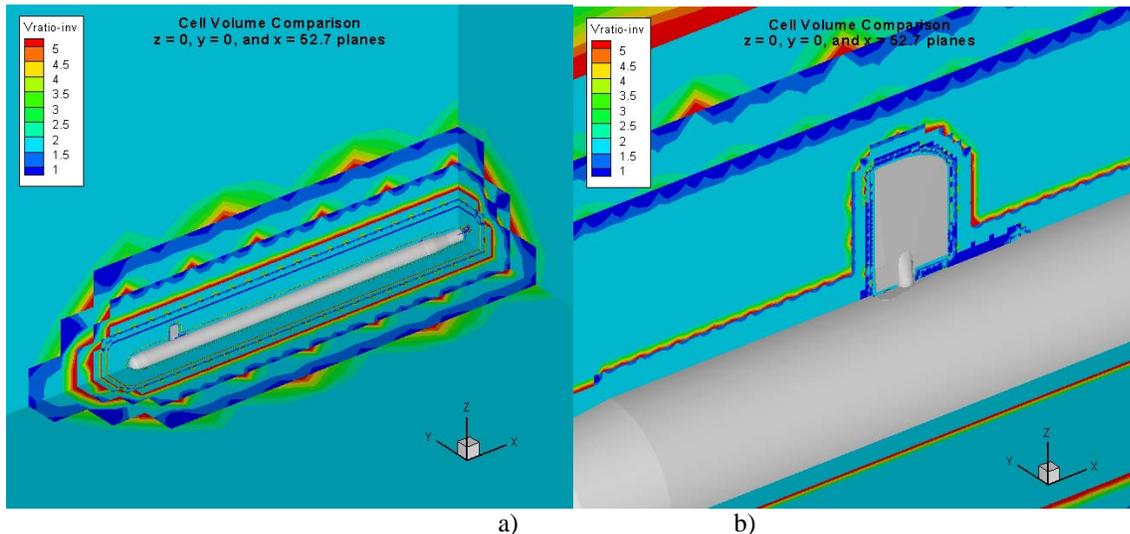


Figure 6. Contour plots of the cell volume ratio between the two unstructured Cartesian mesh levels for a) the entire missile geometry and b) a close-up of the canard (from Roy et al., 2007).

B. Grid Refinement Factor

As discussed earlier, the grid refinement factor is a measure of the ratio of mesh length or time scales in each of the coordinate directions. The refinement factor is simple to compute for structured grids, while for unstructured grids the refinement factor can be computed as follows. Assuming that the mesh has been refined uniformly, the refinement factor can be computed as a function of the total number of cells or elements in the mesh by

$$r_{12} = \left(\frac{N_1}{N_2} \right)^{1/d} \quad (48)$$

where N_k is the number of cells/elements on mesh k and d is the dimensionality of the problem ($d = 1$ for 1D, $d = 2$, for 2D, etc.). Note that in the case where local mesh refinement is applied from one mesh to another, the grid refinement factor will vary over the domain and the numerical solutions should no longer be used in an extrapolation procedure or for the computation of the observed order of accuracy.

C. Fractional Uniform Refinement

It is not necessary to use grid refinement factors of two, a process referred to as grid doubling or grid halving, depending on whether one starts with the fine mesh or the coarse mesh. For simple meshes, grid refinement factors as small as $r = 1.1$ can be employed (Roache, 1998). Since the refinement must always be uniform, we will herein refer to this procedure as fractional uniform refinement. When refinement factors near one are used, one should take special care to ensure that round-off error does not become large relative to the difference between the two solutions. For example, solving a 2D problem on two meshes of size 100×100 and 101×101 elements (i.e., $r = 1.01$) could easily result in solutions that differed by less than 0.01%. Single precision computations generally provide solution errors due to round off of approximately 0.001%, so the expected round-off errors will be only one magnitude smaller than the discretization errors.

Using fractional uniform refinement has the added benefit that it increases the chances of getting discrete solutions on multiple meshes into the asymptotic grid convergence range. However, non-integer grid refinement factors are difficult to apply to complex mesh topologies, especially those involving significant mesh stretching. For simulations using complex, structured meshes, the grid generation can sometimes make up the majority of the overall analysis time. Thus, relying on the original grid generation procedure for grid refinement can be expensive; furthermore, it is difficult to enforce a constant grid refinement factor over the entire domain. Higher-order interpolation can be used for non-integer grid refinement. Here it is again better to start with the fine mesh and then coarsen (at least for structured meshes); however, this approach may not preserve the underlying surface geometry, especially in regions of high curvature.

When a grid refinement factor of two is employed on structured meshes, there is only significant effort involved in generating the fine mesh; the coarser meshes are found by simply removing every other point. The drawback is not only that the fine mesh may be unnecessarily expensive, but there is also an increased chance that the coarse mesh will be outside the asymptotic grid convergence range.

Ideally, one would like the ability to simply create a grid and then have a grid generation tool generate a family of uniformly-refined meshes based on specified (possibly fractional) refinement factors. Such a capability would greatly improve the reliability of extrapolation-based error and uncertainty estimation procedures. To our knowledge, this automatic fractional uniform refinement capability does not exist in any commercially-available grid generators.

Finally, if the extrapolation (or error/uncertainty) estimation method is to be applied to the solution point-wise through the domain, then it is necessary to get fine grid information onto the coarse grid. This procedure is simple for integer refinement since the coarse grid is simply a subset of the fine grid. However, for fractional uniform refinement, fine mesh information must be interpolated to the coarse mesh points, possibly with an interpolation scheme that has a higher order of accuracy than the underlying numerical solutions (Roache and Knupp, 1993).

D. Unidirectional Refinement

It is sometimes the case that the discretization errors come primarily from just one of the coordinate directions. In such cases, it can be helpful to perform independent refinement in the coordinate directions to determine which one is the primary contributor to the overall discretization error. The approach discussed here is similar to the combined space-time method for code order of accuracy verification discussed in Oberkampf and Roy (2010). For independent refinement in x and y , we can write the expansion of the numerical solution on mesh k about the exact solution to the original partial differential equation as

$$f_k = \tilde{f} + g_x(\Delta x_k)^p + g_y(\Delta y_k)^q + \dots \quad (49)$$

where the error terms for each coordinate direction are included. In order to keep the analysis general, the formal order of accuracy in the x direction is assumed to be p and the order of accuracy in the y direction to be q , where the two may or may not be equal. Note that for some numerical schemes (e.g., the Lax-Wendroff scheme), a cross term $g_{xy}(\Delta x)^s(\Delta y)^t$ may also be present.

Consider the case of solutions on two meshes levels (fine: $k = 1$ and coarse: $k = 2$) with refinement only in the x direction by a factor of r_x . As the Δx element size is refined, the term $g_y(\Delta y_k)^q$ will be constant. We are now unable to solve for an estimate of the exact solution \tilde{f} , but instead must solve for the quantity

$$\tilde{f}_x = \tilde{f} + g_y(\Delta y_k)^q \quad (50)$$

which includes the error term due to the Δy discretization. This term will simply be constant on the two meshes since the Δy spacing does not change. Neglecting the higher-order terms in the estimated solution \overline{f} results in the following two equations

$$\begin{aligned} f_1 &= \overline{f}_x + g_x(\Delta x)^p \\ f_2 &= \overline{f}_x + g_x(r_x \Delta x)^p \end{aligned}$$

which can be solved for \overline{f}_x

$$\overline{f}_x = f_1 + \frac{f_1 - f_2}{r_x^p - 1} \quad (51)$$

and the leading x -direction error term

$$g_x(\Delta x)^p = \frac{f_2 - f_1}{r_x^p - 1} \quad (52)$$

Similarly, introducing a third solution ($k = 3$) with coarsening only in the y direction allows us to solve for the y -direction error term

$$g_y(\Delta y)^q = \frac{f_3 - f_1}{r_y^q - 1} \quad (53)$$

The size of the two error terms from Equations (52) and (53) can then be compared to determine the appropriate direction for further mesh refinement. In addition, since g_x and g_y have been estimated, Equation (49) can be used to obtain an estimate of \tilde{f} .

VI. Open Research Issues

There are a number of issues which are currently active areas of research. These include the presence of singularities and discontinuities (especially in hyperbolic problems), oscillatory grid convergence, multiscale models, and approaches for estimating discretization errors on coarse grids which are not in the asymptotic range.

A. Singularities and Discontinuities

Practical scientific computing applications are fraught with singularities and discontinuities. In some cases, they may take the form of relatively weak singularities/discontinuities such as discontinuous surface curvature in a linear elasticity analysis. Examples of strong singularities include shock waves in inviscid flow, the leading edge region in the flow over a sharp-nosed body, and the interface between two different materials in thermal or structural analysis problems. Nature tends to “smooth out” singularities such as the sharp leading edge that takes a more rounded appearance under high magnification or the shock wave that actually transports mass, momentum, and energy in the upstream direction via molecular motion. However, singularities and discontinuities are much more common than is often recognized, especially in idealized geometries and simplified mathematical models. Some examples include

flow over a surface with discontinuous slope or curvature, loading of a structure with point loads, and loading of a plate with angled holes.

The presence of singularities and discontinuities can adversely impact our ability to obtain reliable estimates of the discretization error as well as our ability to compute the numerical solutions themselves. This is because all of the discretization error estimators (and most discretization schemes) require that the solutions be continuous and differentiable. Nevertheless, we still must compute numerical solutions to these problems and then estimate the discretization error in those solutions.

Work by Carpenter and Casper (1999) suggests that for the Euler equations on sufficiently refined meshes, the simulation of flows containing shock waves will reduce down to first-order accuracy, regardless of the formal order of the numerical scheme employed. This reduction to first-order accuracy in the presence of shock waves has also been observed by Roy (2001, 2003) for laminar and inviscid hypersonic flows. In these latter cases, flux limiters were employed to reduce a formally second-order accurate scheme to first order at a shock wave to prevent numerical oscillations.

Banks et al. (2008) provide an excellent explanation of why the formal order of accuracy of any discretization scheme will reduce to first-order accuracy or lower in the presence of discontinuities. They consider the inviscid Euler equations for compressible flows. The Euler equations admit discontinuous solutions such as shock waves, contact discontinuities, and slip lines. However, in the absence of viscosity which enforces the entropy condition in viscous flows, these discontinuous solutions are not unique (i.e., the Euler equations will admit infinitely many jump conditions). It is only in the limit of a vanishing viscosity that the correct Rankine-Hugoniot jump conditions are met. Note that this non-uniqueness generally does not pose a problem for numerical solutions to the Euler equations due to the presence of numerical viscosity in the form of even spatial derivatives in the truncation error. For nonlinear discontinuities such as shock waves, numerical schemes reduce down to first-order accuracy. For linear discontinuities (i.e., linearly degenerate waves) such as contact discontinuities and slip lines, they perform a truncation error analysis to show that most discretization schemes reduce their formal order of accuracy to $p/(p+1)$, where p is the formal order of the method for smooth problems. They also provide numerical examples to confirm their theoretical analysis.

Regardless of the discretization scheme employed, the reduction of the order of accuracy to first order in the presence of discontinuities results in a numerical scheme that is of mixed order. In this case, the scheme will be second-order accurate (or higher) in the smooth regions, but locally first-order accurate at the shock wave. This mixed order behavior can lead to non-monotone convergence of the solutions with grid refinement, as is demonstrated below. In order to analyze mixed first- and second-order schemes, Roy (2001, 2003) proposed the following expansion for the discretization error

$$f_k = \bar{f} + g_1 h_k + g_2 h_k^2 \quad (54)$$

In this approach, terms accounting for the formal order of accuracy in both smooth regions ($p = 2$) and non-smooth regions ($p = 1$) are included. If three mesh levels are available, then Equation (54) can be solved for the two unknown coefficients g_1 and g_2 as well as the estimated exact solution \bar{f} . Consider the case where three discrete solutions f_k are available on fine ($k = 1$), medium ($k = 2$), and coarse ($k = 3$) meshes, where the grid refinement factor is held constant between mesh levels (i.e., $r_{12} = r_{23} = r$). If we further arbitrarily set $h_1 = 1$ (this will simply require the absorbing of constants in the g_k coefficients), then the expressions for the three unknowns become:

$$\begin{aligned} g_1 &= \frac{r^2(f_2 - f_1) - (f_3 - f_2)}{r(r-1)^2} \\ g_2 &= \frac{(f_3 - f_2) - r(f_2 - f_1)}{r(r+1)(r-1)^2} \\ \bar{f} &= f_1 + \frac{(f_3 - f_2) - (r^2 + r - 1)(f_2 - f_1)}{(r+1)(r-1)^2} \end{aligned} \quad (55)$$

These three expressions can be used to not only estimate the discretization error in the numerical solutions, but also to examine the behavior of the two different error terms in the discretization error expansion given by Equation (54).

This mixed-order analysis method has been applied to the dimensionless frictional resistance (i.e., skin friction coefficient) at points along the surface of a sphere-cone geometry in laminar, hypersonic flow (Roy, 2001). The results are shown below in Figure 7a for the behavior of the skin friction coefficient with mesh refinement. The

solution displays non-monotone behavior as the skin friction first decreases with mesh refinement and then begins to increase. Insight into why this non-monotone behavior occurs can be found by examining the discretization error given in Figure 7b which also shows the first- and second-order error terms from Equation (54). In this case, both error terms have the same magnitude near $h = 4$, but they have opposite signs. This leads to error cancellation at $h = 4$ which manifests as a crossing of the estimated exact solution in Figure 7a. Since the two terms have opposite signs, when the mesh is sufficiently refined such that the first order term dominates, the skin friction coefficient displays a local minimum when plotted versus the mesh spacing h .

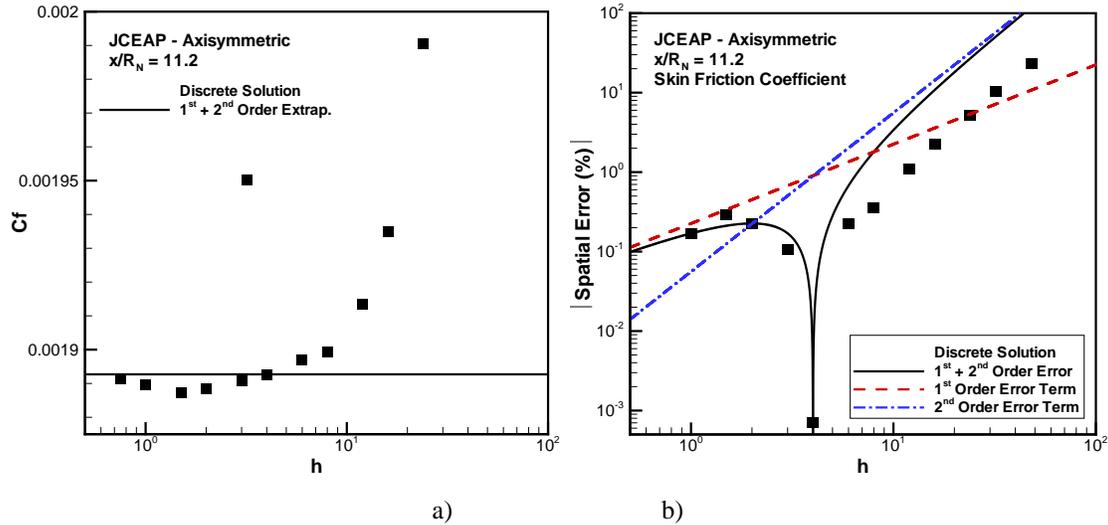


Figure 7. Dimensionless frictional resistance at one location on the body as a function of mesh spacing: a) skin friction coefficient and b) discretization error (from Roy, 2001).

B. Oscillatory Convergence with Mesh Refinement

There has been much discussion in recent years on the presence of oscillatory convergence of numerical solutions with mesh refinement (e.g., see Coleman et al., 2001; Celik et al., 2005; and Brock, 2007). However, examination of the Taylor series expansion of the numerical solution about the exact solution the governing partial differential equations in terms of the mesh spacing parameter as given in Equation (5), or even the more general series expansion from Equation (6), shows that there can be only a single term that dominates the expansion *in the limit* as $h \rightarrow 0$. While this debate may not yet be settled, our opinion is that there can be no oscillatory convergence with mesh refinement in the asymptotic grid convergence range. The solution behavior being interpreted as oscillatory convergence is mostly likely due to a failure to achieve the asymptotic range (as discussed in Section III.A), nonuniform refinement of the mesh (Eca and Hoekstra, 2002), or the presence of non-negligible iterative and/or round-off errors. In the case of the former, we do not dispute the fact that the asymptotic range may be extremely difficult to achieve for complex scientific computing problems.

C. Multiscale Models

A multiscale model is one which resolved different physical phenomena at different mesh resolutions. For some multiscale models, the governing equations actually change as the mesh is refined, thus making it difficult to separate out issues of verification (mathematics) from those of validation (physics). A classic example of a multiscale model is in the Large Eddy Simulation (LES) of turbulent flows. In most LES computations, the mesh itself is used as a filter for the small scale turbulent structures. Thus as one refines the mesh, it is equivalent to solving the Navier-Stokes equations with a different filter size (i.e., a different set of governing equations). The rigorous assessment of numerical errors and physical modeling errors thus becomes extremely difficult since both are tied to the mesh spacing. One possible remedy for this problem is to fix the filter width at the coarse grid spacing while performing mesh refinement to assess numerical errors (Moin, 2007).

D. Coarse Grid Error Estimators

For scientific computing simulations where the physics and/or geometry are complex, the meshes are often under-resolved due to computational resource limitations. This is especially true for 3D time-dependent problems. The ideal discretization error estimator is one that not only provides consistent error estimates as the mesh is refined, but also provides reliable error estimates on under-resolved coarse grids which are outside of the asymptotic grid convergence range. Although often buried deep within the details, all of the discretization error estimation methods discussed in this paper require the solutions to be within the asymptotic range. The reliability of discretization error estimators when the discrete solutions are not in (and possibly far from) the asymptotic range is an area for further investigation.

VII. Acknowledgments

The author would like to thank Jim Stewart, Tim Trucano, and Fred Blottner of Sandia National Laboratories and Luis Eca of the Technical University of Lisbon for their many insightful comments and discussions on discretization error estimation. Partial support for this work was provided by a grant from Sandia National Laboratories through a U.S. Department of Energy Presidential Early Career Award for Scientists and Engineers.

VIII. References

1. Ainsworth, M. and Oden, J. T. (1997). "A Posteriori Error Estimation in Finite Element Analysis," *Computer Methods in Applied Mechanics and Engineering*, Vol. 142, No. 1-2, pp. 1-88.
2. Ainsworth, M. and Oden, J. T. (2000). *A Posteriori Error Estimation in Finite Element Analysis*, Wiley Interscience, New York.
3. Akin, J. E. (2005). *Finite Element Analysis with Error Estimators*, Elsevier, Burlington.
4. Babuska, I. and Rheinboldt, W. C. (1978a). "A Posteriori Error Estimates for the Finite Element Method," *International Journal for Numerical Methods in Engineering*, Vol. 12, No. 10, pp. 1597-1615.
5. Babuska, I. and Rheinboldt, W. C. (1978b). "Error Estimates for Adaptive Finite Element Computations," *SIAM Journal of Numerical Analysis*, Vol. 15, No. 4, pp. 736-754.
6. Babuska, I., Strouboulis, T., Gangaraj, T., and Upadhyay, C. S. (1997). "Pollution Error in the h-Version of the Finite Element Method and Local Quality of the Recovered Derivative," *Computer Methods in Applied Mechanics and Engineering*, Vol. 140, No. 1-2, pp. 1-37.
7. Babuska, I., Strouboulis, T., and Upadhyay, C. S. (1994). "A Model Study of the Quality of A Posteriori Error Estimators for Linear Elliptic Problems. Error Estimation in the Interior of Patchwise Uniform Grids of Triangles," *Computer Methods in Applied Mechanics and Engineering*, Vol. 114, No. 3-4, pp. 307-378.
8. Babuska, I., Zienkiewicz, O. C., Gago, J., Oliveira, E. R. (1986). *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, Wiley, Chichester.
9. Baker, T. J. (2005). "On the Relationship between Mesh Refinement and Solution Accuracy," AIAA Paper 2005-4875, June 2005.
10. Bank, R. E. (1996). "Hierarchical Bases and the Finite Element Method," *Acta Numerica*, Vol. 5, pp. 1-45.
11. Banks, J.W., Aslam, T., and Rider, W. J. (2008). "On Sub-linear Convergence for Linearly Degenerate Waves in Capturing Schemes," *Journal of Computational Physics*, Vol. 227, pp. 6985-7002.
12. Brock, J. S. (2007). "Bounded Numerical Error Estimates for Oscillatory Convergence of Simulation Data," AIAA Paper 2007-4091, June 2007.
13. Cadafalch, J., Perez-Segarra, C. D., Consul, R., and Oliva, A. (2002). "Verification of Finite Volume Computations on Steady-State Fluid Flow and Heat Transfer," *Journal of Fluids Engineering*, Vol. 24, pp. 11-21.
14. Carpenter, M. H., and Casper, J. H. (1999). "Accuracy of Shock Capturing in Two Spatial Dimensions," *AIAA Journal*, Vol. 37, No. 9, pp. 1072-1079.
15. Cavallo, P. A. and Sinha, N. (2007). "Error Quantification for Computational Aerodynamics Using an Error Transport Equation," *Journal of Aircraft*, Vol. 44, No. 6, pp. 1954-1963.
16. Celik, I., Li, J., Hu, G., and Shaffer, C. (2005). "Limitations of Richardson Extrapolation and Some Possible Remedies," *Journal of Fluids Engineering*, Vol. 127, pp. 795-805.
17. Cheng, Z. and Paraschivoiu, M. (2004). "A Posteriori Finite Element Bounds to Linear Functional Outputs of the Three-dimensional Navier-Stokes Equations," *International Journal for Numerical Methods in Engineering*, Vol. 61, No. 11, pp. 1835-1859.
18. Coleman, H. W., Stern, F., Di Mascio, A., and Campana, E. (2001). "The Problem with Oscillatory Behavior in Grid Convergence Studies," *Journal of Fluids Engineering*, Vol. 123, pp. 438-439.

19. Eca, L. and Hoekstra, M. (2002). "An Evaluation of Verification Procedures for CFD Applications," 24th Symposium on Naval Hydrodynamics, Fukuoka, Japan, July 8-13, 2002.
20. Eca, L. and Hoekstra, M. (2009a). "Error Estimation Based on Grid Refinement Studies: A Challenge for Grid Generation," Congress on Numerical Methods in Engineering, Barcelona, Spain, June 29-July 2, 2009.
21. Eca, L. and Hoekstra, M. (2009b). "Evaluation of Numerical Error Estimation Based on Grid Refinement Studies with the Method of the Manufactured Solutions," *Computers and Fluids*, Vol. 38, pp. 1580-1591.
22. Eca, L., Vaz, G. B., Falcao de Campos, J. A. C., and Hoekstra, M. (2004). "Verification of Calculations of the Potential Flow Around Two-Dimensional Foils," *AIAA Journal*, Vol. 42, No. 12, pp. 2401-2407.
23. Eriksson, K. and Johnson, C. (1987). "Error-Estimates and Automatic Time Step Control for Nonlinear Parabolic Problems. Part 1," *SIAM Journal of Numerical Analysis*, Vol. 24, No. 1, pp. 12-23.
24. Estep, D., Larson, M., and Williams, R. (2000). *Estimating the Error of Numerical Solutions of Systems of Nonlinear Reaction-Diffusion Equations*, Memoirs of the American Mathematical Society, Vol. 146, No. 696, American Mathematical Society, Providence.
25. Fehlberg, E. (1969). "Low-Order Classical Runge-Kutta Formulas with Step Size Control and their Application to some Heat Transfer Problems," NASA Technical Report 315, National Aeronautics and Space Administration, July 1969.
26. Ferziger, J. H. and Peric, M. (2002). *Computational Methods for Fluid Dynamics*, 3rd Ed., Springer, Berlin.
27. Garbey, M. and Shyy, W. (2003). "A Least Square Extrapolation Method for Improving Solution Accuracy of PDE Computations," *Journal of Computational Physics*, Vol. 186, No. 1, pp. 1-23.
28. Hirsch, C. (1990). *Numerical Computation of Internal and External Flows: Volume 2 Computational Methods for Inviscid and Viscous Flows*, Wiley, Chichester.
29. Jameson, A. (1988). "Aerodynamic Design via Control Theory," *Journal of Scientific Computing*, Vol. 3, No. 3, pp. 233-260.
30. Jameson, A., Schmidt, W., and Turkel, E. (1981). "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259.
31. Johnson, C. and Hansbo, P. (1992). "Adaptive Finite Element Methods in Computational Mechanics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 101, Nos. 1-3, pp. 143-181.
32. Mastin, C. W. (1999). "Truncation Error on Structured Grids," in *Handbook of Grid Generation*, J. F. Thompson, B. K. Soni, and N. P. Weatherill, Eds., CRC Press, Boca Raton
33. Moin, P. (2007). "Application of High Fidelity Numerical Simulations for Vehicle Aerodynamics," The Aerodynamics of Heavy Vehicles II: Trucks, Buses and Trains, Tahoe City, California, August 26-31, 2007.
34. Oberkampf, W. L. and Roy, C. J. (2010). *Verification and Validation in Scientific Computing*, Cambridge University Press, New York, to appear 2010.
35. Parascioiu, M., Peraire, J., and Patera, A. T. (1997). "A Posteriori Finite Element Bounds for Linear Functional Outputs of Elliptic Partial Differential Equations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 150, No. 1-4, pp. 289-312.
36. Pelletier, D. and Roache, P. J. (2006). "Chapter 13: Verification and Validation of Computational Heat Transfer," *Handbook of Numerical Heat Transfer*, 2nd Ed., Minkowycz, Sparrow, and Murthy, Eds., Wiley, Hoboken, New Jersey.
37. Pierce, N. A. and Giles, M. B. (2000). "Adjoint recovery of superconvergent functionals from PDE approximations," *SIAM Review*, Vol. 42, No. 2, pp. 247-264.
38. Rannacher, R. and Suttmeier, F. T. (1997). "A Feed-Back Approach to Error Control in Finite Element Methods: Application to Linear Elasticity," *Computational Mechanics*, Vol. 19, No. 5, pp. 434-446.
39. Richards, S. A. (1997). "Completed Richardson Extrapolation in Space and Time," *Communications in Numerical Methods in Engineering*, Vol. 13, 1997, pp. 573-582.
40. Richardson, L. F. (1911). "The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, Vol. 210, pp. 307-357.
41. Richardson, L. F. (1927). "The Deferred Approach to the Limit. Part I. Single Lattice," *Philosophical Transaction of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, Vol. 226, pp. 299-349.
42. Richtmyer, R. and Morton, K. (1967). *Difference Methods for Initial-Value Problems*, 2nd Ed., Interscience Publishers, New York.
43. Roache, P. J. (1994). "Perspective: A Method for Uniform Reporting of Grid Refinement Studies," *Journal of Fluids Engineering*, Vol. 116, pp. 405-413.
44. Roache, P. J. (1998). *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, New Mexico.
45. Roache, P. J. (2003). "Conservatism of the Grid Convergence Index in Finite Volume Computations on Steady-State Fluid Flow and Heat Transfer," *Journal of Fluids Engineering*, Vol. 125, No. 4, pp. 731-732.

46. Roache, P. J. (2009). Private Communication, July 13, 2009.
47. Roache, P. J. and Knupp, P. M. (1993). "Completed Richardson Extrapolation," *Communications in Numerical Methods in Engineering*, Vol. 9., No. 5, pp. 365-374.
48. Roy, C. J. (2001). "Grid Convergence Error Analysis for Mixed-Order Numerical Schemes," AIAA Paper 2001-2606, June 2001.
49. Roy, C. J. (2003). "Grid Convergence Error Analysis for Mixed-Order Numerical Schemes," *AIAA Journal*, Vol. 41, No. 4, pp. 595-604.
50. Roy, C. J. (2005). "Review of Code and Solution Verification Procedures for Computational Simulation," *Journal of Computational Physics*, Vol. 205, No. 1, 2005, pp. 131-156.
51. Roy, C. J. (2009). "Strategies for Driving Mesh Adaptation in CFD (Invited)," AIAA Paper 2009-1302, 47th AIAA Aerospace Sciences Meeting, Orlando, Florida, January 5-8, 2009.
52. Roy, C. J. and Blotner, F. G. (2003). "Methodology for Turbulence Model Validation: Application to Hypersonic Transitional Flows," *Journal of Spacecraft and Rockets*, Vol. 40, No. 3, pp. 313-325.
53. Roy, C. J., Heintzelman, C. J., and Roberts, S. J. (2007). "Estimation of Numerical Error for 3D Inviscid Flows on Cartesian Grids," AIAA Paper 2007-0102, Reno, NV, Jan. 2007.
54. Salas, M. D. (2006). "Some Observations on Grid Convergence," *Computers and Fluids*, Vol. 35, pp. 688-692.
55. Salas, M. D. and Atkins, H. L. (2009). "On Problems Associated with Grid Convergence of Functionals," *Computers and Fluids*, Vol. 38, pp. 1445-1454.
56. Shih, T. I-P. and Qin, Y. C. (2007). "A Posteriori Method for Estimating and Correcting Grid-Induced Errors in CFD Solutions Part 1: Theory and Method," AIAA Paper 2007-100.
57. Shih, T. I-P., and Williams, B. R. (2009). "Development and Evaluation of an A Posteriori Method for Estimating and Correcting Grid-Induced Errors in Solutions of the Navier-Stokes Equations," AIAA Paper 2009-1499.
58. Sonar,
59. Stewart, J. R. and Hughes, T. J. R. (1998). "A Tutorial in Elementary Finite Element Error Analysis: A Systematic Presentation of A Priori and A Posteriori Error Estimates," *Computer Methods in Applied Mechanics and Engineering*, Vol. 158, No. 1-2, pp. 1-22.
60. Stern, F., Wilson, R. V., Coleman, H. W., and Paterson, E. G. (2001). "Comprehensive Approach to Verification and Validation of CFD Simulations—Part I: Methodology and Procedures," *ASME Journal of Fluids Engineering*, Vol. 123, No. 4, pp. 793-802.
61. Sonar, T. (1993). "Strong and Weak Norm Refinement Indicators Based on the Finite Element Residual for Compressible Flow Computation: 1. The Steady Case," *Impact of Computing in Science and Engineering*, Vol. 5, No. 2, pp. 111-127.
62. Szabo, B. A. and Babuska, I. (1991). *Finite Element Analysis*, Wiley, New York.
63. Thompson, J. F., Warsi, Z. U. A., and Mastin, C. W. (1985). *Numerical Grid Generation: Foundations and Applications*, Elsevier, New York.
64. Venditti, D. A. and Darmofal, D. L. (2000). "Adjoint Error Estimation and Grid Adaptation for Functional Outputs: Application to Quasi-One Dimensional Flow," *Journal of Computational Physics*, Vol. 164, pp. 204-227.
65. Venditti, D. A. and Darmofal, D. L. (2002). "Grid Adaptation for Functional Outputs: Application to Two-Dimensional Inviscid Flows," *Journal of Computational Physics*, Vol. 176, pp. 40-69.
66. Venditti, D. A. and Darmofal, D. L. (2003). "Anisotropic Grid Adaptation for Functional Outputs: Application to Two-Dimensional Viscous Flows," *Journal of Computational Physics*, Vol. 187, pp. 22-46.
67. Wahlbin, L. B. (1995). *Superconvergence in Galerkin Finite Element Methods*, Volume 1605 of Lecture Notes in Mathematics, Springer-Verlag, Berlin.
68. Whiteman, J. R. (1994). *The Mathematics of Finite Element and Applications: Highlights 1993*, Wiley, New York.
69. Zhang, X. D., Trepanier, J.-Y., and Camarero, R. (2000). "A Posteriori Error Estimation for Finite-Volume Solutions of Hyperbolic Conservation Laws," *Computer Methods in Applied Mechanics and Engineering*, Vol. 185, No. 1, pp. 1-19.
70. Zhang, Z. and Naga, A. (2005). "A New Finite Element Gradient Recovery Method: Superconvergence Property," *SIAM Journal of Scientific Computing*, Vol. 26, No. 4, pp. 1192-1213.
71. Zienkiewicz, O. C. and Zhu, J. Z. (1987). "A Simple Error Estimator and Adaptive Procedure for Practical Engineering Analysis," *International Journal for Numerical Methods in Engineering*, Vol. 24, pp. 337-357.
72. Zienkiewicz, O. C. and Zhu, J. Z. (1992). "The Superconvergent Patch Recovery and A Posteriori Error Estimates, Part 2: Error Estimates and Adaptivity," *International Journal for Numerical Methods in Engineering*, Vol. 33, pp. 1365-1382.