AUTOMATIC RESOLUTION CONTROL FOR THE FINITE-VOLUME METHOD, PART 2: ADAPTIVE MESH REFINEMENT AND COARSENING

H. Jasak
Computational Dynamics Ltd., Hythe House, 200 Shepherds Bush Road, London W6 7NY, England

A. D. Gosman
Department of Mechanical Engineering, Imperial College of Science, Technology and Medicine, Exhibition Road, London SW7 2BX, England

Following the description of a-posteriori error estimates in [1], this article describes an automatic adaptive h-type mesh refinement and coarsening procedure with directional sensitivity, based on the estimated error and solution gradients. Mesh adaptation operates on hexahedra and degenerate hexahedra by hierarchically splitting cells into two, four, or eight new cells and uses “I-irregularity” to preserve mesh smoothness. Arbitrary coarsening is based on cell pairing and controlled by the estimated error distribution and geometric considerations. The method is tested on two test cases in 2-D to examine its error reduction rate and the quality of refined meshes.

1. INTRODUCTION

In the previous article [1], the tools aimed at estimating the magnitude of the discretization error in a numerical solution for the finite-volume method (FVM) were presented. Although knowing the level of discretization error is useful in its own right, the requirements on accuracy are usually of a different kind. A typical situation, from the point of view of an analyst, is that the desired accuracy of the numerical solution is known in advance and the estimated error level is used to modify the discretization to reach the given objective. The desired solution accuracy should be reached with the smallest expense possible, in terms of time and computational resources. The optimal mesh, irrespective of its size, is therefore the one where the error is uniformly distributed through the domain and corresponds to the desired accuracy. In practice, it is virtually impossible to reach this ideal. However, once the solution and hence the a-posteriori error estimate are available, the local discretization accuracy can be adjusted toward this ideal.

The body of literature dealing with adaptive mesh algorithms published to date is considerable. The major categories of mesh adaptation are:

Received 11 January 2000; accepted 28 March 2000.
Address correspondence to Dr. Hrvoje Jasak, Computational Dynamics Ltd., Hythe House, 200 Shepherds Bush Road, London W6 7NY, England. E-mail: h.jasak@cd.co.uk
h-refinement [2–12], where computational points are inserted locally in regions of high numerical error without disturbing the rest of the mesh. Here, we shall discriminate between the “overlapping grid methods” (e.g., [2–5]), where “patches” of refinement are superimposed on the original mesh and a “coupling” procedure is prescribed, and “embedded grids” [6–12], where the complete computational domain is treated in a uniform fashion. 

r-refinement [13–18], where the number of computational points is kept constant, but their distribution is optimized to minimize the discretization error. This approach lacks generality [2], especially in complex 3-D geometries. Also, as the number of computational points remains the same, it may not be possible to reduce the discretization error to the desired level.

p-refinement, where the local order of discretization is adjusted to minimize the error. This method has proven to be efficient for problems with smooth solutions and is popular in conjunction with the finite-element method (FEM) of discretization.

Hybrids of the above, the leading example being the h–p adaptive method in the FEM, e.g., [19–21] and the work by Habashi et al. on h–r methods [22–24]. This work is particularly interesting because it provides higher efficiency by combining the advantages of h- and r-refinement, as well as the option of mesh coarsening.

Automatic resolution control (ARC), consisting of numerical solution–a-posteriori error estimation–mesh adaptation cycles, aims at automatically creating a solution of prespecified accuracy. The issue of a-posteriori error estimation for the FVM has been addressed previously [1]. In this article, options in adaptive mesh refinement for steady-state problems will be examined. Having in mind the characteristics of the FVM on arbitrarily unstructured meshes (see, e.g., [1, 25]), we shall limit ourselves to the control of the local mesh spacing rather than the order of discretization.

The preferred method of adaptation in this study is h-refinement: in conjunction with arbitrarily unstructured mesh design [where the number of neighbors for a control volume (CV) is not specified in advance] and stable and accurate
second-order FV discretization, it gives considerable freedom in the construction of the mesh.

The rest of the article is organized as follows. Section 2 describes the mesh refinement and coarsening algorithms together with the refinement criteria and the solution mapping procedure between meshes. The adaptive algorithm is tested on a problem with an analytical solution in Section 3, where it is possible to examine the error reduction rate for both refinement with and without coarsening. Finally, in Section 4, the algorithm is applied to a supersonic flow test case, where the potential of mesh adaptivity can be clearly seen. The findings of the article are briefly summarized in Section 5.

2. REFINEMENT ALGORITHM

The adaptive local mesh refinement and coarsening procedure used in this study consists of the following steps.

1. An initial computational mesh is created. If this mesh is not fine enough to describe the shape of the computational domain to desired accuracy, an appropriate (discrete) description of the boundary is also provided.
2. The discretized set of equations is solved on the available mesh.
3. The numerical error is estimated. If the desired level of accuracy is reached, the calculation is stopped.
4. The estimated error is used to decide what changes should be made in the mesh in order to remove the high error. In regions of high error, the mesh is locally refined by cell splitting with appropriate alignment with solution gradients. If, in some parts of the domain, the estimated error is much lower than the average, the mesh may also be locally coarsened.
5. The existing mesh is modified according to the specified requirements, including the correction of the boundary shape (if necessary). Some additional changes may also be introduced in order to preserve the mesh quality.
6. The numerical solution is mapped to the new mesh and used as an initial guess for the next calculation.

Steps 2–6 are repeated until the prescribed accuracy is reached. In order to complete the algorithm successfully, we need to examine the following problems:

How to select the regions for refinement and coarsening from a given error distribution
How to modify the mesh and what additional changes are needed to preserve the mesh quality
How to map the numerical solution from one mesh to the other with minimum degradation of the solution accuracy

The first question will be answered in Section 2.1: refinement and coarsening criteria are based on the analysis of the estimated error field. The second question is addressed in Section 2.2: the mesh changes are based on a fully implicit, embedded cell-by-cell refinement and coarsening with additional regularity constraints. Finally, the solution transfer between meshes is briefly discussed in Section 2.3.
2.1. Refinement and Coarsening Regions

The desired local mesh size is determined from the estimated error magnitude $e$, local mesh size $h$, and the order of discretization. If the desired error level is $E_0$, the desired local mesh size $l_0$ for the second-order FVM follows from the error scaling ($e \sim h^2$):

$$l_0 = h \sqrt{E_0 \over e}$$  \hfill (1)

The $l_0$ field from Eq. (1) can be subsequently smoothed to preserve reasonable mesh grading. However, for the proposed algorithm it is more convenient to compare the local and the mean error and use the following criteria:

$$\begin{cases}
    e_{\text{local}} > \lambda_r \bar{e} & \text{cell marked for refinement} \\
    e_{\text{local}} < \lambda_c \bar{e} & \text{cell marked for coarsening}
\end{cases}$$  \hfill (2)

Parameters $\lambda_r > 1$ and $\lambda_c < 1$ control the number of cells and grading of the refined mesh.

A good locally refined mesh takes into account not only the spatial distribution of the error, but also the mesh smoothness and alignment with solution gradients. Lists of cells described above describe only the spatial error distribution. In addition to that, we shall explicitly prescribe the alignment criterion for every cell in the list. The desired refinement direction is obtained from the gradient of the variable $\phi$ on which the error is estimated:

$$R = \frac{\nabla \phi}{|\nabla \phi|}$$  \hfill (3)

If the problem under consideration includes the solution of more than one transport equation, cells for refinement are combined into a single list, with an appropriate adjustment of the directionality information.

2.2. Mesh Adaptation Procedure

A cell marked for refinement is split based on the dot product of the refinement direction vector $R$ and the geometric moment tensor $M = \int_V (x - x_P)^2 \, dV$ for the cell, where $x_P$ is the cell centroid. Consider a hexahedral cell, Figure 1, with vectors $1$, $2$, and $3$ representing the principal axes of inertia (eigenvectors of $M$). The decision on the cell split is based on the magnitude of the dot product of $1$, $2$, and $3$ with $R$. The importance of direction $i$ is estimated as

$$\tau_i = |R \cdot i| \quad i = 1, 2, 3$$  \hfill (4)

If $\tau_i$ for a certain direction is larger than $\xi \bar{\tau}$ (where $\bar{\tau} = \frac{1}{3} \sum_i \tau_i$), the cell is split normal to $\tau_i$. Higher $\xi$ implies stronger alignment to the gradients; lower values result in a more isotropic refinement.

Options on smooth mesh grading with local refinement are limited. In this study, a concept of “1-irregularity” [19] will be used and adapted for hexahedral meshes. The criterion requires a cell to have a maximum of 7 neighbors. If this
is not the case, the refinement is propagated into the neighborhood until the condition is satisfied for all cells in the mesh.

Unlike the cells marked for refinement, where the refinement is forced, cells marked for coarsening can only potentially be removed, depending on the mesh regularity requirements. In this study we shall adopt a general coarsening procedure with the possibility of coarsening beyond the initial mesh.

Mesh coarsening is based on a “cell pair” concept, where the list of cells for coarsening is scanned to create “cell pairs,” which can subsequently be merged into a single cell, subject to the regularity criterion. Cell pairs are constructed in a manner which minimizes the cell surface per unit volume, thus preferentially creating isotropic coarse meshes (a cube is the preferred shape for a merged cell). As the “history” of mesh refinement is not stored, the recovery of the initial mesh after refinement and coarsening is not guaranteed.

2.3. Mapping of Solution between Meshes

In order to speed up the solution of the problem on a refined mesh, a good initial guess is needed. An approximate solution of the same problem is already available and can be used for this purpose.

The mapping of the solution between two meshes is a relatively straightforward task. The assumed variation of the function over each CV is linear. All fields defined on cell centers can be mapped by finding the closest point $T$ in the original mesh for every cell center of the target mesh and using

$$
\phi_p = \phi_T + (x_p - x_T) \cdot (\nabla \phi)_T
$$

The face flux transfer should be performed in such a way that the local mass conservation constraint is enforced on the target mesh. The “obvious” practice of summing up the fluxes of the available solution is not considered satisfactory because, as new faces are introduced, it typically results in a constrained balance problem. Instead, the conservative fluxes are recalculated by reassembling the (mass) conservation equation from the interpolated cell-center values. Details of an appropriate procedure are given in [26].
3. NUMERICAL EXAMPLE

In order to examine the performance of the adaptive algorithm, we shall compare its error reduction rate to that of uniform refinement on a test case with an analytical solution. For this purpose, the 2-D point source in cross-flow test case described in [1] will be revisited.

The test setup consists of a fixed strength point source of a passive scalar in a fixed uniform velocity field in 2-D, on a mesh which is originally aligned with the flow. The analytical solution is given in [27], with the details of the setup identical to [1].

The objective of this test case is to examine the robustness of the proposed adaptive procedure, its sensitivity to the accuracy of the error estimate, and the relative merit of the refinement-only versus refinement with coarsening. We can also visually examine the mesh quality as imposed by “1-irregularity” and the practical limitations in the number of embedded levels of refinement.

3.1. Refinement-Only Calculations

In the first instance, the adaptive refinement starts from a coarse mesh (10×6 CVs). Refinement parameters are set to $\lambda_r = 1.5$ and $\xi = 0.6$, and the moment error estimate (MEE) [1] is used to estimate the discretization error. The initial mesh and several levels of adaptive refinement are shown in Figure 2. The final mesh, with 10 embedded levels of refinement, Figure 2h, consists of 2,564 CVs. The “1-irregularity” principle successfully enforces a smooth transition between the coarse and fine regions. In spite of the very high ratio of minimum to maximum cell volume and second-order-accurate convection discretization, no convergence problems have been encountered.

Existence of the analytical solution allows us to examine the reduction of the “exact” error with refinement, shown in Figure 3, proving the superiority of the adaptive procedure over uniform refinement. Additional benefit in terms of accuracy is achieved by the fact that the adaptively refined mesh is considerably finer in regions of high gradients than its uniform counterpart with the same number of cells.

Clearly, the choice of error estimate influences the performance of mesh adaptivity. In order to determine its effect, the calculation is repeated, basing the refinement criterion on the exact error (available from the analytical solution) and the Taylor series error estimate (TSEE) [1], Figure 3b. The results show a relatively small scatter and some interesting features. The error reduction is slowest when refinement is based on the exact error: this, unlike the error estimates, highlights the error location rather than its source. The error reduction is fastest in early stages of refinement; the final stages are less efficient, presumably as a consequence of mesh-induced discretisation errors. Also, after the 7th level of refinement, the maximum error remains the same. This is attributed to the fact that the error reduction due to improved mesh resolution is counterbalanced by the degradation of the mesh quality.
Figure 2. Adaptive mesh refinement. (a) Initial mesh. (b) First level of refinement. (c) Second level of refinement. (d) Third level of refinement. (e) Fourth level of refinement. (f) Fifth level of refinement. (g) Sixth level of refinement. (h) Tenth level of refinement.
3.2. Refinement with Coarsening

Here, we shall start the ARC from a relatively fine initial mesh and use both refinement and coarsening to improve the mesh quality. In order to examine the performance of the coarsening procedure, both refinement-only and refinement with coarsening calculations with be performed.

The initial mesh consists of $80 \times 40$ CV, and is considered too fine in the smooth parts of the solution and too coarse close to the singularity. The refinement and coarsening parameters are set to $\lambda_r = 1.5, \lambda_c = 0.5,$ and $\zeta = 0.6.$ Figure 4 shows the initial mesh and the adapted mesh after 2 levels of refinement and coarsening. In the region of smooth gradients far downstream from the point source, the estimated error is low and the cells have been merged by coarsening, thus decreasing the cell count. A comparison of the maximum error scaling for refinement-only versus refinement with coarsening is shown in Figure 4d.

4. SUPERSONIC FLOW EXAMPLE

In this section we shall present the application of the ARC to a supersonic flow with strong shocks. The calculations were performed using the FOAM C++ CFD library [28] developed by H. G. Weller et al. in collaboration with the authors. The FOAM library implements a generic second-order-accurate FVM discretization on arbitrarily unstructured meshes [26], consistent with [1]. The pressure-based compressible flow solver uses a segregated solution procedure with a variant of PISO [29], with the Gamma differencing scheme [30] used on all convection terms.

The test setup was introduced by Emery [31] and consists of a Mach 3 flow of an ideal gas hitting a forward-facing step and creating a bow shock and an overexpansion region at the corner, followed by a weak shock originating just downstream of the edge [32]. The shock is reflected from the top boundary and gives a point of shock-to shock interaction. In the first instance, the problem is solved on a coarse and a fine uniform mesh, with 840 and 53000 CVs, respectively, giving the Mach number distribution presented in Figure 5.
Figure 4. Refinement with coarsening. (a) Initial mesh, 3200 CVs. (b) Refinement-only, 6438 CVs. (c) Refinement with coarsening, 4481 CVs. (d) Scaling of the maximum error.
In the adaptive calculations the MEE is used to highlight the error sources. Three levels of mesh refinement starting from the coarse initial mesh are shown in Figure 6a. A detail of the refined mesh, showing the “1-irregularity” principle in action, is shown in Figure 6b. The grading between the coarse and fine parts of the mesh is relatively smooth, preserving the quality of the refined mesh around the shock. If the “1-irregularity” were not obeyed, rapid grading would cause high mesh-induced errors, thus degrading the solution quality.

The Mach number distribution after five refinement levels is shown in Figure 6a. The shock resolution on this mesh with 15,220 CVs is clearly superior to the solution on a uniform mesh with 53,000 CVs, Figure 5b, demonstrating the potential of the adaptive algorithm.

The quality of the solution, however, cannot be judged only by the resolution of strong shocks. The fine uniform mesh resolves the weak shock originating from the corner of the step reasonably well. Although the adaptive solution picks up the same feature, its resolution is not comparable with the resolution of the main shock structure. The key to the problem lies in the fact that the weak shock is not picked up at all on the initial mesh, Figure 5a.
Figure 6. Supersonic flow: adaptive refinement. (a) Adaptive refinement. (b) Detail of the mesh. (c) Mach number distribution for the final mesh.
This situation could be improved with the adaptive procedure starting from a finer mesh and using both mesh refinement and coarsening. Such a mesh is shown in Figure 7a, with the corresponding Mach number distribution in Figure 7b.

The ARC is ideally suited for supersonic flow problems—a wealth of literature on the subject is a good indicator of its popularity. Here, the region of the domain which benefits from increased resolution is easily recognized. In terms of required mesh size, the local refinement reduces the dimensionality of the problem by one:

(a) Refinement/coarsening.

(b) Mach number distribution for the final mesh.
in 2-D, additional refinement is needed on 1-D shock lines, considerably reducing the computational effort compared to uniform mesh refinement.

5. SUMMARY

This article presented some details of the adaptive local mesh refinement and coarsening procedure adopted for the automatic resolution control (ARC) algorithm in conjunction with the error estimates described in [1]. The algorithm for error-driven adaptive mesh refinement procedure consists of several parts: analysis of the error distribution, selection of regions of refinement and coarsening, adaptation of the computational mesh, and mapping of the solution from one mesh to another.

The selected procedure changes the existing computational mesh in an $h$-manner, allowing for both mesh refinement and arbitrary coarsening. Smooth mesh grading is preserved through the concept of “1-irregularity.” The mesh refinement is controlled by three parameters: $\lambda_r$ and $\lambda_c$, controlling the grading between the coarse and fine regions; and $\xi$, determining the level of mesh alignment with the solution gradients. When the new mesh is created, an interpolate of the available solution is used as an initial guess.

A test case with an analytical solution has been solved using the ARC and the performance of the algorithm has been examined in terms of absolute error reduction, showing its superiority over uniform refinement.

The adaptive refinement algorithm is at its best when the error caused by insufficient mesh resolution covers only a small portion of the computational domain. In this case, local refinement efficiently removes the error by augmenting mesh resolution in the affected areas. The error estimation in such cases is also relatively simple, as it is only necessary to highlight the points of high error. A typical example of such a situation is a supersonic flow with shocks.

The benefits of mesh adaptivity in supersonic flows have been recognized for some time. However, the main interest of this study is to establish the potential and limitations of the ARC in problems with smooth solutions and in particular turbulent incompressible flows. The final article in this series [33] will present the application of the adaptive methodology constructed and validated here to laminar and turbulent flows, which presents a considerably more challenging environment for error-driven automatic resolution control.

REFERENCES


