An automatic unstructured grid generation method for viscous flow simulations

Seyed Saied Bahrainian *, Zahra Mehrdoost

Department of Mechanical Engineering, Shahid Chamran University, Ahwaz 61355, Iran

Received 21 August 2010; received in revised form 16 April 2012; accepted 18 May 2012

Available online 11 August 2012

Abstract

High aspect-ratio grids are required for accurate solution of boundary layer and wake flow. An approach for the efficient generation of isotropic and stretched viscous unstructured grids is introduced in this paper. The proposed grid generation algorithm starts with a very coarse initial grid. In far field regions, isotropic cells of excellent quality are produced using a combination of point insertion and cell subdivision techniques. Simultaneously, a directional grid refinement strategy is used to construct highly stretched triangular cells in viscous dominated regions. First, anisotropic unstructured grids are produced in the stream-wise direction. Then, cells close to the solid surface are refined to highly stretched layer of triangles suitable for boundary layer region. The accuracy of the current grid generation approach is assessed by laminar and turbulent compressible flow solutions around NACA0012, RAE2822, and NHLP multi-element airfoils. Numerical flow simulation results are compared with published data. Comparisons point to accuracy of the proposed unstructured viscous grid generation procedure.

Keywords: Unstructured grid; Viscous layers; Stretched cells

1. Introduction

Many computational fluid dynamic problems are concerned with strong gradients in a specific direction. A typical example occurs in the boundary layer region where magnitudes of solution gradients along the normal to solid surfaces are much higher than tangential direction.

In order to accurately resolve large gradients in the boundary layer, grid cells must be clustered in the direction of strong gradients. A limited number of isotropic cells can be arranged in the small boundary layer region. Usually, this is cured by clustering fine isotropic cells that can dramatically increase the computational costs. To overcome this limitation, anisotropic elements with high-aspect-ratio are highly desirable. This can ensure clustering and at the same time maintain a low computational cost since clustering only occurs in a specific direction.

Various methods have been introduced by researchers to generate highly stretched cells in the boundary layer. Most of these methods rely on fully unstructured grids with different point placement techniques [8,14,17,19]. Others have employed a directional grid refinement procedure to enrich their unstructured grid in viscous regions [12,18].

* Corresponding author. Tel.: +98 916 311 1316; fax: +98 611 333 6642.

E-mail addresses: bahrainian@scu.ac.ir, z_mehrooost@yahoo.com (S.S. Bahrainian).

© 2012 IMACS. Published by Elsevier B.V. All rights reserved.

0378-4754/$36.00 © 2012 IMACS. Published by Elsevier B.V. All rights reserved.

http://dx.doi.org/10.1016/j.matcom.2012.05.021
Other methods develop layers of viscous triangular grids around the solid geometry without refining the far field grids [10]. This results in grids very well arranged in highly viscous regions while far field grids are unattended. At a second stage, far field grids are refined and improved using grid quality measures.

Structured and hybrid grids have also been used to address viscous regions. This is accomplished by employing a structured or semi-structured grid close to the solid surface linked to an outer unstructured grid in the rest of domain [5,7,9,11,13,20]. It should be noted that this type of hybrid grid requires a modified flow solver to deal with different element types.

In this paper a procedure is presented that isotropic and anisotropic grids of excellent quality are produced simultaneously. Another feature of the current work is its ability to produce boundary layer and wake grids that enables accurate flow solutions in viscous dominated regions. Isotropic triangular cells in far field regions are produced using a combination of point insertion and cell subdivision techniques. Simultaneously, a directional grid refinement strategy is used to construct highly stretched triangular cells in the viscous regions. The proposed approach is robust and automated so that no user interference is required.

The present grid generation method is outlined in detail using a simple geometry shape for illustrative purposes. The procedure is then applied to a multi element airfoil to show the method’s capability with a more complex geometry. The possibility of extending the current approach to three dimensions is discussed in [1] where sample three dimensional unstructured grid outputs are presented. Finally, accuracy of the current grid generation approach is assessed by laminar and turbulent compressible flow solutions around NACA0012, RAE2822, and NHLP multi-element airfoils. Flow solution results are compared with published numerical and experimental data and are presented in the following sections.

2. Initial grid

The proposed algorithm starts with a very coarse initial grid. The computational domain is initially divided into two regions. First region close to the solid surface is assumed to represent the boundary layer region and is considered for construction of highly stretched cells. The second region represents the rest of computational domain and is considered for generation of isotropic cells. A sample initial grid with a solid body inside a rectangular domain is shown in Fig. 1, where the solid geometry is initially represented by only four points.

This initial grid contains 12 vertices, 28 edges, 8 triangular cells in the anisotropic region, and 8 triangular cells in the isotropic region. Although isotropic and anisotropic cells of the initial grid are separately presented here for descriptive purposes, they are contained within a single unstructured data structure.

3. Data structure

Data structures are used to store unstructured grid connectivity data. Connectivity matrices associated with unstructured grids are usually very large and searches through the matrix can be very time consuming. To minimize the required searches through a single connectivity matrix, a different approach has been taken in the present work.

Two separate data structures are used simultaneously to optimize search operations. An edge-based and cell-based data connectivity matrices are used simultaneously to prevent long searches for grid connectivity data. The algorithm for identifying edges and cells is as follows:

Fig. 1. Initial coarse grid.
Table 1
Various designated edge types.

<table>
<thead>
<tr>
<th>Type of edge</th>
<th>Edge type designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outer boundary</td>
<td>1</td>
</tr>
<tr>
<td>Interior</td>
<td>2</td>
</tr>
<tr>
<td>Surface</td>
<td>3</td>
</tr>
<tr>
<td>Boundary layer</td>
<td>4</td>
</tr>
<tr>
<td>Common boundary</td>
<td>5</td>
</tr>
</tbody>
</table>

Edge-based matrix (EBM) (NE, NV1, NV2, NC1, NC2, edge-type).
Cell-based matrix (CBM) (NC, NV1, NV2, NV3, NE1, NE2, NE3, cell-tag).

The edge-based matrix contains each edge number followed by its forming vertices NV1 and NV2. The two cell numbers NC1 and NC2 that share the edge along with the edge-type are forming the remaining columns. The cell-based matrix contains each triangular cell number followed by its forming vertices NV1, NV2, and NV3. The three edge numbers NE1, NE2, and NE3 that form the triangular cell are then listed in the matrix. A column at the end of the cell-based matrix is reserved for tagging the cells for a specific refinement treatment.

Concurrent use of these matrices has resulted in elimination of long searches among the grid element data. For example, when an edge identified by the edge-based matrix is selected for refinement, the vertices and the edges forming the surrounding cells are readily available by the cell-based matrix.

An integer value is designated and reserved in the edge-based connectivity matrix to represent the type of each edge [1,2]. For example, edge type 3 indicates that the edge is on the solid boundary and special surface treatments must be considered. Another edge type of importance (designated by 5 here) is assigned to edges that are located on the common boundary between isotropic and anisotropic regions. The edge types and their designated integer value are summarized in Table 1.

The edge type designation proposed here for identification of various edge types and their required treatment, can serve as a means to introduce new geometrical bodies. For example, for a complex geometry with various body parts such as aerodynamic configurations, each geometrical body part can be designated with a similar integer designation [1]. Appropriate geometrical treatments can then be performed for each edge based on its type value.

4. Isotropic grid refinement

In each refinement cycle, cells are selected based on their area size. The selected cells are then stored in descending order. Largest edge of the selected cell, from the top of stack, is then chosen for refinement. A new grid point is inserted at the middle of this edge. Construction of new triangular cells using this newly created grid point depends on the type of the original selected edge. If the new grid point is placed on an outer boundary edge (type 1), it is connected to the opposite vertex to form two new triangular cells. If the new grid point is placed on an interior edge (type 2), it is connected to the opposite vertices of triangles that share the edge to form four new triangular cells.

This grid refinement procedure, without considering the small viscous region near the solid surface is illustrated in Fig. 2, where grid outputs at the end of 1st, 2nd, 4th, and 6th refinement cycles are shown.

The above grids contain some poorly shaped elements which are of low value as far as solution stability is concerned. It should be noted, however, that the above grids have been generated without the use of grid quality measures. In the present algorithm, edge swapping and grid smoothing techniques are applied to enhance the overall grid quality [3]. Fig. 3 displays the grid after 6 cycles of refinement with quality improvement.

By comparing the grid of Fig. 3 to that of Fig. 2(d), one can see that the skewed triangles have vanished. Although, the grid of Fig. 3 is of acceptable quality, it does not represent the final output and is only shown for illustrative purposes. Thus far, only interior and outer boundary edges have been illustratively refined. It should be noted that the actual refinement algorithm considers all of the edge types simultaneously.

If the new grid point is placed on a common boundary edge (type 5), a different strategy is used for cell subdivision. The strategy involves simultaneous subdivision of neighboring cells that lay on two different regions within the flow field. One region requires isotropic subdivision while the other requires stretched cells. This is achieved by an algorithm for isotropic and anisotropic grid generation and is detailed in the following sections.
5. Common boundary edge refinement

A special unstructured grid generation algorithm, developed through the current research, is used when a common boundary edge (type 5) is selected for refinement. Since the selected edge is shared between an isotropic and an anisotropic stretched cell, concurrent use of two different algorithms is required. One algorithm is used for treatment of the isotropic cell which was described in the previous section, and another to address the viscous cell which is described here.

A common boundary edge is selected and clearly marked on the grid of Fig. 4. The marking (shown by dashed lines) contains three cells. The isotropic cell that is shown on top shares a common edge (type 5) with one of the two elongated viscous cells. To better describe the procedure involved in refining viscous cells, the above marked cells are illustrated schematically by Fig. 5. The cells in the viscous region are shown by triangles \(acb\) and \(bcd\). The common boundary edge is shown by edge \(ab\) while edge \(cd\) is located on the solid surface geometry.

If a common boundary edge (type 5) is selected for refinement, the neighboring cells that lie in the viscous region are also refined simultaneously. For example, referring to Fig. 5, for refinement of edge \(ab\), in addition to cells \(abe\) and \(acb\), the cell \(bcd\) that has an edge on the solid boundary is also refined. First the edge \(bc\) shared by the two viscous cells is removed. Second, new grid points are inserted in the middle of edges \(ab\) and \(cd\). Third, edges \(ef\) and \(fg\) are constructed by using newly inserted grid points.

![Fig. 2. Isotropic grid after (a) 1st, (b) 2nd, (c) 4th, and (d) 6th cycles of refinement.](image)

![Fig. 3. Isotropic grid after 6th cycle of refinement with quality improvement.](image)
At this point, the three original cells are converted to two isotropic triangular cells and two quadrilateral elements (see Fig. 5(c)). Finally, the refinement procedure continues by directionally converting the two quadrilateral elements into four unstructured viscous grids.

The above detailed procedure is further illustrated by a sample grid shown in Fig. 6(a). A close view of this grid near the solid geometry is also shown in Fig. 6(b).

By far the above grid does not represent the final grid output and is only shown to demonstrate the simultaneous refinement of isotropic and viscous grids. There are other considerations that have to be addressed before a suitable final computational grid is obtained. One such consideration is the way the solid surface geometry takes its final shape.
6. Construction of solid surface geometry

Contrary to most grid generation strategies that start with a very well defined of the solid geometry, here, surface geometry is initially represented by only a few points. These few points that were shown earlier in the initial grid of Fig. 1, must be selected from the actual geometry. Construction of final geometry shape is accomplished through the grid refinement process. Every time a new point is inserted on a surface boundary edge, the coordinates of that point are matched to a prescribed geometry that represents the final surface shape. Calculation of new coordinate points on the surface geometry is either made through an equation that represents the surface geometry or obtained by a cubic-spline routine among surface geometry data.

Although the above procedure may require trivial calculations, the actual movement of surface geometry points is of great importance to the grid integrity. This approach has been extensively used by the authors for isotropic surface grids. Here, an attempt is made to describe this procedure for a more complex situation where both isotropic and viscous vertex points must be moved together.

When edges on the solid surface geometry (type 4) are subdivided, the new surface point coordinate is moved with a specific amount to match the prescribed geometry. This causes the viscous layer thickness to change. To prevent any changes within the viscous layer thickness, the grid point on the common boundary above the surface point is also moved with the exact amount. This procedure is shown by the schematic diagram in Fig. 7.

Coordinate movement and construction of final solid geometry described above is illustrated by a sample grid shown in Fig. 8(a). In this case, coordinates of coarse surface grid are moved to represent an oval body. A close view of this grid near the solid geometry is shown in Fig. 8(b).

By examining close views of viscous cells, it is obvious that further work is required in the boundary layer region.

7. Highly stretched viscous grids

Highly stretched cells are required in the thin boundary layer region. Before starting the process, however, vertices on the common boundary between isotropic and anisotropic regions must be aligned in the normal direction to the solid wall. Displacement process for normal alignment of vertex points that are on the common boundary is shown schematically in Fig. 9.
Normalization of the grid points that lie on the common boundary is obtained based on the maximum spacing allowed for the boundary layer thickness $\lambda$ and coordinates of grid points lying on the solid surface. Grid points laying on the solid surface are specified by $P_1$ and $P_2$, while grid point that lay on the common boundary is specified by $P_3$ and are shown in Fig. 10. The objective is to move point $P_3$ such that it is aligned in the normal direction to the line $P_1P_2$.

The new coordinates of $P_3$ are calculated as follows.

\[
\begin{align*}
  x_{P_3} &= x_{P_2} \pm y_{P_2} \frac{\lambda}{mq} \\
  y_{P_3} &= y_{P_2} + m (x_{P_3} - x_{P_2})
\end{align*}
\]  

(1)

where $x$ and $y$ are coordinate points and values of $m$ and $q$ are calculated by the following equations. It should be noted that a positive sign is used when $|y_{P_2}| > |y_{P_3}|$ in Eq. (1).

\[
\begin{align*}
  m &= \frac{x_{P_2} - x_{P_1}}{y_{P_1} - y_{P_2}} \\
  q &= y_{P_2} \sqrt{1 + m^{-2}}
\end{align*}
\]  

(2)

(3)
After applying the above process to the common boundary grids of Fig. 8, directional cells are now normal to the solid surface and are shown in Fig. 11.

The process of generating high-aspect-ratio cells can now start by dividing the normalized directional cells into highly stretched layer of triangles. Distribution of new grid points for construction of high-aspect-ratio cells is determined by a stretching ratio. In this work, the following function, revised from [19], is used to determine the thickness of each stretched layer.

$$\delta_k = \eta^{k-1}\delta_o$$

where $\delta_k$ is the $k$th layer thickness, $\delta_o$ is the first layer thickness (close to the solid wall), and $\eta$ is a user defined stretching ratio.

Common boundary edges are shared between isotropic cells and the newly generated normalized directional cells. The space between these common boundary edges and the solid surface is now occupied with right triangular cells. Each pair of these viscous directional cells forms a quadrilateral. A pair of viscous directional triangular cell is marked on the grid of Fig. 12(a).

The next step is to generate highly stretched triangular cells in the constant viscous layer region that is formed by directional triangular cells. In the present approach, layers of stretched cells are formed starting from the common boundary edge progressing toward the first layer near the solid surface. New points are inserted on edges normal to surface to form new layers. This procedure is detailed by the schematic diagram of Fig. 12(b).

The $i$th stretched layer is formed by inserting a new grid point $P_i$. The position of this new grid point is determined based on the coordinates of $P_1$ and $P_{i+1}$. $P_1$ is the original point on the surface while $P_{i+1}$ is a grid point that is generated on the previous layer. Coordinates of the point $P_i$ are calculated by the following equation.

$$X_{P_i} = \frac{X_{P_1}\beta_i + X_{P_{i+1}}}{1 + \beta_i}$$

where $\beta_i$ is ratio of the $i$th layer thickness to the thickness of remaining layers toward the surface and is defined as follows.

$$\beta_i = \frac{\delta_o\eta^{i-1}}{\delta_o + \delta_o\eta + \cdots + \delta_o\eta^{i-2}}$$

or,

$$\beta_i = \frac{\eta^{i-1}}{1 + \sum_{m=1}^{i-2}\eta^m}$$

The first layer of stretched cells is formed inside the viscous region near the common boundary. This layer is located farthest away from the solid surface. The procedure for construction of the first layer of stretched triangular elements is shown schematically in Fig. 13.
Fig. 13. Construction of the first layer of stretched cells inside the anisotropic region.

Originally, sets of directional triangular cells are arranged by coordinates $P_1, P'_1$. At this stage, edge $P_1 - P'_1$ lies on the solid surface, while edge $P_{n+1} - P'_{n+1}$ is located on the common boundary. To construct the layer $n - 1$ (near the common boundary), the diagonal edge $P_1 - P'_{n+1}$ is removed. New coordinates $P_n$ and $P'_n$ are inserted to form two new quadrilaterals. Then, diagonal edges $P_n - P'_{n+1}$ and $P_1 - P'_n$ are defined to form new directional triangular cells.

Now, one layer of highly stretched triangular cells is formed near the common boundary. To form the second layer of directional stretched triangular cells, the quadrilateral formed by coordinates $P_1, P_n, P'_n, and P'_1$ is considered.

The above process of constructing highly stretched triangular cells is further described by the grids of Fig. 14. Construction of the first layer of directional elements near the common boundary is shown in Fig. 14(a). Grid refinement after the construction of 3rd viscous directional layer is shown in Fig. 14(b). Grid refinements after the construction of 5th and 7th viscous directional layers are shown in Fig. 14(c) and (d) respectively. Stretched viscous layer construction continues until the desired number of layers specified by the user is obtained.

It should be noted that the above grids are generated for illustrative purposes and suitable grids for actual flow simulations can contain further clustering in the boundary layer region. This can be accomplished by further reducing
the layer size (thickness). The point insertion procedure described above can also start from the solid surface as opposed to starting from the common boundary.

8. Viscous grids at sharp corners

The sharp concave/convex corners can lead to poor quality elements that are not suitable for viscous flow calculations. A common approach to overcome this problem is smoothing the normal direction near the sharp corners. Fig. 15 shows the effect of a Laplacian smoothing operator in the vicinity of a concave region. It should be noted that the normal Laplacian smoothing operation is performed prior to stretch layer development, while final grid outputs of Fig. 15 include the stretched layers.

The Laplacian smoothing operator cannot always guarantee the required grid resolution near the corners. An alternative procedure is proposed to better discretize sharp regions.

To apply the proposed procedure, slight modification of the initial grid is required. The grid of Fig. 16(a) displays an ordinary initial grid arrangement at a sharp corner. To apply the proposed procedure, a set of directional triangular cells are inserted at the sharp concave corner of the initial grid and can be seen in Fig. 16(b). As discussed in the previous sections, the next step is to generate pairs of directional triangular cells in the anisotropic region while isotropic cells are generated elsewhere. This can be seen in Fig. 16(c) where the two added directional cells enable a continuous distribution of directional cells around a sharp corner. When layers of stretched triangles are created, the two triangular cells in the concave corner are refined with the same manner. The final corner grid is shown in Fig. 16(d).
Fig. 17. Construction of stretched triangular layers at a concave corner.

Although the refinement procedure from directional triangular cells to highly stretched viscous layers has been discussed earlier, an attempt has been made to display the construction of stretched triangular cells at a concave corner. These are shown in Fig. 17, where subsequent stretched triangular layer generation is displayed. Similar shape grids in concave corners are produced by other researchers [12].

The same procedure can be used to handle convex corners. Examples of computational grid adjacent to concave and convex corners are shown in Fig. 18.

Fig. 18. Grid close to (a) concave corner and (b) convex corner.
9. Wake region considerations

The procedure for generating highly stretched triangular cells in the wake region is similar to the procedure that is used to generate viscous grids near solid surfaces. To consider generating highly stretched viscous cells in the wake region, the edges in that region have to be designated similar to the solid boundary. This requires a more complicated initial grid that allows the domain to be divided into different edge-type regions. Fig. 19 shows a close view of an initial grid that considers both the solid surface and the wake region.

The solid geometry in the above initial grid is numbered with vertices 9, 10, 11, and 12. The eight skewed triangles surrounding the solid geometry are designated to represent the boundary layer region. Similarly, the (shaded) skewed triangles downstream the solid boundary with vertex numbers 11, 19, and 15 for example, are arranged and designated to represent the wake region.

Since the edge-type designation, detailed in Table 1, is assigned as boundary layer and common boundary for both the boundary layer and wake regions, layers of stretched triangular cells are generated throughout the refinement cycles. Furthermore, since both regions are assigned with similar edge-type designations, the refinement procedure for the wake region is identical to that of the boundary layer region which was discussed in previous sections. It should be noted that the boundary layer grids surround a solid surface and grid points may be moved to match a prescribed geometry. In contrast, the grids in the wake region may only be adjusted according to a wake line. Fig. 20 shows a close view of a sample grid output near the wake region.

10. Applications to complex geometries

The present unstructured grid generation algorithm can be applied to complex multi body geometries with concave and convex sections and three dimensional configurations. This section provides sample grid outputs to illustrate the algorithm’s grid generation capabilities as applied to more complex geometries.

Since the current grid generation approach is fully automated, the procedure for generating a suitable grid around multi element geometries is similar to those discussed earlier with single body geometry. The only difference is in the
Table 2
Various designated edge types for generating a grid around a three element airfoil.

<table>
<thead>
<tr>
<th>Type of edge</th>
<th>Edge type designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outer boundary</td>
<td>1</td>
</tr>
<tr>
<td>Interior</td>
<td>2</td>
</tr>
<tr>
<td>Surface 1 (main airfoil)</td>
<td>3</td>
</tr>
<tr>
<td>Boundary layer</td>
<td>4</td>
</tr>
<tr>
<td>Common boundary</td>
<td>5</td>
</tr>
<tr>
<td>Surface 2 (slat)</td>
<td>6</td>
</tr>
<tr>
<td>Surface 3 (flap)</td>
<td>7</td>
</tr>
</tbody>
</table>

definition of surface edges in the initial grid. To generate a grid around a three element airfoil, edges on the surface of each element have to be designated by a specific integer value for geometry treatment. Table 2 shows the designated integer value associated with edge types used in this case.

It should be noted that the only difference between the values given in this table as opposed to those given earlier in Table 1, lie in definition of additional new surfaces. Edge type 3 is now assigned to the airfoil’s main body. New surface edges designated by integer values 6 and 7 now refer to new surface bodies. These surface bodies represent the...
Fig. 22. Close views of final grid around a three element airfoil near leading and trailing edges.

slat and flap of the multi element airfoil. The grid refinement procedure for this multi element geometry is illustrated in Fig. 21, where 3rd, 6th, and 9th cycles of refinement as well as the final grid output are shown.

To further illustrate the quality of unstructured grids generated around this three element airfoil, close views near the leading and trailing edges of the airfoil are shown in Fig. 22.

An analogous algorithm to the two-dimensional approach presented here has been developed to generate grids in three dimensional spaces. Tetrahedral cells, as opposed to triangular cells, are generated for this purpose. Details of three dimensional grid generation approach are given in [1,2].

11. Viscous flow computations

Compressible viscous flow computations are carried out to assess the performance of the proposed viscous grid generation approach. Three well known test cases were selected and are presented here to show their solution accuracy with the proposed approach. The first case includes laminar flow around a NACA0012 airfoil. The second case includes transonic turbulent flow around a RAE2822 airfoil. The third and last test case involves turbulent flow around a multi-element NHLP airfoil.

A finite volume cell-centered scheme is used to solve compressible viscous flow equations on the proposed unstructured triangular viscous grids. Repetitive mathematical finite volume formulations are omitted. Details of mathematical finite volume formulations along with solution techniques including initial conditions, boundary conditions, convergence acceleration, and application of turbulence models can be found in [2,15].

11.1. Laminar flow around a NACA0012 airfoil

As an initial test case, laminar flow over a NACA0012 airfoil is computed. The flow conditions including free stream Mach number, Reynolds number, angle of attack, and the grid details are summarized in Table 3.

The computational grid employed for this case is shown in Fig. 23. This grid contains 14,223 isotropic and viscous anisotropic directional stretched cells. The normal spacing of the first viscous layer from the airfoil surface is 0.001% of the airfoil’s chord. This thickness increases for the cells that are located in viscous layers further away from the airfoil surface. This is to obtain a proper cell size distribution while ensuring a smooth transition of viscous cells to the

Table 3
Geometry, flow conditions and viscous grid details for a laminar test case.

<table>
<thead>
<tr>
<th>Airfoil</th>
<th>Ma</th>
<th>Re</th>
<th>$\alpha$</th>
<th>Number of nodes</th>
<th>Number of cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>NACA0012</td>
<td>0.5</td>
<td>5000</td>
<td>0°</td>
<td>7244</td>
<td>14,223</td>
</tr>
</tbody>
</table>
isotropic grid region. To examine the quality of viscous unstructured grids, a close view of the stretched cells on the upper surface of the airfoil is also shown in Fig. 23.

Mach number contours in the flow field for this laminar test case are displayed in Fig. 24. Distributions of the surface pressure coefficient and of the skin friction coefficients are given in Fig. 25. The computational results are compared with numerical solutions given by Mavriplis and Jameson [15].

### 11.2. Turbulent flow around RAE2822 airfoil

The second test case was carried out to investigate the capabilities of the proposed grid generation algorithm for accurate and efficient solution of turbulent flow. The flow conditions including free stream Mach number, Reynolds number, angle of attack, and the grid details are summarized in Table 4. The computational grid employed for this

<table>
<thead>
<tr>
<th>Airfoil</th>
<th>Ma</th>
<th>$Re$</th>
<th>$\alpha$</th>
<th>Number of nodes</th>
<th>Number of cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAE2822</td>
<td>0.73</td>
<td>$6.5 \times 10^6$</td>
<td>3.19$^\circ$</td>
<td>12,867</td>
<td>25,344</td>
</tr>
</tbody>
</table>
turbulent test case is shown in Fig. 26. This grid contains 25,344 isotropic and viscous anisotropic directional stretched cells. The normal spacing of the first viscous layer from the airfoil surface is 0.00004% of the airfoil’s chord. This small viscous layer thickness was selected to enable accurate prediction of shock location on the airfoil surface.

To examine the viscous grids, close views of the RAE2822 grids are shown near the airfoil’s leading and trailing edges in Fig. 27.

Viscous compressible transonic flow solutions are obtained by using the highly stretched elements around RAE2822 airfoil. In this case, a strong shock wave is present on the upper surface of the airfoil. Plots of Mach number contours reveal this shock wave and are shown in Fig. 28. Flow simulations have made the use of Baldwin–Lomax [4] turbulence model to compute the eddy viscosity. Transition point for this test case was fixed at 0.03 chord length.

To show accuracy of the present grid generation and flow solution approach, comparisons are made with published experimental data. Fig. 29 shows comparison of computed surface pressure and skin friction distributions with the experimental data of Cook et al. [6]. Excellent agreement of pressure coefficient plot with that of experimental data, points to accuracy and robustness of the present approach. This is especially true by observing the vertical line at about 60% of chord, which indicates accurate prediction of shock location.
Fig. 27. Close view of RAE2822 airfoil grid near the leading and trailing edges.

Fig. 28. Mach number contours around RAE2822 airfoil at $Ma = 0.73$, $Re = 6.5 \times 10^6$ and $\alpha = 3.19^\circ$.

Fig. 29. Pressure coefficients (a) and skin friction coefficients (b) on the RAE2822 airfoil at $Ma = 0.73$, $Re = 6.5 \times 10^6$ and $\alpha = 3.19^\circ$. 
Fig. 30. Mach number contours and stream lines around NHLP 3-element airfoil at Ma = 0.197, Re = 3.52 × 10^6 and α = 4.01°, (a) and (b), close views near the slat are shown in (c) and (d) while close views near the flap are shown in (e) and (f).
Table 5
Flow conditions and grid details for the turbulent NHLP test case.

<table>
<thead>
<tr>
<th>Airfoil</th>
<th>Ma</th>
<th>Re</th>
<th>( \alpha )</th>
<th>Number of nodes</th>
<th>Number of cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>NHLP 3-element</td>
<td>0.197</td>
<td>( 3.52 \times 10^6 )</td>
<td>4.01(^{\circ})</td>
<td>68,722</td>
<td>34,881</td>
</tr>
</tbody>
</table>

Fig. 31. Pressure coefficient distribution on the NHLP 3-element airfoil at Ma = 0.197, Re = \( 3.52 \times 10^6 \) and \( \alpha = 4.01^{\circ} \).

11.3. Turbulent flow around NHLP 3-element airfoil

The third and final test case presented here was carried out to investigate the capabilities of the proposed grid generation algorithm as applied to a more complex geometry. The flow conditions including free stream Mach number, Reynolds number, angle of attack, and the grid details for this NHLP 3-element airfoil are summarized in Table 5.

The computational grid for this turbulent test case is shown in Figs. 22 and 23. The grid of NHLP 3-element airfoil with a slat and single-slotted flap configurations employed 68,722 cells and 34,881 vertices. Viscous compressible flow solutions are obtained using this grid output. Mach number contours on a grid background and flow streamlines are illustrated in Fig. 30. Details of viscous flow behavior in the boundary layer and wake regions can be seen in the illustrated figures. The viscous flow behavior is especially evident in the close up views given between the slat and airfoil’s leading edge as well as the flap and airfoil’s trailing edge.

To show solution accuracy for this 3-element airfoil, distributions of surface pressure coefficients are compared with experimental measurement values of Moir [16] and are shown in Fig. 31. