9
Solution adaptation

The previous chapter focused on the estimation of numerical errors and uncertainties due to the discretization. In addition to estimating the discretization error, we also desire methods for reducing it when either it is found to be too large or when the solutions are not yet in the asymptotic range and therefore the error estimates are not reliable. Applying systematic mesh refinement, although required for assessing the reliability of all discretization error estimation approaches, is not the most efficient method for reducing the discretization error. Since systematic refinement, by definition, refines by the same factor over the entire domain, it generally results in meshes with highly-refined cells or elements in regions where they are not needed. Recall that for 3-D scientific computing applications, each time the mesh is refined using grid halving (a refinement factor of two), the number of cells/elements increases by a factor of eight. Thus systematic refinement for reducing discretization error can be prohibitively expensive.

Targeted, local solution adaptation is a much better strategy for *reducing* the discretization error. After a discussion of factors affecting the discretization error, this chapter then addresses the two main aspects of solution adaptation:

(1) methods for determining which regions should be adapted and
(2) methods for accomplishing the adaption.

This chapter concludes with a comparison of different approaches for driving mesh adaptation for a simple 1-D scalar problem.

9.1 Factors affecting the discretization error

There are three factors that affect the discretization error for uniform meshes: the chosen discretization scheme, the mesh resolution, and the local solution behavior (including solution derivatives). For nonuniform meshes, the mesh quality also plays a role. This section begins with a review of the relationship between the discretization error and the truncation error that was initially developed in Chapter 5. Then the truncation error is examined for a simple finite difference numerical scheme applied to the 1-D Burgers' equation; first on a uniform mesh and then on a nonuniform mesh using a global transformation. The truncation error developed in transformed coordinates clearly shows the interplay between mesh resolution, mesh quality, and the solution behavior for the chosen discretization scheme. Implications for anisotropic mesh adaption are also discussed.

9.1.1 Relating discretization error to truncation error

Recall that in Chapter 8 we derived both the continuous discretization error transport equation (Eq. (8.30)) and the discrete discretization error transport equation (Eq. (8.32)). These equations are repeated here for convenience

$$L(\varepsilon_h) = -TE_h(u_h), \tag{9.1}$$

$$L_h(\varepsilon_h) = -TE_h(\tilde{u}), \tag{9.2}$$

where $L(\cdot)$ is the original (continuous) mathematical operator and $L_h(\cdot)$ is the discrete operator. These error transport equations can only be derived if the operators are linear or have been linearized. These two equations show that the discretization error can be both locally generated by the truncation error and also transported from other regions of the domain. Transport error is often called pollution error by the finite element community. Since the truncation error serves as the local source in the discretization error transport equations, reducing the truncation error should result in a commensurate reduction in the discretization error. Furthermore, any solution adaptation scheme should adapt only on the locally-generated error and not on the error transported from other regions in the domain.

9.1.2 1-D truncation error analysis on uniform meshes

The process of examining the truncation error first requires that a mathematical model be defined along with its discrete approximation. Consider the steady-state form of Burgers' equation given by

$$L(\tilde{u}) = \tilde{u}\frac{\mathrm{d}\tilde{u}}{\mathrm{d}x} - \nu\frac{\mathrm{d}^2\tilde{u}}{\mathrm{d}x^2} = 0, \tag{9.3}$$

where the first term is a nonlinear convection term, the second term is a diffusion term (multiplied by the constant viscosity ν), and $\tilde{u}$ is the exact solution to this differential equation.

A simple second-order accurate finite difference discretization for the steady-state Burgers' equation is

$$L_h(u_h) = u_i\left(\frac{u_{i+1} - u_{i-1}}{2\Delta x}\right) - \nu\left(\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}\right) = 0, \tag{9.4}$$

where u_h denotes the exact solution to this discrete equation where we have assumed a Cartesian mesh with constant node spacing $h = \Delta x$. To find the truncation error associated with this discretization method, we first find the Taylor series expansions of u_{i+1} and u_{i-1}

expanded about u_i:

$$u_{i+1} = u_i + \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_i \Delta x + \left.\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right|_i \frac{\Delta x^2}{2} + \left.\frac{\mathrm{d}^3u}{\mathrm{d}x^3}\right|_i \frac{\Delta x^3}{6} + \left.\frac{\mathrm{d}^4u}{\mathrm{d}x^4}\right|_i \frac{\Delta x^4}{24} + O\left[\Delta x^5\right], \tag{9.5}$$

$$u_{i-1} = u_i - \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_i \Delta x + \left.\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right|_i \frac{\Delta x^2}{2} - \left.\frac{\mathrm{d}^3u}{\mathrm{d}x^3}\right|_i \frac{\Delta x^3}{6} + \left.\frac{\mathrm{d}^4u}{\mathrm{d}x^4}\right|_i \frac{\Delta x^4}{24} + O\left[\Delta x^5\right]. \tag{9.6}$$

Plugging these expansions into the discrete operator $L_h(\cdot)$ and rearranging gives

$$L_h(u) = \underbrace{u_i \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_i - \nu \left.\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right|_i}_{L(u)} + \underbrace{u_i \left.\frac{\mathrm{d}^3u}{\mathrm{d}x^3}\right|_i \frac{\Delta x^2}{6} - \nu \left.\frac{\mathrm{d}^4u}{\mathrm{d}x^4}\right|_i \frac{\Delta x^2}{12} + O\left[\Delta x^4\right]}_{TE_h(u)}. \tag{9.7}$$

The right hand side of Eq. (9.7) can be thought of as the actual differential equation that is represented by the discretization scheme. It contains the original partial differential equation, higher-order derivatives of the solution, and coefficients which are functions of the grid spacing to different powers. Since the leading terms are on the order of Δx^2, we find that the formal order of accuracy of this method is second order. Equation (9.7) also tells us that this discretization is consistent since the discrete equations approach the partial differential equation in the limit as Δx goes to zero. Equation (9.7) can be rewritten as the *generalized truncation error expression* (developed in Chapter 5) by employing our operator notation:

$$L_h(u) = L(u) + \mathrm{TE}_h(u). \tag{9.8}$$

Equation (9.8) states that the discretized equation is equal to the continuous mathematical model (in this case Burgers' equation) plus the truncation error (TE) associated with the discretization $h = \Delta x$. Note that in this form of the equation we have not specified the form that u will take. We will make extensive use of Eq. (9.8) in the following discussion.

9.1.3 1-D truncation error analysis on nonuniform meshes

One approach for examining the truncation error on nonuniform (anisotropic) meshes is to first perform a global transformation of the governing equation. An additional motivation for performing such a global transformation is that it will provide insight into the role of mesh quality on the solution discretization and truncation errors. Following Thompson *et al.* (1985) and Roy (2009), the first derivative in physical space can be transformed into a uniform computational space using the transformation $\xi = \xi(x)$ and the truncation error for the centered central difference method can be found as follows:

$$\begin{aligned}\frac{\mathrm{d}u}{\mathrm{d}x} = \underbrace{\frac{1}{x_\xi}\frac{\mathrm{d}u}{\mathrm{d}\xi}}_{L(u)} &= \underbrace{\frac{1}{x_\xi}\left(\frac{u_{i+1} - u_{i-1}}{2\Delta\xi}\right)}_{L_h(u)} - \underbrace{\frac{1}{6}\frac{x_{\xi\xi\xi}}{x_\xi}\frac{\mathrm{d}u}{\mathrm{d}x}\Delta\xi^2}_{Small} - \underbrace{\frac{1}{2}x_{\xi\xi}\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\Delta\xi^2}_{Stretching} \\ &\quad - \underbrace{\frac{1}{6}x_\xi^2\frac{\mathrm{d}^3u}{\mathrm{d}x^3}\Delta\xi^2}_{Standard\ TE} + O(\Delta\xi^4).\end{aligned} \tag{9.9}$$

Here we have assumed a fixed transformation $\xi = \xi(x)$ and thus $\Delta\xi \to 0$ with systematic mesh refinement. The derivatives of x with respect to ξ (e.g., x_ξ, $x_{\xi\xi}$) are called metrics of the transformation and are functions only of the mesh transformation itself. On the right hand side of this equation, the first term is the finite difference equation in transformed coordinates, the second term involving $x_{\xi\xi\xi}$ is zero when the discrete form of the metrics is used with the same central difference approximation (Mastin, 1999), the third term involving $x_{\xi\xi}$ is a grid stretching term, and the fourth term involving the square of x_ξ is the standard leading truncation error term that appears when the equations are discretized on a uniform mesh. The higher order terms (order $\Delta\xi^4$ and higher) can be neglected in the limit as the mesh spacing goes to zero. The grid stretching term is zero for a uniformly spaced mesh, but gets large as the mesh is stretched (e.g., from coarse to fine spacing).

This approach for examining the truncation error of the transformed equation illustrates some important features of the truncation error for anisotropic meshes. First, the truncation error is affected by the mesh resolution ($\Delta\xi$), the mesh quality ($x_{\xi\xi}$), and the local solution derivatives. Second, the quality of a mesh can only be assessed in the context of the local solution behavior and the mesh resolution since the grid stretching term $x_{\xi\xi}$ is multiplied by the second derivative of the solution and $\Delta\xi^2$.

