SANDIA REPORT
SAND98–2222
Unlimited Release
Printed October 1998

Determination of Solution Accuracy of Numerical Schemes as Part of Code and Calculation Verification

F. G. Blottner and A. R. Lopez

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

NTIS price codes
Printed copy: A05
Microfiche copy: A01

SAND98-2222
Unlimited Release
Printed October 1998

Determination of Solution Accuracy of Numerical Schemes as Part of Code and Calculation Verification

F. G. Blottner and A. R. Lopez
Aerosciences and Compressible Fluid Mechanics Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-0825

Abstract

This investigation is concerned with the accuracy of numerical schemes for solving partial differential equations used in science and engineering simulation codes. Richardson extrapolation methods for steady and unsteady problems with structured meshes are presented as part of the verification procedure to determine code and calculation accuracy. The local truncation error determination of a numerical difference scheme is shown to be a significant component of the verification procedure as it determines the consistency of the numerical scheme, the order of the numerical scheme, and the restrictions on the mesh variation with a non-uniform mesh. Generation of a series of co-located, refined meshes with the appropriate variation of mesh cell size is investigated and is another important component of the verification procedure. The importance of mesh refinement studies is shown to be more significant than just a procedure to determine solution accuracy. It is suggested that mesh refinement techniques can be developed to determine consistency of numerical schemes and to determine if governing equations are well posed. The present investigation provides further insight into the conditions and procedures required to effectively use Richardson extrapolation with mesh refinement studies to achieve confidence that simulation codes are producing accurate numerical solutions.

Acknowledgment

The authors thank Bennie Blackwell and Kambiz Salari for their insightful comments and constructive suggestions during the review of this report.

Contents

Figures

Tables

Nomenclature

A	Surface area in control volume formulation
A, B, C	Coefficients in discretization error difference Eq. (9)
	or coefficients in difference equation (See examples 1, 2, 3)
$A_i(\Delta t)$	Temporal coefficient in discretization error expansion
a, b, c	Spatial coefficients in discretization error expansion for 3D case
$\bar{a}, \bar{b}$	Local truncation error terms defined in Eq. (27)
$\tilde{a}, \tilde{b}$	Local truncation error terms defined in Eq. (43)
d	Exponential mesh parameter = κ_{i+1}/κ_i
E	Flux in x direction in integral formulation
Ee	Global solution percent error relative to exact solution, see Eq. (14)
Er	Global solution percent error relative to reference value, see Eq. (14)
e_i^n	Discretization error at time n and spatial node location i
$F(t, x)$	Exact solution of partial differential equation
F_e	Exact solution of governing equation in Richardson extrapolation formulation
F_r	Reference value of solution for use in determining solution error
f_i^n	Exact solution of difference equation at time n and spatial node location i
f_m	Computational solution on mesh m in Richardson extrapolation formulation
$f_{m, m+1, \ldots}$	Richardson extrapolation solution from numerical solutions on mesh m, $m+1$, ...
H	Value of x at end of solution domain
h	Uniform cell size
h_i	Non-uniform cell size = $(x_i - x_{i-1})$
h_m	Uniform cell size on mesh m
I_T	Total number of spatial node points in solution domain
$L(F)$	Representation of partial differential equation
l	Diffusion coefficient, e.g., viscosity or thermal conductivity
M	Mesh number of finest mesh

m	Mesh number as refinement is performed ($m = 0$ is the coarsest mesh)
Q	Dependent variable in PDE for box scheme, Eq. (35)
q	Dependent variable in difference equation, Eq. (36)
r	Mesh refinement ratio = h_m / h_{m+1}
$S(x)$	Source term in partial differential equation
t	Time
W	Right-hand side of PDE for box scheme, Eq. (35)
w	Right-hand side of difference equation for box scheme, Eq. (36)
x, y, z	Cartesian coordinates
x_i, y_j, z_k	Node location

Greek letters

α	Weight factor, see Eq. (25)
α, β	Spatial coefficient in discretization error expansion for Richardson extrapolation or coefficients in differential Eq. (28)
$\Delta(f_i^n)$	Representation of difference equation
Δ_0	Difference relation expansion for time derivative
Δ_1	Difference relation expansion for first spatial derivative
Δ_2	Difference relation expansion for second spatial derivative
Δt	Time step
$\Delta x, \Delta y, \Delta z$	Cell size in x, y, and z coordinate directions
$\delta(m)$	Smallest cell size in mesh m
θ	Parameter used to determine if time marching is explicit ($\theta = 0$) or implicit ($\theta = 1$)
κ_i	Spatial mesh ratio = $\Delta x_{i+1} / \Delta x_i$
λ	Parameter used in solution of difference equation, see Section 4.1
ξ	Curvilinear coordinate, which is a function of x
σ	Step-size parameter = $\Delta t / \Delta x^2$
τ_i^n	Local truncation error at time n and node location i

τ_0	Local truncation error for time derivative
τ_1	Local truncation error for first spatial derivative
τ_2	Local truncation error for second spatial derivative
φ	Use to represent any dependent variable

Other symbols

$\%Ee_i$	Solution percent error relative to the exact solution at node i, see Eq. (12)
$\%Er_i$	Solution percent error relative to a reference value at node i, see Eq. (13)

Superscripts

n	Time index, $t^n = n\Delta t$
p	Order of leading term in discretization error expansion, Eq. (34)

Subscripts

avg	Average
d	Mid-point location between $i-1$ and i
e	Mid-point location between i and $i+1$
I, J, K	Maximum value of grid indices
i, j, k	Grid indices or node locations along x, y, and z coordinates
Fd	Truncation error in cell d of PDE with F as the dependent variable
Fe	Truncation error in cell e of PDE with F as the dependent variable
Qd	Truncation error in cell d of PDE with Q as the dependent variable
Qe	Truncation error in cell e of PDE with Q as the dependent variable
t	Indicates a derivative with respect to t
x	Indicates a derivative with respect to x

Executive Summary

The verification and validation process is required to obtain confidence that a science and engineering simulation code is producing accurate numerical results. These processes are defined as:

Verification: The process of determining that a model implementation accurately represents the developer's conceptual description of the model and the solution to the model.

Validation: The process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model.

The major efforts required for verification and validation of simulation codes and solutions are as follows: code verification, calculation verification, governing equation validation, and code model validation. These four efforts are defined and briefly described in the introduction. This report is concerned with part of the verification process where numerical solution error is evaluated with Richardson extrapolation which requires solutions on several refined meshes. The importance of mesh refinement studies on numerical solution verification and the potential impact on determining if the governing equations are mathematically well posed is also indicated. A literature review on papers concerned with evaluation of numerical accuracy is provided. This review covers the formulation of the governing equations in Cartesian coordinates, non-uniform Cartesian meshes, generalized curvilinear coordinates, and non-uniform and unstructured meshes.

For structured meshes, the Richardson extrapolation method is demonstrated to be the most appropriate approach to estimate solution error. The Richardson extrapolation method requires a numerical scheme that is stable and consistent. The difference equation of a consistent numerical scheme accurately represents the partial differential equation as the mesh spatial sizes go to zero. In addition, Richardson extrapolation method is more effective if knowledge of the behavior of the discretization error is available. The local truncation error of the numerical difference equation is shown to be proportional to the discretization error for linear difference equations. As the local truncation error can be determined analytically for finite difference and control volume methods, its evaluation is a necessary part of the verification process. Procedures to determine the local and global solution errors are also given. Three simple examples are used to illustrate the errors for steady-state problems with uniform meshes. The examples have both exact solutions to the differential equations and the numerical difference equations. One example shows that the solution error relative to the exact solution goes to infinity at the location where the solution passes through zero even through the numerical solution is reasonably accurate.

The effective use of Richardson extrapolation requires a series of refined meshes with co-located nodes at the coarser mesh nodes. For non-uniform meshes, the appropriate approach to refine the mesh is dependent on the numerical scheme that is being used. With some numerical schemes, a coarse to fine mesh approach is appropriate, whereas in other cases a fine to coarse approach is needed. Most problems require a non-uniform mesh to obtain sufficiently accurate results with a reasonable number of mesh cells. The order of numerical schemes can be adversely impacted by non-uniform meshes, especially if the mesh cell size varies too rapidly. The procedure to refine a mesh with a constant ratio of adjacent cell sizes κ is described, and analytical results are presented

on the variation of κ as the mesh is refined. Meshes with algebraic and exponential stretching are defined and illustrated. It is important to understand that non-uniform meshes become more uniform as the mesh is refined.

The truncation error for a steady-state, one-dimensional problem with a non-uniform mesh with four forms of the governing equations and numerical schemes has been investigated. The form of the governing equation and the resulting local truncation errors in terms of the cell size h are:

- Physical coordinates: $\tau = C_1(\kappa - 1)h + O(h^2)$
- Transformed coordinate approach: $\tau = C_2\Delta\xi^2 + O(\Delta\xi^4)$
- Box scheme: $\tau_Q = \bar{C}_2(\kappa h)^2 + O[(\kappa h)^4] \qquad \tau_F = \tilde{C}_2(\kappa h)^2 + O[(\kappa h)^4]$
- Control volume method: $\tau = \bar{C}_1(\kappa - 1)h + O(h^2)$

The physical coordinate and control volume methods are first order on non-uniform meshes, whereas the transformed coordinate approach and box scheme are second order. Although two of these methods are formally first-order on non-uniform meshes, Ferziger and Peric concluded that numerical schemes are second-order asymptotically as the mesh is refined. This statement is clearly appropriate when the ratio of adjacent cell sizes $\kappa = 1 + h$ or the mesh stretching is algebraic with κ near one.

The Richardson extrapolation procedure for steady-state problems is described for multi-mesh refinement, for the de Vahl Davis approach, which determines the actual order of the numerical scheme in the code, and for multi-dimensional problems. Four examples are used to illustrate the Richardson extrapolation procedure. The examples are turbulent boundary layer flow, natural convective flow in a cavity, hypersonic flow over a blunt body, and transonic wing flow. The Richardson extrapolation procedure for transient problems is presented. The impact of both temporal and spatial mesh refinement on solution error is determined. An unsteady problem with an exact solution to the differential equation and an exact solution to the difference equation is used to illustrate the Richardson extrapolation procedure. Richardson extrapolation method is only applicable when the mesh is sufficiently refined such that higher order terms in the discretization error expansion can be neglected. The required solutions can be very computationally expensive.

There are several items that have not been investigated and need further study:

1. A capacity to obtain the local truncation error for new numerical schemes is required. A symbolic manipulation code, such as Mathematica, is needed to properly analyze the local truncation error for 2D and 3D difference schemes.
2. The influence of non-uniform meshes on discretization error or solution accuracy of numerical schemes of current interest need to be evaluated.
3. Investigate and develop a technique to determine the consistency of difference schemes with a numerical method rather than using the usual analytical evaluation of the truncation error.
4. Determine how mesh refinement studies can be quantitatively used in the evaluation of the solution properties of the governing equations. Are the governing equations well conditioned with a unique solution?

1. Introduction

1.1 Code & Calculation Verification and Equation & Model Validation

Code verification is concerned with establishing that a code is accurately solving the governing equations that are being used to simulate a physical problem. There are many steps to ensure that the numerical solution technique is appropriate (stable and consistent), and the coding has been performed correctly; however, this work is concerned with the development of techniques to obtain the solution accuracy from mesh refinement studies. It is assumed that the initial steps of code verification have been completed before the code is used in the present solution accuracy studies. Also, it is assumed that the numerical scheme is consistent and will provide an accurate solution of the governing equations as the mesh is refined. The present investigation is concerned with local truncation error which has a direct impact on determining if a numerical scheme is consistent. The discretization error of the numerical solution is the issue being addressed in this work and is determined from mesh refinement studies. The present approach is to obtain the solution on three or more meshes and then use Richardson extrapolation [1], [2], [3] to obtain a sufficiently accurate result that can be used to judge the accuracy or error of the numerical solutions. *The application of Richardson extrapolation requires that the order of the numerical scheme is known.* The order of the numerical scheme can be determined from mathematical analysis of the numerical method and its local truncation error or can be determined as part of the grid refinement studies. If the order of the numerical scheme is known, the solution error can be determined more effectively with fewer mesh solutions required. This is the value in knowing the behavior of the local truncation error. In addition, the behavior of the local truncation error as the mesh cell size goes to zero determines if the numerical solution approaches the solution of the partial differential equation or, in other words, determines if the *numerical scheme is consistent.*

Calculation verification is concerned with determining the discretization error or solution accuracy with a verified code to ensure that the solution has the desired accuracy. The numerical accuracy of the solution from a simulation model must be determined as the first step after confidence has been established that the numerical solution technique is appropriate and the coding in the computer program is correct. This effort must be performed for each problem that is of a different class. When one has established the behavior and solution accuracy with a code for one type of problem, then additional mesh refinement studies on similar problems are probably unnecessary.

Governing Equation validation is needed to demonstrate that a well-posed mathematical problem has been formulated, and a unique solution exists.

Code model validation is required to establish the accuracy of the model used to simulate the physical problem being investigated. Verified solutions from a verified code are required for the evaluation of the models used in the code; verification must precede validation. The validation issue is not considered in this report.

Mesh refinement studies have significant value in the verification process for the following reasons:

1. With numerical solutions obtained on several meshes, Richardson extrapolation can be used effectively to obtain an estimate of the local exact solution of the problem and then an estimate of the local solution error can be determined.
2. The Richardson extrapolation procedure can also be used to obtain a more accurate solution than available from the finest mesh solution. The usual Richardson extrapolation method provides more accurate results only at the coarse or medium mesh point locations. The improved results can be obtained at the fine mesh points with the **Complete Richardson Extrapolation** method developed by Roache and Knupp [4]. This improved accuracy is obtained with little additional cost.
3. The numerical properties of the difference scheme can be evaluated to determine if the solution is consistent with the partial differential equation. The classic example of an inconsistent numerical scheme is the Du Fort and Frankel (1953) method, in reference [5], of solving the unsteady diffusion equation. If the ratio of the time step to the spatial step-size is held constant as the mesh is refined, the difference equation is inconsistent with the original partial differential equation. A more recent example has been pointed out by Turkel [6], [7] where the original difference scheme developed in reference [8] becomes inconsistent for non-uniform meshes. Turkel then suggested a modified numerical scheme that is consistent and more accurate on non-uniform meshes. Consistency of a numerical scheme has always been evaluated from truncation error analysis. A numerical approach to determine the consistency of a numerical scheme has not been used or developed.
4. The mathematical properties of the governing equations can be investigated to see if the formulation is well conditioned. Numerical solutions on coarse meshes can have significant dissipation and appear to be well behaved, but as the mesh is refined the solutions can become unstable or not approach a unique solution. Two examples of this behavior are found in the Parabolized Navier-Stokes (PNS) model [9] and the Baldwin-Barth turbulence model [10], [11]. The PNS model was used by Lubard and Helliwell (1978) to calculate viscous flow over a sharp cone and the solution procedure becomes unstable as the mesh is refined in the marching direction. A more rigorous analysis of the governing equation by Vigneron, Rakich and Tannehill (1978) determined the appropriate formulation of the flow direction pressure gradient terms in the momentum equations. This model is stable and converges to a unique solution as the mesh is refined. According to Menter [10], Rogers (1994) could not obtain grid independent solutions for airfoil flows with the Baldwin-Barth turbulence model using an incompressible Navier-Stokes code. With the Baldwin-Barth model, the properties of the solution at the interface between the turbulent and laminar flow regions appears to cause difficulties. There is no solution to the model equations at the shear layer edge interface. The problem was only identified by testing the model with a simple one-dimensional analysis tool, rather than in a full Navier-Stokes code, as very fine mesh solutions where necessary.

This report is concerned with developing a capability to evaluate local truncation errors of finite difference and finite volume schemes analytically. In addition, the procedures for mesh refinement studies with Richardson extrapolation to evaluate numerical solution error is documented and investigated. In Chapter 2, the reason that Richardson extrapolation is used to estimate the exact solution is discussed. In Chapter 3, the discretization error and local truncation error are defined and how they are related is presented. Also local and global errors are defined. Three simple problems are presented to illustrate the issues discussed in this chapter. These examples are for steady-state solutions with a uniform mesh. In Chapter 4, the stretching used in non-uniform meshes is considered and this information is needed for the next chapter. Chapter 5 is concerned with determining the local truncation error on non-uniform meshes with four numerical schemes. Chapter 6 presents the Richardson extrapolation procedure for multi-mesh, for the de Vahl Davis extrapolation approach, and for multi-dimensional problems. Four benchmark numerical problems are used to illustrate these procedures for determining solution accuracy. Chapter 7 presents how to use Richardson extrapolation to determine the solution accuracy of transient problems. A simple problem with exact solutions to the governing equation and difference equation is used to illustrate how the solution accuracy can be determined from the numerical results. In Chapter 8, a summary of this study is given. This report covers the initial part of developing the desired capability of determining numerical solution accuracy.

1.2 Review of Previous Work

There have been many studies concerned with accuracy of numerical schemes and the evaluation of numerical accuracy of simulations. The literature reviewed is related to computational fluid dynamics. A book by Roache [12] provides a complete review of the verification and validation procedure for computational science and engineering codes and became available as this work was being completed.

1.2.1 Uniform Cartesian Mesh

Initially, Richardson extrapolation was used to obtain more accurate numerical results from several coarse grid solutions of boundary layer flows. This procedure was used by Cooke and Mangler [13] with a Crank-Nicolson numerical scheme with a uniform mesh. The motivation for using Richardson extrapolation was the need to improve solution accuracy with a limited number of mesh points due to insufficient computer capabilities available at that time. The accuracy of the numerical solution of laminar boundary layer flows was estimated by Blottner [14] with the use of Richardson extrapolation. The flows were obtained on several meshes and then Richardson extrapolation was used to obtain a more accurate solution where the first term in the error expansion is retained. In this study, the extrapolated solution is assumed to be sufficiently close to the exact solution so that it can be used to judge the error of the numerical solutions. The natural convective flow in a square cavity has been solved by de Vahl Davis [15] on several meshes. The flow is governed by the Navier-Stokes equations and a central difference scheme was used. Richardson extrapolation was also used to obtain more accurate benchmark results and the error of the results have also been estimated.

1.2.2 Non-Uniform Cartesian Mesh

Keller and Cebeci used the box scheme with a non-uniform mesh for solving laminar [16] and then turbulent boundary layer flow [17]. In this work, Richardson extrapolation is used to improve solution accuracy with results obtained on a coarse mesh that was refined several times. The natural convective flow in a square cavity has also been solved by Hortmann, Peric and Scheuerer [18]. They have obtained more accurate results than previously obtained by de Vahl Davis. A cell centered control volume approach is used with both uniform and non-uniform meshes. The difference scheme is second order for the convective terms but only formally first-order for the diffusion terms. The mesh refinement studies indicated the scheme has second-order behavior. Turbulent flow over a backward facing-step problem with a non-uniform mesh has been investigated by Celik and Zhang [19] and Celik and Karatekin [20]. A commercial code was used and the method did not give monotonic spatial convergence for all of the variables in all regions of the solution domain. Richardson extrapolation could not be used for these cases. Also, mesh refinement studies discovered inappropriate application of turbulent boundary conditions at the wall. In the second paper, Richardson extrapolation method is modified to handle oscillatory spatial convergence.

1.2.3 Generalized Curvilinear Coordinates

There have been several Navier-Stokes codes developed with the governing equations written in curvilinear coordinates. With this approach, the physical space is transformed into the computational space where a uniform mesh is used. Therefore, the method is formally second-order when central differences are used. The truncation error for this approach has been addressed in Chapter V of the book by Thompson, Warsi, and Mastin [21]. They consider the order of these schemes as the number of points in the field increases with a fixed relative distribution of points, as the metric evaluation method changes, and as the non-uniform mesh spacing is varied to reduce solution error. A code with curvilinear coordinates has been used by Blottner [22] to solve the axisymmetric hypersonic flow over a blunt body where mesh refinement studies were performed. Properties of the numerical solution were evaluated with Richardson extrapolation method and solution errors were predicted. In this work, the mesh was refined with several approaches. For this two-dimensional problem, the mesh spacing in one coordinate direction was held fixed while the mesh spacing in the other direction was decreased by a factor of two. Then, this procedure was reversed for the other coordinate direction. In addition, the mesh spacing in both coordinate directions were simultaneously decreased by a factor of two. The transonic flow over two wings has been solved by Bonhaus and Wornom [23] with two compressible Navier-Stokes codes. This is a large three-dimensional problem which required significant computational effort and provides realistic simulation of flow conditions over an airplane wing. Four meshes were used for each wing solution but the mesh refinement approach resulted in insufficient information for the correct application of the Richardson extrapolation method. Also, it is not clear if both codes will produce the same numerical results as the mesh cell size approaches zero; however, the results from the two codes on the finest mesh are within a few percent. Zingg [24], [25] has performed a detailed investigation into the effects of grid clustering and refinement on the prediction of lift and drag of viscous airfoil flows obtained with a Navier-Stokes code. Richardson extrapolation is used to

obtain more accurate results and is used to estimate truncation error. The results provide useful guidelines for determining the levels of grid refinement and the outer boundary position required to achieve a given level of accuracy for a range of flow conditions. Hall and Zingg [26] use the solution accuracy predictions to develop a solution adaptive grid approach. Formulas for the local truncation error of convective terms in a curvilinear coordinate system have been obtained by Lee and Tsuei [27]. The truncation errors are a function of grid size, grid size ratio, angle between the grid lines, angle of velocity vector relative to the grid line, grid uniformity, and grid skewness.

1.2.4 Non-Uniform and Unstructured Meshes

For compressible flow problems, the aeronautics industry mainly uses finite volume methods for solving the governing equations numerically. The governing difference equation for a control volume cell is independent of whether the meshes are structured or unstructured. The origins of many of the present finite volume numerical techniques are related to two papers presented at the AIAA Fifth Computational Fluid Dynamics Conference in 1981. Ni [28] developed a 2D numerical scheme for quadrilateral cells where the dependent variables are located at the cell vertices. Jameson, Schmidt, and Turkel [8] developed a 2D numerical scheme for quadrilateral cells where the dependent variables are located at the cell centers. The accuracy of the Jameson, Schmidt, and Turkel scheme with quadrilateral cells was first investigated by Turkel [6] in 1985 and the numerical scheme was discovered to be inconsistent for non-uniform meshes. In addition, it is shown for many schemes that the accuracy of the space discretization for time dependent problems reduces to first order. Further work by Turkel, Yaniv, and Landau [7] have shown that a numerical scheme can be developed that has improved accuracy and at least first-order accuracy for general meshes. The accuracy of cell-vertex schemes with quadrilateral cells has been investigated by Roe [29]. A theoretical analysis of the local truncation error of these schemes has been performed. The results show that this approach obtained second-order accuracy on smooth meshes but the same accuracy cannot be obtained on arbitrary meshes. However, the cell-vertex scheme is more tolerant of non-uniform meshes than the cell centered scheme. Accuracy of the finite volume approach with triangular cells has been investigated by Giles [30] and Lindquist and Giles [31]. These authors concluded that triangular schemes perform as well as quadrilateral schemes, numerical dissipation influences the order of accuracy, and the solution error is second-order even though the local truncation error is first order. Morton and Paisley [32] investigated the cell centered and cell vertex finite volume formulations and determined that the cell vertex scheme is preferred because it is more accurate and has fewer spurious solution modes. Truncation error analysis for the finite volume method with the cell centered formulation for quadrilateral cells has been performed by Jeng and Chen [33]. They have considered a model steady convective equation with special attention given to the accuracy of the numerical approximation of the diffusion term on non-uniform meshes. The phase error of various finite difference schemes on regular triangular meshes has been determined by Zingg and Lomax [34]. An accuracy analysis of the finite volume method for triangular cells with the incompressible Navier-Stokes equations has been performed by Hwang [35]. This work showed that a numerical scheme that he developed and published in 1995 is a conditionally consistent formulation which depends on the properties of the non-uniform mesh being used.

2. Determination of Solution Error

One needs the exact solution to the problem to determine the solution discretization error, but the exact solution is not known except for a limited number of special cases. How accurate must the exact solution be known to obtain reasonable estimates of the numerical solution error? Insight into this issue will be addressed later.

First, two approaches are considered to estimate the exact solution of problems that can be solved only with numerical schemes. These two approaches are

1. Use a very fine mesh solution to determine accuracy of the more coarse mesh solutions.

2. Obtain solutions on several meshes and use Richardson extrapolation to obtain a sufficiently accurate result so that it can be used to judge the *accuracy of all of the numerical solutions.*

These two approaches are illustrated in Figure 2.1 with data that will be described later in Section 6.3.1 and is given in Table 6.2 for six meshes. Each of the numerical solutions in Figure 2.1 is obtained on uniform meshes with different mesh refinements. In each mesh refinement, the cell size is reduced by a factor of 2 in each coordinate direction. The curve labeled "Bench Mark" uses two very fine mesh solutions (i.e., mesh 5 = 320 x 320 and mesh 6 = 640 x 640) on a non-uniform mesh and then uses Richardson extrapolation to obtain a very accurate estimate of the exact solution of the problem. This solution is used to determine the error of the four numerical solutions given in the figure and is considered to be an accurate prediction of the solution error. The curve labeled "Fine Mesh" uses the solution with mesh 5 as an estimate of the exact solution to determine the solution errors with the different meshes. This approach gives some error on the finer mesh solutions (i.e., meshes 3 and 4). Also, the error of the finest mesh (i.e., mesh 5) solution can not be estimated with this approach. The curve labeled "Richardson" uses this method to extrapolate the solution from the solutions with the two finest meshes (i.e., meshes 3 and 4) shown in the figure. With this approach, the error of all of the numerical solutions (i.e., meshes 1 to 4) can be estimated and the estimated solution error is close to the Bench Mark exact value. The use of a further refined mesh, as in the Fine Mesh curve, to estimate the solution error of the coarser mesh solutions will work but a more expensive fine mesh solution that requires a large amount of computer storage and time is necessary. In addition, the use of a refined mesh solution to evaluate the order of the numerical scheme requires a solution on a mesh where the cell size has been refined by a factor of perhaps four times smaller. Overall, the Richardson extrapolation approach is the more appropriate method to obtain an estimate of the exact solution as only the solutions on mesh 3 and 4 are required to obtain an accurate prediction of the solution error.

In addition, a *one mesh approach* can be used with the adaptive mesh technique to estimate local solution error. This procedure obtains the solution on the mesh with two numerical schemes with different and known orders of accuracy. The higher-ordered scheme solution is used as the exact solution to estimate the solution error of the lower-ordered scheme. If a code with a higher-ordered scheme is available, then this scheme would most likely be the appropriate numerical method to use to obtained the desired solutions. The one mesh approach does not provide a technique to

determine the solution error of the higher-ordered scheme. Again, Richardson extrapolation approach is the most appropriate method to obtain an estimate of the exact solution with the higher-ordered scheme and to obtain an accurate prediction of the solution error.

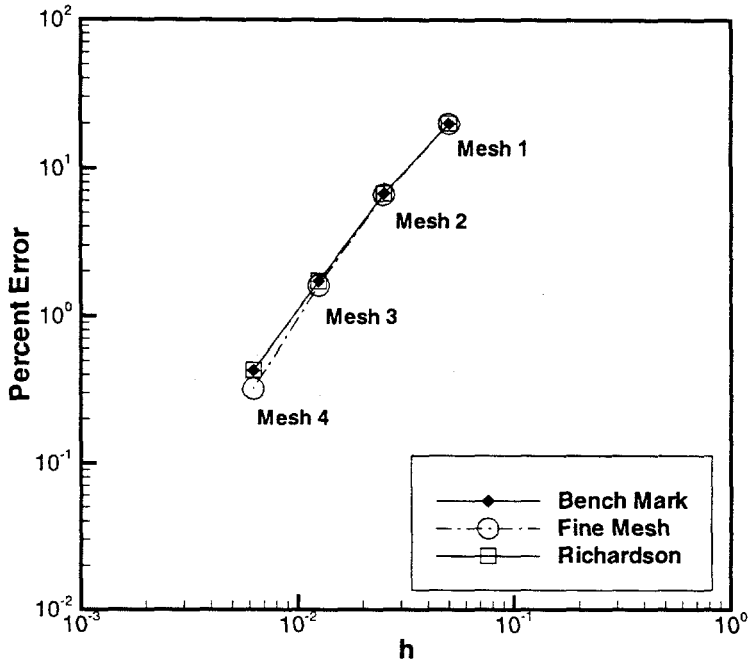


Figure 2.1 Two Approaches to Estimate Discretization Solution Error

3. Discretization Error for Richardson Extrapolation

3.1 Discretization Error, Local Truncation Error, and Consistency

The Richardson extrapolation procedure assumes that the exact solution f_i^n of the difference equation is related to the exact solution $F(t, x)$ of the differential equation at time $t^n = n\Delta t$ and location $x_i = i\Delta x$ with an expression of the form (for one-dimensional case)

$$f_i^n - F(t^n, x_i) = e_i^n = \alpha_i(\Delta x)^p + \ldots + A_i(\Delta t)^P + \ldots = \textbf{Discretization error} \qquad (1)$$

Morton [36] calls the above relation the **global error**. A *stable and consistent numerical scheme* is assumed in the development of Eq. (1). The justification for this type of expression for the discretization error is illustrated by considering the following partial differential equation (PDE):

$$L(F) = \frac{\partial F}{\partial t} - \frac{\partial^2 F}{\partial x^2} - S(x) = 0 \tag{2}$$

The numerical solution of the PDE is approximated with an explicit difference approach with a uniform mesh spacing which gives the **difference equation**

$$\Delta(f_i^n) = \left(\frac{f^{n+1} - f^n}{\Delta t}\right)_i - \left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right)^n - S_i^n = 0 \tag{3}$$

The amount by which the exact solution $F(t, x)$ of the partial differential equation does not satisfy the difference equation at the point (t^n, x_i) is called the **local truncation error** τ_i^n. For the present explicit difference scheme, the local truncation error becomes

$$\tau_i^n = \Delta(F_i^n) = \left(\frac{F^{n+1} - F^n}{\Delta t}\right)_i - \left(\frac{F_{i+1} - 2F_i + F_{i-1}}{\Delta x^2}\right)^n - S_i^n \tag{4}$$

The truncation error of the individual derivatives in the PDE is initially evaluated with the exact solution determined from the following Taylor's series:

$$F_i^{n+1} = F_i^n + \left(\frac{\partial F}{\partial t}\right)_i^n \Delta t + \frac{1}{2}\left(\frac{\partial^2 F}{\partial t^2}\right)_i^n \Delta t^2 + \ldots$$

$$F_{i+1}^n = F_i^n + \left(\frac{\partial F}{\partial x}\right)_i^n \Delta x + \frac{1}{2}\left(\frac{\partial^2 F}{\partial x^2}\right)_i^n \Delta x^2 + \frac{1}{6}\left(\frac{\partial^3 F}{\partial x^3}\right)_i^n \Delta x^3 + \frac{1}{24}\left(\frac{\partial^4 F}{\partial x^4}\right)_i^n \Delta x^4 + \ldots$$

$$F_{i-1}^n = F_i^n - \left(\frac{\partial F}{\partial x}\right)_i^n \Delta x + \frac{1}{2}\left(\frac{\partial^2 F}{\partial x^2}\right)_i^n \Delta x^2 - \frac{1}{6}\left(\frac{\partial^3 F}{\partial x^3}\right)_i^n \Delta x^3 + \frac{1}{24}\left(\frac{\partial^4 F}{\partial x^4}\right)_i^n \Delta x^4 - \ldots$$

The difference relation and difference relation expansion of the time derivative are

$$\left(\frac{\partial F}{\partial t}\right)_i^n \approx \left(\frac{f^{n+1} - f^n}{\Delta t}\right)_i \qquad \Delta_0 = \left(\frac{F^{n+1} - F^n}{\Delta t}\right)_i = \left(\frac{\partial F}{\partial t}\right)_i^n + \tau_0 \qquad \tau_0 = \frac{1}{2}\left(\frac{\partial^2 F}{\partial t^2}\right)_i^n \Delta t + \ldots$$

The local truncation error resulting from the difference approximation to the time derivative is τ_0. The difference relation and difference relation expansion of the second derivative are

$$\left(\frac{\partial^2 F}{\partial x^2}\right)_i^n \approx \left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right)^n$$

$$\Delta_2 = \left(\frac{F_{i+1} - 2F_i + F_{i-1}}{\Delta x^2}\right)^n = \left(\frac{\partial^2 F}{\partial x^2}\right)_i^n + \tau_2 \qquad \tau_2 = \frac{1}{12}\left(\frac{\partial^4 F}{\partial x^4}\right)_i^n \Delta x^2 + \ldots$$

The local truncation error resulting from the difference approximation to the spatial derivative is τ_2. The local truncation error of the difference equation for the partial differential Eq. (2) becomes:

$$L(F_i^n) = \Delta(F_i^n) - \tau_i^n = 0 \qquad \tau_i^n = \Delta_0 - \Delta_2 - S(x) = \left(\frac{\partial F}{\partial t}\right)_i^n - \left(\frac{\partial^2 F}{\partial x^2}\right)_i^n - S(x) + \tau_0 - \tau_2 \tag{5}$$

$$\tau_i^n = \tau_0 - \tau_2 = \frac{1}{2}(F_{tt})_i^n \Delta t - \frac{1}{12}(F_{xxxx})_i^n \Delta x^2 + \ldots$$

In the above equation, it is assumed that the source term is evaluated exactly. At least for simple differential equations and difference equations, the local truncation error can be readily determined. The difference Eq. (3) has a local truncation error $O(\Delta t) + O(\Delta x^2)$ and this is referred to as a first-order accurate method in time and second-order accurate method in the spatial coordinate. The present **numerical scheme is consistent** since the local truncation error $\tau_i^n \to 0$ as $\Delta t \to 0$ and $\Delta x \to 0$. In addition, the difference equation approaches the PDE as can be seen from Eq. (5).

3.2 Relation Between Discretization and Local Truncation Errors

For Richardson extrapolation, the discretization error must be evaluated. For a transient solution, the approach is addressed in Chapter 7. The present analysis is concerned with the steady-state solution of (2). For this case, the local truncation error of the PDE is obtained from Eq. (4) which gives

$$\tau_i = \Delta(F_i) = \left(\frac{F_{i+1} - 2F_i + F_{i-1}}{\Delta x^2}\right) + S_i \qquad \tau_i = \frac{1}{12} F_{xxxx} \Delta x^2 + \ldots \tag{6}$$

The exact solution of the difference equation is obtained from Eq. (3) which gives

$$\Delta(f_i) = \left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right) + S_i = 0 \tag{7}$$

The discretization error equation is obtained by subtracting Eq. (6) from Eq. (7)

$$\left(\frac{e_{i+1} - 2e_i + e_{i-1}}{\Delta x^2}\right) + \tau_i = 0 \tag{8}$$

If it is assumed that the boundary conditions are at $x = \pm H$ or $i = \pm I$ with the discretization error zero at these locations, the solution to Eq. (8) with *constant truncation error* $\tau_i = \tau$ is (see Boole [37] and Hildebrand [38], [39] for information on the analytical solution of difference equations)

$$e_i = A + Bi + Ci^2 \qquad A = -CI^2 \qquad B = 0 \qquad C = -\frac{1}{2}\Delta x^2 \tau \tag{9}$$

The results for A and B are obtained from the boundary conditions while the result for C is required to satisfy the difference equation. The above equation is rewritten as

$$e_i = \frac{1}{2}H^2\left[1 - \left(\frac{x_i}{H}\right)^2\right]\tau_i \qquad x_i = i\Delta x \qquad H = I\Delta x \tag{10}$$

It is important to notice that the **discretization error is proportional to the local truncation error**. Using the truncation error as given in Eq. (6), the discretization error becomes

$$e_i = \alpha_i \Delta x^2 + \ldots \qquad \alpha_i = \frac{1}{24}H^2\left[1 - \left(\frac{x_i}{H}\right)^2\right]\left(\frac{\partial^4 F}{\partial x^4}\right)_i \tag{11}$$

For linear difference problems, the discretization difference equation in general becomes

$$\Delta(e_i) = \Delta(f_i) - \Delta(F_i) = -\tau_i$$

The difference equation for e_i is the same as the numerical difference equation for f_i except the local truncation error is added as a source term on the right hand side. For nonlinear difference problems, numerical results indicate that the local truncation error of the numerical scheme determines the behavior of the discretization error.

3.3 Local and Global Errors

It is usually desirable to write the solution error as a **local percent error**. The local percent error of the numerical solution relative to the *local exact solution* is defined as follows:

$$\%\, Ee_i = 100(f_i - F_i)/F_i = 100e_i/F_i = 100[\alpha_i/F_i]\Delta x^2 + \ldots \tag{12}$$

The local percent error of the numerical solution relative to some *representative solution* value F_r is defined as follows:

$$\%\, Er_i = 100(f_i - F_i)/F_r = 100e_i/F_r = 100[\alpha_i/F_r]\Delta x^2 + \ldots \tag{13}$$

In order to measure the **global solution percent error**, the two errors above can be summed over all of the mesh points i using an L_2 norm or least square as follows:

$$Ee = \sqrt{\sum_i (\%\, Ee_i)^2 / I_T} = 100(\Delta x)^2 \sqrt{\sum_i (\alpha_i / F_i)^2 / I_T} + \ldots$$
$$Er = \sqrt{\sum_i (\%\, Er_i)^2 / I_T} = \frac{100}{F_r}(\Delta x)^2 \sqrt{\sum_i (\alpha_i)^2 / I_T} + \ldots \qquad (14)$$

The total number of mesh points used in the solution is I_T.

3.4 Illustration of Steady-State Errors with a Uniform Mesh

The following examples show how the above errors can be determined for some simple one-dimensional problems.

3.4.1 Example 1: Numerical Solution That Is Exact

The foregoing analysis is evaluated for the steady-state form of the partial differential equation with the exact solution and boundary conditions indicated

$$\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + 2\frac{F_0}{H^2} = 0 \qquad F = F_0\left[1 - \left(\frac{x}{H}\right)^2\right] \qquad F = 0 \text{ at } x = \pm H$$

The difference equation and boundary conditions become

$$\left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right) + 2\frac{F_0}{H^2} = 0 \qquad f_I = f_{-I} = 0$$

The solution of the difference equation is of the form

$$f_i = A + Bi + Ci^2 \qquad A = -CI^2 \qquad B = 0 \qquad C = -F_0\left(\frac{\Delta x}{H}\right)^2$$

and the difference solution becomes

$$f_i = F_0\left[1 - \left(\frac{x_i}{H}\right)^2\right]$$

The solution of the difference equation is the same as the exact solution for this problem. In addition, the discretization error and the local truncation error are zero for this problem. This is a excellent problem to determine if a numerical scheme is second-order on a uniform mesh.

3.4.2 Example 2: Constant Truncation Error Case

The foregoing analysis is evaluated for the steady-state form of the partial differential equation with the exact solution and boundary conditions indicated

$$\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + 12\frac{F_0}{H^2}\left(\frac{x}{H}\right)^2 = 0 \qquad F = F_0\left[1-\left(\frac{x}{H}\right)^4\right] \qquad F = 0 \text{ at } x = \pm H$$

The difference equation and boundary conditions becomes

$$\left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right) + 12\frac{F_0}{H^2}\left(\frac{x}{H}\right)^2 = 0 \qquad f_I = f_{-I} = 0$$

The solution of the difference equation is of the form

$$f_i = A + Bi + Ci^2 + Di^3 + Ei^4 \qquad A = -CI^2 - EI^4 \qquad B = 0$$

$$C = -E \qquad D = 0 \qquad E = -F_0\left(\frac{\Delta x}{H}\right)^4$$

and the difference solution becomes

$$f_i = F_i + e_i \qquad F_i = F_0\left[1-\left(\frac{x_i}{H}\right)^4\right] \qquad e_i = -F_0\left[1-\left(\frac{x_i}{H}\right)^2\right]\left(\frac{\Delta x}{H}\right)^2 \tag{15}$$

The solution of the difference equation is the same as the exact solution plus a discretization error. In addition, the local truncation error from Eq. (5) is

$$\tau_i^n = -\frac{1}{12}\left(\frac{\partial^4 F}{\partial x^4}\right)_i^n \Delta x^2 + \ldots = 2\frac{F_0}{H^2}\left(\frac{\Delta x}{H}\right)^2$$

The above local truncation error is constant and approaches zero as $\Delta x \to 0$ which shows that the numerical scheme is consistent. The discretization error can be determined from Eq. (11) which gives

$$f_i - F_i = e_i = \alpha_i \Delta x^2 \qquad \alpha_i = -\frac{F_0}{H^2}\left[1-\left(\frac{x_i}{H}\right)^2\right]$$

As expected, this result is the same as obtained from the difference solution given in Eq. (15). The local percent error of the numerical solution is obtained from Eq. (12) and Eq. (13) with $F_r = F_0$ as follows:

$$\% Ee_i = -100(\Delta x/H)^2[1-(x_i/H)^2](F_0/F_i) = -100(\Delta x/H)^2/[1+(x_i/H)^2]$$

$$\% Er_i = -100(\Delta x/H)^2[1-(x_i/H)^2]$$

The *upper half of the solution* ($0 \le x_i \le H$) for this example is given in Table 3.1 for the case of step-size $\Delta x/H = 0.2$. The exact solution, discretization error, and percent errors are given at the mesh points in this table. The use of the exact solution to determine the error becomes indeterminate at $x_i = H$ and requires the use of the second form for an accurate evaluation. The global percent errors for this case are

$$Ee = 3.112\ \% \qquad Er = 2.785\ \%$$

and are a reasonable indication of the solution accuracy.

Table 3.1: Example 2 Results

i	x_i/H	F_i/F_0	e_i/F_0	$\% Ee_i$	$\% Er_i$
5	1.0	0.0000	0.0	–2.000	0.0
4	0.8	0.5904	-1.44×10^{-2}	–2.439	–1.44
3	0.6	0.8704	-2.56×10^{-2}	–2.941	–2.56
2	0.4	0.9744	-3.36×10^{-2}	–3.448	–3.36
1	0.2	0.9984	-3.84×10^{-2}	–3.846	–3.84
0	0.0	1.0000	-4.00×10^{-2}	–4.000	–4.00

3.4.3 Example 3: Variable Truncation Error Case

The foregoing analysis is evaluated for the steady-state form of the partial differential equation with the exact solution and boundary conditions indicated

$$\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} - 20\frac{x^3}{H^5} = 0 \qquad F = \left(\frac{x}{H}\right)^5 \qquad \begin{matrix} F = 1 \text{ at } x = H \\ F = -1 \text{ at } x = -H \end{matrix}$$

The difference equation becomes

$$(f_{i+1} - 2f_i + f_{i-1}) - 20\left(\frac{\Delta x}{H}\right)^5 i^3 = 0$$

The solution of the difference equation is of the form

$$f_i = A + Bi + Ci^2 + Di^3 + Ei^4 + Fi^5 \qquad A = 0 \qquad B = \frac{5}{3}(\Delta x/H)^2$$

$$C = 0 \qquad D = -\frac{5}{3}(\Delta x/H)^5 \qquad E = 0 \qquad F = (\Delta x/H)^5$$

and the difference solution becomes

$$f_i = (x_i/H)^5 + e_i \qquad e_i = -\frac{5}{3}(x_i/H)[(x_i/H)^2 - 1](\Delta x/H)^2$$

The local truncation error is obtained from Eq. (5) and varies with x

$$\tau_i^n = -\frac{1}{12}\left(\frac{\partial^4 F}{\partial x^4}\right)_i^n \Delta x^2 + \ldots = 10\frac{x_i}{H^3}\left(\frac{\Delta x}{H}\right)^2 + \ldots$$

The local truncation error goes to zero as $\Delta x \to 0$ and the numerical scheme is consistent. The discretization error difference equation is determined from Eq. (8) which gives

$$(e_{i+1} - 2e_i + e_{i-1}) + 10\left(\frac{\Delta x}{H}\right)^5 i = 0$$

The solution for the discretization error is

$$e_i = \alpha_i \Delta x^2 \qquad \alpha_i = -\frac{5}{3}\left(\frac{x_i}{H^3}\right)\left[\left(\frac{x_i}{H}\right)^2 - 1\right]$$

The local percent error of the numerical solution is obtained from Eq. (12) and Eq. (13) with $F_r = 1.0$

$$\%\, Ee_i = -100\left(\frac{5}{3}\right)(\Delta x/H)^2 [(x_i/H)^2 - 1]/(x_i/H)^4$$

$$\%\, Er_i = -100\left(\frac{5}{3}\right)(\Delta x/H)^2 [(x_i/H)^2 - 1](x_i/H)$$

The *upper half of the solution* for this example is given in Table 3.2 for the case of step-size $\Delta x/H = 0.2$. The exact solution, discretization error, and percent errors are given at the mesh points in this table. The global percent errors for this case are

$$Ee = \text{Infinity} \qquad Er = 1.681\ \%$$

The use of the exact solution to obtain the local percent error for this case is inappropriate. The exact solution goes to zero at $x = 0$ with the local percent error going to infinity. The previous example and this example illustrate that when the exact solution is zero or near zero, the solution error $\% Ee_i$ can be very inaccurate or very large. The use of $\% Er_i$ with a reference value of the solution (average solution value or maximum solution value) may be more appropriate for this situation.

Table 3.2: Example 3 Results

i	x_i/H	F_i	e_i	$\% Ee_i$	$\% Er_i$
5	1.0	1.0000	0.0	0.0	0.0
4	0.8	3.2768×10^{-1}	1.92×10^{-2}	-5.86	-1.92
3	0.6	7.776×10^{-2}	2.56×10^{-2}	-32.92	-2.56
2	0.4	1.024×10^{-2}	2.24×10^{-2}	-218.75	-2.24
1	0.2	3.200×10^{-4}	1.28×10^{-2}	-4000.0	-1.28
0	0.0	0.0	0.0	Infinity	0.0

4. Mesh Stretching

Since numerical solution error is a function of the numerical scheme and the mesh employed, this section is included to give some background information on non-uniform meshes with different cell size variation. For non-uniform meshes, the local truncation error is a function of the cell size variation as will be illustrated in Section 5. In many cases, the local truncation error is proportional to $(\kappa_i - 1)$ where κ_i is defined in Eq. (17).

A variety of methods have been used to vary the mesh spacing within the solution region. A commonly used method is to use a constant ratio of the mesh interval along a coordinate direction. The mesh increment or interval along the x coordinate is related to the mesh coordinates as follows:

$$\begin{aligned} &x_i \text{ is specified at } \quad i = 0, 1, 2, \ldots, I \\ &h_i = x_i - x_{i-1} \qquad i = 1, 2, \ldots, I \end{aligned} \tag{16}$$

The *spatial mesh ratio* is defined as ratio of adjacent mesh increments which is expressed as

$$\kappa_i = h_{i+1}/h_i \qquad i = 1, 2, \ldots, I-1 \tag{17}$$

One approach is to keep the spatial mesh ratio constant while other approaches specify some variation of this ratio.

4.1 Constant Spatial Mesh Stretching

When $\kappa_i = \kappa$, the mesh coordinates are determined from Eq. (16) and Eq. (17) which gives the difference relation

$$x_{i+1} - (1+\kappa)x_i + \kappa x_{i-1} = 0$$

This mesh is related to the geometric progression series. The solution of this equation is of the form

$$x_i = C\lambda^i \qquad \lambda^2 - (1+\kappa)\lambda + \kappa = 0 \quad \text{and} \quad \lambda = \kappa \text{ or } 1$$

When the end points of the region and the spatial mesh ratio are specified, the location of the mesh nodes becomes

$$x_i = x_0 + (x_I - x_0)\left[\frac{\kappa^i - 1}{\kappa^I - 1}\right] \qquad i = 0, 1, 2, \ldots, I \tag{18}$$

When the first mesh increment and the spatial mesh ratio are specified, the location of the mesh nodes becomes

$$x_i = x_0 + h_1\left[\frac{\kappa^i - 1}{\kappa - 1}\right] \qquad i = 0, 1, 2, \ldots, I$$

When the end points of the region and the first mesh increment are specified, the spatial mesh ratio must first be determined from the following relation in an iterative manner:

$$\kappa = \left[1 + (\kappa - 1)\left(\frac{x_I - x_0}{h_1}\right)\right]^{(1/I)}$$

This mesh stretching approach can be readily used to determine a series of refined meshes, m, which overlap or have co-located nodes. The spatial mesh ratio for a refined mesh is obtained from

$$\kappa(m+1) = \sqrt{\kappa(m)} \qquad m = 0, 1, \ldots, M \tag{19}$$

where the number of mesh cell is doubled with each refinement.

4.2 Algebraic or Quasi-Uniform Mesh Stretching

In the paper of Turkel [6] on the accuracy of numerical schemes with non-uniform meshes, the definition of algebraic and exponential mesh spacings are introduced. These definitions are used in determining the order of the local truncation error of difference schemes on non-uniform meshes. An **algebraic mesh** is defined with a spatial mesh ratio such that

$$\kappa_i = h_{i+1}/h_i = 1 + O(h^p) \qquad p > 0 \qquad h = max(h_i) \qquad i = 1, 2, \ldots, I-1$$

The mesh spacing for this case is evaluated for $x_0 = 0$ and $x_I = 1$. The value of stretching parameter is specified as

$$\kappa_i = 1 + h_I^p = \kappa = \text{Constant}$$

With this assumption, the algebraic mesh becomes the same as the constant spatial mesh stretching case of Section 4.1. The algebraic mesh defines a method to specify the value of κ. The difference equation and solution for the mesh increment become

$$h_{i+1} - \kappa h_i = 0 \qquad h_i = \kappa^{i-1} h_1 = x_i - x_{i-1} \tag{20}$$

When the above difference equation is solved for the node locations, the result is given by Eq. (18). The sum of the mesh increments gives

$$\sum_{i=1}^{I} h_i = h_1(1 + \kappa + \kappa^2 + \kappa^3 + \ldots + \kappa^{I-1}) = 1$$

The sum of a geometric series is known and is used to obtain

$$h_1(\kappa^I - 1) = \kappa - 1 = h_I^p \tag{21}$$

For the case with p = 1 and $I = 10$, the first and last increments are obtained from Eq. (20) and Eq. (21)

$$h_I = 1/(1 + h_I)^{I-1} = 0.1974914 \qquad \kappa = 1 + h_I = 1.1974914$$

$$h_1 = (\kappa - 1)h_I = h_I^2 = 0.03900285 \qquad h_I/h_1 = 5.06$$

An iterative solution is required for the last increment h_I. The mesh increments and node locations for this algebraic mesh spacing is given in Table 4.1.

For the case with p = 2 and $I = 10$, the first and last increments are obtained from Eq. (20) and Eq. (21)

$$h_I = (\kappa^I - 1)/\kappa^{I-1} = 0.1050064 \qquad \kappa = 1 + h_I^2 = 1.011026$$

$$h_1 = h_I/\kappa^{I-1} = h_I^2/(\kappa^I - 1) = 0.0951379 \qquad h_I/h_1 = 1.104$$

An iterative solution is again required for the last increment h_I. The mesh increments and node locations for this algebraic mesh spacing is given in Table 4.1. As p becomes larger, the mesh becomes more uniform as illustrated in this table. The stretching with $p = 2$ is insufficient for aerodynamic flow problems.

Table 4.1: Algebraic Meshes

i	$p = 1$ $\kappa = 1.1974914$		$p = 2$ $\kappa = 1.011026$	
	h_i	x_i	h_i	x_i
0		0		0
1	0.0390	0.0390	0.0951	0.0951
2	0.0467	0.0857	0.0962	0.1913
3	0.0559	0.1416	0.0973	0.2886
4	0.0670	0.2086	0.0983	0.3869
5	0.0802	0.2888	0.0994	0.4863
6	0.0961	0.3849	0.1005	0.5868
7	0.1150	0.4999	0.1016	0.6884
8	0.1377	0.6376	0.1027	0.7911
9	0.1649	0.8025	0.1039	0.8950
10	0.1975	1.0000	0.1050	1.0000

4.3 Exponential Mesh Stretching

An **exponential mesh** is defined with a mesh spacing such that

$$\kappa_{i+1}/\kappa_i = 1 + O(h) \qquad i = 1, 2, \ldots, I-2$$

The mesh spacing for this case is evaluated for $x_0 = 0$ and $x_I = 1$. The value of the spatial mesh ratio parameter is determined from

$$\kappa_{i+1}/\kappa_i = 1 + h_I = d = \text{Constant}$$

The difference equation and solution for the spatial mesh ratio become

$$\kappa_{i+1} - d\kappa_i = 0 \qquad \kappa_i = Cd^i \qquad i = 1, 2, \ldots, I-1 \tag{22}$$

The difference equation and solution for the mesh increment become

$$h_{i+1} - Cd^i h_i = 0 \qquad h_i = C^{i-1} d^{s_i} h_1 \qquad h_I = C^{I-1} d^{s_I} h_1 \tag{23}$$

where $s_i = s_{i-1} + i - 1$ and $s_1 = 0$ $\qquad i = 2, 3, \ldots, I$

The sum of the mesh increments gives

$$\sum_{i=1}^{I} h_i = h_1(1 + Cd + C^2 d^3 + C^3 d^6 + C^4 d^{10} + \ldots + C^{I-1} d^{s_I}) = 1$$

The series is rewritten as a relation for the last increment which gives

$$h_I = 1/(1 + C^{-1} d^{(s_{I-1} - s_I)} + C^{-2} d^{(s_{I-2} - s_I)} + \ldots) \tag{24}$$

For the case with $I = 10$ and $C = 1$, the first and last increments are obtained from Eq. (23) and Eq. (24)

$$h_I = 1/(1 + d^{-9} + d^{-17} + d^{-24} + \ldots) = 0.9980255 \qquad d = 1 + h_I = 1.9980255$$

$$h_1 = h_I/(1 + h_I)^{s_I} = 2.965478 \times 10^{-14} \qquad s_I = 45 \qquad h_I/h_1 = 3.36 \times 10^{13}$$

An iterative solution is required for the last increment h_I. The mesh increments and node locations for this exponential mesh spacing is given in Table 4.2. This case appears to provide more stretching than is used in aerodynamic flow problems where usual mesh stretching requires $h_I/h_1 \approx 10^3$ to 10^6.

Table 4.2: Exponential Mesh

i	s_i	κ_i	h_i	x_i
0				0
1	0	1.998	2.965×10^{-14}	2.965×10^{-14}
2	1	3.992	5.924×10^{-14}	8.889×10^{-14}
3	3	7.976	2.365×10^{-13}	3.254×10^{-13}
4	6	15.937	1.886×10^{-12}	2.211×10^{-12}
5	10	31.842	3.006×10^{-11}	3.227×10^{-11}
6	15	63.622	9.573×10^{-10}	9.896×10^{-10}
7	21	127.12	6.090×10^{-8}	6.190×10^{-8}
8	28	253.99	7.742×10^{-6}	7.805×10^{-6}
9	36	507.49	1.966×10^{-3}	1.975×10^{-3}
10	45		9.980×10^{-1}	1.0000

5. Local Truncation Error with Non-Uniform Meshes

The local truncation error of the numerical difference scheme has been defined in Eq. (4). If the truncation error of each difference approximation used to replace derivatives is known, then the total truncation error of the numerical approximation can be evaluated. The difficulty of obtaining second-order numerical schemes on a non-uniform mesh is illustrated by considering first and second derivatives and a ordinary differential equation at the local node i. In the following sections, the notation is simplified with some of the variables at i being written without a subscript; for example, $\kappa = \kappa_i$.

5.1 Finite Difference Method in Physical Coordinates

Two forms that are used to approximate the first derivative are as follows:

$$\left(\frac{\partial F}{\partial x}\right)_i \approx \alpha\left(\frac{f_{i+1}-f_i}{\kappa h_i}\right)+(1-\alpha)\left(\frac{f_i-f_{i-1}}{h_i}\right) \qquad h_i = (x_i - x_{i-1}) \qquad h_{i+1} = \kappa h_i$$

$$\Delta_1 = \alpha\left(\frac{F_{i+1}-F_i}{\kappa h_i}\right)+(1-\alpha)\left(\frac{F_i-F_{i-1}}{h_i}\right) = \left(\frac{\partial F}{\partial x}\right)_i + \tau_1$$

$$\tau_1 = \frac{1}{2}[\alpha(1+\kappa)-1]F_{xx}h_i + \ldots$$

If $\alpha = \kappa/(1+\kappa)$, then the approximation appears to be first-order as shown below

$$\left(\frac{\partial F}{\partial x}\right)_i \approx \left(\frac{\kappa}{1+\kappa}\right)\left(\frac{f_{i+1}-f_i}{\kappa h_i}\right)+\left(\frac{1}{1+\kappa}\right)\left(\frac{f_i-f_{i-1}}{h_i}\right) = \frac{f_{i+1}-f_{i-1}}{x_{i+1}-x_{i-1}}$$

$$\Delta_1 = \frac{F_{i+1}-F_{i-1}}{x_{i+1}-x_{i-1}} = \left(\frac{\partial F}{\partial x}\right)_i + \tau_1 \tag{25}$$

$$\tau_1 = \frac{1}{2}F_{xx}(\kappa-1)h_i + \frac{1}{6}F_{xxx}(\kappa^2-\kappa+1)h_i^2 + O(h_i^3)$$

If $\alpha = 1/(1+\kappa)$, then the approximation is second-order as shown below

$$\left(\frac{\partial F}{\partial x}\right)_i \approx \left(\frac{1}{1+\kappa}\right)\left(\frac{f_{i+1}-f_i}{\kappa h_i}\right)+\left(\frac{\kappa}{1+\kappa}\right)\left(\frac{f_i-f_{i-1}}{h_i}\right)$$

$$\Delta_1 = \left(\frac{1}{1+\kappa}\right)\left(\frac{F_{i+1}-F_i}{\kappa h_i}\right)+\left(\frac{\kappa}{1+\kappa}\right)\left(\frac{F_i-F_{i-1}}{h_i}\right) = \left(\frac{\partial F}{\partial x}\right)_i + \tau_1 \qquad \tau_1 = \frac{1}{6}F_{xxx}\kappa h_i^2 + \ldots$$

If the mesh is uniform with $\kappa = 1$, then both approximations are the same and second-order. In Eq. (25), the first term in the truncation error goes to zero and the second term is of order h_i^2 for a uniform mesh.

The second derivative usually involves a parameter l which is the fluid viscosity or the thermal conductivity of the material. The second derivative is approximated as follows with the notation given in the figure on the next page:

$$\left[\frac{\partial}{\partial x}\left(l\frac{\partial F}{\partial x}\right)\right]_i \approx \frac{2}{(1+\kappa)h_i}\left[l_e\left(\frac{f_{i+1}-f_i}{\kappa h_i}\right)-l_d\left(\frac{f_i-f_{i-1}}{h_i}\right)\right] \qquad (26)$$

$$l_d = (l_i + l_{i-1})/2 \qquad l_e = (l_{i+1} + l_i)/2$$

The difference relation expansion and local truncation error for this difference expression is

$$\Delta_2 = \left[\frac{\partial}{\partial x}\left(l\frac{\partial F}{\partial x}\right)\right]_i + \tau_2 \qquad \tau_2 = \bar{a}_i(\kappa-1)h_i + \bar{b}_i(\kappa^2-\kappa+1)h_i^2 + \ldots$$

$$\bar{a}_i = \frac{1}{12}[lF_{xxx} + 3(lF_x)_{xx}]_i \qquad \bar{b}_i = \frac{1}{24}[(lF_{xxx})_x + (lF_x)_{xxx}]_i \qquad (27)$$

With this analysis, the second derivative is second-order only if the mesh is uniform (i.e., $\kappa = 1$) or the mesh varies such that $\kappa = 1 + O(h)$.

The local truncation error of this method is best illustrated by the second-order differential equation with the indicated boundary conditions

$$\frac{\partial}{\partial x}\left(l\frac{\partial F}{\partial x}\right) - \alpha F - \beta\frac{\partial F}{\partial x} = 0 \qquad F(0) = 0 \qquad F(x_e) = 1 \qquad 0 \le x \le x_e \qquad (28)$$

The difference equation for this differential equation is

$$\Delta(f_i) = \frac{2}{(1+\kappa)h_i}\left[l_e\left(\frac{f_{i+1}-f_i}{\kappa h_i}\right)-l_d\left(\frac{f_i-f_{i-1}}{h_i}\right)\right] - \alpha f_i - \beta\left[\frac{f_{i+1}-f_{i-1}}{(\kappa+1)h_i}\right] = 0$$

The local truncation error for this difference equation at x_i is

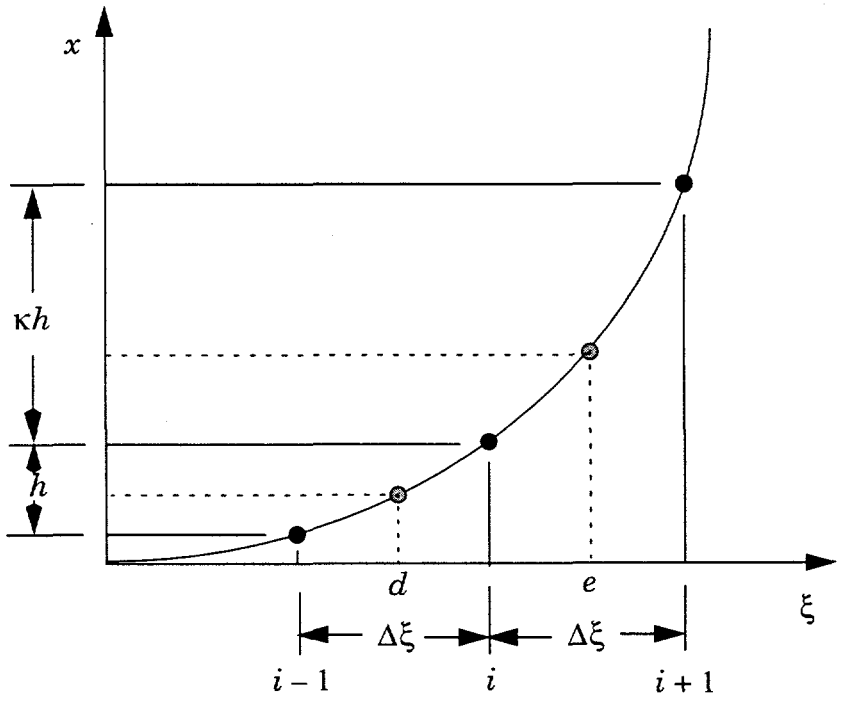
$$\Delta(F_i) = [(lF_x)_x - \alpha F - \beta F_x] + \tau$$

$$\tau = \left(\bar{a} - \frac{1}{2}\beta F_{xx}\right)(\kappa-1)h + \left(\bar{b} - \frac{1}{6}\beta F_{xxx}\right)(\kappa^2-\kappa+1)h^2 + \ldots \qquad (29)$$

The first term on the right hand side is the governing Eq. (28) and this term is zero. The leading term in the local truncation error is of order of the mesh size; therefore, the difference approximation to the governing Eq. (28) is first-order. The scheme is second-order only if the mesh is uniform (i.e., $\kappa = 1$) or the mesh varies such that $\kappa = 1 + O(h)$.

5.2 Transformed Coordinate Approach

A common approach to obtain a second-order numerical scheme is to use a uniform mesh but not in the physical coordinates x. This approach transforms the governing equations into a new coordinate system ξ where $x = x(\xi)$ and the mesh spacing $\Delta\xi$ is uniform. See references [21], [22], and [27] for more information on this subject. Also, it can be shown that the coordinate transformation approach can be equivalent to the finite volume approach. The relation between the two coordinates and the notation is illustrated below. The mesh points $d = i - 1/2$ and $e = i + 1/2$ are the mid-point locations for the ξ coordinate system.



The first derivative becomes

$$\left(\frac{\partial F}{\partial x}\right)_i = \left\{(x_\xi)^{-1}\frac{\partial F}{\partial \xi}\right\}_i \approx \left[\left(\frac{f_{i+1} - f_{i-1}}{2\Delta\xi}\right)\right](x_\xi)_i^{-1} = \Delta_1(f_i)$$

$$\Delta_1(F_i) = \left(\frac{F_\xi}{x_\xi}\right)_i + \tau_1 \qquad \tau_1 = \frac{1}{6}\left(\frac{F_{\xi\xi\xi}}{x_\xi}\right)_i \Delta\xi^2 + O(\Delta\xi^4)$$

The second derivative becomes

$$\left[\frac{\partial}{\partial x}\left(l\frac{\partial F}{\partial x}\right)\right]_i \equiv \left\{(x_\xi)^{-1}\frac{\partial}{\partial \xi}\left[l\left(\frac{\partial x}{\partial \xi}\right)^{-1}\frac{\partial F}{\partial \xi}\right]\right\}_i$$

$$\approx (x_\xi \Delta\xi)_i^{-1}\left[\left(\frac{l}{x_\xi}\right)_e\left(\frac{f_{i+1}-f_i}{\Delta\xi}\right)-\left(\frac{l}{x_\xi}\right)_d\left(\frac{f_i-f_{i-1}}{\Delta\xi}\right)\right] = \Delta_2(f_i)$$

$$\Delta_2(F_i) = \frac{1}{(x_\xi)_i}\left[\frac{1}{(x_\xi)_e}-\frac{1}{(x_\xi)_d}\right]\left\{\frac{lF_\xi}{\Delta\xi}+\left[\frac{1}{4}(lF_\xi)_\xi+\frac{1}{6}lF_{\xi\xi\xi}\right]\Delta\xi\right\}+\frac{1}{2(x_\xi)_i}\left[\frac{1}{(x_\xi)_e}+\frac{1}{(x_\xi)_d}\right](lF_\xi)_\xi$$

The coordinate derivatives are approximated as

$$(x_\xi)_i \approx \left(\frac{x_{i+1}-x_{i-1}}{2\Delta\xi}\right) \qquad (x_\xi)_e \approx \left(\frac{x_{i+1}-x_i}{\Delta\xi}\right) \qquad (x_\xi)_d \approx \left(\frac{x_i-x_{i-1}}{\Delta\xi}\right)$$

$$\frac{x_{i+1}-x_{i-1}}{2\Delta\xi} = (x_\xi)_i+\frac{1}{6}x_{\xi\xi\xi}\Delta\xi^2+\ldots \qquad \frac{x_{i+1}-x_i}{\Delta\xi} = (x_\xi)_i+\frac{1}{2}x_{\xi\xi}\Delta\xi+\frac{1}{6}x_{\xi\xi\xi}\Delta\xi^2+\ldots$$

The coordinate derivatives are used in the above to obtain the following difference relations for the derivatives, difference relation expansion, and the local truncation error:

$$\left(\frac{\partial F}{\partial x}\right)_i \approx \frac{f_{i+1}-f_{i-1}}{x_{i+1}-x_{i-1}} \qquad \Delta_1 = \left[\frac{F_\xi}{x_\xi}\right]_i+\tau_1 \qquad \tau_1 = \frac{1}{6}\left(\frac{F_{\xi\xi\xi}}{x_\xi}-\frac{x_{\xi\xi\xi}F_\xi}{x_\xi^2}\right)_i\Delta\xi^2+O(\Delta\xi^4)$$

$$\left[\frac{\partial}{\partial x}\left(l\frac{\partial F}{\partial x}\right)\right]_i \approx \frac{2}{(x_{i+1}-x_{i-1})}\left[l_e\left(\frac{F_{i+1}-F_i}{x_{i+1}-x_i}\right)-l_d\left(\frac{F_i-F_{i-1}}{x_i-x_{i-1}}\right)\right] \tag{30}$$

$$\Delta_2 = \left[\frac{(lF_\xi/x_\xi)_\xi}{x_\xi}\right]_i+\tau_2 \qquad \tau_2 = \tau_{22}\Delta\xi^2+\ldots$$

The evaluation of τ_{22} has not been obtained as it requires extensive algebra with $\Delta\xi^3$ terms included in the expansion. The local truncation error for the ordinary differential Eq. (28) becomes

$$\boxed{\Delta(F_i) = \left[\frac{1}{x_\xi}\left(\frac{l}{x_\xi}F_\xi\right)_\xi-\alpha F-\beta\frac{F_\xi}{x_\xi}\right]+\tau \qquad \tau = \left[\tau_{22}-\frac{1}{6}\beta\left(\frac{F_{\xi\xi\xi}}{x_\xi}-\frac{x_{\xi\xi\xi}F_\xi}{x_\xi^2}\right)\right]\Delta\xi^2+\ldots} \tag{31}$$

The first term in the difference relation expansion is the transformed governing Eq. (28) which is equal to zero. This approach gives a second-order method without any restrictions on the mesh other than $x_\xi \neq 0$.

The first derivative in Eq. (30) has the same difference approximation as Eq. (25) while the second derivative in Eq. (30) has the same difference approximation as Eq. (26). In Eq. (30), the truncation errors are rewritten with

$$F_\xi = x_\xi F_x \qquad F_{\xi\xi} = x_{\xi\xi}F_x + x_\xi^2 F_{xx} \qquad F_{\xi\xi\xi} = x_{\xi\xi\xi}F_x + 3x_\xi x_{\xi\xi}F_{xx} + x_\xi^3 F_{xxx}$$

The local truncation error for the first and second derivatives become

$$\Delta_1 = \left(\frac{\partial F}{\partial x}\right)_i + \tau_1 \qquad \tau_1 = \frac{1}{6}(3x_{\xi\xi}F_{xx} + x_\xi^2 F_{xxx})\Delta\xi^2 + O(\Delta\xi^4)$$
$$\Delta_2 = \left[\frac{\partial}{\partial x}\left(l\frac{\partial F}{\partial x}\right)\right]_i + \tau_2 \qquad \tau_2 = O(\Delta\xi^2) \tag{32}$$

When the coordinate derivatives are written in difference form as

$$x_{\xi\xi} = (x_{i+1} - 2x_i + x_{i-1})/\Delta\xi^2 = [(\kappa - 1)h_i]/\Delta\xi^2$$
$$x_\xi^2 = \left(\frac{x_{i+1} - x_{i-1}}{2\Delta\xi}\right)^2 = \left(\frac{h_{i+1} + h_i}{2\Delta\xi}\right)^2 = \left(\frac{\kappa^2 + 2\kappa + 1}{4}\right)\left(\frac{h_i}{\Delta\xi}\right)^2 \approx (\kappa^2 - \kappa + 1)\left(\frac{h_i}{\Delta\xi}\right)^2$$

then the truncation error for the first derivative in Eq. (25) and Eq. (32) is the same and the error is order of $\Delta\xi^2$. The truncation error of the second derivative in the physical coordinates [see Eq. (29)] is of order $(\kappa - 1)h$ and this term can be written as

$$(\kappa - 1)h_i = (h_{i+1} - h_i) = (x_{i+1} - 2x_i + x_{i-1}) = (x_{\xi\xi})_i\Delta\xi^2 + O(\Delta\xi^4)$$
$$h_i = \Delta\xi(x_\xi)_{i+1/2} + O(\Delta\xi^2) \tag{33}$$

This shows that the truncation error τ_2 is order of $\Delta\xi^2$. The transformed coordinate approach gives also a useful procedure to express the local discretization error. With a variable mesh spacing h_i, Eq. (1) becomes with the use of Eq. (33) the following:

$$e_i = f_i - F_i = \alpha_i(h_i)^p + \ldots = \bar{\alpha}_i(\Delta\xi)^p + \ldots \qquad i = 0, \ldots, I \qquad \Delta\xi = I^{-1} \tag{34}$$

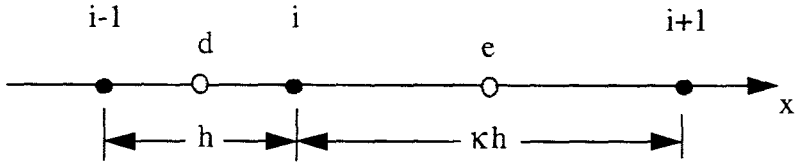
If the number of mesh cells or control volumes I is specified, then the transformed mesh size $\Delta\xi$ is known and the Richardson extrapolation procedure for a uniform mesh can be readily applied to the non-uniform mesh case. With the transformed coordinate approach, the mesh should be refined such that the variable spacing function $x(\xi)$ is used to determine the physical mesh spacing.

5.3 Box Scheme Method

The local truncation error of this method is best illustrated by the second-order differential Eq. (28), which is written as the two first-order differential equations

$$\frac{\partial Q}{\partial x} = W \qquad l\frac{\partial F}{\partial x} = Q \qquad W = \alpha F + \beta\frac{\partial F}{\partial x} \tag{35}$$

The above governing equations are written as difference equations with the mid-point scheme. The notation is illustrated below with d and e at the mid-point location of the cells.



The difference relation for ***cell*** e becomes

$$\left(\frac{q_{i+1}-q_i}{\kappa h}\right)-w_e=0 \qquad w_e=\alpha\left(\frac{f_{i+1}+f_i}{2}\right)+\beta\left(\frac{f_{i+1}-f_i}{\kappa h}\right)$$
$$l_e\left(\frac{f_{i+1}-f_i}{\kappa h}\right)-\left(\frac{q_{i+1}+q_i}{2}\right)=0 \tag{36}$$

The truncation error for the above relations has been evaluated with a Taylor series expansion of the dependent variables about the mid-point location which gives (φ is either of the dependent variables)

$$\left(\frac{\varphi_{i+1}-\varphi_i}{\kappa h}\right)=\left(\frac{\partial\varphi}{\partial x}\right)_e+\tau_1 \qquad \tau_1=\frac{1}{6}(\varphi_{xxx})_e\left(\frac{\kappa h}{2}\right)^2+\ldots$$
$$\left(\frac{\varphi_{i+1}+\varphi_i}{2}\right)=\varphi_e+\tau_a \qquad \tau_a=\frac{1}{2}(\varphi_{xx})_e\left(\frac{\kappa h}{2}\right)^2+\ldots$$

The truncation errors for the two equations become

$$\Delta_{Qe}=\left(\frac{Q_{i+1}-Q_i}{\kappa h}\right)-W_e=[Q_x-W]_e+\tau_{Qe}$$
$$\tau_{Qe}=\frac{1}{6}(Q_{xxx}-3\alpha F_{xx}-\beta F_{xxx})_e\left(\frac{\kappa h}{2}\right)^2+\ldots \qquad W_e=\alpha\left(\frac{F_{i+1}+F_i}{2}\right)+\beta\left(\frac{F_{i+1}-F_i}{\kappa h}\right) \tag{37}$$

$$\Delta_{Fe}=l_e\left(\frac{F_{i+1}-F_i}{\kappa h}\right)-\left(\frac{Q_{i+1}+Q_i}{2}\right)=[lF_x-Q]_e+\tau_{Fe}$$
$$\tau_{Fe}=\frac{1}{6}(lF_{xxx}-3Q_{xx})_e\left(\frac{\kappa h}{2}\right)^2+\ldots \qquad Q_{xx}=(lF_x)_{xx} \qquad Q_{xxx}=(lF_x)_{xxx}$$

Similar difference equations can be written for ***cell*** d and are

$$\left(\frac{q_i - q_{i-1}}{h}\right) - w_d = 0 \qquad w_d = \alpha\left(\frac{f_i + f_{i-1}}{2}\right) + \beta\left(\frac{f_i - f_{i-1}}{h}\right)$$
$$l_d\left(\frac{f_i - f_{i-1}}{h}\right) - \left(\frac{q_i + q_{i-1}}{2}\right) = 0 \tag{38}$$

The truncation errors in cell d are

$$\Delta_{Qd} = \left(\frac{Q_i - Q_{i-1}}{h}\right) - W_d = [Q_x - W]_d + \tau_{Qd}$$
$$\tau_{Qd} = \frac{1}{6}(Q_{xxx} - 3\alpha F_{xx} - \beta F_{xxx})_d\left(\frac{h}{2}\right)^2 + \ldots \qquad W_d = \alpha\left(\frac{F_i + F_{i-1}}{2}\right) + \beta\left(\frac{F_i - F_{i-1}}{h}\right) \tag{39}$$

$$\Delta_{Fd} = l_d\left(\frac{F_i - F_{i-1}}{h}\right) - \left(\frac{Q_i + Q_{i-1}}{2}\right) = [lF_x - Q]_d + \tau_{Fd}$$
$$\tau_{Fd} = \frac{1}{6}(lF_{xxx} - 3Q_{xx})_d\left(\frac{h}{2}\right)^2 + \ldots$$

The dependent variables can be solved from the difference equations given by Eq. (36) and Eq. (38). If $i = 1, 2, \ldots, I$, then the unknown dependent variables are

$$f_2, f_3, \ldots, f_{I-1} \quad \text{and} \quad q_1, q_2, \ldots, q_I$$

This gives $2(I-1)$ unknowns where f_1 and f_I are known from the boundary conditions. There are $2(I-1)$ difference equations for the $(I-1)$ mesh cells. The number of difference equations matches the number of unknowns and the linear difference equations can be readily solved. This scheme has a local truncation error of order h^2 and the method is ***second-order for any mesh spacing***. The *mesh refinement for this approach* takes the coarse mesh and divides each of the basic mesh cells into a uniform number of cells. For example, the first refinement would have 2 cells of equal size within each basic mesh cell and the next refinement would have 4 cells of equal size within each basic mesh cell.

How is this approach related to the previous methods? The two difference relations given in Eq. (37) for cell e are combined with Q_{i+1} eliminated and this gives a difference equation with the indicated local truncation error

$$l_e\left(\frac{F_{i+1} - F_i}{\kappa h}\right) - Q_i = \left(\frac{\kappa h}{2}\right)W_e + \Delta_e \qquad i = 1, 2, \ldots, I-1$$
$$\Delta_e = \Delta_{Fe} + \left(\frac{\kappa h}{2}\right)\Delta_{Qe} = \left[lF_x - Q + \left(\frac{\kappa h}{2}\right)(Q_x - W)\right]_e + \tau_e$$
$$\tau_e = \tau_{Fe} + \left(\frac{\kappa h}{2}\right)\tau_{Qe} = \frac{1}{6}(lF_{xxx} - 3Q_{xx})_e\left(\frac{\kappa h}{2}\right)^2 + O(h^3)$$

The leading term in the local truncation error results from the derivative term which is related to the second derivative in the original equation. The two difference relations given in Eq. (39) for cell d are combined with Q_{i-1} eliminated and this gives a difference equation with the indicated local truncation error

$$-l_d\left(\frac{F_i - F_{i-1}}{h}\right) + Q_i = \left(\frac{h}{2}\right)W_d - \Delta_d \qquad i = 2, 3, \ldots, I$$

$$\Delta_d = \Delta_{Fd} - \left(\frac{h}{2}\right)\Delta_{Qd} = \left[lF_x - Q - \frac{h}{2}(Q_x - W)\right]_d + \tau_d$$

$$\tau_d = \tau_{Fd} - \left(\frac{h}{2}\right)\tau_{Qd} = \frac{1}{6}(lF_{xxx} - 3Q_{xx})_d\left(\frac{h}{2}\right)^2 + O(h^3)$$

The addition of the above two equations eliminates Q_i and gives a difference equation only involving the dependent variable F_i which is

$$l_e\left(\frac{F_{i+1} - F_i}{\kappa h}\right) - l_d\left(\frac{F_i - F_{i-1}}{h}\right) = \left(\frac{\kappa h}{2}\right)W_e + \left(\frac{h}{2}\right)W_d + \Delta$$

$$\left(\frac{\kappa h}{2}\right)W_e + \left(\frac{h}{2}\right)W_d = \alpha\frac{h}{4}[\kappa F_{i+1} + (1+\kappa)F_i + F_{i-1}] + \frac{\beta}{2}[F_{i+1} - F_{i-1}] \qquad (40)$$

The difference relation expansion and the local truncation error of the difference Eq. (40) are

$$\Delta = \Delta_e - \Delta_d = \left[lF_x - Q + \left(\frac{\kappa h}{2}\right)(Q_x - W)\right]_e - \left[lF_x - Q - \left(\frac{h}{2}\right)(Q_x - W)\right]_d + \tau$$

$$\tau = \frac{1}{6}[(lF_{xxx} - 3Q_{xx})_e\kappa^2 - (lF_{xxx} - 3Q_{xx})_d]\left(\frac{h}{2}\right)^2 + O(h^3)$$

or expanding the e and d terms about the i location gives

$$\Delta = \Delta_e - \Delta_d = \left[lF_x - Q + \left(\frac{\kappa h}{2}\right)(Q_x - W)\right]_e - \left[lF_x - Q - \left(\frac{h}{2}\right)(Q_x - W)\right]_d + \tau$$

$$\tau = \frac{1}{24}[lF_{xxx} - 3(lF_x)_{xx}]_i(\kappa^2 - 1)h^2 + O(h^3)$$

The local truncation error appears to be second-order with an arbitrary mesh spacing with the formulation as given in Eq. (40).

Eq. (40) is rewritten in a form consistent with the governing equation Eq. (28) which gives the new difference equation

$$\frac{2}{(1+\kappa)h_i}\left[l_e\left(\frac{f_{i+1}-f_i}{\kappa h_i}\right)-l_d\left(\frac{f_i-f_{i-1}}{h_i}\right)\right]$$
$$-\alpha\left[\frac{\kappa f_{i+1}+(1+\kappa)f_i+f_{i-1}}{2(1+\kappa)}\right]-\beta\left[\frac{f_{i+1}-f_{i-1}}{(1+\kappa)h_i}\right]=0 \qquad (41)$$

The leading term in the local truncation error results from the second derivative and the local truncation error for Eq. (41) becomes

$$\bar{\Delta} = [(lF_x)_x - \alpha F - \beta F_x] + \bar{\tau} \qquad \bar{\tau} = \frac{1}{12}[lF_{xxx} - 3(lF_x)_{xx}]_i(\kappa - 1)h + O(h^2) \qquad (42)$$

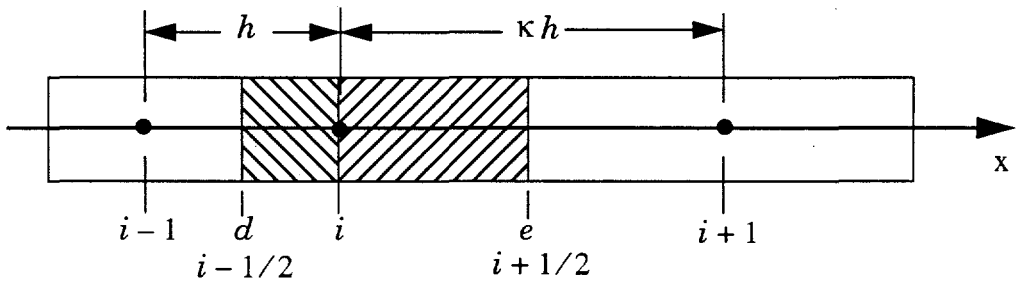
Comparison of the above with the local truncation error for the physical coordinate case as given in Eq. (29) shows that the same terms appear but there are some differences. In addition, the local truncation error for both methods appears to be first-order for a non-uniform mesh. However, if the box scheme is written as Eq. (40), the method is second-order in this form.

5.4 Control Volume Method

The governing Eq. (28) is written in integral form as

$$\int_{Surface} E dA + \int_{Volume} \alpha F dV = 0 \qquad E = \beta F - l\frac{\partial F}{\partial x} = \text{Flux}$$

The notation and control volume for this case is illustrated below:



The control volume around the node i goes from $i - 1/2 = d$ to $i + 1/2 = e$. The surfaces of the control volume are at the mid-points between the nodes. The area of these surfaces is A. The control volume round the node i is divided into the two subcontrol volumes which go from d to i and i to e. This formulation approach allows boundary conditions to be more readily applied. The integral equation for the subcontrol volume from d to i is approximated as

$$A(E_i - E_d) + A\alpha\int_{-h/2}^{0} F dx = 0 \qquad F \approx f_i + (f_i - f_{i-1})(x/h) \qquad -\frac{h}{2} \le x \le 0$$

$$E_d \approx \beta\left(\frac{f_{i-1} + f_i}{2}\right) - l_d\left(\frac{f_i - f_{i-1}}{h}\right)$$

and the difference equation for the subcontrol volume becomes

$$l_d\left(\frac{f_i - f_{i-1}}{h}\right)A + \alpha\left(\frac{f_{i-1} + 3f_i}{4}\right)\left(\frac{hA}{2}\right) - \beta\left(\frac{f_{i-1} + f_i}{2}\right)A + E_i A = 0$$

The local truncation error for this difference equation is obtained with a Taylor's series expansion about the point i and becomes

$$\frac{\Delta_d}{A} = (lF_x - \beta F) + \tau_d \qquad \tau_d = [-(lF_x)_x + \alpha F + \beta F_x]\frac{h}{2} + \frac{1}{2}\tilde{a}h^2 - \frac{1}{2}\tilde{b}h^3 + \ldots$$

The coefficients in this expansion are

$$\tilde{a} = \bar{a} - \frac{1}{4}\alpha F_x - \frac{1}{2}\beta F_{xx} \qquad \tilde{b} = \bar{b} - \frac{1}{8}\alpha F_{xx} - \frac{1}{6}\beta F_{xxx} \tag{43}$$

where the coefficient $\bar{a}$ and $\bar{b}$ are defined in Eq. (27). The first term in the above local truncation error expansion is zero as this is the governing differential Eq. (28).

The integral equation for the subcontrol volume from i to e is approximated as

$$A(E_e - E_i) + A\alpha\int_{0}^{\kappa h/2} F dx = 0 \qquad F \approx f_i + (f_{i+1} - f_i)(x/\kappa h) \qquad 0 \le x \le \frac{\kappa h}{2}$$

$$E_e \approx \beta\left(\frac{f_i + f_{i+1}}{2}\right) - l_e\left(\frac{f_{i+1} - f_i}{\kappa h}\right)$$

and the difference equation for the subcontrol volume becomes

$$-l_e\left(\frac{f_{i+1} - f_i}{\kappa h}\right)A + \alpha\left(\frac{3f_i + f_{i+1}}{4}\right)\left(\frac{\kappa hA}{2}\right) + \beta\left(\frac{f_i + f_{i+1}}{2}\right)A - E_i A = 0$$

The local truncation error for this difference equation is obtained with a Taylor's series expansion about the point i and becomes

$$\frac{\Delta_e}{A} = (-lF_x + \beta F) + \tau_e \qquad \tau_e = [-(lF_x)_x + \alpha F + \beta F_x]\frac{\kappa h}{2} - \frac{1}{2}\tilde{a}(\kappa h)^2 - \frac{1}{2}\tilde{b}(\kappa h)^3 + \ldots$$

The difference equation for the complete control volume around the mode i is scaled by the volume of the control volume $A(1 + \kappa)h/2$ which gives

$$\Delta(f_i) = \frac{2}{(1+\kappa)h}\left[l_e\left(\frac{f_{i+1}-f_i}{\kappa h}\right) - l_d\left(\frac{f_i - f_{i-1}}{h}\right)\right]$$
$$-\alpha\left[\frac{\kappa f_{i+1} + 3(1+\kappa)f_i + f_{i-1}}{4(1+\kappa)}\right] - \beta\left(\frac{f_{i+1}-f_{i-1}}{(1+\kappa)h}\right) = 0 \tag{44}$$

This scaling is required to obtain the correct local truncation error of the difference approximation. Comparison of the control volume scheme with Eq. (41) shows that the diffusion (second-derivative) terms and the convective (first derivative) terms are the same. The source terms use different weighting of the three mode values. The local truncation error for the difference equation for the complete control volume is

$$\Delta(F_i) = -\frac{2(\Delta_d + \Delta_e)}{A(1+\kappa)h} = [(lF_x)_x - \alpha F - \beta F_x] + \tau$$
$$\tau = \tilde{a}(\kappa - 1)h + \tilde{b}(\kappa^2 - \kappa + 1)h^2 + \ldots$$

The first term in the difference relation expansion is the differential Eq. (28) which is zero. The truncation error is second-order for a uniform mesh and variable meshes with $\kappa = 1 + O(h)$. A comparison of this truncation error with Eq. (42) shows that both are of the same order but have different coefficients. The coefficients in the above equation are given in Eq. (43).

5.5 Ferziger and Peric Approach

The authors Ferziger and Peric [40] have pointed out that a good numerical method makes the local truncation error approximately constant over the solution domain and this requires a non-uniform mesh. They investigated the local truncation error of the first derivative approximated with a central difference scheme. The mesh is defined as

$$h_i = x_i - x_{i-1} \qquad h_{i+1} = \kappa(m)h_i$$

The difference approximation, difference relation expansion, and local truncation error become [see Eq. (25)]

$$\frac{\partial f}{\partial x} = \frac{f_{i+1} - f_{i-1}}{(1+\kappa)h_i} \qquad \Delta_1 = \frac{F_{i+1} - F_{i-1}}{(1+\kappa)h_i} = \frac{\partial F}{\partial x} + \tau_1$$
$$\tau_1 = \frac{1}{2}F_{xx}(\kappa - 1)h_i + \frac{1}{6}F_{xxx}(\kappa^2 - \kappa + 1)h_i^2$$

If the truncation error mesh coefficient $|\kappa - 1|h_i = h_i^2$, then the difference approximation is second-order; otherwise, the scheme is first-order. Ferziger and Peric investigated the behavior of the above difference approximation as the mesh is refined with the following two approaches given in Sections 5.5.1 and 5.5.2.

5.5.1 Add Nodes Between Coarse Nodes - Coarse to Fine Mesh Approach

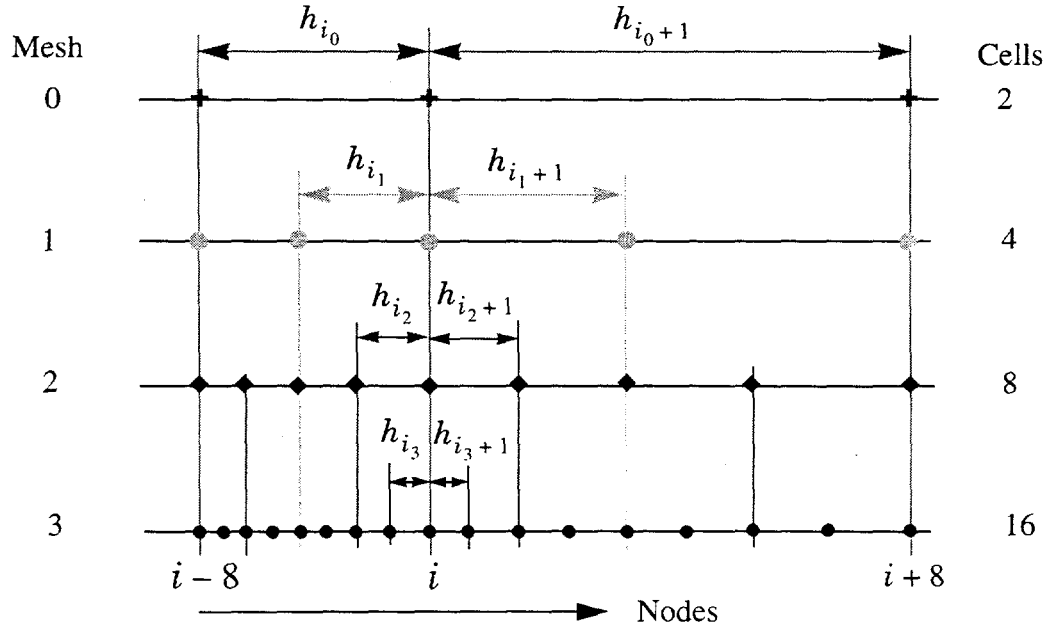
This mesh refinement approach is used with the Box Scheme Method (see section 5.3). In this approach after several mesh refinements, the mesh is uniform everywhere except at the original coarse mesh points. The authors indicate that global error decreases just a bit more slowly than in a true second-order scheme. No definitive results are presented to justify this conclusion. However, results for this type of mesh refinement has been investigated by Blottner [41] and results obtained for boundary layer flows indicate second-order behavior.

5.5.2 Insert Nodes to Maintain Constant Spatial Mesh Ratio - Fine to Coarse Mesh Approach

With this approach, the mesh stretching parameter $\kappa(m)$ is constant for a mesh m but will vary as the mesh is refined. In Chapter 4 on Mesh Stretching, it is shown in Eq. (19) that the mesh refinement parameter varies as

$$\kappa(m+1) = \sqrt{\kappa(m)} \qquad m = 0, 1, 2, \ldots$$

where $\kappa(0)$ is the coarsest mesh. The mesh spacing for an arbitrary point i for 4 meshes and with $\kappa(3) = 1.1$ is illustrated below. This mesh has co-located nodes at the coarse node locations. The value of i_m used in the following relations is the i node location for the mesh m with $i_m = 0, 1, \ldots, I_m$.



On the finest mesh $m = 3$, the stretching parameter is specified as $\kappa(3) = \kappa > 1$. Then the stretching parameter for the other meshes is determined from Eq. (19) and is given in Table 5.1. For mesh m, the smallest cell occurs at $i_m = 1$ or $\delta_m = h_1$. If $\delta_3 = h$, the size of the smallest cell for each of the meshes is given in Table 5.1. The cell size h_{i_m} is obtained from Eq. (18) which gives

$$h_{i_m} = \delta_m[\kappa(m)]^{i_m - 1} \qquad i_m = 0, 1, \ldots, I_m$$

$$\delta_m = x_1 - x_0 = (x_I - x_0)[\kappa(m) - 1]/[\kappa(m)^{I_m} - 1]$$

The *truncation error mesh coefficient* $|\kappa(m) - 1|h_{i_m}$ is readily determined and is tabulated in Table 5.1.

Table 5.1: Truncation Error Mesh Coefficient with Non-Uniform Mesh

Mesh (m)	$\kappa(m)$	δ_m	h_{i_m}	$[\kappa(m) - 1]h_{i_m}$
0	κ^8	$(\kappa^4 + 1)\delta_1$	$\delta_0\kappa^{8(i_m - 1)}$	$\delta_0(\kappa^8 - 1)\kappa^{8(i_m - 1)}$
1	κ^4	$(\kappa^2 + 1)\delta_2$	$\delta_1\kappa^{4(i_m - 1)}$	$\delta_1(\kappa^4 - 1)\kappa^{4(i_m - 1)}$
2	κ^2	$(\kappa + 1)\delta_3$	$\delta_2\kappa^{2(i_m - 1)}$	$\delta_2(\kappa^2 - 1)\kappa^{2(i_m - 1)}$
3	κ	$\delta_3 = h$	$\delta_3\kappa^{(i_m - 1)}$	$\delta_3(\kappa - 1)\kappa^{(i_m - 1)}$

The foregoing mesh refinement is illustrated at one node location i_m for the following conditions:

$$\begin{aligned} &x_0 = 0 \qquad x_I = 1 \qquad \text{On mesh 3: } I_3 = 80 \qquad \kappa(3) = 1.1 \qquad i_3 = 40 \\ &\text{On all 4 meshes: } I_m = 10, 20, 40, 80 \qquad i_m = 5, 10, 20, 40 \qquad m = 0, 1, 2, 3 \end{aligned} \tag{45}$$

The results for this example are given in Table 5.2 for the four meshes. At each mesh refinement, the number of mesh cell doubles. The cell size is decreasing roughly by a factor of 2 as the mesh is refined. The truncation error mesh coefficient $|\kappa(m) - 1|h_{i_m}$ is decreasing roughly by a factor of 4 as the mesh is refined. In the last column of the table, the truncation error mesh coefficient has been normalized to one for the finest mesh. The parameter ϕ^2 is the expected variation of the normalized truncation error mesh coefficient for second-order behavior. The normalized truncation error mesh coefficient should be one for a second-order behavior. The example shows that even with a large variation of the cell size (approximately factor of two), the non-uniform mesh has a small impact on the second-order behavior of the local truncation error when $\kappa \cong$ constant.

Table 5.2: Example of Truncation Error Mesh Coefficient

Mesh (m)	$\kappa(m)$	i_m	I_m	ϕ	h_{i_m}	$[\kappa(m)-1]h_{i_m}$	$\frac{\lvert\kappa(m)-1\rvert h_{i_m}}{\lvert\kappa-1\rvert h_{i_3}\phi^2}$
0	2.1435881	5	10	8	1.179324×10^{-2}	1.349×10^{-2}	1.0486
1	1.4641	10	20	4	7.007216×10^{-3}	3.252×10^{-3}	1.0114
2	1.21	20	40	2	3.836530×10^{-3}	8.057×10^{-4}	1.0023
3	1.1	40	80	1	2.009611×10^{-3}	2.010×10^{-4}	1.0000

As pointed out by Ferziger and Peric, as the mesh is refined, the stretching parameter $\kappa \rightarrow 1$ and convergence becomes second-order asymptotically. Ferziger and Peric state, "Although central difference schemes become formally first-order-accurate on non-uniform grids, the error is reduced in nearly a second-order manner as the grid is refined." Turbulent boundary layer results have been obtained by Blottner [42], [43] with a variable grid scheme on non-uniform meshes and the results indicate second-order behavior.

5.6 Paradox of Truncation Error with Non-Uniform Meshes

From a formal analysis of the Transformed Coordinate Approach and the Box Scheme Method, the leading term in the expansion for the local truncation error can be shown to be second-order. The local truncation error is proportional to the transformed cell size squared or is proportional to the inverse of the total number of cells squared. For the difference form of the partial differential equation in physical coordinates and the Control Volume Method, the local truncation error is formally first-order. Yet from evaluation of numerical results, all of the methods show second-order behavior if appropriate difference approximations and meshes are employed. It appears that the degradation of a second-order scheme to first-order on non-uniform meshes for the one-dimensional case is not very significant for meshes with reasonable cell size variation. More work is needed to determine how much larger than $\kappa = 1 + O(h)$ can be tolerated without changing the order of the scheme.

6. Richardson Extrapolation Procedure and Examples for Steady-State Problems

6.1 Extrapolation with Multi-Mesh Refinement

The solution of the governing equations should be obtained on a series of meshes to be sure that the errors are sufficiently small and the higher order truncations can be neglected in the Richardson extrapolation relations. In addition, the discretization error of the extrapolated result can be decreased by using several extrapolations. This procedure is illustrated for the case of the solution on 3 meshes, but can be used on a larger series of refined meshes. The following development assumes a one-dimensional problem or a multi-dimensional problem where the mesh is refined in all directions by the same factor. The computational result f_m is related to the exact solution F_e plus a discretization error. The discretization error for central difference schemes on ***uniform meshes*** has the following expansion in terms of the spatial step-size h_m on the mesh m:

$$f_m = F_e + Error \qquad Error = \alpha h_m^2 + \beta h_m^4 + \ldots \qquad m = 0, 1, 2 \tag{46}$$

The numerical scheme is second-order as the first term in the error expansion involves the step-size squared. The second term in the expansion is fourth-order. The behavior of the discretization error for the difference scheme being used is needed and the impact of non-uniform mesh spacing on the truncation error is important. The 0 mesh has the largest spatial step-size and the step-size decreases as the value of m increases. The above relation is evaluated for the 0 and 1 meshes which gives the known numerical solution values as a function of the unknowns F_e, α, and β

$$f_0 = F_e + \alpha h_0^2 + \beta h_0^4 + \ldots \qquad f_1 = F_e + \alpha h_1^2 + \beta h_1^4 + \ldots$$

These two relations are combined to obtain the Richardson extrapolation value, $f_{0,1}$, from the numerical solutions with meshes 0 and 1

$$f_{0,1} = \frac{h_0^2 f_1 - h_1^2 f_0}{h_0^2 - h_1^2} = F_e - \beta h_0^2 h_1^2 + O(h^6) \tag{47}$$

A similar relation is determined for meshes 1 and 2 which gives

$$f_{1,2} = \frac{h_1^2 f_2 - h_2^2 f_1}{h_1^2 - h_2^2} = F_e - \beta h_1^2 h_2^2 + O(h^8) \tag{48}$$

From solutions on two meshes, the *Richardson extrapolation value gives an estimate of the exact solution* if the fourth and higher-order terms are neglected. This is expressed as

$$f_{0,1} = f_1 + \left[\frac{h_1^2}{h_0^2 - h_1^2}\right](f_1 - f_0) = F_e + O(h^4) \approx F_e$$
$$f_{1,2} = f_2 + \left[\frac{h_2^2}{h_1^2 - h_2^2}\right](f_2 - f_1) = F_e + O(h^4) \approx F_e \quad (49)$$

Eq. (47) and Eq. (48) are combined to obtain an improved Richardson extrapolation value where six-order terms are neglected

$$f_{0,1,2} = \frac{h_0^2 f_{1,2} - h_2^2 f_{0,1}}{h_0^2 - h_2^2} = f_{1,2} + \left[\frac{h_2^2}{h_0^2 - h_2^2}\right](f_{1,2} - f_{0,1}) = F_e + O(h^6) \approx F_e \quad (50)$$

6.1.1 Example 4: Turbulent Boundary Layer Flow

The procedure is illustrated with results obtained by Keller and Cebeci [17] where the box scheme has been used to solve a turbulent boundary layer flow problem. The flow solution has been obtained with 3 meshes with the number of mesh points in the flow direction held constant and the grid refined across the layer. In addition, a fine mesh solution has been obtained and this result is close to the exact result. From these solutions, the skin friction $f = c_f$ at $Re_x = 1.88 \times 10^7$ is tabulated in Table 6.1. The extrapolated skin friction results from solutions on two meshes are obtained from Eq. (49) which give

$$f_{0,1} = 3.2536 \times 10^{-3} \qquad f_{1,2} = 3.2330 \times 10^{-3} \quad (51)$$

Using these results, an improved extrapolated result is obtained from Eq. (50) for the skin friction

$$f_{0,1,2} = 3.2304 \times 10^{-3} \quad (52)$$

The Richardson extrapolated results and the fine mesh result have been used to estimate the accuracy or percent error of the skin friction which is defined as

$$\% \text{ Error} = 100 \times (f_m - F_e)/F_e \quad \text{on mesh } m \quad (53)$$

The fine mesh solution is believed to provide a result that is close to the exact solution. The error results are given in Table 6.1 where it can be seen that a reasonable estimate of the skin friction error can be obtained with the use of either of the extrapolated results. This illustrates that Richardson extrapolation can be used to obtain a reasonable estimate of the numerical solution error.

Table 6.1: Skin Friction for Turbulent Boundary Layer

Mesh m	Mesh Size	h_m	$c_f \times 10^3$	%Error $f_{1,2}$	%Error $f_{0,1,2}$	%Error fine mesh
0	18×11	1/10	3.6599	13.20	13.30	13.41
1	18×21	1/20	3.3552	3.78	3.86	3.97
2	18×31	1/30	3.2873	1.68	1.76	1.86
Fine grid	18×231	1/230	3.2272			

The value of the exact solution and the coefficients in the quadratic relation Eq. (46) are determined from three numerical solutions with different mesh sizes. The result for the value of the exact solution is the same as given by Eq. (50) and this is expressed as

$$F_e \approx f_{0,1,2} = f_{1,2} + \left[\frac{h_2^2}{h_0^2 - h_2^2}\right](f_{1,2} - f_{0,1}) = 3.2304 \times 10^{-3} \tag{54}$$

The coefficients in the quadratic relation Eq. (46) have the following values:

$$\alpha = \frac{1}{h_0^2 - h_2^2}\left\{-(h_1^2 + h_2^2)\left[\frac{f_0 - f_1}{h_0^2 - h_1^2}\right] + (h_0^2 + h_1^2)\left[\frac{f_1 - f_2}{h_1^2 - h_2^2}\right]\right\} = 52.24 \times 10^{-3} \tag{55}$$

$$\beta = \frac{1}{h_0^2 - h_2^2}\left\{\left[\frac{f_0 - f_1}{h_0^2 - h_1^2}\right] - \left[\frac{f_1 - f_2}{h_1^2 - h_2^2}\right]\right\} = -929.4 \times 10^{-3} \tag{56}$$

With the result in Eq. (52) used as the exact solution and the coefficients in Eq. (55) and Eq. (56), the expected variation of the numerical solution or the solution error with different mesh sizes is determined. This solution error is represented as the curve labeled "1st and 2nd terms" in Figure 6.1. The line labeled slope = 2 is the linear part of Eq. (46). All of the ways to estimate the errors of the three numerical solutions give reasonable values for the numerical errors. It should be pointed out that the percent error for two finest meshes will fall on the slope = 2 line with Richardson extrapolation method with one term in the expansion. Therefore, the coarsest mesh result shows the real deviation from slope = 2.

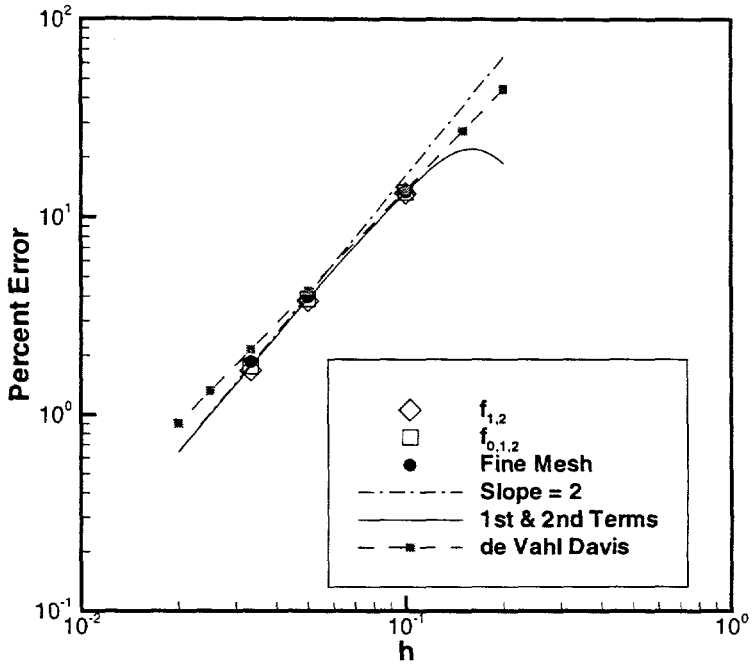


Figure 6.1: Error in Boundary Layer Skin Friction at $Re_x = 1.88 \times 10^7$.

6.2 de Vahl Davis Approach for Extrapolated Solutions

If the order of the solution scheme is not known, de Vahl Davis [15] determined the value of the discretization error order from the results on three different meshes that have the mesh points co-located at the coarse mesh point locations. With this approach the numerical solution is related to the exact solution and the discretization error by the relation

$$f_m = F_e + \alpha h_m^p + \ldots \qquad m = 0, 1, 2 \tag{57}$$

From the three numerical solutions, the order of accuracy of the numerical scheme, p, as the mesh cell size goes to zero is obtained from

$$r_1^p \left(\frac{r_0^p - 1}{r_1^p - 1} \right) = \left[\frac{f_0 - f_1}{f_1 - f_2} \right] \qquad r_0 = \frac{h_0}{h_1} \qquad r_1 = \frac{h_1}{h_2} \tag{58}$$

For the special case of $r = r_0 = r_1$, the above becomes

$$p = \ln\left[\frac{f_0 - f_1}{f_1 - f_2}\right] / \ln r \tag{59}$$

The estimated exact solution and the coefficient in Eq. (57) become

$$F_e = f_2 + \left[\frac{1}{r_1^p - 1}\right](f_2 - f_1) \qquad \alpha = \frac{f_1 - f_2}{h_2^p (r_1^p - 1)} \tag{60}$$

For the skin friction results given in Table 6.1, the order of accuracy of the numerical scheme is obtained from Eq. (58) with an iterative solution of the equation

$$p = \ln\left\{\left[\frac{r_1^p - 1}{r_0^p - 1}\right]\left[\frac{f_0 - f_1}{f_1 - f_2}\right]\right\} / \ln r_1 = 1.689509$$

The estimated exact solution and the coefficient become

$$F_e = 3.218285 \times 10^{-3} \qquad \alpha = 21.60492 \times 10^{-3}$$

The expected error for various mesh spacing with the de Vahl Davis approach is given in Figure 6.1. This approach will improve when the three meshes have been refined further. However, the present result is not accurate in estimating the solution error when the mesh sizes are very small. One should use four levels of mesh refinement to ascertain that the asymptotic order of accuracy of the numerical scheme has been obtained.

In the work of Celik and Karatekin [20], the above equation for p is modified by taking the absolute value of the right hand side. This is necessary in order to ensure extrapolation toward decreasing step-size. For oscillatory convergence, the coarse and fine mesh solutions are given by Eq. (57) and the medium mesh solution is written as

$$f_1 = F_e - \alpha h_1^p + \ldots$$

The resulting equation for p becomes

$$p = \left| \left\{ \ln\left[\frac{f_0 - f_1}{f_2 - f_1}\right] - \ln\left[\frac{r_0^p + 1}{r_1^p + 1}\right] \right\} / \ln r_1 \right| \tag{61}$$

When the calculated value of p is unrealistic, the value of p is bounded within its theoretical limits. This approach is used to avoid unreasonable extrapolated solution values and solution error estimates.

6.3 Multi-Dimensional Problems

6.3.1 Same Mesh Refinement in All Coordinate Directions

The appropriate Richardson extrapolation relation to use for a three-dimensional problem with a second-order numerical scheme is the following:

$$f_m = F_e + a\Delta x_m^2 + b\Delta y_m^2 + c\Delta z_m^2 + \ldots \qquad m = 0, 1, \ldots, M \tag{62}$$

In the use of this relation, it is assumed that the spatial step-sizes are sufficiently small such that the higher order terms can be neglected. The mesh is refined in each coordinate direction by the *mesh refinement ratio* r as follows:

$$\Delta x_m = r\Delta x_{m+1} \qquad \Delta y_m = r\Delta y_{m+1} \qquad \Delta z_m = r\Delta z_{m+1} \tag{63}$$

In the above, if the mesh step-size in each coordinate direction is halved, then $r = 2$. The numerical solutions can be written as

$$\begin{aligned} f_m &= F_e + \alpha h_m^2 \qquad \alpha h_m^2 = a\Delta x_m^2 + b\Delta y_m^2 + c\Delta z_m^2 \\ f_{m+1} &= F_e + \alpha h_{m+1}^2 = F_e + \frac{1}{r^2}\alpha h_m^2 \qquad h_{m+1} = h_m/r \end{aligned} \tag{64}$$

The relations in Eq. (64) are the same form as Eq. (46) and the extrapolation relations given in Eq. (49) and Eq. (50) can be used.

6.3.2 Example 5: Natural Convective Flow in a Cavity

This grid refinement approach is illustrated with the results obtained by Hortmann, Peric and Scheuerer [18] for the natural convective flow in a cavity. The solution results for the Nusselt number for the heat flux across the cavity for a Rayleigh number of 10^6 is given in Table 6.2. A uniform mesh is used for this solution. Additional solutions for this case with a non-uniform mesh have been obtained with a finer mesh (640×640). A very accurate value of the Nusselt number has been obtained from the two finest mesh results with Richardson extrapolation which gives

$$F_e = \overline{Nu}_{exact} = 8.82511 \tag{65}$$

This solution is considered exact to the number of decimal places presented and can be used to accurately evaluate the error of the numerical solutions given in the table. It is assumed that the solutions are only available from meshes 1, 2, and 3. If only the first term in the expansion is retained in Eq. (46), then this equation and its error become

$$f_m = F_e + \alpha h_m^2 \qquad m = 2, 3 \qquad \%\,Error = 100\left(\frac{f_m - F_e}{F_e}\right) = \left(\frac{100\alpha}{F_e}\right)h_m^2 \tag{66}$$

The Richardson extrapolated exact solution with the *first term expansion* is determined from

$$F_e = f_3 + \left[\frac{h_3^2}{h_2^2 - h_3^2}\right](f_3 - f_2) = 8.829030$$

The coefficient is obtained from

$$\alpha = \frac{f_2 - f_3}{h_2^2 - h_3^2} = 948.224$$

The percent error of each mesh solution from Eq. (66) is given in Table 6.2 as the first term result.

Table 6.2: Nusselt Number for Heat Flux Across Cavity with Uniform Mesh

Mesh m	Mesh Cells	h_m	$F_m = \overline{Nu}$	% Error F_e	% Error 1st Term	% Error 1st + 2nd Terms	% Error de Vahl Davis
0	10×10	0.1000	8.46097	–4.126	107.4	–41.36	57.44
1	20×20	0.0500	10.5976	20.085	26.85	18.80	21.71
2	40×40	0.0250	9.42167	6.760	6.712	6.520	8.207
3	80×80	0.0125	8.97719	1.723	1.678	1.744	3.102
4	160×160	0.00625	8.86302	0.4293	0.4195	0.4431	1.173
5	320×320	0.003125	8.83459	0.1074	0.1049	0.1112	0.4432

When the first and second terms in the error expansion are retained, the Richardson extrapolated exact solution and coefficients in Eq. (46) are obtained from Eq. (54), Eq. (55), and Eq. (56) using the same 3 meshes. The results for the *first and second term expansion* are

$$F_e = 8.815652 \qquad \alpha = 1055.244 \qquad \beta = -1.369862 \times 10^5$$

The predicted error from Eq. (46) is given in the table as the first plus second term result.

The approach of de Vahl Davis has been used with the same 3 meshes to determine the Richardson extrapolated exact solution, coefficient, and p in Eq. (57) with the use of Eq. (59) and Eq. (60). These parameters become

$$F_e = 8.707089 \qquad \alpha = 126.68 \qquad p = 1.4036$$

The percent error for this case becomes

$$\% \; Error = \left(\frac{100\alpha}{F_e}\right) h_m^p$$

and is tabulated in the table as the de Vahl Davis result. The error predictions for the four cases are given in Figure 6.2. The de Vahl Davis approach gives poor estimate of the solution error when the mesh size is small while all of the other methods give a reasonable estimate. All of the methods approach the same error estimate as the mesh is refined. When the mesh is coarse, only the error estimated with the first and second terms included gives the correct behavior. See the coarse mesh solution in the above Table 6.2 where the error is negative for this mesh.

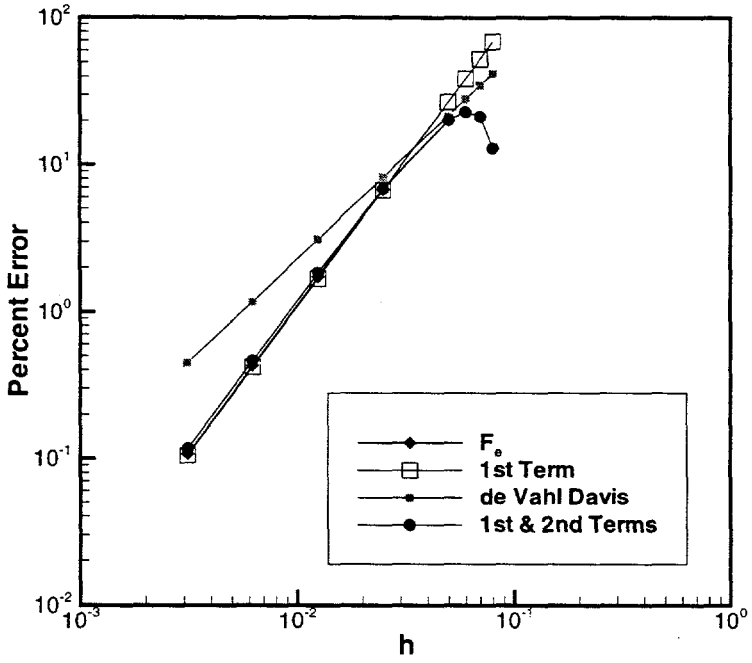


Figure 6.2: Error of Nusselt Number for Cavity Problem at $Ra = 10^6$.

6.3.3 Mesh Refinement in One Coordinate Direction

To obtain more information about the solution error, it is necessary to refine the mesh in one coordinate direction at a time. The mesh step-size in each coordinate direction should be of a size that results in solution errors of the same magnitude for each coordinate direction. The mesh can be refined from the coarse mesh where $l = m = n = 0$ to the fine mesh where $l = L$, $m = M$, and $n = N$. In the present approach, only one of the grid indices of l, m, n is varied

while the other two are held fixed. The previous numerical solution expansion Eq. (62) is now extended. The numerical solution, exact solution, and discretization error for a three-dimensional problem can be expressed as

$$f_{lmn} = F_e + a\Delta x_l^2 + b\Delta y_m^2 + c\Delta z_n^2 + \ldots$$

where the first term in the discretization error expansion has been retained. The subscripts l, m, n are used to indicate the mesh refinement being used in each coordinate direction. The numerical solution relations for mesh refinement in the x coordinate direction, y coordinate direction and z coordinate direction are the following:

$$\begin{aligned}
f_{lMN} &= F_e + a\Delta x_l^2 + A_{MN} + \ldots \qquad A_{MN} = b\Delta y_M^2 + c\Delta z_N^2 \qquad l = 0, 1, \ldots, L \\
f_{LmN} &= F_e + b\Delta y_m^2 + B_{LN} + \ldots \qquad B_{LN} = a\Delta x_L^2 + c\Delta z_N^2 \qquad m = 0, 1, \ldots, M \\
f_{LMn} &= F_e + c\Delta z_n^2 + C_{LM} + \ldots \qquad C_{LM} = a\Delta x_L^2 + b\Delta y_M^2 \qquad n = 0, 1, \ldots, N
\end{aligned} \tag{67}$$

In the above relations, when the mesh is being refined in one coordinate direction, the mesh spacing in the other two directions is the finest mesh spacing. This is the best approach to use, but is computational very expensive. When the solution is obtained on two meshes in each coordinate direction, then Eq. (67) is used to obtain the value of the expansion coefficients as follows:

$$\begin{aligned}
a &= [f_{L-1} - f_L]_{MN} / (\Delta x_{L-1}^2 - \Delta x_L^2) \\
b &= [f_{M-1} - f_M]_{LN} / (\Delta y_{M-1}^2 - \Delta y_M^2) \\
c &= [f_{N-1} - f_N]_{LM} / (\Delta z_{N-1}^2 - \Delta z_N^2)
\end{aligned} \tag{68}$$

In the above relations, the fine mesh solution is always used as one of the two solutions. The Richardson extrapolation solution relations for this approach become

$$\begin{aligned}
f_{(L-1)MN,LMN} &= f_{LMN} + \left[\frac{\Delta x_L^2}{\Delta x_{L-1}^2 - \Delta x_L^2}\right][f_L - f_{L-1}]_{MN} = F_e + A_{MN} + O(\Delta x^4) \\
f_{L(M-1)N,LMN} &= f_{LMN} + \left[\frac{\Delta y_M^2}{\Delta y_{M-1}^2 - \Delta y_M^2}\right][f_M - f_{M-1}]_{LN} = F_e + B_{LN} + O(\Delta y^4) \\
f_{LM(N-1),LMN} &= f_{LMN} + \left[\frac{\Delta z_N^2}{\Delta z_{N-1}^2 - \Delta z_N^2}\right][f_N - f_{N-1}]_{LM} = F_e + C_{LM} + O(\Delta z^4)
\end{aligned} \tag{69}$$

For example, as the mesh is refined in the x coordinate direction, the Richardson extrapolated solution always has an error of A_{MN} but it can be used to determine solution errors for this one

coordinate mesh refinement. This Richardson extrapolated result only indicates what value the solution is approaching as $\Delta x \rightarrow 0$ with Δy and Δz held constant. The Richardson extrapolated solution for an improved estimation of the numerical solution as all of the spatial step-sizes go to zero is obtained from

$$f_{RE} = f_{LMN} - a\Delta x_L^2 - b\Delta y_M^2 - c\Delta z_N^2 + \ldots \tag{70}$$

6.3.4 Example 6: Hypersonic Flow Over a Blunt Body

An example of this type of study for a two-dimensional problem is given in Table 6.3. These results are from Blottner [22] for Navier-Stokes equation results for the hypersonic flow over a spherical body. The governing equations are written in generalized coordinates before the numerical scheme is applied. Three mesh refinements in each coordinate direction were performed in this paper. The numerical results of surface heat flux at 90° from the stagnation point is tabulated for each mesh. The error of the heat flux is the quantity to be evaluated with Richardson extrapolation. The first extrapolated results are obtained with mesh refinement in the x direction only and are obtained from the first relation in Eq. (69)

$$f_{02,\,12} = 8.39487 \qquad f_{12,\,22} = 8.41037 \qquad \Delta y = 1/(M-1) = 1/96$$

Further extrapolation of these results is obtained from Eq. (50) which gives the more accurate result

$$f_{02,\,12,\,22} = 8.41140 \qquad \Delta y = 1/96 \tag{71}$$

The percent error of the heat flux for the first 3 meshes in the table is estimated with the above result used as the exact value.

The second extrapolated results are obtained with mesh refinement in the y direction only and is obtained from the second relation in Eq. (69)

$$f_{20,\,21} = 8.41832 \qquad f_{21,\,22} = 8.48805 \qquad \Delta x = 1/(L-1) = 1/88$$

Further extrapolation of these results is obtained from Eq. (50) which gives the more accurate result

$$f_{02,\,12,\,22} = 8.49270 \qquad \Delta x = 1/88 \tag{72}$$

The percent error of the heat flux for the last 3 meshes in the table is estimated with the above result used as the exact value. The error in this case is negative as the numerical results are less than the above extrapolated value. The Richardson extrapolation of heat flux value when the mesh spacing in both coordinate direction is refined is obtained from Eq. (70). The coefficients in this equation are obtained from

$$a = \frac{f_{12} - f_{22}}{\Delta x_1^2 - \Delta x_2^2} = 122.23 \qquad b = \frac{f_{21} - f_{22}}{\Delta y_1^2 - \Delta y_2^2} = -570.50$$

The extrapolated value becomes

$$f_{RE} = 8.42615 - 0.01578 + 0.06190 = 8.47227 \quad (73)$$

The error of the numerical prediction of the heat flux in Table 6.3 can be estimated with the value from Eq. (73) but as the results show in the table, the estimates are erroneous and this approach is inappropriate. For the meshes with refinement in the x coordinate direction, the extrapolated value in Eq. (70) is used to determine the error of the numerical predictions. For the meshes with refinement in the y coordinate direction, the extrapolated value in Eq. (71) is used to determine the error of the numerical predictions.

Table 6.3: Heat Flux on Spherical Nosetip

Mesh	Mesh Size	l	m	Δx_l	Δy_m	Heat Flux kw/m^2	% Error $f_{02,\,12,\,22}$	% Error $f_{20,\,21,\,22}$	% Error f_{RE}
02	23×97	0	2	1/22	1/96	8.70938	3.543		2.799
12	45×97	1	2	1/44	1/96	8.47350	0.738		0.015
22	89×97	2	2	1/88	1/96	8.42615	0.175	-0.784	-0.544
21	89×49	2	1	1/88	1/48	8.24044		-2.970	-2.736
20	89×25	2	0	1/88	1/24	7.70679		-9.254	-9.035

The solution error with mesh refinement in the x coordinate direction is shown in Figure 6.3. When the error is evaluated with the extrapolated value given in Eq. (71), the error has the appropriate second-order behavior when obtained with the finest mesh. The use of the more exact solution as given by Eq. (73), where the mesh is refined in both coordinate directions at the same time, gives an inappropriate estimate of the solution error. The solution error with mesh refinement in the y coordinate direction is shown in Figure 6.4. When the error is evaluated with the extrapolated value given in Eq. (72), the error has the appropriate second-order behavior when obtained with the finest mesh. The use of the more exact solution as given by Eq. (73), where the mesh is refined in both coordinate directions at the same time, gives a reasonable estimate of the solution error in this case. *If you know the exact solution of 2D or 3D problems, the present approach of mesh refinement in one coordinate direction at a time is not appropriate.* The correct approach is to refine the mesh by the same factor in all of the coordinate directions.

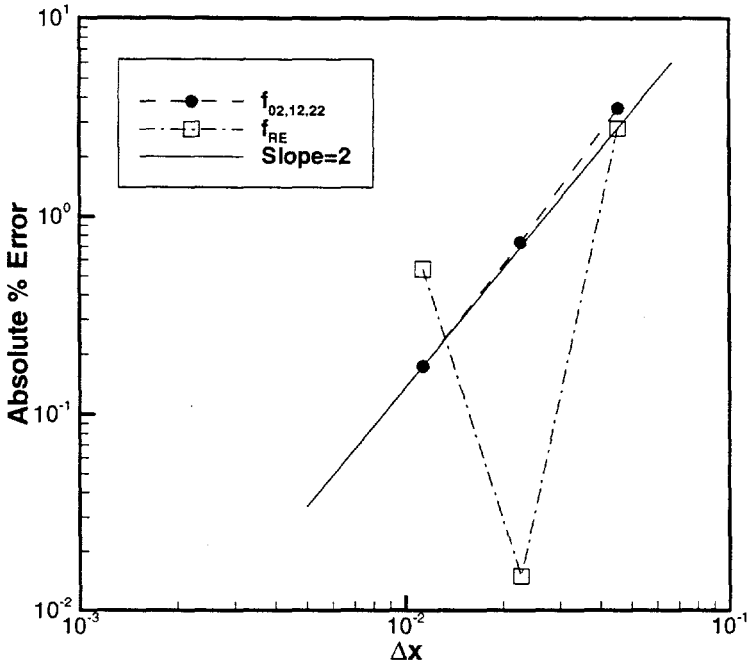


Figure 6.3: Error of Heat Flux for Blunt Body Problem with Mesh Refinement in x Coordinate Direction.

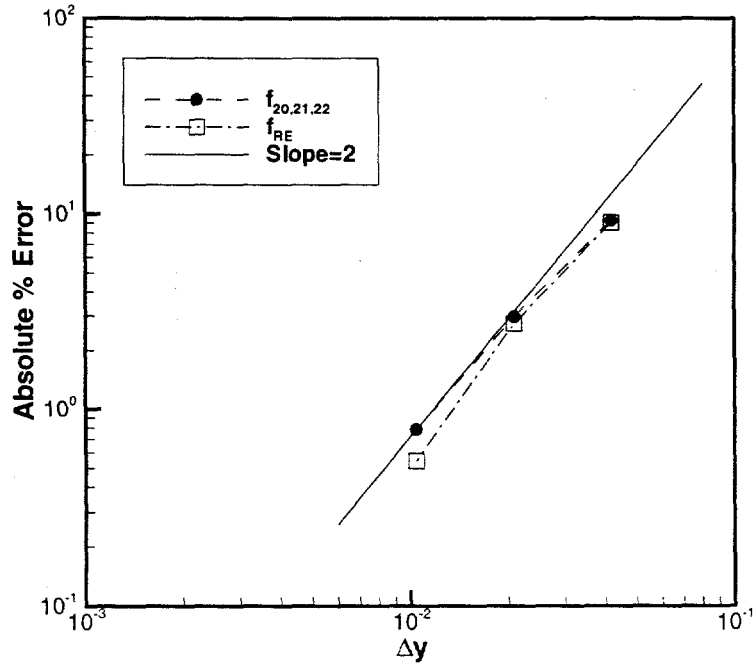


Figure 6.4: Error of Heat Flux for Blunt Body Problem with Mesh Refinement in y Coordinate Direction

6.3.5 Example 7: Transonic Wing Flow

Another investigation on the accuracy of two Navier-Stokes codes has been performed by Bonhaus and Wornom [23]. The codes evaluated are CFL3D and TLNS3D which are cell centered finite volume codes and are believed to be second-order numerical schemes. CFL3D uses an upwind approach while TLNS3D uses a central difference methodology with dissipation terms added to stabilize the numerical scheme. Three-dimensional flow over two wings has been calculated for transonic Mach numbers. The governing equations are the same in both codes where a perfect gas model is used with the Baldwin-Lomax algebraic turbulence model. Solutions have been obtained with four mesh levels as shown in Table 6.4. The mesh in the x and z coordinate directions has been refined by the same factor while a different factor is used in the y coordinate direction. The use of Richardson extrapolation is limited with these mesh solutions as the mesh is not refined the same in the three coordinate directions. The average mesh size for each of the meshes is obtained from

$$h_{avg} = \left(\frac{\Delta x_m}{\Delta x_0} \times \frac{\Delta y_m}{\Delta y_0} \times \frac{\Delta z_m}{\Delta z_0}\right)^{1/3} = (I_0 \times J_0 \times L_0/\text{Cells in Field})^{1/3}$$

and is given in Table 6.5. In the Bonhaus and Wornom paper, the numerical error of various quantities is plotted as a function of the reciprocal of the average mesh size. This is an approximation for the meshes used in this study and is not an appropriate choice.

Table 6.4: Meshes Used for Wing Problem

Mesh m	Mesh $I_m \times J_m \times L_m$	$\frac{\Delta x_m}{\Delta x_0}$	$\frac{\Delta y_m}{\Delta y_0}$	$\frac{\Delta z_m}{\Delta z_0}$
0	96x24x16	1	1	1
1	144x32x24	2/3	3/4	2/3
2	192x48x32	1/2	1/2	1/2
3	288x64x48	1/3	3/8	1/3

In the Bonhaus and Wornom paper the lift, pitching moment and drag coefficients have been tabulated for the two codes and for the four meshes. In the present discussion of this work, the drag coefficient is used to illustrate the accuracy of the results which makes $f = C_D$. The numerical results for the drag coefficient and errors are given in Table 6.5.

Since $\Delta z_m/\Delta z_0 = \Delta x_m/\Delta x_0$, Eq. (62) is written as

$$f_m = F_e + (a+c)\Delta x_0^2\left(\frac{\Delta x_m}{\Delta x_0}\right)^2 + b\Delta y_0^2\left(\frac{\Delta y_m}{\Delta y_0}\right)^2 + \ldots \qquad m = 1, 2, 3 \tag{74}$$

Table 6.5: Numerical Results for Drag Coefficient for Lockheed Wing B

Mesh m	f_m		% Error with F_e		$(h_m)_{avg}$	$(h_m)_{eff}$	
	CFL3D	TLNS3D	CFL3D	TLNS3D		CFL3D	TLNS3D
0	0.02770	0.02937	9.055	14.19	1.0	1.1062	1.0006
1	0.02689	0.02784	5.866	8.243	0.6934	0.6934	0.6934
2	0.02631	0.02685	3.583	4.393	0.5	0.5531	0.5003
3	0.02578	0.02625	1.496	2.061	0.3467	0.3467	0.3467

The above equation is evaluated for the three finest meshes and these equations are solved for the coefficients and exact value which gives

$$(a+c)\Delta x_0^2 = 3(f_1 - f_3) - \frac{81}{64} b\Delta y_0^2 \qquad b\Delta y_0^2 = \frac{256}{17}\left[\frac{5}{12}(f_1 - f_3) - (f_2 - f_3)\right]$$
$$F_e = f_3 - (a+c)\Delta x_0^2\left(\frac{\Delta x_3}{\Delta x_0}\right)^2 - b\Delta y_0^2\left(\frac{\Delta y_3}{\Delta y_0}\right)^2 \tag{75}$$

For the CFL3D code, the results for the parameters in Eq. (75) are

$$(a+c)\Delta x_0^2 = 4.616471 \times 10^{-3} \qquad b\Delta y_0^2 = -1.016471 \times 10^{-3} \qquad F_e = 0.02540 \tag{76}$$

For the TLNS3D code, the results for the parameters in Eq. (75) are

$$(a+c)\Delta x_0^2 = 3.578824 \times 10^{-3} \qquad b\Delta y_0^2 = 9.411765 \times 10^{-4} \qquad F_e = 0.02572 \tag{77}$$

The estimated exact values F_e (extrapolated values) are used to determine the error of the numerical results in Table 6.5. These extrapolated values are different than the values given by Bonhaus and Wornom. This cause of this difference will be discussed subsequently.

Another approach to evaluate the results is to determine the error behavior as the mesh is refined in one coordinate direction at a time. This approach has been developed in Eq. (67) and for the present case is expressed as

$$f_{l3} = F_e + (a+c)\Delta x_l^2 + A_3 + \ldots \qquad l = m = 0, 1, 2, 3$$
$$f_{3m} = F_e + b\Delta y_m^2 + B_3 + \ldots \qquad m = 0, 1, 2, 3 \tag{78}$$

where the following coefficients should be constant if the mesh is refined properly

$$A_3 = b\Delta y_0^2\left(\frac{\Delta y_m}{\Delta y_0}\right)^2 \qquad B_3 = (a+c)\Delta x_0^2\left(\frac{\Delta x_l}{\Delta x_0}\right)^2$$

With the present meshes, these terms are varying and have been determined for the four meshes. When the mesh is refined in the x coordinate direction, the value of $F_e + A_3$ is used as the exact value to determine the solution error. When the mesh is refined in the y coordinate direction, the value of $F_e + B_3$ is used as the exact value to determine the solution error. The results of the one-coordinate mesh refinement behavior for the CFL3D and TLNS3D codes are given in Table 6.6 and Table 6.7, respectively. A plot of these errors is given in Figure 6.5. The error variation indicates the methods are second-order in the x coordinate direction but further mesh refinement in the y coordinate direction is needed to confirm second-order behavior.

Table 6.6: Estimated Error with One-Coordinate Mesh Refinement for CFL3D

Mesh m	F_{lm}	$\frac{\Delta x_m}{\Delta x_0}$	$F_e + A_3$	% Error	$\frac{\Delta y_m}{\Delta y_0}$	$F_e + B_3$	% Error
0	0.02770	1	0.02438	-13.62	1	0.03002	7.728
1	0.02689	2/3	0.02483	-8.296	3/4	0.02745	2.040
2	0.02631	1/2	0.02515	-4.612	1/2	0.02655	0.9040
3	0.02578	1/3	0.02526	-2.059	3/8	0.02591	0.5017

Table 6.7: Estimated Error with One-Coordinate Mesh Refinement for TLNS3D

Mesh m	F_{lm}	$\frac{\Delta x_m}{\Delta x_0}$	$F_e + A_3$	% Error	$\frac{\Delta y_m}{\Delta y_0}$	$F_e + B_3$	% Error
0	0.02937	1	0.02666	10.17	1	0.02930	0.2389
1	0.02784	2/3	0.02625	6.057	3/4	0.02731	1.941
2	0.02685	1/2	0.02596	3.428	1/2	0.02662	0.8640
3	0.02625	1/3	0.02585	1.547	3/8	0.02612	0.4977

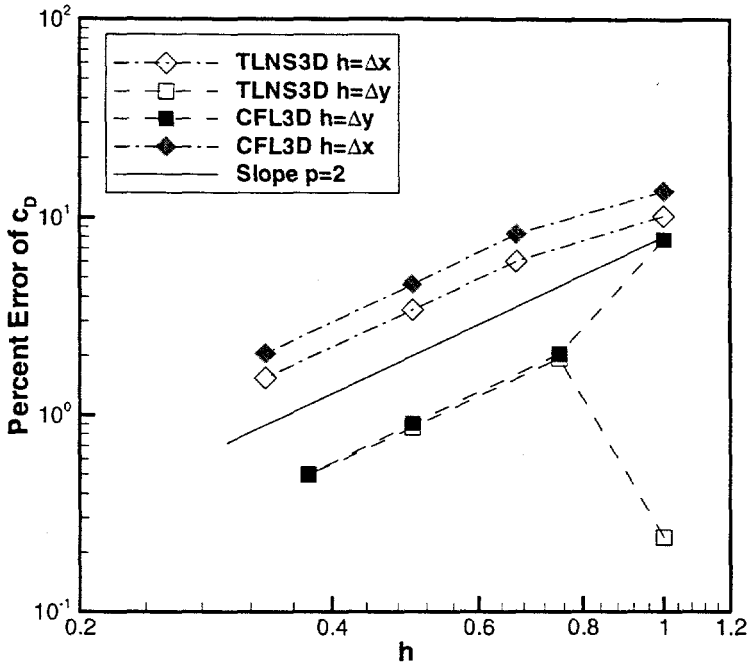


Figure 6.5: Estimated Error of Drag Coefficient with One-Coordinate Mesh Refinement.

As indicated earlier, Bonhaus and Wornom have used h_{avg} in a plot to illustrate the behavior of the errors in Table 6.5 as the mesh is refined. This is illustrated in Figure 6.6. The results for the CFL3D code do not show second-order behavior as the average mesh step size is not the correct mesh size to use. The solution error is correctly expressed as

$$E_m = \frac{f_m - F_e}{F_e} = \alpha_m (h_m)_{eff}^2 + \ldots \qquad \alpha_m = \left[(a+c)\Delta x_0^2 \left(\frac{\Delta x_m}{\Delta x_0}\right)^2 + b\Delta y_0^2 \left(\frac{\Delta y_m}{\Delta y_0}\right)^2 \right] \frac{(h_m)_{eff}^{-2}}{F_e}$$

On the finest mesh ($m = 3$), the value of α_m becomes with the effective step size set equal to the average step size

$$\alpha_3 = 0.01457/(h_3)_{eff}^2 = 0.12123 \quad \text{for CFL3D code} \qquad (h_3)_{eff} = 0.34668$$

$$\alpha_3 = 0.02061/(h_3)_{eff}^2 = 0.17148 \quad \text{for TLNS3D code} \qquad (h_3)_{eff} = 0.34668$$

On the mesh $m = 1$, the effective step size is

$$(h_1)_{eff} = 2(h_3)_{eff} = 0.69336$$

The effective mesh size on the other meshes are obtained from

$$(h_2)_{eff} = \sqrt{E_2/\alpha_3} \qquad (h_0)_{eff} = 2(h_2)_{eff}$$

where E_2 is the sum of the percent errors in each coordinate direction found in Table 6.6 and Table 6.7. The parameter α_3 was used instead of α_2 since these values should be constant. The effective step sizes are given in Table 6.5. The errors given in this table are plotted as a function of the effective step sizes in Figure 6.6. The error of the CFL3D solution has second-order behavior but additional mesh refinement is needed to be sure the higher order terms in Eq. (74) are sufficiently small. It is interesting to note that the two codes do not appear to be approaching the same value for the drag coefficient as the mesh is refined (see Eq. (76) and Eq. (77)). There is a difference of 1.2% between the drag coefficient results from the two codes. Again, further mesh refinement is required to confirm this conclusion.

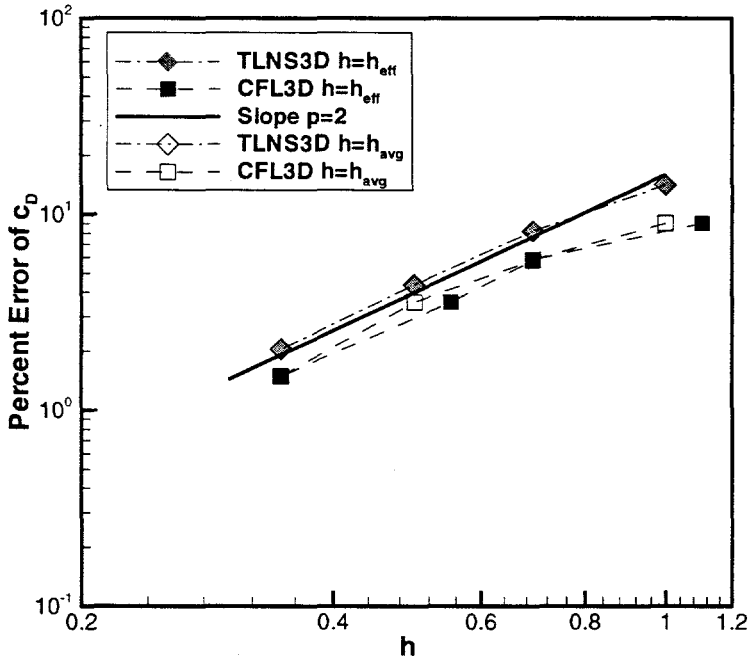


Figure 6.6: Appropriate Mesh Size to Use with Study of Bonhaus and Wornom.

6.4 Appropriate Method of Mesh Modification and Mesh Cell Size

For Cartesian coordinates, there are two ways to obtain a series of co-located meshes with a changing number of nodes as discussed in Section 5.5. With the box scheme, the appropriate mesh modification procedure is the **coarse to fine mesh approach**. With this approach the spacing of the coarse mesh is decreased by adding nodes at the mid-points in the physical coordinate. The next mesh refinement is obtained by adding mesh points at the mid-points between the nodes of the previous mesh in physical space. With this approach, you start with the coarse

mesh and divide each coarse mesh cell into equal size cells as the mesh is refined. The nodes for the coarsest mesh will be co-located with the nodes of the finer meshes. The resulting mesh cells will be of uniform size between each of the nodes of the coarse mesh. The effective mesh cell size for Richardson extrapolation is

$$\begin{aligned} \Delta x_{eff} &= 1/(I-1) \qquad I = \text{Number of nodes along } x \text{ coordinate} \\ \Delta y_{eff} &= 1/(J-1) \qquad J = \text{Number of nodes along } y \text{ coordinate} \\ \Delta z_{eff} &= 1/(K-1) \qquad \text{K} = \text{Number of nodes along } z \text{ coordinate} \end{aligned} \tag{79}$$

The cell size has been normalized by the length of the computational domain in the coordinate direction. This approach is mathematically correct for the Box Scheme Method but has not been justified for other solution approaches on non-uniform meshes.

With the governing equations written in curvilinear coordinates, the resulting difference equations are for a uniform mesh or cell size in the computational coordinate. Mesh modification is obtained with the **fine to coarse mesh approach (agglomeration)**. With this approach, one first obtains the mesh on the finest mesh that is going to be used and then removes every other mesh node to obtain the next coarser mesh. The next coarser mesh is obtained with every other mesh node removed from the previous mesh. The nodes for the coarsest mesh will be co-located with the nodes of the finer meshes. This mesh modification approach can also be obtained from the coarse mesh with the addition of nodes at the mid-points in the computational coordinate where a uniform mesh is always retained. In physical space, the mesh will have a variable spacing. The effective mesh cell size for Richardson extrapolation is again Eq. (79).

The appropriate mesh modification approach for the finite volume method on structured meshes could follow either of the above approaches. The appropriate approach should be determined from the behavior of the local truncation error of the numerical scheme that is being used. More work is required on how Richardson extrapolation is influenced by the mesh modification procedure. It appears that after sufficient mesh refinement, either of the above mesh modification approaches will give solution errors that behave as determined by the order of the difference scheme. However, the coarse mesh solution error behavior could be sensitive to the mesh refinement approach. The effective mesh cell size for Richardson extrapolation should again be Eq. (79).

The appropriate mesh modification approach for the finite volume method on unstructured meshes could also follow the ideas of either of the above approaches. The techniques used in multigrid solution techniques can provide guidance on how to perform mesh modification. In the agglomeration approach, coarse meshes are constructed by fusing together neighboring fine mesh control volumes to form the coarse mesh with a smaller number of larger and more complex control volumes with more surfaces. The appropriateness of multigrid mesh modification approaches to Richardson extrapolation appears to be untested. It is not clear what the effective mesh cell size for Richardson extrapolation is for unstructured meshes. Could it be as simple as the total number of mesh cells in the calculation?

7. Richardson Extrapolation Procedure for Transient Solutions

The solution of transient problems can be improved and the accuracy evaluated with the Richardson extrapolation method. Consider a partial differential equation which is represented in differential or integral form as

$$\frac{\partial F}{\partial t} = R(F) \qquad \text{or} \qquad F(t^N) = \int_0^{t^N} R\,dt \tag{80}$$

where R depends upon F and can contain spatial derivatives and a source term. The dependent variable F is the exact solution of the partial differential equation or integral equation. The Richardson extrapolation method for integration has been described by Fox and Mayers [44]. If the numerical value of the integral is obtained with the trapezoidal rule, the approximation becomes with a uniform interval Δt the following:

$$f^N = \Delta t\left(\frac{1}{2}R^0 + R^1 + R^2 + \ldots + R^{N-1} + \frac{1}{2}R^N\right) \qquad t^n = n\Delta t \qquad n = 0, 1, \ldots, N \tag{81}$$

The superscripts are used to indicate the function is evaluated at the specified time. The true value of the integral and the error with the trapezoidal rule ($p = 2$) are

$$f^N = F(t^N) + Error \qquad Error = f^N - F(t^N) = \alpha\Delta t^p + \ldots \tag{82}$$

The integral can be evaluated with step-sizes Δt_0 and Δt_1 which gives

$$\begin{aligned} f^{N_0} &= F(t^N) + \alpha\Delta t_0^p + \ldots \qquad & t^N &= N_0\Delta t_0 \\ f^{N_1} &= F(t^N) + \alpha\Delta t_1^p + \ldots \qquad & t^N &= N_1\Delta t_1 \end{aligned} \tag{83}$$

A more accurate estimate of the integral is obtained with Richardson extrapolation which gives

$$f^{N_0, N_1} = f^{N_1} + \left(\frac{\Delta t_1^p}{\Delta t_0^p - \Delta t_1^p}\right)[f^{N_1} - f^{N_0}] \quad \text{if } \Delta t_1 = \frac{1}{2}\Delta t_0, \left(\frac{\Delta t_1^p}{\Delta t_1^p - \Delta t_0^p}\right) = \frac{1}{2^p - 1} \tag{84}$$

When the numerical solution is obtained from the difference Eq. (80), the governing difference equation is expressed as

$$\left(\frac{f^{n+1} - f^n}{\Delta t}\right) - [\theta R^{n+1} + (1-\theta)R^n] = 0 \tag{85}$$

with θ varying as follows:

$\theta = 0$ Explicit (first-order $p = 1$)

$\theta = 1/2$ Crank-Nicolson (second-order $p = 2$)

$\theta = 1$ Implicit (first-order $p = 1$)

The variable f is the exact solution of the difference equation and will be different than the exact solution F of the differential equation. At time $t^N = N\Delta t$, the relation between f and F becomes as before

$$f^N = F(t^N) + Error \tag{86}$$

The solution F at different times is obtained from Taylor series expansions which gives the difference relation expansion for the time derivative

$$\Delta_0 = \frac{F^{n+1} - F^n}{\Delta t} = \left(\frac{\partial F}{\partial t}\right)^{n+\frac{1}{2}} + \tau_0 \qquad \tau_0 = \frac{1}{24}\left(\frac{\partial^3 F}{\partial t^3}\right)^{n+\frac{1}{2}} \Delta t^2 + \ldots \tag{87}$$

The right side of Eq. (80) is also evaluated with Taylor series expansions which gives

$$\begin{aligned} \Delta_R &= [\theta R^{n+1} + (1-\theta)R^n] = (R)^{n+\frac{1}{2}} + \tau_R \\ \tau_R &= \left(\theta - \frac{1}{2}\right)\left(\frac{\partial R}{\partial t}\right)^{n+\frac{1}{2}} \Delta t + \frac{1}{8}\left(\frac{\partial^2 R}{\partial t^2}\right)^{n+\frac{1}{2}} \Delta t^2 + \ldots \end{aligned} \tag{88}$$

When R is completely evaluated, there are spatial errors that have not been included in this analysis. The amount the exact solution F does not satisfy the difference Eq. (85) is the *local truncation error.* For the present case with the use of Eq. (87) and Eq. (88), the local truncation error becomes

$$\begin{aligned} \Delta^n &= \frac{F^{n+1} + F^n}{\Delta t} - [\theta R^{n+1} + (1-\theta)R^n] = \left(\frac{\partial F}{\partial t}\right)^{n+\frac{1}{2}} - (R)^{n+\frac{1}{2}} + \tau_0 - \tau_R \\ \tau^n &= \tau_0 - \tau_R = \left(\theta - \frac{1}{2}\right)\gamma\Delta t + \delta\Delta t^2 + \ldots \end{aligned} \tag{89}$$

where

$$\gamma = -\left(\frac{\partial R}{\partial t}\right)^{n+\frac{1}{2}} \qquad \delta = -\frac{1}{8}\left(\frac{\partial^2 R}{\partial t^2}\right)^{n+\frac{1}{2}} + \frac{1}{24}\left(\frac{\partial^3 F}{\partial t^3}\right)^{n+\frac{1}{2}}$$

The exact solution at any time step is obtained from Eq. (89) which gives

$$F^{n+1} = F^n + \Delta t[\theta R^{n+1} + (1-\theta)R^n] + \Delta t\tau^n \qquad 0 \le n \le N-1 \tag{90}$$

It is assumed that γ and δ are constant as this allows the solution behavior to be readily evaluated. The exact solution at time $t = t^N$ with uniform time steps Δt becomes

$$F(t^N) = F^N = F^0 + \Delta t \sum_{n=1}^{N-1} [\theta R^{n+1} + (1-\theta)R^n] + t^N\left[\left(\theta - \frac{1}{2}\right)\gamma\Delta t + \delta\Delta t^2 + \ldots\right] \tag{91}$$

while the numerical solution at the same time is obtained from Eq. (85) and becomes

$$f^N = f^0 + \Delta t \sum_{n=1}^{N-1} [\theta R^{n+1} + (1-\theta)R^n] \tag{92}$$

The initial conditions for the exact and numerical solutions are the same, so that $f^0 = F^0$.

As $\Delta t \to 0$ and $N \to \infty$, the local truncation error goes to zero and the exact solution becomes

$$F^N = F^0 + \Delta t \sum_{n=1}^{N-1} [\theta R^{n+1} + (1-\theta)R^n] \tag{93}$$

while the solution with finite Δt is expressed as Eq. (86) which gives

$$f^N = F(t^N) + Error \qquad Error = -t^N\left[\left(\theta - \frac{1}{2}\right)\gamma\Delta t + \delta\Delta t^2 + \ldots\right] \tag{94}$$

The above relation shows that the various methods given in Eq. (85) have the order of accuracy as indicated. The form of Eq. (94) is the same as Eq. (46) and the ideas presented for Richardson extrapolation for steady-state solutions are applicable to the transient problem. For transient solutions, the solutions are obtained with several values of Δt and then the exact solution can be estimated with Richardson extrapolation as discussed previously or as illustrated in Eq. (84).

7.1 Example 8: Unsteady Problem

The governing equation has been given previously in Eq. (2) and is

$$L(F) = \frac{\partial F}{\partial t} - \frac{\partial^2 F}{\partial x^2} - S(x) = 0 \qquad S(x) = 0$$

with initial and boundary conditions

$$F(0,x) = F_0 \sin s\pi x \qquad F(t,0) = F(t,1) = 0 \qquad 0 \le t \le \infty \qquad 0 \le x \le 1$$

The parameter $s = 1, 2, \ldots$ and will be later set to one. If the initial condition is of the form $F(0, x) = F(x)$, then a Fourier series approach can be used to approximate the function $F(x)$ with an infinite series involving the s parameter.

The **exact solution of the PDE** is

$$F(t, x) = F_0 e^{-(s\pi)^2 t} \sin(s\pi x) \quad \text{or} \quad F_i^n = F_0 e^{-(s\pi)^2 t^n} \sin(s\pi x_i) \tag{95}$$

where the mesh point location and time are given by $x_i = i\Delta x$ and $t^n = n\Delta t$. An explicit ($\theta = 0$) or implicit ($\theta = 1$) numerical scheme is used for this problem which gives the following difference equation:

$$\Delta(f_i^n) = \left(\frac{f^{n+1} - f^n}{\Delta t}\right)_i - (1-\theta)\left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right)^n - \theta\left(\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}\right)^{n+1} = 0$$

The above difference equation assumes that $S(x) = 0$ and is rewritten with $\sigma = \Delta t/\Delta x^2$ as

$$(f_i^{n+1} - f_i^n) - (1-\theta)\sigma(f_{i+1} - 2f_i + f_{i-1})^n - \theta\sigma(f_{i+1} - 2f_i + f_{i-1})^{n+1} = 0 \tag{96}$$

The difference equation is solved with the separation of variables approach which uses

$$f_i^n = u^n v_i \tag{97}$$

where u^n is independent of x_i and v_i is independent of t^n. The substitution of Eq. (97) into Eq. (96) gives the relations

$$\frac{u^{n+1}}{u^n} = \frac{\sigma v_{i+1} + (1-2\sigma)v_i + \sigma v_{i-1}}{v_i} = \lambda_\theta \qquad \text{when } \theta = 0$$

$$-\frac{u^n}{u^{n+1}} = \frac{\sigma v_{i+1} - (1+2\sigma)v_i + \sigma v_{i-1}}{v_i} = -\lambda_\theta \qquad \text{when } \theta = 1$$

Since λ_θ must be a constant, the above is written as two difference equations for the new variables

$$u^{n+1} - \lambda_\theta u^n = 0 \qquad \sigma v_{i+1} + (1 - 2\sigma - \lambda_\theta)v_i + \sigma v_{i-1} = 0 \qquad \theta = 0$$

$$u^{n+1} - \left(\frac{1}{\lambda_\theta}\right)u^n = 0 \qquad \sigma v_{i+1} - (1 + 2\sigma - \lambda_\theta)v_i + \sigma v_{i-1} = 0 \qquad \theta = 1$$

The solution for u^n is

$$u^n = C_0(\lambda_\theta)^n \qquad \lambda_0 = 1 - 2\sigma(1 - \cos\alpha) = 1 - 4\sigma\sin^2(\alpha/2) \qquad \theta = 0$$

$$u^n = C_0(\lambda_\theta)^{-n} \qquad \lambda_1 = 1 + 2\sigma(1 - \cos\alpha) = 1 + 4\sigma\sin^2(\alpha/2) \qquad \theta = 1$$

The second difference equation is rewritten for both cases of θ as

$$v_{i+1} - 2v_i\cos\alpha + v_{i-1} = 0 \qquad \begin{aligned} 1 - 2\sigma - \lambda_0 &= -2\sigma\cos\alpha \qquad \theta = 0 \\ 1 + 2\sigma - \lambda_1 &= 2\sigma\cos\alpha \qquad \theta = 1 \end{aligned}$$

with the solution

$$v_i = C_1\sin(i\alpha) + C_2\cos(i\alpha) \qquad \text{when } \theta = 0 \text{ or } \theta = 1$$

The solutions for u^n and v_i are combined to obtain the difference solution that satisfies the initial conditions and boundary conditions which requires $\alpha = s\pi\Delta x$. The **exact solution to the difference equation** is

$$\begin{aligned} f_i^n &= F_0(\lambda_0)^n\sin(i\alpha) = F_0\left[1 - 4\sigma\sin^2\left(\frac{s\pi\Delta x}{2}\right)\right]^{t^n/\Delta t}\sin(s\pi x_i) \qquad \theta = 0 \\ f_i^n &= F_0(\lambda_1)^{-n}\sin(i\alpha) = F_0\left[1 + 4\sigma\sin^2\left(\frac{s\pi\Delta x}{2}\right)\right]^{-t^n/\Delta t}\sin(s\pi x_i) \qquad \theta = 1 \end{aligned} \tag{98}$$

where $C_2 = 0$ and $C_0C_1 = F_0$. When the exact difference solutions in Eq. (98) are compared to the PDE solution in Eq. (95), these solutions appear to be very different. For small values of Δx, the time dependent term is written as

$$\left[1 \pm 4\sigma\sin^2\left(\frac{s\pi\Delta x}{2}\right)\right]^{t^n/\Delta t} = \exp\left\{\frac{t^n}{\sigma\Delta x^2}\ln\left[1 \pm 4\sigma\sin^2\left(\frac{s\pi\Delta x}{2}\right)\right]\right\}$$

With Δx very small, the above is expanded to obtain the difference solution as

$$\begin{aligned} f_i^n &= \left[1 - \frac{1}{2}(s\pi)^4\Delta x^2\left(\sigma - \frac{1}{6}\right)t^n + O(\Delta x^4)\right]F_i^n \qquad \theta = 0 \\ f_i^n &= \left[1 + \frac{1}{2}(s\pi)^4\Delta x^2\left(\sigma + \frac{1}{6}\right)t^n + O(\Delta x^4)\right]F_i^n \qquad \theta = 1 \end{aligned} \tag{99}$$

With σ and t^n held fixed and $\Delta x \to 0$, the above solution approaches the solution as given in Eq. (95). Unstable solutions which oscillate with exponentially increasing amplitude are avoided when $\theta = 0$ with the explicit scheme if

$$4\sigma\sin^2\left(\frac{s\pi\Delta x}{2}\right) \le 1 \qquad \sigma \le 1/2 \qquad \Delta t \le \frac{1}{2}\Delta x^2$$

The **discretization error** for this example with $\theta = 0$ or 1 is

$$e_i^n = f_i^n - F_i^n = (s\pi)^4 t^n\left[\frac{1}{2}(2\theta - 1)\Delta t + \frac{1}{12}\Delta x^2 + O(\Delta t^2) + O(\Delta x^4)\right]F_i^n \tag{100}$$

The method is first-order in time and second-order in space. The discretization error in terms of the **local truncation error** for this example can be obtained from Eq. (5) which gives

$$e_i^n = -t^n \tau_i^n \qquad \tau_i^n = -(s\pi)^4 \left[\frac{1}{2}(2\theta - 1)\Delta t + \frac{1}{12}\Delta x^2 + O(\Delta t^2) + O(\Delta x^4) \right] F_i^n$$

The difference equation discretization error needs to be evaluated from the numerical solutions. The appropriate procedures are illustrated by considering the solution at $t = 0.1$ and $x = 0.5$ with $s = 1$. The exact solution of the PDE at this time and location is obtained from Eq. (95) and is

$$F(0.1, 0.5)/F_0 = 0.3727078 \tag{101}$$

The difference equation discretization error can be evaluated by fixing the spatial mesh step-size $\Delta x = 0.1$ and then obtaining solutions with different time steps. This approach is shown in Table 7.1 for the two difference schemes. As the time step is halved, the percent error is reduced by a factor of 2 for both schemes and this indicates the temporal scheme is first-order. Richardson extrapolation is used to estimate the exact solution to determine the percent error of the numerical results shown in Table 7.1. Since the temporal difference scheme is first-order, the appropriate relation and result from the explicit and implicit schemes is

$$f^{23}/F_0 = [f^3 + (f^3 - f^2)]/F_0 = 0.3757332$$

The discretization error as given by Eq. (100) and with $\Delta t = 0$ it is used to estimate the exact difference solution with $\Delta x = 0.1$ which gives

$$f_i^n/F_0 = \left[1 + \frac{1}{12}(s\pi)^4 t^n \Delta x^2 + O(\Delta x^4) \right] (F_i^n/F_0) = 0.3757332$$

The explicit scheme error is negative and has the same magnitude as the error of the implicit scheme. The explicit scheme gives a solution that is smaller than the exact solution while the implicit scheme gives a solution that is larger than the exact solution. These two solutions provide a technique to obtain an upper and lower bound on the solution with a specified spatial step-size.

Table 7.1: Difference Solution and Error with $\Delta x = 0.1$

Temporal Mesh	Δt	Explicit		Implicit	
		f_i^n/F_0	$\% Er$	f_i^n/F_0	$\% Er$
0	4×10^{-3}	0.3684722	−1.933	0.3829943	1.933
1	2×10^{-3}	0.3721027	−0.9662	0.3793637	0.9662
2	1×10^{-3}	0.3739180	−0.4831	0.3775485	0.4831
3	5×10^{-4}	0.3748256	−0.2416	0.3766409	0.2416

The difference equation discretization error can be evaluated by fixing the temporal mesh step-size $\Delta t = 0.0001$ and then obtaining solutions with different spatial step-sizes at $t = 0.1$ and $x = 0.5$. This approach is shown in Table 7.2 for the two difference schemes. As the spatial step is halved, the percent error is reduced by a factor of 4 for both schemes and this indicates the spatial scheme is second-order. Richardson extrapolation is used to estimate the exact solution to determine the percent error of the numerical results shown in Table 7.2. Since the spatial difference scheme is second-order, the appropriate relation and result from the explicit and implicit schemes is

$$f_{12}/F_0 = \left[f_2 + \frac{1}{3}(f_2 - f_1)\right]/F_0 = \begin{cases} 0.3725262 & \text{explicit scheme} \\ 0.3728893 & \text{implicit scheme} \end{cases}$$

The discretization error as given by Eq. (100) and with $\Delta x = 0$ is used to estimate the exact solution with $\Delta t = 0.0001$ which gives

$$f_i^n/F_0 = \left[1 \mp \frac{1}{2}(s\pi)^4 t^n \Delta t + O(\Delta t^2)\right](F_i^n/F_0) = \begin{cases} 0.3725263 & \text{explicit scheme} \\ 0.3728893 & \text{implicit scheme} \end{cases}$$

The explicit and implicit errors are nearly the same.

Table 7.2: Difference Solution and Error with $\Delta t = 0.0001$

Spatial Mesh	Δx	Explicit		Implicit	
		f_i^n/F_0	$\%\ Er$	f_i^n/F_0	$\%\ Er$
0	0.1	0.3755517	0.8121	0.3759148	0.8114
1	0.05	0.3732826	0.2032	0.3736457	0.2029
2	0.025	0.3727154	0.05076	0.3730784	0.05071

The difference equation discretization error can be evaluated by refining both the temporal and spatial meshes simultaneously. For the present difference scheme, the temporal mesh is refined by a factor of 4 while the spatial mesh is refined by a factor of 2. This makes the total solution error decrease by a factor of 4 as the mesh is refined and gives an effective second-order behavior. This approach is illustrated in Table 7.3 at $t = 0.1$ and $x = 0.5$. Since the difference scheme acts as a second-order method, the appropriate Richardson relation and results from the explicit and implicit schemes are

$$f_{23}/F_0 = \left[f_3 + \frac{1}{3}(f_3 - f_2)\right]/F_0 = \begin{cases} 0.3727078 & \text{explicit scheme} \\ 0.3727077 & \text{implicit scheme} \end{cases}$$

With this approach, the exact solution that should be used to determine the discretization error is the exact solution of the PDE which is given in Eq. (101) and this value is very close to the above

values. It should remembered that the implicit scheme is numerically stable while the explicit scheme becomes unstable if the time step is too large, $\Delta t/\Delta x^2 \leq 0.5$. For the second-ordered schemes and for very accurate spatial solutions, the explicit scheme requires very small time steps while the implicit scheme has no time step restrictions.

Table 7.3: Difference Solution and Error with Temporal and Spatial Refinement

Solution Mesh	Δt	Δx	Explicit		Implicit	
			f_i^n/F_0	$\%\ Er$	f_i^n/F_0	$\%\ Er$
0	4×10^{-3}	0.2	0.3775485	1.299	0.3920705	5.195
1	1×10^{-3}	0.1	0.3739180	0.3247	0.3775485	1.299
2	2.5×10^{-4}	0.05	0.3730103	0.08116	0.3739180	0.3247
3	6.25×10^{-5}	0.025	0.3727834	0.02028	0.3730103	0.08116

8. Conclusions and Future Work

The necessity and importance of mesh refinement studies for code and calculation verification has been discussed. A review of previous work on determining solution accuracy with different governing equation formulations and meshes has been presented. It is demonstrated that Richardson extrapolation is the appropriate procedure for estimating solution error for structured meshes. The evaluation of local truncation error is an important part of verification as it determines the consistency of the numerical scheme, the order of the numerical scheme, and the restrictions on mesh variation with a non-uniform mesh.

Mesh generation is an important part of the verification process as a series of appropriately refined meshes are required for use with Richardson extrapolation. The series of meshes must have co-located nodes in order to perform Richardson extrapolation at all the nodes of the coarsest mesh. A non-uniform mesh is usually necessary in order to reduce the total number of mesh cells required while obtaining a solution error that is nearly uniform throughout the computational domain. The mesh spatial variation must be sufficiently smooth such that the order of accuracy of the numerical scheme does not change from the order of accuracy on a uniform mesh. For structured non-uniform meshes, the refinement approach must be consistent with the numerical scheme such that an appropriate average cell size in each coordinate direction can be obtained. For unstructured meshes, further work is required to determine the appropriate mesh refinement procedure and the appropriate average cell size to use with Richardson extrapolation.

The local truncation error for a steady-state diffusion type partial differential equation with a one-dimensional non-uniform mesh has been investigated. Formal second-order accuracy can be obtained for some governing equation formulations and numerical difference schemes. However, other governing equation formulations and numerical difference schemes show formal first-order accuracy. For practical application, all of the approaches demonstrate asymptotic second-order behavior for the numerical examples as the spatial mesh is refined. The approach to obtain the local truncation error of the governing equation written in integral form is also described.

The Richardson extrapolation procedure for steady-state problems with multi-mesh refinements and for multi-dimensional problems is documented. The de Vahl Davis approach to determine numerical scheme order and solution error from mesh refinement results is also described. Four examples are presented where the foregoing procedures are illustrated. The Richardson extrapolation procedure for transient problems is discussed and illustrated with an example problem with exact analytical and numerical solutions.

There are several items that have not been investigated and need further study:

1. A capacity to obtain the local truncation error for new numerical schemes is required. A symbolic manipulation code, such as Mathematica, is needed to properly analyze the local truncation error for 2D and 3D difference schemes.
2. The influence of non-uniform meshes on discretization error or solution accuracy of numerical schemes of current interest need to be evaluated.
3. Investigate and develop a technique to determine the consistency of difference schemes with a numerical method rather than using the usual analytical evaluation of the truncation error.
4. Determine how mesh refinement studies can be quantitatively used in the evaluation of the solution properties of the governing equations. Are the governing equation well conditioned with a unique solution?

References

[1] L. F. Richardson, "The Approximate Arithmetic Solution by Finite Differences of Physical Problems Involving Differential Equations, With Applications to the Stresses in a Masonry Dam," *Philosophical Transactions of the Royal Society of London*, Series A, Vol. 210, pp. 307-357, 1910.

[2] L. F. Richardson and J. A. Gaunt, "The Deferred Approach to the Limit," *Transactions of the Royal Society of London*, Series A, Vol. 226, pp. 299-361, 1927.

[3] D. C. Joyce, "Survey of Extrapolation Processes in Numerical Analysis," *SIAM Review*, Vol. 13, pp. 435-490, October 1971.

[4] Patrick J. Roache and Patrick M. Knupp, "Completed Richardson Extrapolation," Communications in Numerical Methods in Engineering, Vol. 9, pp. 365-374, 1993.

[5] Robert D. Richtmyer and K. W. Morton, *Difference Methods for Initial-Value Problems*, Interscience Publishers, Division of John Wiley & Sons, New York, 1967.

[6] Eli Turkel, "Accuracy of Schemes with Nonuniform Meshes for Compressible Flows," *Applied Numerical Mathematics*, Vol. 2, pp. 529-550, 1986. Also ICASE Report No. 85-43.

[7] E. Turkel, S. Yaniv, and U. Landau, "Accuracy of Schemes for the Euler Equations with Non-Uniform Meshes," AIAA Paper 86-0341 and ICASE Report No. 85-59.

[8] A. Jameson, W. Schmidt, and E. Turkel, "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA-81-1259, AIAA 14th Fluid and Plasma Dynamics Conference, June 23-25, 1981.

[9] Stanley G. Rubin and John C. Tannehill, "Parabolized/Reduced Computational Techniques," *Annual Review of Fluid Mechanics*, Vol. 24, pp. 117-144, 1992.

[10] F. R. Menter, "Eddy Viscosity Transport Equations and Their Relation to the $k-\varepsilon$ Model," *Journal of Fluids Engineering*, Vol. 119, pp. 876-884, Dec. 1997.

[11] F. R. Menter, H. Grotjans, and F. Unger, "Numerical Aspects of Turbulence Modeling for the Reynolds Averaged Navier-Stokes Equations," *Computational Fluid Dynamics*, von Karman Institute for Fluid Dynamics Lecture Series 1997-02, March 3-7, 1997.

[12] Patrick J. Roache, *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, PO Box 9110, Albuquerque, New Mexico 87119-9110, 1998.

[13] J. C. Cooke and K. W. Mangler, "The Numerical Solution of the Laminar Boundary Layer Equations for an Ideal gas in Two and Three Dimensions," Royal Aircraft Establishment Technical Memorandum Aero 999, August 1967. Also presented at AGARD Seminar on

Numerical Methods for Viscous Flows, National Physical Laboratory, Teddington, England, Sept. 18-21, 1967.

[14] F. G. Blottner, "Investigation of Some Finite-Difference Techniques for Solving the Boundary Layer Equations," *Computer Methods in Applied Mechanics and Engineering,* Vol. 6, pp. 1-30, 1975.

[15] G. de Vahl Davis, "Natural Convection of Air in a Square Cavity: A Bench Mark Numerical Solution," *International Journal for Numerical methods in Fluids*, Vol. 3, pp. 249-264, 1983.

[16] H. B. Keller and T. Cebeci, "Accurate Numerical Methods for Boundary Layers, I: Two-Dimensional Laminar Flows," In Lecture Notes in Physics, Proceedings of Second International Conference on Numerical Methods in Fluid Dynamics, Springer-Verlag, Berlin, 1971.

[17] Herbert B. Keller and Tuncer Cebeci, "Accurate Numerical Methods for Boundary Layers, II: Two-Dimensional Turbulent Flows," *AIAA Journal*, Vol. 1, September 1972.

[18] M. Hortmann, M. Peric, and G. Scheuerer, "Finite Volume Multigrid Predictions of Laminar Natural Convection: Bench-Mark Solutions," *International Journal for Numerical methods in Fluids*, Vol. 11, pp. 189-207, 1990.

[19] Ismail Celik and Wei-Ming Zhang, "Calculation of Numerical Uncertainty Using Richardson Extrapolation: Application to Some Simple Turbulent Flow Calculations," *Journal of Fluids Engineering,* Vol. 117, pp. 439-445, Sept. 1995.

[20] Ismail Celik and Ozgur Karatekin, "Numerical Experiments on Application of Richardson Extrapolation With Nonuniform Grids," *Journal of Fluids Engineering,* Vol. 119, pp. 584-590, Sept. 1997.

[21] Joe F. Thompson, Z. U. A. Warsi, and C. Wayne Mastin, *Numerical Grid Generation Foundations and Applications*, North-Holland, New York, NY, 1985.

[22] F. G. Blottner, "Accurate Navier-Stokes Results for the Hypersonic Flow Over a Spherical Nosetip," *Journal of Spacecraft and Rockets*, Vol. 27, pp. 113-122, March-April, 1990.

[23] Daryl L. Bonhaus and Stephen F. Wornom, "Relative Efficiency and Accuracy of Two Navier-Stokes Codes for Simulating Attached Transonic Flow Over Wings," Paper at AIAA 8th Applied Aerodynamics Conference, Paper No. AIAA-90-3078-CP, p. 713, August 20-22, 1990.

[24] D. W. Zingg, "Viscous Airfoil Computations Using Richardson Extrapolation," AIAA 91-1559-CP, AIAA 10th Computational Fluid dynamics Conference, June 1991.

[25] D. W. Zingg, "Grid Studies for Thin-Layer Navier-Stokes Computations of Airfoil Flowfields," *AIAA Journal*, Vol. 30, pp. 2561-2564, 1992.

[26] D. J. Hall and D. W. Zingg, "Viscous Airfoil Computations Using Adaptive Grid Redistribution," *AIAA Journal*, Vol. 33, pp. 1205-1210, July 1995.

[27] D. Lee and Y. M. Tsuei, "A Formula for Estimation of Truncation Errors of Convection Terms in a Curvilinear Coordinate System," *Journal of Computational Physics*, Vol. 98, pp. 90-100, 1992.

[28] Ron-Ho Ni, "A Multiple-Grid Scheme for Solving the Euler Equations," *AIAA Journal*, Vol. 20, pp. 1565-1571, 1982.

[29] P. L. Roe, "Error Estimates for Cell-Vertex Solutions of the Compressible Euler Equations," ICASE Report 87-6, January 1987.

[30] Michael B. Giles, "Accuracy of Node-Based Solutions on Irregular Meshes," 11th International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics 323, Editors D. L. Dwoyer, M. Y. Hussaini, and R. G. Voigt, Springer-Verlag, 1988.

[31] Dana R. Lindquist and Michael B. Giles, "A Comparison of Numerical Schemes on Triangular and Quadrilateral Meshes," 11th International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics 323, Editors D. L. Dwoyer, M. Y. Hussaini, and R. G. Voigt, Springer-Verlag, 1988.

[32] K. W. Morton and M. F. Paisley, "A Finite Volume Scheme with Shock Fitting for the Steady Euler Equations," Journal of Computational Physics, Vol. 80, pp. 168-203, 1989.

[33] Yih Nen Jeng and Jiann Lin Chen, "Truncation Error Analysis of the Finite Volume Method for a Model Steady Convective Equation," *Journal of Computational Physics*, Vol. 100, pp. 64-76, 1992.

[34] David W. Zingg and Harvard Lomax, "Finite-Difference Schemes on Regular Triangular Grids," *Journal of Computational Physics*, Vol. 108, pp. 306-313, 1993.

[35] Yao-Hsin Hwang, "Stability and Accuracy Analyses for the Incompressible Navier-Stokes Equations on the Stagged Triangular Grid," *Numerical Heat Transfer, Part B*, Vol. 32, pp. 321-349, 1997.

[36] K. W. Morton, *Numerical Solution of Convection-Diffusion Problems*, Chapman & Hall, 2-6 Boundary Row, London SE1 8HN, United Kingdom, 1996.

[37] George Boole, *A Treatise on the Calculus of Finite Differences*, 2nd edition, Dover Publication Inc., New York, 1960. (first edition published in 1860)

[38] F. B. Hildebrand, *Methods of Applied Mathematics*, Prentice-Hall, Inc., Engle Cliffs, NJ, 1952.

[39] Francis B. Hildebrand, *Finite-Differences Equations and Simulation*, Prentice-Hall, Inc., Engle Cliffs, NJ, 1968.

[40] Joel H. Ferziger and Milovan Peric, "Further Discussion of Numerical Errors in CFD," *International Journal for Numerical Methods in Fluids*, Vol. 23, pp. 1263-1274, 1996.

[41] F. G. Blottner, "Variable Grid Scheme for Discontinuous Grid Spacing and Derivatives," *Computers and Fluids*, Vol. 8, pp. 421-434, 1980.

[42] Frederick G. Blottner, "Nonuniform grid Method for Turbulent Boundary Layers," in Proceedings of the Fourth International Conference on Numerical Methods in Fluid Dynamics, Edited by Robert D. Richtmyer, Springer-Verlag, 1974.

[43] F. G. Blottner, "Variable Grid Scheme Applied to Turbulent Boundary Layers," *Computer Methods in Applied Mechanics and Engineering*, Vol. 4, pp. 179-194, 1974.

[44] L. Fox and D. F. Mayers, *Computing Methods for Scientists and Engineers*, Clarendon Press, Oxford University Press, London, 1968.