

A h -Adaptive Algorithm Using Residual Error Estimates for Fluid Flows

N. Ganesh¹ and N. Balakrishnan^{2,*}

¹ *Department of Mechanical Engineering, Indian Institute of Technology Guwahati, Guwahati 781039, Assam, India.*

² *Computational Aerodynamics Laboratory, Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560012, India.*

Received 17 August 2011; Accepted (in revised version) 21 February 2012

Available online 28 June 2012

Abstract. Algorithms for adaptive mesh refinement using a residual error estimator are proposed for fluid flow problems in a finite volume framework. The residual error estimator, referred to as the \mathfrak{R} -parameter is used to derive refinement and coarsening criteria for the adaptive algorithms. An adaptive strategy based on the \mathfrak{R} -parameter is proposed for continuous flows, while a hybrid adaptive algorithm employing a combination of error indicators and the \mathfrak{R} -parameter is developed for discontinuous flows. Numerical experiments for inviscid and viscous flows on different grid topologies demonstrate the effectiveness of the proposed algorithms on arbitrary polygonal grids.

AMS subject classifications: 65M08, 65M50, 65Z05, 76N15

Key words: Adaptive refinement, error estimator, residual, error indicator.

1 Introduction

The primary focus in numerical simulations of practical engineering problems which invariably involve complexities in both geometry and flow, is to obtain accurate numerical solutions at shorter turnaround times. Unfortunately, the computational effort in obtaining the solutions and the solution accuracy are at conflict: finer meshes lead to accurate solutions at greater computational effort, while coarse meshes involve lower computational effort but result in inaccurate solutions. *Adaptive Mesh Refinement* (AMR) algorithms constitute a class of computational methods that strike the right balance between numerical solution accuracy and the associated computational effort and provide

*Corresponding author. *Email addresses:* n.ganesh@iitg.ernet.in (N. Ganesh), nbalak@aero.iisc.ernet.in (N. Balakrishnan)

a viable and economical approach for simulation of complex flow problems. AMR algorithms have been employed by several researchers for a variety of flow problems ranging from shock hydrodynamics [1] and compressible flows [2] to multi-phase flows [3] and astrophysical applications [4].

Sensors employed in the AMR algorithm can be broadly classified as *Error Indicators* and *Error Estimators*. Several adaptive algorithms in the past have relied on error indicators, mainly due to their ability to precisely detect flow phenomena. Some of the popular error indicators include curl and divergence of velocity [5] and divided differences in density [6,7] for shocked flows and vorticity gradient [8] for vortex dominated flows. Error indicators, however, provide no information on error levels in the domain which can result in a termination criterion for refinement. Consequently, all indicator based adaptive refinement algorithms resort to heuristic considerations involving user-defined parameters that are problem-dependent. Error estimators, on the other hand, provide a reasonable estimate of some error distribution in the domain and can therefore be exploited to derive a suitable termination criterion. Error estimators can be broadly classified as global error based estimators, adjoint based estimators and residual estimators. Since the focus of this work is on residual error estimation, we shall discuss only this class of estimators in detail. For a review on other error estimators, refer to [9,10] and references therein.

Residual error estimation have been extensively studied in the finite element framework [11], though only few studies in the context of finite volume framework are reported in literature. Particularly in the context of finite volume computations, the residual error estimators provide an estimate of the local truncation error, which is the extent to which the discrete algebraic equation differs from the partial differential equation it models. Jasak and Gosman [12] proposed the element residual estimate for finite volume discretisations. Aftsomis and Berger [13] compute the local truncation error using an approximation of the exact solution. Hay and Visonneau [14] have proposed the use of a higher order reconstruction operator to compute the residual, while Karni and Kurganov [15] have introduced the concept of weak local residuals to compute the truncation error. Roy and Sinclair [16] have developed an interesting approach where the analytical expression for the numerical solution obtained using curve-fitting techniques is employed to estimate the truncation error. The authors in one of their earlier works have also proposed an estimator called the \mathfrak{R} -parameter which is a novel and generic approach to residual estimation in a finite volume framework [17–20].

There are several merits to the use of \mathfrak{R} -parameter in an adaptive strategy as amply demonstrated in reference [17]: (i) The computation of \mathfrak{R} -parameter is cheap and does not require any approximation to the exact solution (in contrast to the estimators of the global error). (ii) It is possible to obtain theoretically the rate of fall of the \mathfrak{R} -parameter with grid refinement and this is not so obvious in the case of global errors. (iii) Apart from this, the \mathfrak{R} -parameter marks the source of errors in problems involving error transport and therefore offer better control over the global error (the ultimate objective of any adaptive calculation) as compared to global error based mesh adaptation. (iv) One of the

most important advantages resulting from the use of \mathfrak{R} -parameter is that it is possible to evolve length scales for adequately resolving the local flow features. This obviates heuristic user defined inputs in order to effect mesh adaptation. Also, this provides an automatic termination criterion to the adaptive process, at least in regions where the flow is continuous. In order to realize this objective, in the earlier work, the authors have introduced the concept of dissipation function, which is effectively made use of in arriving at length scales required to resolve local flow features [17]. The present work can be considered as yet another way of exploiting the \mathfrak{R} -parameter wherein the tools required for computing the aforementioned length scales can be considerably simplified. Adaptive algorithms based on this alternate methodology are developed for continuous and discontinuous flows and are investigated for inviscid and viscous fluid flow problems on arbitrary polygonal meshes.

The organization of the paper is as follows. Section 2 discusses the basic solution methodology. For sake of completeness, aspects of the \mathfrak{R} -parameter are briefly discussed in Section 3. The adaptive algorithm for continuous flows based on the \mathfrak{R} -parameter is presented in Section 4, while a hybrid adaptive algorithm for discontinuous flows is discussed in Section 5. Numerical experiments using the proposed adaptive algorithms and the efficiency of these algorithms constitute Section 6.

2 Solution methodology

The governing equations for two-dimensional unsteady compressible fluid flows in an inertial frame of reference can be written as

$$U_t + f_x + g_y = 0,$$

where $U = [\rho \ \rho u \ \rho v \ e]^T$ is the vector of conserved variables and f and g are the flux vectors in x and y directions respectively. The governing equations are complemented by the ideal gas equation of state, $p = \rho \bar{R} T$, where \bar{R} is the gas constant. This system of non-linear, coupled governing laws are solved in an unstructured data based finite volume framework. In the present study, a cell-centered finite volume procedure [22] is employed to obtain the numerical solutions. Inviscid flux computations are performed using the approximate Riemann solver of Roe [23] or the Flux Vector Splitting scheme of van Leer [24], while the viscous fluxes are discretised using Green-Gauss procedure [21, 25, 26]. Linear reconstruction [27, 28] is employed to enhance solution accuracy. A least squares based reconstruction procedure is employed for inviscid computations [22, 27] and a diamond path based reconstruction for the viscous computation [21]. Venkatakrishnan limiter is used to enforce solution monotonicity [29] and an implicit scheme using the symmetric Gauss-Seidel procedure [22] is employed for advancing the solution in time. No-slip boundary condition is enforced on solid boundaries for viscous flows, while mirror boundary condition is employed for inviscid flows [30]. Characteristic boundary conditions [31] are enforced on the farfield boundaries. Supersonic inlet and outlet boundary conditions [31] are made use of for internal supersonic flows.

3 Residual error estimation

A new residual error estimator, referred to as \mathfrak{R} -parameter, for hyperbolic conservation laws in a finite volume framework was developed by the authors in an earlier work [17]. This error estimator is a posteriori and gives a measure of the local truncation error. In this section, we briefly review the theory and computation of the \mathfrak{R} -parameter as well as its salient properties, for sake of completeness.

3.1 Theory of error estimation

Consider the governing partial differential equation be represented in operator form as

$$I[U]=0, \quad (3.1)$$

where I represents the exact operator and U the exact solution. The discrete solution u satisfies the discrete conservation law given by

$$\delta^1[u]=0, \quad (3.2)$$

where δ^1 is the discrete operator. Evidently, the use of the exact operator on the numerical solution would lead to an imbalance and then we have

$$I[u]=R^1[u]. \quad (3.3)$$

In Eq. (3.3), R^1 represents the local truncation error (LTE). On a discretised domain, we can obtain an estimate of this truncation error only by employing another discrete operator δ^2 approximating I . This results in

$$\delta^2[u]=R^1[u]-R^2[u]=\mathfrak{R}[u]. \quad (3.4)$$

The quantity $R^2[u]$ is the error due to the discretisation of the exact operator I while $R^1[u]$ is the imbalance due to the use of the exact operator on the numerical solution. The quantity $\delta^2[u]$ or $\mathfrak{R}[u]$ is referred to as \mathfrak{R} -parameter. If $R^1[u] \sim \mathcal{O}(h^m)$ and $R^2[u] \sim \mathcal{O}(h^n)$, the \mathfrak{R} -parameter is an estimate of the local truncation error for $m < n$. This implies that the numerical operator δ^2 should necessarily be of higher order accuracy as compared to δ^1 .

3.2 Computation of the \mathfrak{R} -parameter

The theory of the \mathfrak{R} -parameter presented above is generic and can be applied to any system of conservation laws. For the Euler/Navier-Stokes equations in a finite volume framework the computation of the \mathfrak{R} -parameter reduces to the computation of the flux imbalance (also known as the *residual*) given by

$$\mathfrak{R}_i = \frac{1}{\Omega_i} \sum_k \left[\sum_l w_l \vec{F}_{k,l} \cdot \hat{n}_k \right] \Delta s_k, \quad (3.5)$$

where l is the number of Gaussian points used in the flux quadrature, w_l are the Gaussian weights and \hat{n}_k and Δs_k are the unit normal and edge length of the face k of volume Ω_i . A least squares based quadratic polynomial reconstruction procedure is used to estimate the \mathfrak{R} -parameter while the solution is obtained using a linear reconstruction procedure as discussed in Section 2. While the linear reconstruction procedure offers formal first order accuracy on irregular grids, the quadratic reconstruction procedure offers second order accuracy [17,27,28]. This ensures that the errors due to discretisation of the exact operator does not dominate the error due to the imbalance. It must be emphasized that the same stencil of points are employed for the reconstruction associated with the state update and higher order reconstruction associated with the \mathfrak{R} -parameter.

3.3 Properties of the \mathfrak{R} -parameter

The \mathfrak{R} -parameter satisfies certain properties which makes it an attractive choice for driving mesh refinement algorithms. These properties are summarized here for sake of completeness and the interested reader is referred to [17] for a detailed discussion on the same.

1. The \mathfrak{R} -parameter is consistent in smooth flow regions which are free of limiters but is rendered inconsistent in limiter-active and discontinuous flow regions.
2. The \mathfrak{R} -parameter is a measure of "Cell error" and therefore detects the sources of error. This property renders the \mathfrak{R} -parameter based adaptive strategies with excellent global error control.
3. The \mathfrak{R} -parameter works equally well on structured and unstructured meshes and is inexpensive to compute. It requires no additional information than what is required by the solution procedure itself and the effort to compute the \mathfrak{R} -parameter is comparable to one explicit iteration of the flow solver.

4 Residual Adaptive Strategy (RAS) for continuous flows

Adaptive refinement algorithms for fluid flow problems are critically influenced by the refinement/derefinement criteria, which in turn depend on the choice of sensors employed in the algorithm. It has already been established that the \mathfrak{R} -parameter is consistent everywhere for smooth and continuous flows devoid of limiting. For such flows, it is possible to employ the \mathfrak{R} -parameter itself to derive refinement/derefinement criteria, leading to a purely-residual based adaptive strategy referred to as RAS algorithm.

Resolution of flow features are linked to the error levels on the mesh on which they are resolved. Sufficient resolution of flow features on a given mesh results in error levels lesser than some specified threshold value. It is therefore possible to define a threshold value for the error and then determine a length scale based on this threshold that resolves the flow phenomena to the desired extent on a given mesh. As in the previous work of the authors [17], the design of refinement (or derefinement) criterion involves defining

a threshold value of the \mathfrak{R} -parameter, \mathfrak{R}^* and recovering a threshold length scale h_c^* for every volume using \mathfrak{R}^* , for adequately resolving the local flow features. The decision to refine or derefine a given volume is taken by comparing h_c^* with the length scale associated with the volume. The RAS algorithm hinges on the following hypothesis.

Hypothesis. For a grid l^p at any level p ($p \geq 1$), the volumes which continue to remain at the base level ($p=0$), have the lowest values of the \mathfrak{R} -parameter.

Define

$$\|\mathfrak{R}\|_1 = \frac{\sum_{c \in l^0} |\mathfrak{R}_c|}{N_{l^0}}, \quad \|\mathfrak{R}\|_\infty = \max_{c \in l^0} |\mathfrak{R}_c|, \quad (4.1)$$

where l^0 refers to the set of cells at the initial (or base) level, N_{l^0} is the number of undivided volumes at the base level and $\|\mathfrak{R}\|_1$ and $\|\mathfrak{R}\|_\infty$ denote the L_1 and L_∞ norms of the \mathfrak{R} -parameter respectively. We then define a global threshold value of the \mathfrak{R} -parameter as

$$\mathfrak{R}^* = \frac{1}{2} (\|\mathfrak{R}\|_1 + \|\mathfrak{R}\|_\infty).$$

It is important to note that this definition of \mathfrak{R}^* is not unique; this is merely a pragmatic approach resulting in a balanced refinement and derefinement strategy. In order to derive the threshold length scale h_c^* , we begin by rewriting the relationship[†] between the \mathfrak{R} -parameter and the characteristic length scale as $\mathfrak{R}_c = Ch_c^q$. Defining a threshold value for the \mathfrak{R} -parameter, \mathfrak{R}^* as described earlier, we also have $\mathfrak{R}^* = Ch_c^{*q}$. It follows that

$$h_c^* = h_c \left(\frac{\mathfrak{R}^*}{\mathfrak{R}_c} \right)^{\frac{1}{q}}. \quad (4.2)$$

The important distinction between the present methodology and that in [13] among others, is that the exponent ' q ' is non-integral in nature. The functional form of h_c^* in Eq. (4.2) accounts for the non-integral nature of the exponent, while the dissipation function accounts for the same. The threshold length scale can be determined if the value of the exponent ' q ' is known. However, computing the value of ' q ' numerically at a cell-level is complicated by the fact that the grid undergoes both refinement and derefinement simultaneously. Therefore, rather than determining the exponent ' q ' explicitly in each cell, we employ its approximate limits to estimate the threshold length scale. The \mathfrak{R} -parameter being an estimate of the local truncation error merely mimics the behaviour of the leading truncation term. Therefore the approximate limiting values of the exponent q are 1 in case of irregular grids and 2 in case of the regular grids, for a solution procedure employing a linear reconstruction. The following three different possibilities are now analysed to evolve a threshold length scale.

[†]It should be noted that this representation of \mathfrak{R}_c is different from our earlier representation [17], $\mathfrak{R}_c = X(h)h$ where $X(h)$ is the dissipation function.

- Case 1.* $\mathfrak{R}^* > \mathfrak{R}_c$ This is the case with volumes that are candidates for derefinement, since the levels of local truncation error are lower than the threshold \mathfrak{R} -parameter. It follows from Eq. (4.2) that for a given volume 'c', the minimum and maximum values of h_c^* occur when $q=2$ and $q=1$ respectively.
- Case 2.* $\mathfrak{R}^* < \mathfrak{R}_c$ This condition arises in volumes which are to be refined, since the truncation error levels are more than the threshold \mathfrak{R} -parameter. From Eq. (4.2) it is easy to see that the minimum and maximum values of h_c^* for a given volume 'c' occur when $q=1$ and $q=2$ respectively.
- Case 3.* $\mathfrak{R}^* \sim \mathfrak{R}_c$ This condition arises when the cell is neither to be refined nor derefined. We then have $h_c^* \sim h_c$, which is not surprising and the exponent q becomes insignificant.

In order that flow features of disparate strengths are equally resolved, a conservative adaptive strategy should encourage more refinement and discourage derefinement. This in turn means that h_c^* should be kept minimum in order to result in "maximal" refinement and "minimal" derefinement. It follows that the threshold length scale h_c^* can be computed as

$$h_c^* = \min \left(h_c \left(\frac{\mathfrak{R}^*}{\mathfrak{R}_c} \right), h_c \sqrt{\frac{\mathfrak{R}^*}{\mathfrak{R}_c}} \right). \quad (4.3)$$

The refinement and derefinement criteria for the RAS algorithm are given below.

Refinement criterion. A volume c on a grid l^p ($p \geq 1$) is flagged for refinement iff the characteristic length scale h_c associated with the volume satisfies $h_c > h_c^*$.

Derefinement criterion. A volume c on a grid l^p ($p \geq 1$) is flagged for derefinement iff the characteristic length scale h_c associated with the volume satisfies $2 \times h_c \leq h_c^*$.

A different refinement criterion is necessary on the initial level mesh. This is achieved by flagging a fixed percentage of cells where the \mathfrak{R} -parameter is maximum. Additional cells are flagged to ensure good grid quality and are based on considerations as outlined in [32]. Mesh refinement is based on *h-refinement* strategy which involves cell division/agglomeration. In particular, quadrilateral cells are isotropically divided into four quads and triangular cells into four triangles. The *h-refinement* algorithm is implemented using a unstructured data framework akin to the flow solver [32].

The proposed algorithm automatically evolves a threshold length scale and termination criteria for refinement/derefinement, is free of heuristic (user-defined and tunable) parameters and is easy to implement.

5 Hybrid Adaptive Strategy (HAS) for compressible flows

Compressible flows involve discontinuities such as shocks and contacts where limiters are inevitably operational leading to inconsistency of the \mathfrak{R} -parameter. The error indicators like curl and divergence of velocity, precisely mark these regions, where the

\mathfrak{R} -parameter loses its consistency. It is therefore natural to develop an algorithm that employs error indicators for refinement and \mathfrak{R} -parameter for derefinement. This leads to a “hybrid” algorithm which effectively exploits the mutual exclusivity of regions of validity of the error indicators and the \mathfrak{R} -parameter [17].

A major drawback that plagues error indicator-based algorithms is the lack of a “termination criterion” for refinement. In the absence of a stopping criterion, stronger flow features would get repeatedly refined, while the weaker flow features are not resolved. Fortunately, for cells flagged for refinement based on the indicators and where limiters are not operational, a threshold length scale can be derived based on the \mathfrak{R} -parameter itself. However, the termination criterion for cells marked for refinement where limiters are operational must be based on some heuristic considerations. For cells flagged for refinement by divergence of velocity and where limiters are operational, an explicit length scale based on the geometry is defined; e.g. 10^{-3} times the chord length could be useful for shocked flow past airfoils. For turbulent flow computations, empirical evidence suggest that the cells in the wall cells are limiter-operational; the y^+ values itself can serve as a termination parameter.

The computation of the threshold length scales using the \mathfrak{R} -parameter for the HAS algorithm is similar to that of the RAS algorithm (discussed in Section 4) with the notable exception that the hypothesis is modified to exclude cells at the initial level (cells constituting l^0) where limiters are operational. Also, in order to promote refinement of weaker flow features, a conservative estimate for the threshold value of the \mathfrak{R} -parameter is employed for refinement only. Consequently, the definition of the threshold value of the \mathfrak{R} -parameter, \mathfrak{R}^* , for the HAS algorithm is defined as

$$\mathfrak{R}^* = \begin{cases} \|\mathfrak{R}\|_1, & \text{for refinement,} \\ (\|\mathfrak{R}\|_1 + \|\mathfrak{R}\|_\infty)/2, & \text{for derefinement.} \end{cases} \quad (5.1)$$

This definition of the \mathfrak{R} -parameter is employed to recover the threshold length scales employed in the termination criterion for refinement and derefinement. Derefinement criterion is based on the residual error estimator as explained for the RAS algorithm. A *h-refinement* strategy is used to effect mesh refinement and coarsening, as in the case of the RAS algorithm.

6 Numerical simulations and AMR efficiency

6.1 AMR efficiency

The success of any adaptive algorithm is primarily gauged by its ability to obtain physically accurate solutions to fluid flow problems at a lower computational effort. Solution contours, aerodynamic coefficients and surface pressure and shear distributions provide a qualitative assessment of the adaptive algorithm. The AMR efficiency [38] is a quantitative measure of the computational efficiency of the AMR algorithm. The AMR efficiency

is defined as

$$\eta_{AMR} = 1 - \frac{N}{N_u}, \quad (6.1)$$

where N is the number of volumes in the adapted grid (at any given level) and N_u is the number of volumes on a uniformly adapted grid (at the same level). A higher value of the AMR efficiency points to an efficient adaptive algorithm. It must however be emphasized that both the AMR efficiency and solution quality are equally important to the success of the adaptive algorithm.

6.2 Numerical simulations using the RAS algorithm

The efficacy of the RAS algorithm is tested on inviscid and viscous flows devoid of discontinuities and limiters. The details pertaining to the test cases and the mesh are given in Table 1.

Table 1: Grid details and AMR efficiencies for test cases using the adaptive algorithms.

Test Case	Grid Type	Number of cells at initial level	Number of cells at final level	AMR efficiency η_{AMR}
Inviscid flow past NACA0012 airfoil	Structured grid	2450	22946	0.964
Laminar flow past NACA0012 airfoil	Structured grid	2450	20537	0.869
Inviscid flow past scramjet intake	Structured grid	7230	81921	0.823
Laminar flow past bump in a channel	Hybrid grid	2299	51778	0.912

6.2.1 Case 1: Inviscid subsonic flow past NACA0012 airfoil $M_\infty=0.63$, $\alpha=2^\circ$

The inviscid subsonic flow past a symmetric airfoil is investigated. Starting from an initial coarse grid, four levels of refinement and derefinement are performed. The initial, intermediate and final meshes along with the mach contours are shown in Fig. 1. The adaptive algorithm initially refines a larger region of the computational domain but eventually concentrates more points in the stagnation regions while coarsening the mesh in other regions of the domain. This manifests as a reduction in numerical entropy generation which is evident in the mach contours (see Fig. 1). The good agreement of the lift coefficient prediction with the AGARD data [33] and previous computations [34] also reflects the efficacy of the adaptive strategy (see Table 2).

Table 2: Comparison of lift coefficient for subsonic inviscid flow past NACA 0012 airfoil.

	AGARD data [33]	Previous computation [34]	RAS algorithm
C_L	0.3335	0.3047	0.3142

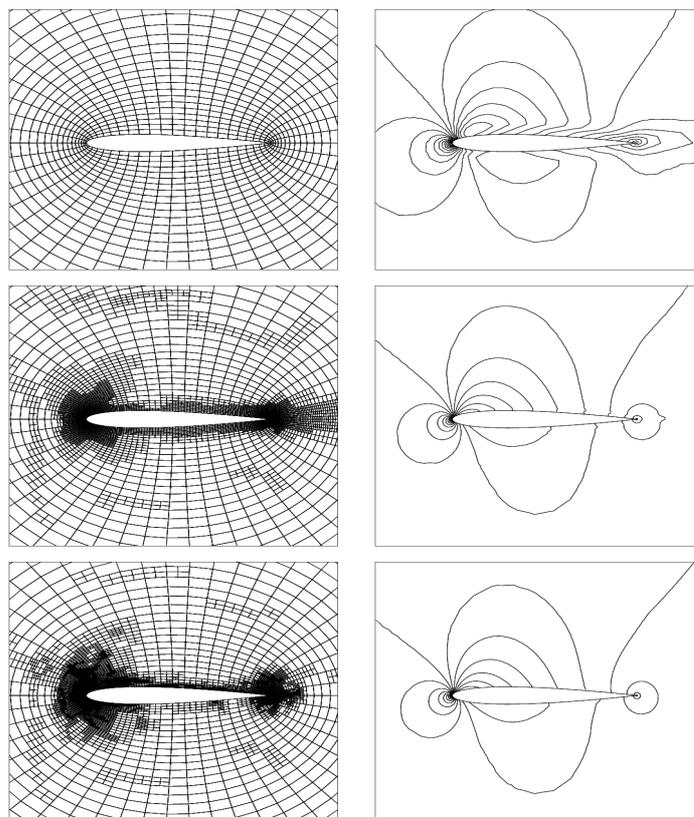


Figure 1: Grids (left) and Mach contours (right) with grid adaptation for subsonic inviscid flow past NACA0012 airfoil. Contour levels are 0:0.05:1.0

6.2.2 Case 2: Laminar subsonic flow past NACA0012 airfoil, $M_\infty = 0.5$, $Re_\infty = 5000$, $\alpha = 0^\circ$

The RAS algorithm is applied to the viscous flow past NACA0012 airfoil to study its ability to predict the boundary layer separation. Three levels of refinement-derefinement are performed starting from a coarse O-grid. The sequence of meshes and the corresponding mach contours are shown in Fig. 2. The pressure and skin friction distribution on the final adapted mesh shown in Fig. 3 indicate that the peak in suction pressure as well as the point of flow separation are accurately predicted. Comparisons of the predicted drag coefficient and separation point (Table 3) are in favourable agreement with previous computations [35], demonstrating the success of the adaptive algorithm.

Table 3: Comparison of aerodynamic coefficients for subsonic laminar flow past NACA 0012 airfoil.

	Venkatakrishnan [35]	Present method
C_D	0.0554	0.0560
Separation point	81.0	81.4

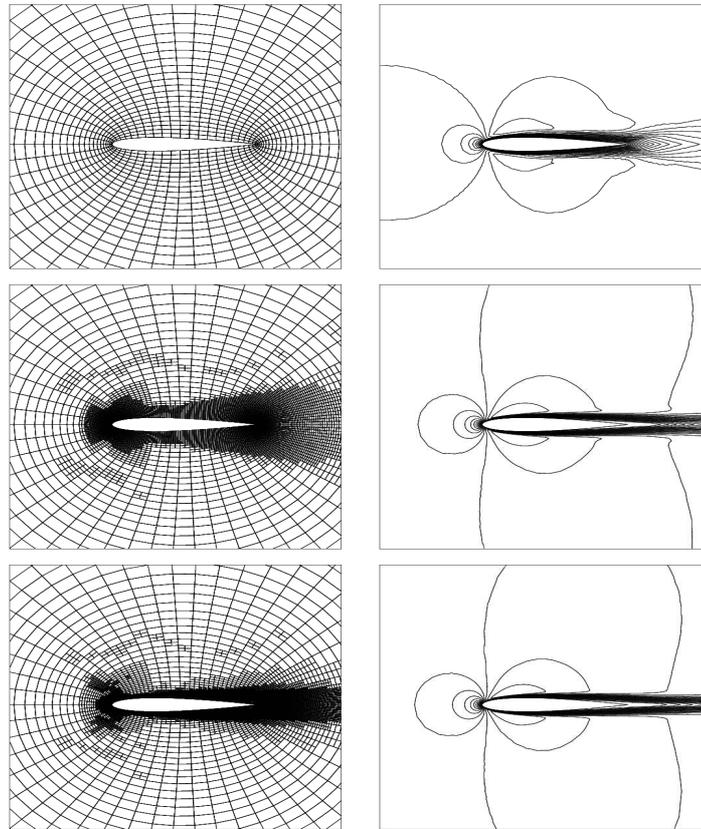


Figure 2: Grids (left) and Mach contours (right) with grid adaptation for subsonic laminar flow past NACA0012 airfoil. Contour levels are 0:0.03:0.6.

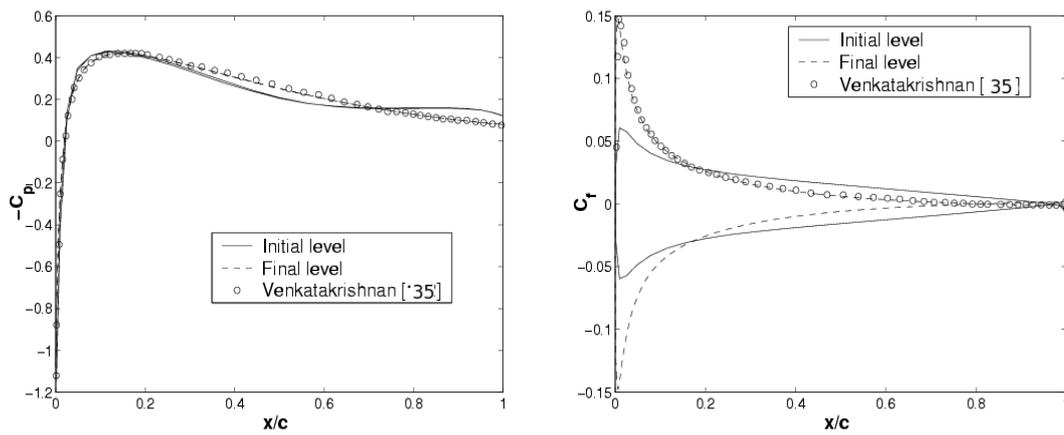


Figure 3: Pressure distribution (left) and Skin friction distribution (right) for the laminar flow past NACA airfoil.

6.3 Numerical simulations using the HAS algorithm

The effectiveness of the hybrid adaptive algorithm is studied using two representative test problems. These problems involve complex flow phenomena and provide the opportunity to judge the performance of the algorithm in handling compressible flows on arbitrary polygonal grids. The details of the grid are given in Table 1, while Table 4 provides the information on grid refinement.

Table 4: Grid refinement details for test cases using the hybrid adaptive algorithms.

Test Case	Error indicator(s)	Fraction marked by divergence [†]	Total percentage of cells divided [§]
Case 1	Divergence and curl	0.7,0.7,0.7	60,40,30
Case 2	Divergence and curl	0.5,0.5,0.5,0.5	70,45,30,20

[†] The fraction of cells marked by divergence for case where curl and divergence are employed as indicators, in each level of refinement.

[§] The total percentage of cells divided in the domain in each level of refinement.

6.3.1 Case 1: Inviscid hypersonic flow through a scramjet intake

The first test case is the inviscid hypersonic flow ($M_\infty=5$) through a two-dimensional scramjet intake [37]. The geometry for this case is shown in Fig. 4 and the flow is characterized by a complex pattern of shocks (and shock reflections), expansions and contact discontinuities. Starting from a structured mesh, three levels of refinement-derefinement are performed.

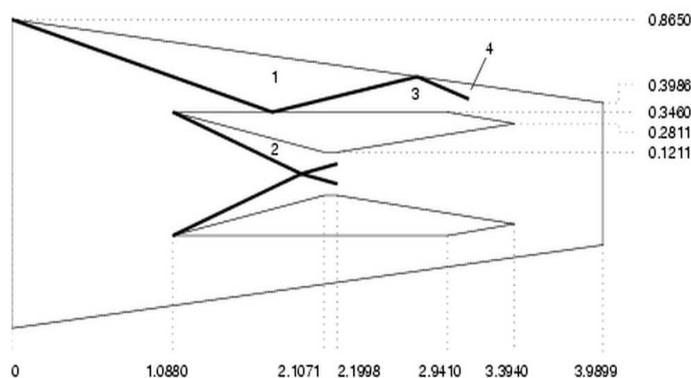


Figure 4: Geometry of two-dimensional scramjet intake (from [37]).

Fig. 5 shows the initial, intermediate and final meshes and the corresponding density contours. A zoomed view of the mesh and corresponding density contours on the final level mesh are shown in Fig. 6. The shock reflections are smeared on the unadapted mesh, while on the final level adapted mesh the incident shock and multiple shock reflections

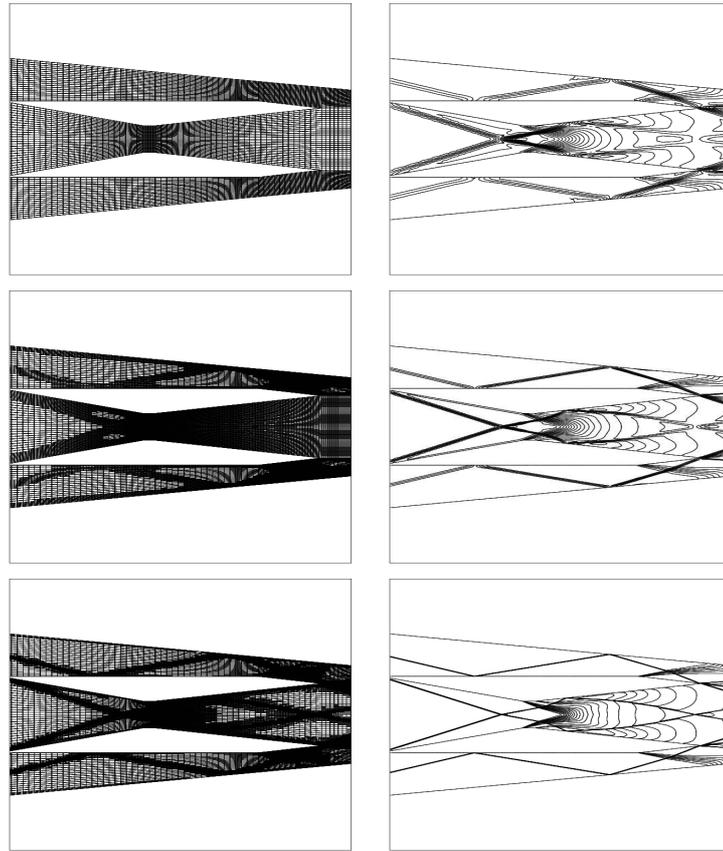


Figure 5: Grids (left) and Density contours (right) with grid adaptation for inviscid hypersonic flow past scramjet intake. Contour levels are 0.35:0.25:5.54.

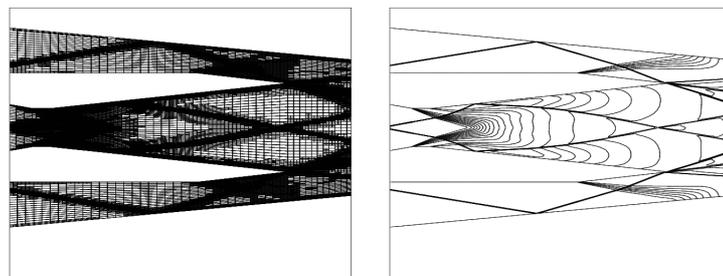


Figure 6: Zoomed view of the final level adapted mesh (left) and density contours (right). Contour levels are 0.35:0.25:5.54.

are captured accurately. Expansion fans and their interactions with shocks as well as the slipstreams are also well resolved. It is observed that the adaptation preserves mesh symmetry, which is in line with the associated symmetry of this problem.

6.3.2 Case 2: Supersonic laminar flow past bump in a channel

Supersonic laminar flow ($M_\infty=1.4$, $Re_\infty=8000$) past a 4% thick circular arc bump in a channel is investigated. This test case [36] is a challenging problem because of the presence of disparate flow features. A schematic view of the geometry with the relevant boundary conditions is shown in Fig. 7. The attached shock at the nose of the bump reflects from the top wall, forming a small mach stem. The reflected shock from the upper wall induces boundary layer separation and reflects as expansion waves. The boundary layer “lift-off” results in an effective compression corner from where compression waves emanate before coalescing into a shock and reflecting again from the upper wall. This reflected shock interacts with the expansion waves from the shock-boundary layer interaction region. The slip line emanating at the triple point near the upper wall, which interacts with the compression and expansion waves.

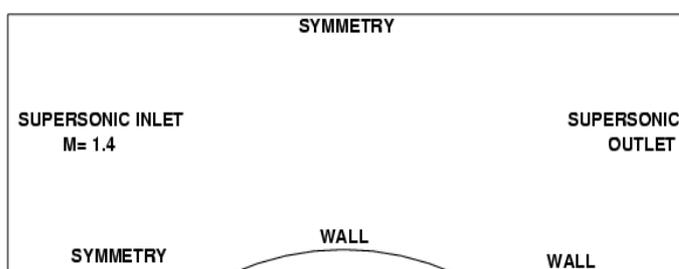


Figure 7: Schematic figure (not to scale) showing the geometry and boundary conditions for supersonic laminar flow past bump.

In order to study the ability of the HAS algorithm to handle mixed elements for viscous flows, a hybrid mesh is employed for this test case. Four levels of refinement-derefinement are performed starting from a coarse initial mesh. The initial, intermediate and final meshes and the corresponding mach contours are shown in Fig. 8. The shock emanating at the bump nose and its reflection as well as the mach stem are well captured by both the algorithms. The adaptive algorithms also do well to capture the boundary layer “lift-off” as also the compression waves generated from the “lift-off” region and its subsequent reflection from the upper wall. Both algorithms also detect the expansion waves and the recirculation zone at the rear of the bump. Fig. 9(a) shows a close-up view of the recirculation zone on the final level mesh. The slip discontinuity emanating from the triple point is comparatively a weaker feature, but the adaptive algorithm is successful even in resolving this flow feature. The skin friction distribution on the lower wall is shown in Fig. 9(b) indicate that the separation and reattachment points are in good agreement with the previous computations [36]. The ability of the adaptive algorithm to resolve the detailed flow physics in the presence of multiple flow phenomena of disparate strengths, is a testimonial to the fact that the proposed HAS algorithm is a powerful tool to obtain fast and accurate solutions for complex flow problems.

The AMR efficiencies on the final level mesh for all the test cases are shown in Table 1.

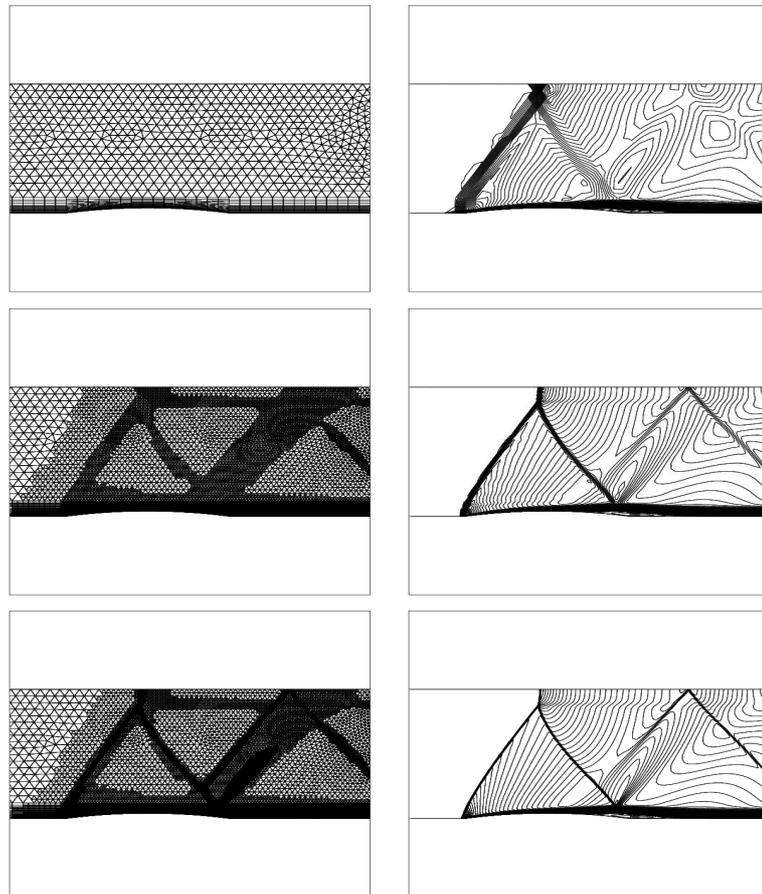


Figure 8: Grids (left) and Mach contours (right) with grid adaptation for supersonic laminar flow past bump in a channel. Contour levels are 0:0.2:1.4.

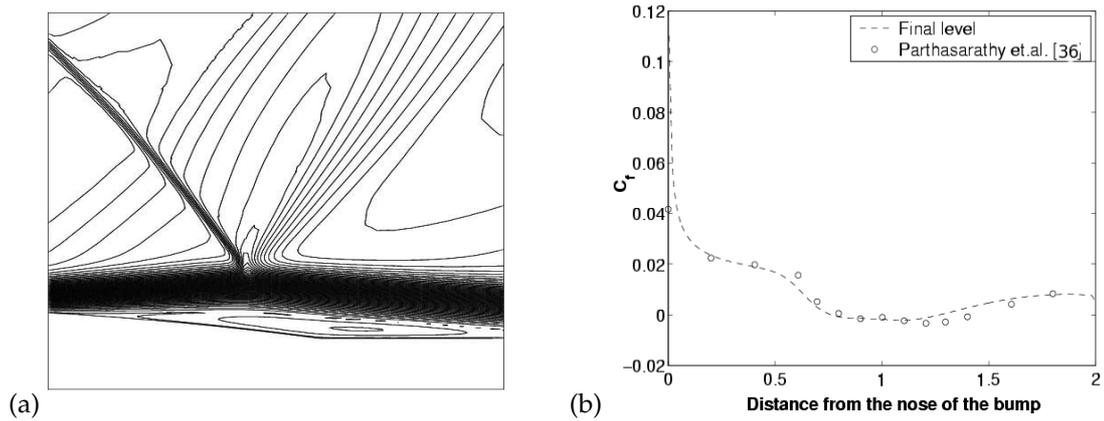


Figure 9: (a) Close-up of shock/boundary layer interaction. (b) Wall skin friction distribution.

The AMR efficiency is high for RAS and HAS adaptive algorithms for the test cases considered in this paper. In addition, these algorithms have been able to accurately resolve flow phenomena for internal and external fluid flow problems on arbitrary polygonal meshes. The high AMR efficiency and accurate solutions are clear indications that the proposed adaptive algorithms are effective and efficient for inviscid and viscous fluid flows.

7 Conclusions and future work

Adaptive algorithms based on a residual error estimator known as \mathfrak{R} -parameter have been proposed for inviscid and viscous compressible fluid flows. For smooth flows devoid of limiters, the RAS adaptive algorithm based purely on the residual estimator has been proposed. The hybrid HAS algorithm employing error indicators in conjunction with the \mathfrak{R} -parameter has been employed for compressible flows with discontinuities and limiting. Numerical simulations on complex fluid flow problems on arbitrary polygonal grids are indicative of effectiveness of the proposed adaptive strategies. The proposed adaptive algorithms can be extended to three-dimensional and unsteady fluid flows in the future.

References

- [1] Berger, M.J., Olinger, J., Adaptive mesh refinement for hyperbolic partial differential equations, *Journal of Computational Physics*, Vol. 53, pp. 484–512, 1984.
- [2] Kallinderis, Y.G., and Baron, R.J., Adaptation methods for a new Navier-Stokes algorithm, *AIAA Journal*, Vol. 27(1), pp. 37–43, 1989.
- [3] Sussman, M., A parallelized adaptive algorithm for multiphase flows in general geometries, *Computers and Structures*, Vol. 83(6), pp. 435–444, 2005.
- [4] Bryan, G., Abel, T., and Norman, L.M., Achieving extreme resolution in numerical cosmology using adaptive mesh refinement: Resolving primordial star formation, *Book of Abstracts*, p. 13, ACM/IEEE SC 2001 Conference, 2001.
- [5] De Zeeuw, D., Powell, K. G., Euler calculations of axisymmetric under-expanded jets by an adaptive-refinement method, *AIAA Paper 92-0321*, 1992.
- [6] Lohner, R., An adaptive finite element scheme for transient problems in CFD, *Computational Methods in Applied Mechanics and Engineering*, North Holland, Vol. 61, pp. 323–338, 1987.
- [7] Sun, M., Takayama, K., Conservative smoothing on adaptive quadrilateral grid, *Journal of Computational Physics*, Vol. 150, pp. 143–180, 1996.
- [8] Oh, W.S., Kim, J.S., and Kwon, O.J., Time-accurate Navier-Stokes simulation of vortex convection using an unstructured dynamic mesh procedure, *Computers and Fluids*, Vol. 32, pp. 727–749, 2003.
- [9] Laforest, M., Christon, M.A., and Voth, T.E., A Survey of error indicators and error estimators for hyperbolic problems, Report, Sandia National Laboratories, New Mexico, 2002.
- [10] Zhang, X.D., Pelletier, D., Trepanier, J.Y., and Camarero, R., Verification of error estimators for Euler equations, *AIAA Paper 2000-1001*, 2000.

- [11] Ainsworth, M., Oden, J.T., A unified approach to a posteriori error estimation using element residual methods, *Numerische Mathematik*, Vol. 65, pp. 23–50, 1993.
- [12] Jasak, H., Error Analysis and Estimation for the Finite Volume Method with Application to Fluid Flows, Ph.D. Dissertation, Imperial College, 1996.
- [13] Aftosmis, M.J., Berger, M.J., Multi-Level Error estimation and h-refinement for cartesian meshes with embedded boundaries, *AIAA Paper 2002-0863*, 2002.
- [14] Hay, A., Visonneau, M., Adaptive finite-volume solution of complex turbulent flows, *Computers and Fluids*, Vol. 36(8), pp. 1347–1363, 2007.
- [15] Karni, S., Kurganov, A., Local error analysis for approximate solutions of hyperbolic conservation laws, *Advances in Computational Mathematics*, Vol. 178, pp. 79–99, 2005.
- [16] Roy, C., Sinclair, M., On the generation of exact solutions for evaluating numerical schemes and estimating discretization error, *Journal of Computational Physics*, Vol. 228(5), pp. 1790–1802, 2009.
- [17] Ganesh, N., Shende, N.V., and Balakrishnan, N., \mathfrak{R} -parameter: A local truncation error based adaptive framework for finite volume compressible flow solvers, *Computers & Fluids*, Vol. 38(9), pp. 1799–1822, 2009.
- [18] Ganesh, N., Shende, N.V., and Balakrishnan, N., A new adaptation strategy for compressible flows, *Proceedings of the 7th AeSI CFD Symposium, India*, July 2004.
- [19] Ganesh, N., Shende, N.V., and Balakrishnan, N., A residual estimator based adaptation strategy for compressible flows, *Computational Fluid Dynamics*, Springer, 2006.
- [20] Ganesh, N., Balakrishnan, N., Residual adaptive computations of complex turbulent flows, *Computational Fluid Dynamics*, Springer, 2008.
- [21] Jawahar, P., Kamath, H., A High-Resolution procedure for Euler and Navier-Stokes computations on unstructured grids, *Journal Of Computational Physics*, Vol. 164, pp. 165–203, 2000.
- [22] Shende, N., Balakrishnan, N., New migratory memory algorithm for implicit finite volume solvers, *AIAA Journal*, Vol. 42(9), pp. 1863–1870, 2004.
- [23] Roe, P.L., Approximate Riemann solvers, parameter vectors and difference schemes, *Journal of Computational Physics*, Vol. 104, pp. 56–58, 1983.
- [24] van Leer, B., Flux Vector Splitting for Euler equations, Report No. 82-30, NASA Langley Research Centre, Hampton, Virginia, 1982.
- [25] Munikrishna, N., Balakrishnan, N., On viscous flux discretization procedures for cell center finite volume schemes, Fluid Mechanics Report No. 2004 FM 04, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India, 2004.
- [26] Holmes, D.G., Connel, S.D., Solution of the 2D Navier-Stokes equations on unstructured adaptive grids, *AIAA Paper 89-1932-CP*, 1989.
- [27] Balakrishnan, N., Deshpande, S.M., Reconstruction on unstructured meshes with upwind solvers, *Proceedings of the First Asian CFD conference*, Vol. 1, W.H. Hui, Y.K. Kwok and J.R. Chasnov (eds.), Department of Mathematics, The Hong Kong University of Science & Technology, Hong Kong, pp. 359-364, 1995.
- [28] Barth, T.J., A 3-D Least-squares upwind Euler solver for unstructured meshes, *Proceedings of 13th International Conference on Numerical Methods in Fluid Dynamics*, Lecture Notes in Physics, 414, M. Napolitano and F. Sabetta (eds.), Springer-Verlag, Berlin, p. 240, 1992.
- [29] Venkatakrishnan, V., Convergence to steady state solutions of the euler equations on unstructured grids with limiters, *Journal of Computational Physics*, Vol. 118, pp. 120–130, 1995.
- [30] Balakrishnan, N., Fernandez, G., Wall boundary conditions for inviscid compressible flows on unstructured meshes, *International Journal for Numerical Methods in Fluids*, Vol. 28, pp.

1481–1501, 1998.

- [31] Hirsch, C., Numerical Computation of Internal and External Flows, Vols. 1 and 2, John Wiley & Sons, 1990.
- [32] Shende, N., Garg, U., Karthikeyan, D., and Balakrishnan, N., An embedded grid adaptation strategy for unstructured data based finite volume computations, Computational Fluid Dynamics, Springer, 2002.
- [33] Anon, Inviscid flow field methods, Fluid Dynamics Panel Working Group, AGARD Advisory Report No. 211, 1985.
- [34] Mondal, P., Munikrishna, N., and Balakrishnan, N., Cartesian like grids using a novel grid stitching algorithm for viscous flow computations, Journal of Aircraft, Vol. 44(5), pp. 1598–1609, 2007.
- [35] Venkatakrisshnan, V., Computations using a direct solver, Computers and Fluids, Vol. 18(2), pp. 191–204, 1990.
- [36] Parthasarathy, V., Kallinderis, Y., Directional viscous multigrid using adaptive prismatic meshes, AIAA Journal, Vol. 33(1), pp. 69–78, 1995.
- [37] Kumar, A., Numerical analysis of the scramjet-inlet flow field by using two-dimensional Navier-Stokes equations, NASA Technical paper 1940, 1981.
- [38] Groth, C.P.T., Northrup, S.A., Parallel implicit adaptive mesh refinement scheme for body-fitted multi-block mesh, AIAA Paper 2005-5333, 2005.