In a similar manner, the truncation error for the second-order accurate centered second derivative in non-conservative form is:

$$\frac{\mathrm{d}^2 u}{\mathrm{d}x^2} = \overbrace{\frac{1}{x_\xi^2}\frac{\mathrm{d}^2 u}{\mathrm{d}\xi^2} - \frac{x_{\xi\xi}}{x_\xi^3}\frac{\mathrm{d}u}{\mathrm{d}\xi}}^{L(u)} = \overbrace{\frac{1}{x_\xi^2}\left(\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta\xi^2}\right) - \frac{x_{\xi\xi}}{x_\xi^3}\left(\frac{u_{i+1} - u_{i-1}}{2\Delta\xi}\right)}^{L_h(u)}$$
$$+ \overbrace{\frac{1}{12}\left(\frac{2x_{\xi\xi}x_{\xi\xi\xi} - x_\xi x_{\xi\xi\xi\xi}}{x_\xi^3}\right)\frac{\mathrm{d}u}{\mathrm{d}x}\Delta\xi^2}^{Small} + \overbrace{\frac{1}{12}\left(\frac{3x_{\xi\xi}^2 - 4x_\xi x_{\xi\xi\xi}}{x_\xi^2}\right)\frac{\mathrm{d}^2 u}{\mathrm{d}x^2}\Delta\xi^2}^{Small}$$
$$- \underbrace{\frac{1}{3}x_{\xi\xi}\frac{\mathrm{d}^3 u}{\mathrm{d}x^3}\Delta\xi^2}_{Stretching} - \underbrace{\frac{1}{12}x_\xi^2\frac{\mathrm{d}^4 u}{\mathrm{d}x^4}\Delta\xi^2}_{Standard\ TE} + O(\Delta\xi^4). \tag{9.10}$$

The third and fourth terms on the right hand side will usually be small when the discrete form of the metrics and smooth transformations are used. The fifth term involving $x_{\xi\xi}$ is a grid stretching term and the sixth term is the standard leading truncation error term that appears when the equations are discretized in physical space on a uniform grid.

Combining these two truncation error expressions and neglecting the small terms results in the following truncation error expression for the transformed Burgers' equation:

$$
\begin{aligned}
u\frac{\mathrm{d}u}{\mathrm{d}x} - \nu\frac{\mathrm{d}^2u}{\mathrm{d}x^2} &= \overbrace{\left(\frac{u}{x_\xi} + \frac{\nu x_{\xi\xi}}{x_\xi^3}\right)\frac{\mathrm{d}u}{\mathrm{d}\xi} - \frac{\nu}{x_\xi^2}\frac{\mathrm{d}^2u}{\mathrm{d}\xi^2}}^{L(u)} \\
&= \overbrace{\left(\frac{u_i}{x_\xi} + \frac{\nu x_{\xi\xi}}{x_\xi^3}\right)\left(\frac{u_{i+1}-u_{i-1}}{2\Delta\xi}\right) - \frac{\nu}{x_\xi^2}\left(\frac{u_{i+1}-2u_i+u_{i-1}}{\Delta\xi^2}\right)}^{L_h(u)} \\
&\quad + \underbrace{x_{\xi\xi}\left(\frac{\nu}{3}\frac{\mathrm{d}^3u}{\mathrm{d}x^3} - \frac{u_i}{2}\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)\Delta\xi^2}_{\textit{Stretching}} + \underbrace{x_\xi^2\left(\frac{\nu}{12}\frac{\mathrm{d}^4u}{\mathrm{d}x^4} - \frac{u_i}{6}\frac{\mathrm{d}^3u}{\mathrm{d}x^3}\right)\Delta\xi^2}_{\textit{Standard TE}} + O(\Delta\xi^4).
\end{aligned}
\tag{9.11}
$$

Thus the truncation error for second-order accurate central differences applied to Burgers' equation contains two types of term. The first is due to mesh quality (stretching in 1-D) and the second is due to the mesh resolution. For a uniform grid, the mesh stretching term is zero and the standard truncation error terms are exactly equal to the truncation error for Burgers' equation discretized in physical space on a uniform grid as given in Eq. (9.7).

Extending these mesh transformation procedures to higher dimensions results in additional truncation error terms related to the mesh quality such as mesh skewness, aspect ratio, etc. (Mastin, 1999). While the above procedures rely on a global transformation of the mathematical model, similar procedures could be developed for unstructured grid methods which rely on local transformations centered about each cell or element.

9.1.4 Isotropic versus anisotropic mesh adaptation

As shown above, the mesh quality plays a role in the truncation error which acts as the local source term for the discretization error. Some argue that isotropic (i.e., uniform or nearly uniform) meshes should be used in order to minimize the impact of the mesh quality on the solution. However, examination of Eq. (9.11), developed for the steady Burgers' equation, shows that the mesh quality (mesh stretching in this case) appears in the truncation error multiplied by second and third derivatives of the solution. Therefore, in regions where these solution derivatives are small, large variations in mesh stretching can be tolerated. Alternatively, the magnitude of the mesh stretching term can be reduced by increasing the mesh resolution. In our opinion, these two observations provide a sound argument for allowing significant mesh anisotropy in regions where the solution is well behaved or there is significant mesh clustering.

Two examples are now given for hyperbolic problems that employ mesh adaptation with large anisotropy. The first example is from Laflin (1997) who examined laminar hypersonic flow over a compression corner using structured grids. Figure 9.1a shows the adapted mesh and Figure 9.1b the Mach number contours. The hyperbolic free-stream region, where no

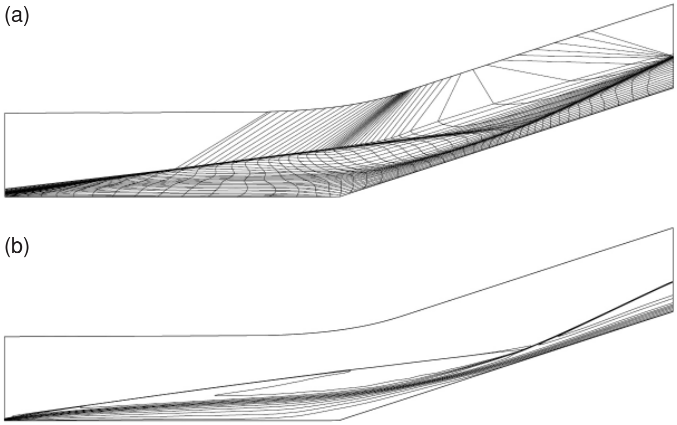

Figure 9.1 Example of anisotropic mesh adaption using structured grids for the laminar hypersonic flow over a compression corner: (a) r-adapted mesh and (b) Mach number contours (reproduced from Laflin, 1997).

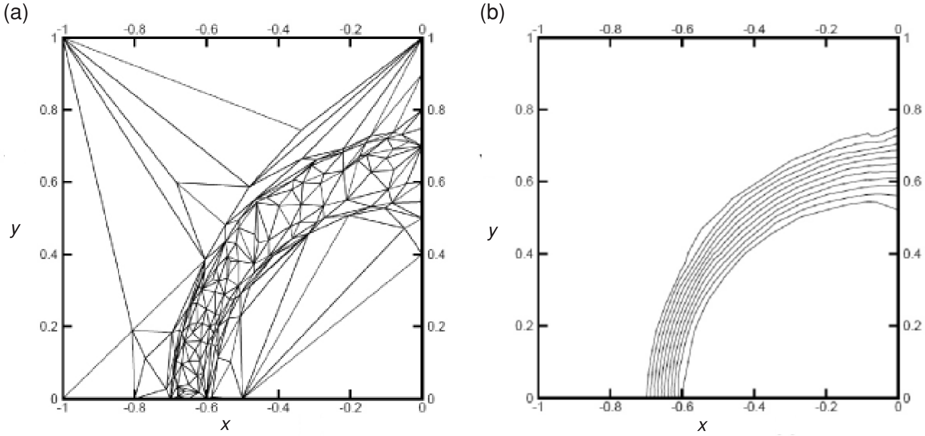

Figure 9.2 Example of anisotropic mesh adaption using unstructured grids for a 2-D linear advection equation: (a) r-adapted mesh and (b) velocity contours (reproduced from Wood and Kleb, 1999).

solution gradients occur, is contained by a single layer of large, high-aspect ratio, skewed cells. While these cells have very poor mesh quality, the fact that there are no gradients in this region means that no discretization errors are introduced. The second example is from Wood and Kleb (1999) who examined the 2-D linear wave equation using unstructured mesh adaptation. The adapted mesh is given in Figure 9.2a and the corresponding solution in Figure 9.2b. Again the mesh is extremely anisotropic away from the gradient regions and clustered only in the regions of the solution variation.

9.2 Adaptation criteria

One of the most difficult aspects of solution adaptation is finding a good criterion for driving the adaptation process. The least defensible methods for adaptation are based on either solution features or estimates of the discretization error. Another approach based on higher-order reconstructions of solution values or gradients (e.g., finite element recovery methods) is also commonly used. The most rigorous methods are based on evaluating or estimating the local truncation error, or equivalently the residual (e.g., from finite elements). We refer to these latter approaches as residual-based methods, and they account for the local cell/element contributions to the discretization error. In their basic form, residual-based methods, and to a lesser extent, the reconstruction methods, can provide for only globally "good" discrete solutions. For adaptation targeted to a specific system response quantity, an adjoint problem must also be solved in order to find the sensitivity of the discretization error in the system response quantity to the local errors generated at each cell/element.

9.2.1 Solution features

A commonly-used method for driving the solution adaptation process is to use solution features such as solution gradients, solution curvature, or even specific solution features to drive the adaptation process. When there is only one dominant feature to be resolved in a problem, then feature-based adaptation can often improve the solution. However, when multiple features are present (e.g., shock waves, expansion fans, contact discontinuities, and boundary layers in a compressible flow problem), feature-based adaptation often results in some features being over-refined while other features are not refined enough. In such cases, feature-based adaptation can fail "disastrously" (Ainsworth and Oden, 2000). An example of the failure of solution gradient-based adaptation is given in Section 9.2.5.

9.2.2 Discretization error

Since it is the discretization error that one wishes to reduce with solution adaptation, on the surface it might appear the discretization error, or perhaps its estimate, would serve as an appropriate driver for the adaption process. However, as will be shown later, the total discretization error is not an appropriate solution adaption criterion. One should instead adapt based on the *local contributions* to the discretization error, i.e., the truncation error. Due to the fact that the discretization error is also transported (i.e., convected and diffused) from other regions of the domain, it is possible that the total discretization error may be large in a region where the local contributions to the discretization error are small (e.g., near, but not at, a singularity). Similarly, it is possible for the total discretization error to be small even when the local contribution to the discretization error is large. For these reasons, adaptation based on the total discretization error is not recommended.

An example of the failure of mesh adaptation based on the total discretization error was given by Gu and Shih (2001) who examined the flow in an incompressible lid-driven cavity.

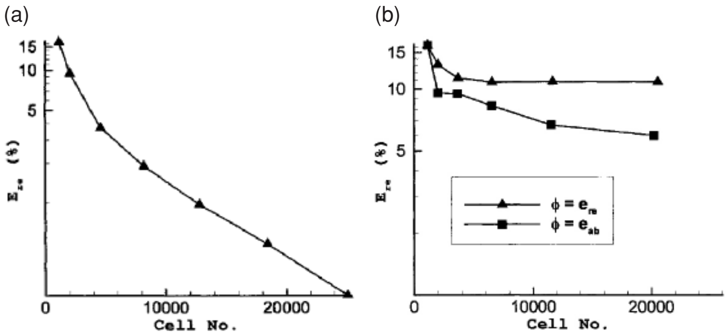

Figure 9.3 Example of Cartesian grid-based h-adaptation for a lid-driven cavity: (a) uniform refinement and (b) refinement based on the total discretization error (reproduced from Gu and Shih, 2001).

The cavity aspect ratio (width to height) was two, and the Reynolds number based on the cavity width was 2000. Figure 9.3 shows the changes in the L1 norm of the discretization error in velocity with the total number of cells. For the case of uniform refinement (Figure 9.3a), the error was reduced at the expected formal order of accuracy of two with refinement. In the case of mesh adaptation (Figure 9.3b), the solutions were nonconvergent when the relative discretization error was used, whereas only a small error reduction occurred when the absolute discretization error was used. This example highlights the dangers of adapting based on total discretization error which also includes error transported from other regions.

9.2.3 Recovery methods

An error indicator that is frequently used for solution adaptation for finite element methods is gradient recovery or reconstruction. In this approach, gradients of the finite element solution are compared to the gradients found from post-processing patches of neighboring elements. Larger mismatches between these two gradient computations serve as an indicator of larger local errors. Recall from Chapter 8 that recovery-based methods in finite elements such as the superconvergent patch recovery (SPR) method (Zienkiewicz and Zhu, 1992a) can provide higher-order accurate estimates of the solution gradients when the superconvergence property is attained. In this case, the SPR error estimator will act like an estimator of the total discretization error in the solution gradients. One would thus expect the SPR method to be a poor criterion on which to base a solution adaptation strategy; however, as was the case for discretization error estimation, the SPR method has been shown to be effective for driving solution adaptation, at least for elliptic problems (Ainsworth and Oden, 2000).

An example of mesh adaptation based on the SPR method in a linear elasticity problem was given by Zienkiewicz and Zhu (1992b). They examined an L-shaped domain in a plane stress condition as shown in Figure 9.4a. For this problem, the SPR method will provide local estimates of the discretization error in the stresses. By using adaptive remeshing with

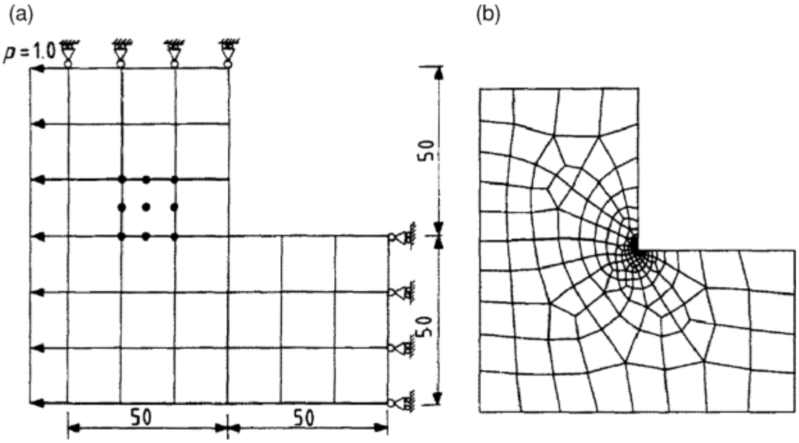

Figure 9.4 Application of the SPR recovery-based error estimator for estimating plane stress in an L-shaped linear elastic domain: (a) problem set up and initial mesh and (b) final adapted mesh with 229 elements (reproduced from Zienkiewicz and Zhu, 1992b).

h-refinement (discussed in Section 9.3), an adapted mesh with 229 elements is shown in Figure 9.4b with maximum discretization errors in the stress of under 1%. For an application of the SPR method to compressible turbulent flows with adaptive remeshing, see Ilinca *et al.* (1998).

Laflin (1997) and McRae and Laflin (1999) have developed a similar driver for adaptation based on solution interpolation error. The implementation along with *r*-adaptation (see Section 9.3.2.2) on structured meshes has been successful for a wide range of 2-D problems. The basic idea behind the approach is that the solution values are compared to values interpolated from a patch of neighboring cells. One difference between their approach and finite element-based recovery methods described above is that they deal with solution values rather than solution gradients. Examples of meshes and solutions found using their solution interpolation error approach along with *r*-adaptation were given in Figure 9.1; an additional example will be presented later in Figure 9.12. Both examples show a significant amount of anisotropic mesh adaptation.

9.2.4 Truncation error/residuals

The formal relationship between the discretization error and the truncation error comes from the discretization error transport equations discussed earlier. It was shown that the truncation error provides the local element's contribution to the discretization error. As such, the truncation error is a good indicator of where mesh adaptation should occur. The general concept behind truncation error-based adaptation is to reduce the truncation error where its magnitude is large in order to reduce the total discretization error. Baker (1997) notes that although the idea of truncation error-based adaptation is fundamentally sound, it has “surprisingly” seen little use. The factors affecting the truncation error for a given discretization scheme include the element size, mesh quality, and the local solution behavior.

9.2.4.1 General truncation error/residual-based methods

For simple discretization schemes, the truncation error can be computed directly. For more complex schemes where direct evaluation of the truncation error is difficult, an approach for its estimation is needed. Three approaches for estimating the truncation error were discussed in the last chapter in the context of error transport equations for estimating the discretization error. In the first approach, the exact solution to the mathematical model (or an approximation thereof) is substituted into the discretized equations, with the nonzero remainder (i.e., the discrete residual) approximating the truncation error. This approach has been used by Berger and Jameson (1985) for the Euler equations in fluid dynamics by approximating the exact solution to the mathematical model using Richardson extrapolation. In the second approach for estimating the truncation error, the exact solution to the discrete equations is inserted into the continuous mathematical model equations. This continuous residual can be easily evaluated if a continuous representation of the discrete solution is available, for example, in the finite element method. In fact, this is exactly the residual used in the finite element method, which measures how well the finite element solution satisfies the weak form of the mathematical model. The final approach for estimating the truncation error is to approximate it with any extra numerical dissipation used to stabilize the computations. The upwind schemes used for compressible flow problems fall into this category since any upwind scheme can be written as a central difference scheme plus a numerical diffusion term (e.g., see Hirsch, 1990).

9.2.4.2 Finite element residual-based methods

The residual-based methods for estimating discretization error in finite element solutions discussed in the last chapter also provide estimates of the local contributions to the total discretization error. The initial formulation of residual-based methods, including their use for mesh adaptation, was given by Babuska and Rheinboldt (1978a,b). Since explicit residual methods neglect the transported component of the error, their local behavior makes them suitable for solution adaptation. Implicit residual methods directly treat both the local and the transported components of the discretization error that appear in the residual equation. When applied in the context of solution adaptation, only the local component should be used. A review of explicit and implicit residual methods for solution adaption is given by Verfurth (1999), who also addresses recovery-based methods and hierarchical bases.

Stewart and Hughes (1996) provide an example of an explicit residual method applied to the Helmholtz equation for acoustic applications. The specific case examined is nonuniform acoustic radiation in an infinite rigid cylinder. They note that when used for mesh adaptation, the global constant in the explicit residual-based error estimates need not be computed, thus saving computational effort. Adaption is performed using adaptive remeshing (see

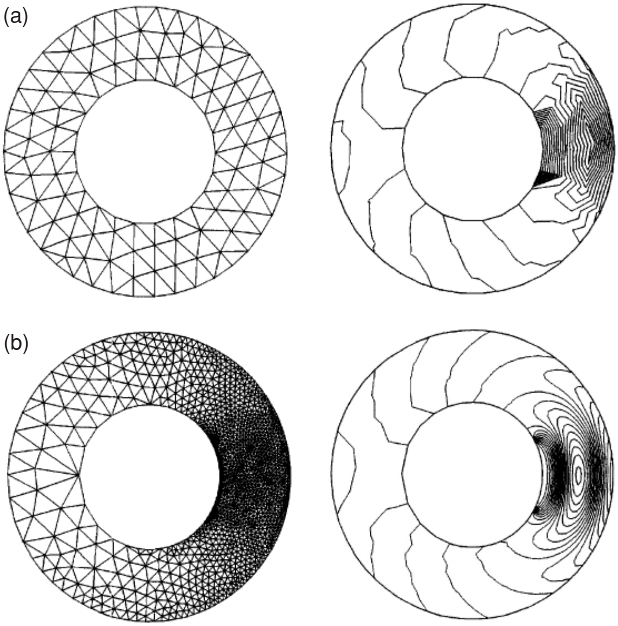

Figure 9.5 Nonuniform acoustic radiation in an infinite rigid cylinder: (a) initial uniform mesh and solution and (b) adapted mesh and solution (reproduced from Stewart and Hughes, 1996).

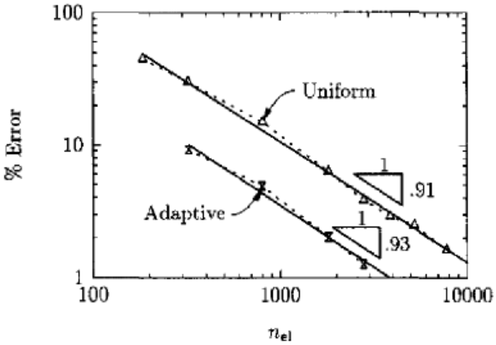

Figure 9.6 Global energy norms of the discretization error with increasing number of elements for both uniform and adaptive mesh refinement for acoustic radiation in an infinite rigid cylinder (reproduced from Stewart and Hughes, 1996).

Section 9.3.1). The initial uniform mesh and solution are presented in Figure 9.5a, while an adapted mesh and solution using five levels of refinement are given in Figure 9.5b. It is clear that the solution is significantly better resolved with adaptation. Figure 9.6 shows the behavior of the global energy norm of the discretization error with both uniform and adaptive mesh refinement, and both approaches appear to converge at close to the formal order of accuracy of two. To achieve global energy norms of 1%, the adaptive procedure requires fewer than 4000 elements, whereas approximately 13 000 elements are required for uniform mesh refinement.

9.2.5 Adjoint-based adaptation

A promising approach for solution adaptation to reduce the discretization error in a system response quantity is the adjoint method. Most adjoint methods for mesh adaptation use the truncation error/residual weighted by the adjoint sensitivities as indicators of where mesh adaptation should occur (see Section 8.2.2.3). Adjoint methods thus provide targeted mesh adaption for the chosen system response quantity. The main drawback to adjoint methods is their complexity and code intrusiveness, as evidenced by the fact that adjoint-based adaption has not yet found its way into commercial scientific computing codes. This section provides examples of adjoint-based adaptation methods based on the truncation error/residual or approximations thereof.

An example of a truncation error approximation method combined with an adjoint method for system response quantities is given by Dwight (2008), who considered the inviscid transonic flow over an airfoil using an unstructured finite-volume discretization. The discretization scheme employed a central-type flux quadrature combined with an artificial viscosity method for numerical stability. The adjoint method was formulated to provide the sensitivities of the total drag force to the numerical parameters used in the artificial viscosity stabilization technique. The underlying assumption is that these sensitivities will be large where the artificial viscosity is large and thus dominates the standard central-type truncation error terms.

Figure 9.7 shows the discretization error in the drag coefficient as a function of the number of nodes. Uniform refinement (squares) shows second-order convergence on the coarser grids, then a reduction to first order on the finer grids, likely due to the presence of the shock discontinuities (e.g., see Banks *et al.*, 2008). Feature-based adaptation using solution gradients (triangles) initially shows a reduction in the discretization error for the drag coefficient, but then subsequent adaptation steps show an increase in the error. The adjoint-based artificial dissipation estimators give the best results, especially when the drag coefficient is corrected by its discretization error estimate (circles).

The adapted grids for the gradient-based and adjoint-based adaption strategies are given in Figure 9.8a and b, respectively. The gradient-based adaptation refines the shock waves on the upper and lower surface as well as the region downstream of the trailing edge. The adjoint-based method also refines the mesh near the surface and in the region above the airfoil containing the two acoustic waves that emanate from the leading edge region and impinge on the trailing edge. In many cases, the mesh resulting from adjoint-based adaptation can be used to provide insight into the physical mechanisms that impact a given system response quantity, in this case the importance of the acoustic waves in determining the airfoil drag.

Another example of adjoint-based adaptation used to reduce discretization errors in a viscous, compressible flow problem was given by Venditti and Darmofal (2003). They used

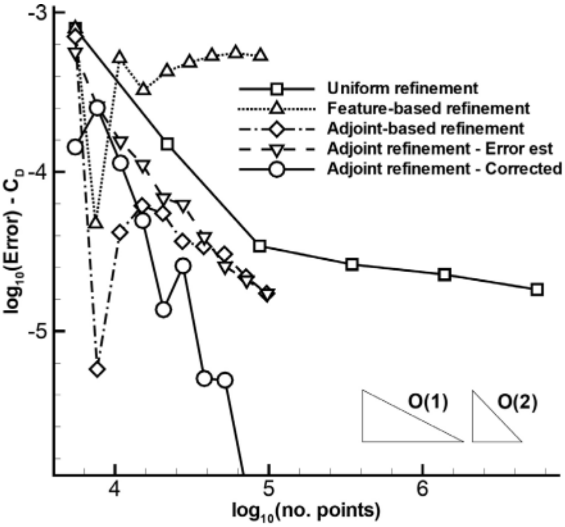

Figure 9.7 Discretization error in the drag coefficient for transonic flow over an airfoil comparing uniform refinement, local gradient-based adaptation, and adjoint, artificial dissipation-based adaptation (reproduced from Dwight, 2008).

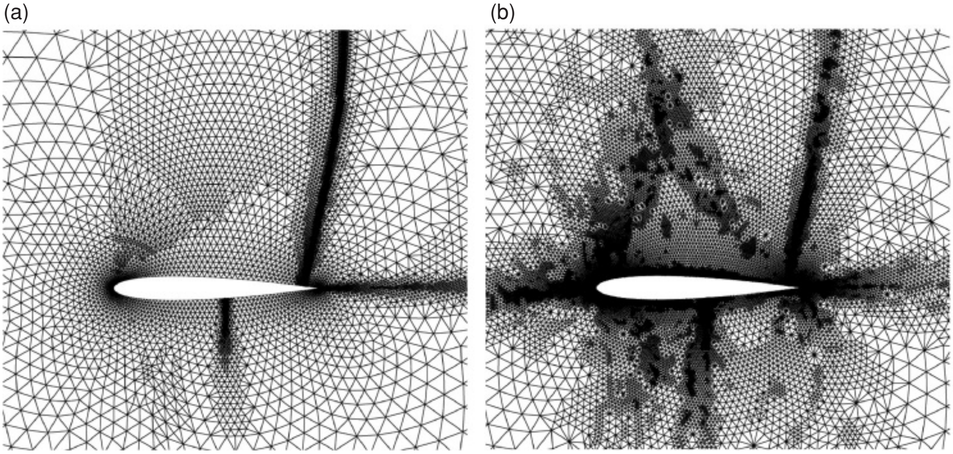

Figure 9.8 Adapted grids for transonic flow over an airfoil: (a) solution gradient-based adaptation and (b) adaptation with an adjoint-based artificial dissipation estimator for the drag force (reproduced from Dwight, 2008).

an unstructured grid finite volume scheme, and examined both the laminar and turbulent viscous flow over an airfoil. Their study also included a comparison to feature-based adaptation using the solution curvature (Hessian). A plot of the drag versus the number of mesh nodes is presented in Figure 9.9. The adjoint- (output-) based adaption converges much faster with mesh adaptation than the solution curvature-based (Hessian) method.

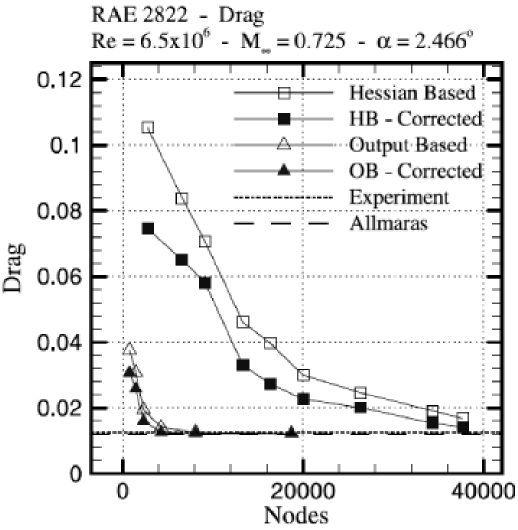

Figure 9.9 Drag for viscous transonic flow over an airfoil using solution curvature-based (Hessian) and adjoint-based adaptation (reproduced from Venditti and Darmofal, 2003).

The adapted grids for this case are presented in Figure 9.10 and show that adjoint-based adaptation results in significantly different final adapted meshes.

Rannacher and Suttmeier (1997) examined the case of a square elastic disk with a crack subjected to a constant boundary traction acting along a portion of the upper boundary as shown in Figure 9.11a. They then employed an explicit residual method both alone and combined with an adjoint method to drive a mesh adaptation routine. The system response quantity for the example shown here was the average normal stress over the clamped bottom and right boundaries. Final adapted meshes for the residual-based adaption and the adjoint method are given in Figure 9.11b and c, respectively. While both approaches adapted to the crack tip (near the center), the adjoint method also adapted to the lower boundary. The adjoint-based adaption was found to provide much lower discretization error levels in the average normal stress than the residual-based adaption. In fact, the adjoint-based method achieved similar error levels as the 65 000 elements residual-based method using under 9000 elements.

9.3 Adaptation approaches

Once the approach for driving the adaptation process has been implemented, the end result is typically a weighting function that varies over the spatial domain. In some cases, the weighting function may be a vector quantity indicating directionality preferences for the adaptation. With the weighting function in hand, local solution adaptation can be achieved in a variety of different ways. Adaptive remeshing generally starts with a coarse mesh, then

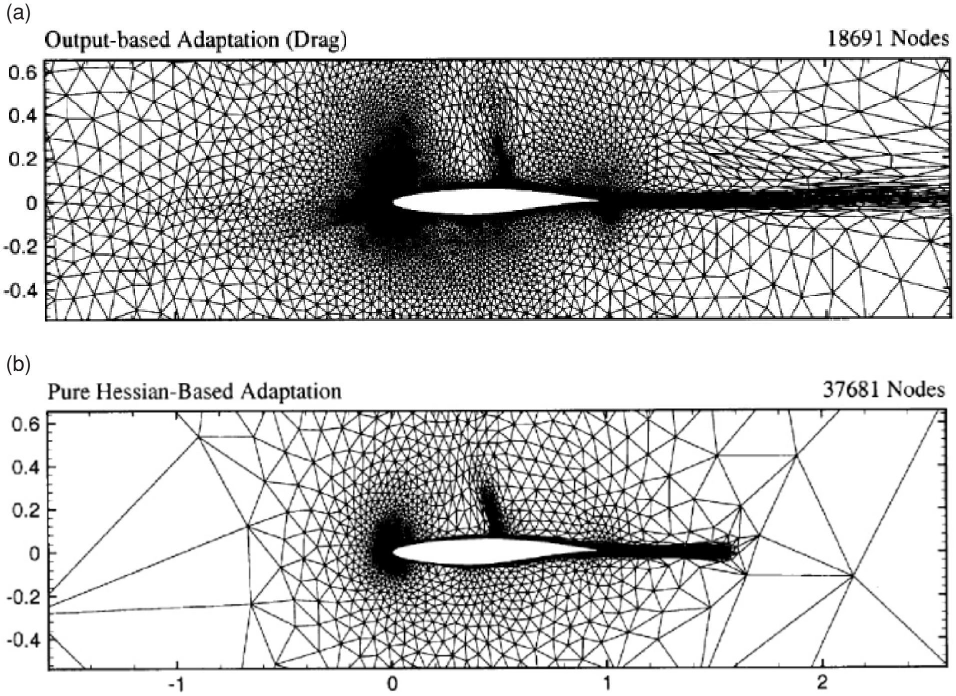

Figure 9.10 Adapted grids for subsonic flow over an airfoil: (a) adjoint-based adaptation for drag and (b) solution curvature-based (Hessian) adaptation (reproduced from Venditti and Darmofal, 2003).

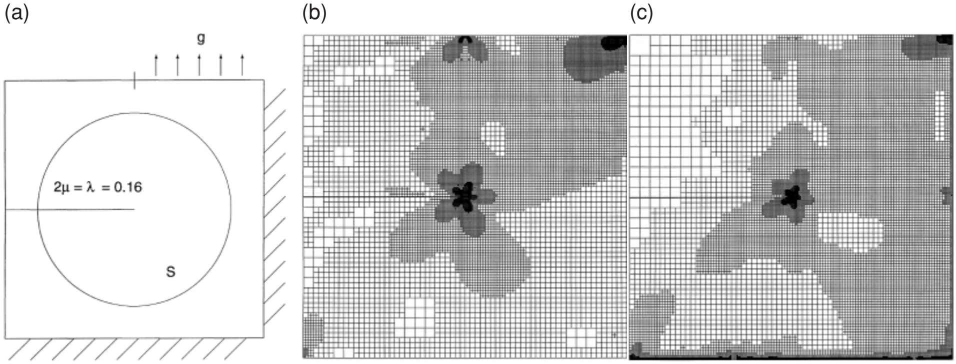

Figure 9.11 Linear elasticity considering the average normal stress over the clamped boundaries: (a) problem set up showing the crack emanating from the center and extending to the left boundary, (b) final mesh for residual-based adaption, and (c) final mesh for adjoint-based adaption (reproduced from Rannacher and Suttmeier, 1997).

employs a mesh generation tool in a recursive manner to improve the mesh. Alternatively, one could seek to adapt the mesh by adding additional cells/elements (*h*-adaptation) or moving them from one region to another while maintaining the original mesh connectivity (*r*-adaptation). Order refinement (*p*-adaptation) increases the formal order of accuracy of the discretization scheme in regions of high weighting function. Hybrid combinations of these approaches are also used such as mixed *h* and *r* adaption (mesh movement and refinement) and mixed *h* and *p* adaption (e.g., the *hp*-version of finite elements). For general unstructured grid methods, adaptive remeshing and *h*-adaptation are the most popular. For structured grid methods, the *r*-adaption approach is the most common since *h*-adaptation cannot be performed locally due to the required *i*, *j*, *k* ordering of the mesh. In addition to mesh refinement, other issues that should be considered when adapting a mesh are mesh quality, local solution behavior, and the alignment of the element faces.

9.3.1 Adaptive remeshing

Initial approaches to adaptation were based on adaptive remeshing, where a grid generation tool (often the same one used to create the initial mesh) is employed to adaptively improve the mesh. The weighting functions are usually accounted for as constraints in the remeshing process. An early example of adaptive remeshing for compressible fluid flow problems was given by Peraire *et al.* (1987), while a more recent application for solid mechanics is given by Bugeda (2006).

9.3.2 Mesh adaptation

There are two approaches for adapting a mesh that are not based on remeshing. Mesh refinement (*h*-adaptation) selectively sub-divides cells where the weighting function is large, and possibly coarsening cells where it is small. Mesh movement (*r*-adaptation) maintains the same number of nodes and node-to-node connectivity, but moves elements towards regions targeted for adaptation. In some cases, both procedures are used together in a mixed *h*- and *r*-adaptation process. These approaches to mesh adaptation are discussed below.

9.3.2.1 Local mesh refinement/coarsening (h-adaptation)

Local mesh refinement, or *h*-adaptation, is achieved by sub-dividing cells that have been targeted for refinement. One usually starts with an initial coarse mesh, and then begins refining, and possibly coarsening (a process known as agglomeration) the mesh in an iterative manner. The total number of local mesh refinement levels is usually specified, for example, by allowing a coarse grid cell to be refined up to a maximum of six times. Another constraint that is often used is that neighboring cells should be within one level of refinement of each other. This prevents the possibility of having a highly-refined cell adjacent to an extremely coarse cell. In some cases, *h*-adaptation employs edge swapping

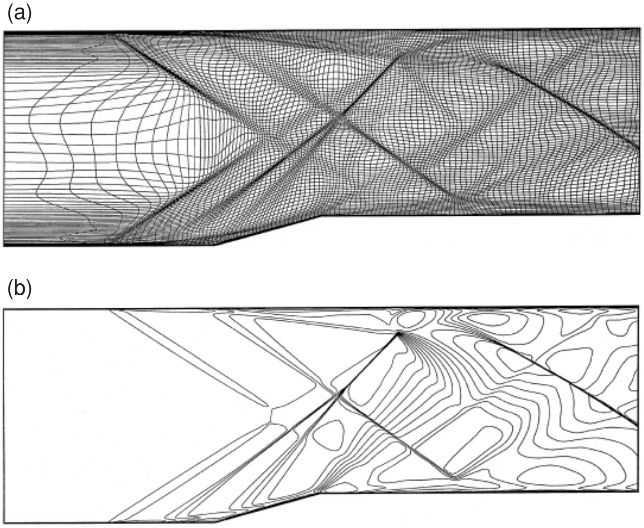

Figure 9.12 Structured grid adaptation for laminar flow through a supersonic inlet: (a) *r*-adapted mesh and (b) density contours (reproduced from McRae, 2000).

where the edge connecting two cells is moved to either improve mesh quality or to better align the faces with the solution features. One drawback of using pure *h*-adaptation is that the mesh quality is adversely affected at the interface between adapted and unadapted regions since the ratio of adjacent cell length scales (i.e., stretching factor) approaches two.

9.3.2.2 Mesh movement (r-adaptation)

Mesh movement, or *r*-adaptation, is more common in structured grids since local mesh refinement (*h*-adaptation) by sub-dividing cells results in an unstructured mesh. When mesh movement is used, the current number of cells and the mesh connectivity is retained, but cells/elements are moved in space according to the weighting function. The most common approaches for achieving mesh movement (Burg, 2006) are:

(1) the Laplacian and modified Laplacian (i.e., elliptic adaptation),
(2) the linear elasticity equations,
(3) the linear spring analogy,
(4) the torsional spring analogy, and
(5) variational methods.

An example of mesh movement is given in Figure 9.12 for laminar flow in a supersonic inlet (McRae, 2000). An elliptic adaption scheme is used based on the solution interpolation error, which is then modified to include grid quality measures. The figure shows both the solution adapted mesh and the density contours. The shock wave formed by the compression ramp separates the laminar boundary layer, leading to another shock forming upstream.

The impingement of these combined shocks on the upper wall also causes the upper-wall boundary layer to separate, producing another shock that emanates from the top wall. It is interesting to note that the shock waves and the boundary layers are clearly evident simply by examining the mesh. The mesh used in this example consists of only 121×91 nodes and the solution was found to provide similar solution quality to one computed on a mesh with approximately 700×900 evenly spaced (i.e., nearly isotropic) cells.

9.3.2.3 Mixed mesh refinement (r- and h-adaptation)

A number of researchers have found that a combination of different solution adaptation strategies provides the best approach for reducing discretization error. For example, Baker (2005) found a combination of local mesh refinement/coarsening and mesh movement to be effective for temporally varying (i.e., unsteady) problems on unstructured meshes.

9.3.3 Order refinement (p-adaptation)

Order refinement (or *p*-adaptation) can also be used to adaptively improve the solution. Order refinement consists of increasing the formal order of accuracy of the discretization scheme in regions targeted for refinement. While this approach is natural to implement with the *p*-version of finite elements, applying it to finite difference and finite volume discretizations is more difficult due to the difficulties of formulating higher-order discretizations for both the governing equations and the boundary conditions. In the finite element method, *p*-adaption and *h*-adaption are often combined in what is known as *hp*-adaptive finite elements. For a review of *hp*-adaptive finite elements, see Patra and Oden (1997).

9.4 Comparison of methods for driving mesh adaptation

The following example compares four different strategies for driving mesh adaption for the 1-D steady Burgers' equation (Roy, 2009). The approach used for adapting the mesh is an *r*-adaptation procedure where the nodal points are assumed to be connected by linear springs (e.g., see Gnoffo, 1982). The strength of these springs is specified by the weighting function, which is in turn determined by the chosen strategy for driving the adaptation. Four different strategies are examined:

- solution gradients,
- solution curvature,
- discretization error, and
- truncation error.

The first two approaches are both feature-based methods. The discretization error-based adaption is implemented in an exact manner since the exact solution to the mathematical model is available. The truncation error for this case was developed in Section 9.1.3 for the nonuniform mesh encountered during the *r*-adaptation process. The derivatives appearing in the truncation error are evaluated using finite-difference expressions formulated in transformed coordinates. The availability of an exact solution to the mathematical model is crucial for the unambiguous assessment of different adaptation strategies.

9.4.1 Mathematical model

The steady Burgers' equation is a quasi-linear ordinary differential equation of the form

$$u\frac{\mathrm{d}u}{\mathrm{d}x} = \nu\frac{\mathrm{d}^2u}{\mathrm{d}x^2}, \tag{9.12}$$

where $u(x)$ is a scalar velocity field, x is the position, and ν is the viscosity. Since mesh adaptation will be used, we employ a global transformation $\xi = \xi(x)$ onto uniformly spaced computational coordinates. The steady-state form of Burgers' equation in transformed coordinates is thus

$$\left(\frac{u}{x_\xi} + \frac{\nu x_{\xi\xi}}{x_\xi^3}\right)\frac{\mathrm{d}u}{\mathrm{d}\xi} - \frac{\nu}{x_\xi^2}\frac{\mathrm{d}^2u}{\mathrm{d}\xi^2} = 0, \tag{9.13}$$

where x_ξ and $x_{\xi\xi}$ are metrics of the mesh transformation. Since Burgers' equation in transformed coordinates is mathematically equivalent to the equation in physical coordinates, the exact solution discussed below will solve both forms of Burgers' equation.

9.4.2 Exact solution

We employ the steady-state viscous shock wave exact solution for the results presented here (Benton and Platzman, 1972). Dirichlet boundary conditions are used as found from the exact solution, which is given by

$$u'(x) = \frac{-2\sinh(x')}{\cosh(x')}, \tag{9.14}$$

where the prime denotes a non-dimensional variable. The Reynolds number for Burgers' equation can be defined as

$$\mathrm{Re} = \frac{u_{\mathrm{ref}}L_{\mathrm{ref}}}{\nu}, \tag{9.15}$$

where u_{ref} is taken as the maximum value for $u(x, t)$ in the domain (here $u_{\mathrm{ref}} = 2$ m/s), L_{ref} is the domain width (here $L_{\mathrm{ref}} = 8$ m), and the choice for ν specifies the Reynolds number.

This exact solution can be related to dimensional quantities by the following transformations:

$$x' = x/L_{\mathrm{ref}} \quad \text{and} \quad u' = uL_{\mathrm{ref}}/\nu. \tag{9.16}$$

Furthermore, the exact solution is invariant to scaling by a constant α:

$$\bar{x} = x/\alpha \quad \text{and} \quad \bar{u} = \alpha u. \tag{9.17}$$

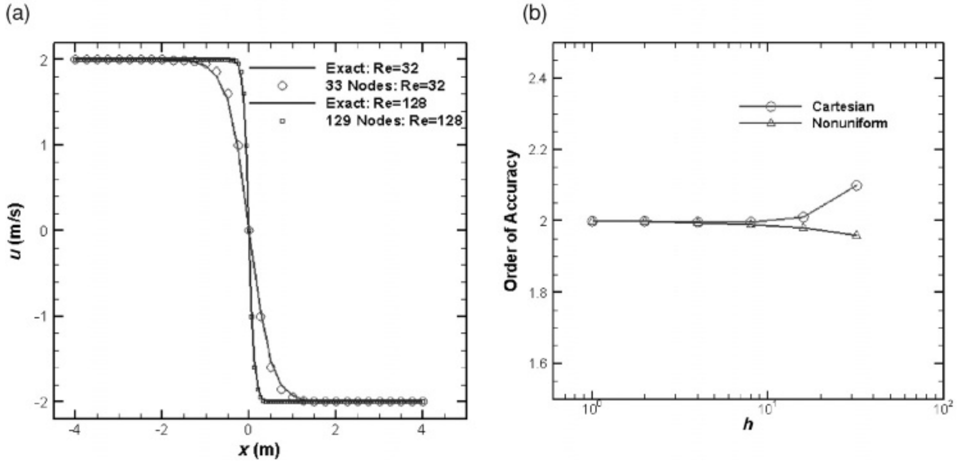

Figure 9.13 (a) Numerical and exact solutions for Burgers' equation in Cartesian coordinates for Reynolds numbers 32 and 128 and (b) order of accuracy of the L_2 norms of the discretization error for Reynolds number 8 (from Roy, 2009).

For simplicity, we find it convenient to solve Burgers' equation in dimensional coordinates on the physical domain $-4 \text{ m} \le x \le 4$ m, choosing α such that the limiting u values vary between -2 m/s and 2 m/s.

9.4.3 Discretization approach

A fully implicit finite-difference code was developed to solve the steady-state form of Burgers' equation using the following discretization:

$$\left(\frac{u_i^n}{x_\xi} + \frac{\nu x_{\xi\xi}}{x_\xi^3}\right)\left(\frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2\Delta\xi}\right) - \frac{\nu}{x_\xi^2}\left(\frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta\xi^2}\right) = 0. \tag{9.18}$$

The nonlinear term is linearized and the resulting linear tridiagonal system is solved directly using the Thomas algorithm. This fully implicit method is formally second-order accurate in space for both the convection and diffusion terms, as was shown in Eq. (9.11). The resulting equations are iterated until the nonlinear system is converged to machine zero, an approximately 12 order of magnitude reduction in the iterative residual since double precision computations are used. Thus round-off and iterative error are neglected.

Numerical and exact solutions for Reynolds numbers of 32 and 128 are given in Figure 9.13a. These numerical solutions were computed on uniform meshes with 33 and 129 nodes for Re = 32 and 128, respectively. In order to verify the Burgers' equation code (see Chapter 5), numerical solutions were run for a Reynolds number of 8 using both uniform and nonuniform meshes. The choice of a lower Reynolds number for this code verification study was made to ensure that the convection and diffusion terms were of similar magnitude. The order of accuracy of the discrete L_2 norms of the discretization

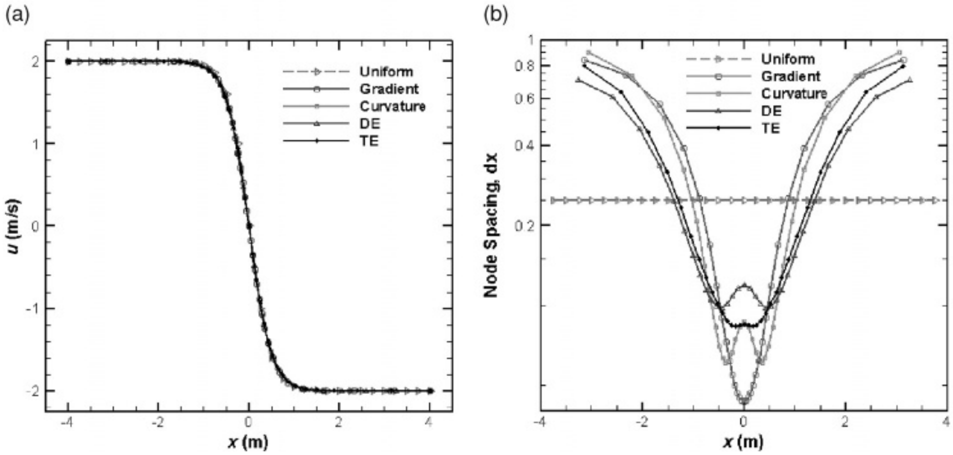

Figure 9.14 Different adaption schemes applied to Burgers' equation for Reynolds number 32: (a) numerical solutions and (b) local nodal spacing Δx (from Roy, 2009).

error is given in Figure 9.13b for increasingly refined meshes up to 513 nodes ($h = 1$). The numerical solutions quickly approach an observed order of accuracy of two with mesh refinement, thus providing evidence that there are no mistakes in the code which will affect the discretization error.

9.4.4 Results

In this section, different methods for driving the mesh adaption are analyzed and compared to the case without adaption (i.e., a uniform mesh). The four different methods for driving the mesh adaptation are: adaption based on solution gradients, adaption based on solution curvature, adaption based on the discretization error (DE), and adaption based on the truncation error (TE). Numerical solutions to the steady-state Burgers' equation for Reynolds number 32 are given in Figure 9.14a for the uniform mesh and the four mesh adaption approaches, all using 33 nodes. The final local node spacing is given in Figure 9.14b for each method and shows significant variations in the vicinity of the viscous shock ($x = 0$).

The discretization error is evaluated by subtracting the exact solution from the numerical solution. Figure 9.15a gives the discretization error for all five cases over the entire domain. The uniform mesh has the largest discretization error, while all mesh adaption approaches give discretization errors that are at least one-third of that from the uniform mesh. The different mesh adaption approaches are compared to each other in Figure 9.15b which shows an enlargement of the region $-2.5 \leq x \leq 0$. Note that the solutions and the discretization errors are all skew-symmetric about the origin, so we may concentrate on only one half of the domain. Truncation error-based adaption gives discretization errors that are less than half of that found from gradient-based adaption, while the other approaches fall in between the two.

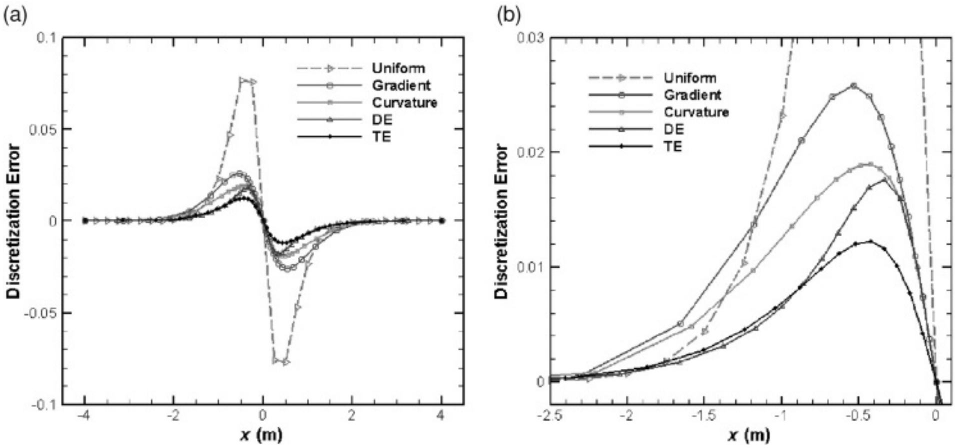

Figure 9.15 Discretization error for different adaptation schemes applied to Burgers' equation for Reynolds number 32: (a) entire domain and (b) enlargement of the region $-2.5 \leq x \leq 0$ (from Roy, 2009).

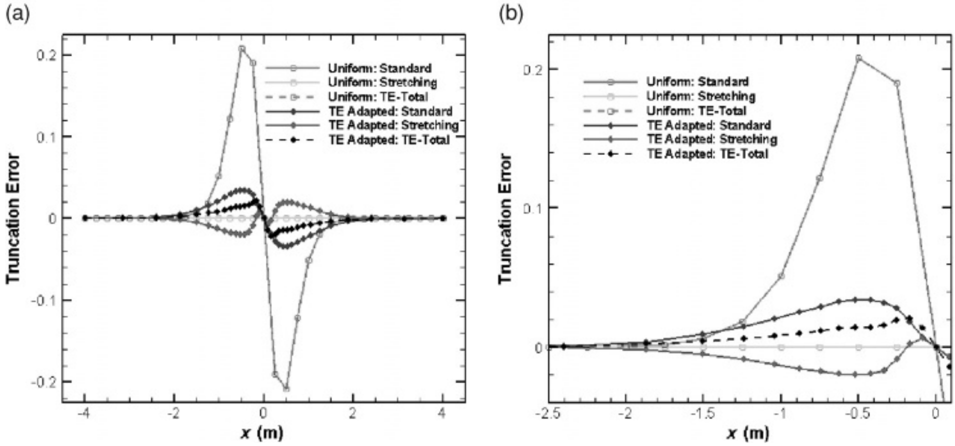

Figure 9.16 Truncation error for a uniform mesh and truncation error-based adaptation (TE) applied to Burgers' equation for Reynolds number 32: (a) entire domain and (b) enlargement of the region $-2.5 \leq x \leq 0$ (from Roy, 2009).

As discussed previously, the truncation error acts as a local source term for the discretization error. Thus it is also instructive to examine the truncation error and its components. The truncation error for the uniform mesh case and the case of truncation error-based adaption is given in Figure 9.16a. The truncation error terms shown are the standard truncation error term, the stretching term, and their sum (TE-Total) as defined by Eq. (9.11). For the uniform mesh, the stretching truncation error term is exactly zero, but the standard truncation error term is large. For the adapted case, the standard and stretching terms are much smaller,

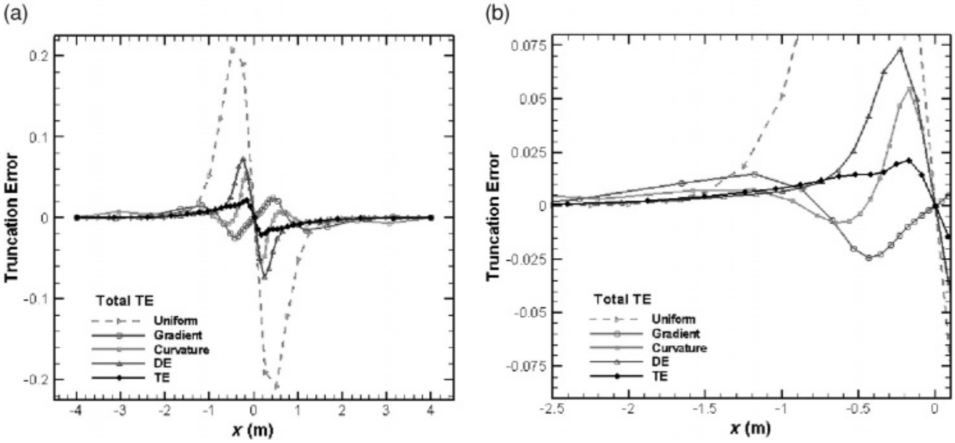

Figure 9.17 Total truncation error for various adaptation approaches applied to Burgers' equation for Reynolds number 32: (a) entire domain and (b) enlargement of the region $-2.5 \leq x \leq 0$ (from Roy, 2009).

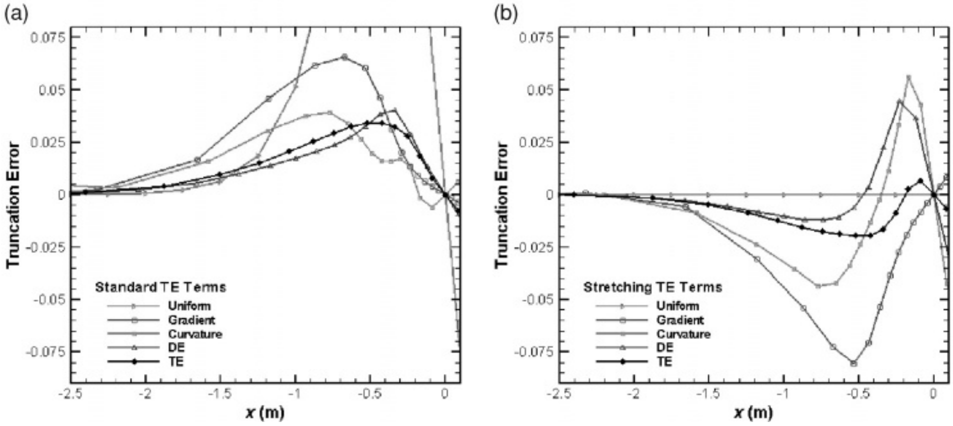

Figure 9.18 Truncation error terms for various adaptation approaches applied to Burgers' equation for Reynolds number 32: (a) standard terms and (b) stretching terms (from Roy, 2009).

and are generally of opposite sign, thus providing some truncation error cancellation. An enlarged view is presented in Figure 9.16b, which shows that the total truncation error (i.e., the sum of the standard and stretching terms) is an order of magnitude smaller for the adapted case than for the uniform mesh.

The total truncation error for all cases is shown in Figure 9.17a, with an enlargement showing only the adaption cases given in Figure 9.17b. The magnitude of the total truncation error is smallest and shows the smoothest variations for the truncation error-based adaption case, while the other three cases show spikes at either $x = -0.4$ m (gradient) or $x = -0.2$ m (DE and curvature).

The two components of the truncation error, the standard terms and the stretching terms, are given in Figure 9.18a and b, respectively. The standard truncation error terms are actually quite similar for all mesh adaption cases, with the gradient-based adaption showing a slightly larger magnitude than the other cases. The biggest differences are seen in the stretching truncation error terms (Figure 9.18b), where the feature-based adaption cases (gradient and curvature) show large mesh stretching contributions. The truncation error-based adaption approach provides the smallest magnitude of the truncation error due to mesh stretching.

9.5 References

Ainsworth, M. and J. T. Oden (2000). *A Posteriori Error Estimation in Finite Element Analysis*, New York, Wiley Interscience.

Babuska, I. (1986). *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, New York, Wiley.

Babuska, I. and W. C. Rheinboldt (1978a). A posteriori error estimates for the finite element method, *International Journal for Numerical Methods in Engineering*. **12**, 1597–1615.

Babuska, I. and W. C. Rheinboldt (1978b). Error estimates for adaptive finite element computations, *SIAM Journal of Numerical Analysis*. **15**(4), 736–754.

Babuska, I., T. Strouboulis, S. K. Gangaraj, and C. S. Upadhyay (1997). Pollution error in the h-version of the finite element method and local quality of the recovered derivatives, *Computer Methods in Applied Mechanics and Engineering*. **140**, 1–37.

Baker, T. J. (1997). Mesh adaptation strategies for problems in fluid dynamics, *Finite Elements in Analysis and Design*, **25**, 243–273.

Baker, T. J. (2005). Adaptive modification of time evolving meshes, *Computer Methods in Applied Mechanics and Engineering*. **194**, 4977–5001.

Bank, R. E. (1996). Hierarchical bases and the finite element method, *Acta Numerica*. **5**, 1–45.

Banks, J. W., T. Aslam, and W. J. Rider (2008). On sub-linear convergence for linearly degenerate waves in capturing schemes, *Journal of Computational Physics*. **227**, 6985–7002.

Benton, E. R. and G. W. Platzman (1972). A table of solutions of the one-dimensional Burgers' equation, *Quarterly of Applied Mathematics*. **30**, 195–212.

Berger, M. J. and A. Jameson (1985). Automatic adaptive grid refinement for the Euler equations. *AIAA Journal*, **23**(4), 561–568.

Bugeda, G. (2006). A new adaptive remeshing scheme based on the sensitivity analysis of the SPR point wise error estimation, *Computer Methods in Applied Mechanics and Engineering*. **195**(4–6), 462–478.

Burg, C. (2006). Analytic study of 2D and 3D grid motion using modified Laplacian, *International Journal for Numerical Methods in Fluids*. **52**, 163–197.

Dwight, R. P. (2008). Heuristic a posteriori estimation of error due to dissipation in finite volume schemes and application to mesh adaptation, *Journal of Computational Physics*. **227**(5), 2845–2863.

Gnoffo, P. (1982). *A Vectorized, Finite-Volume, Adaptive Grid Algorithm Applied to Planetary Entry Problems*, AIAA Paper 1982–1018.

Gu, X. and T. I.-P. Shih (2001). *Differentiating between Source and Location of Error for Solution-Adaptive Mesh Refinement*, AIAA Paper 2001–2660.

Hirsch, C. (1990). *Numerical Computation of Internal and External Flows: Volume 2, Computational Methods for Inviscid and Viscous Flows*, Chichester, Wiley.

Ilinca, F., D. Pelletier, and L. Ignat (1998). Adaptive finite element solution of compressible turbulent flows, *AIAA Journal*. **36**(12), 2187–2194.

Johnson, C. and P. Hansbo (1992). Adaptive finite element methods in computational mechanics, *Computer Methods in Applied Mechanics and Engineering*. **101**(1–3), 143–181.

Laflin, K. R. (1997). *Solver-Independent r-Refinement Adaptation for Dynamic Numerical Simulations*, Doctoral Thesis, North Carolina State University.

Mastin, C. W. (1999). Truncation error on structured grids, in *Handbook of Grid Generation*, J. F. Thompson, B. K. Soni, and N. P. Weatherill, eds., Boca Raton, CRC Press.

McRae, D. S. (2000). r-refinement grid adaptation algorithms and issues, *Computer Methods in Applied Mechanics and Engineering*. **189**, 1161–1182.

McRae, D. and K. R. Laflin (1999). Dynamic grid adaption and grid quality, in *Handbook of Grid Generation*, J. F. Thompson, B. K. Soni, and N. P. Wetherill, eds., Boca Raton, FL, CRC Press, 34-1–34-33.

Patra, A. and J. T. Oden (1997). Computational techniques for adaptive hp finite element methods, *Finite Elements in Analysis and Design*. **25**(1–2), 27–39.

Peraire, J., M. Vahdati, K. Morgan, and O. Zienkiewicz (1987). Adaptive remeshing for compressible flow computations, *Journal of Computational Physics*. **72**(2), 449–466.

Rannacher, R. and F. T. Suttmeier (1997). A feed-back approach to error control in finite element methods: application to linear elasticity, *Computational Mechanics*. **19**(5), 434–446.

Roy, C. J. (2003). Grid convergence error analysis for mixed-order numerical schemes, *AIAA Journal*. **41**(4), 595–604.

Roy, C. J. (2009). *Strategies for Driving Mesh Adaptation in CFD*, AIAA Paper 2009–1302.

Stewart, J. R. and T. J. R. Hughes (1996). A posteriori error estimation and adaptive finite element computation of the Helmholtz equation in exterior domains, *Finite Elements in Analysis and Design*. **22**(1), 15–24.

Thompson, J. F., Z. U. A. Warsi, and C. W. Mastin (1985). *Numerical Grid Generation: Foundations and Applications*, New York, Elsevier (www.erc.msstate.edu/publications/gridbook/).

Venditti, D. A. and D. L. Darmofal (2003). Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows, *Journal of Computational Physics*. **187**, 22–46.

Verfurth, R. (1999). A review of a posteriori error estimation techniques for elasticity problems, *Computer Methods in Applied Mechanics and Engineering*. **176**(1–4), 419–440.

Wood, W. A. and W. L. Kleb (1999). *On Multi-dimensional Unstructured Mesh Adaption*, AIAA Paper 1999–3254.

Zienkiewicz, O. C. and J. Z. Zhu (1992a). The superconvergent patch recovery and a posteriori error estimates, Part 2: Error estimates and adaptivity, *International Journal for Numerical Methods in Engineering*. **33**, 1365–1382.

Zienkiewicz, O. C. and J. Z. Zhu (1992b). Superconvergent patch recovery (SPR) and adaptive finite element refinement, *Computer Methods in Applied Mechanics and Engineering*. **101**(1–3), 207–224.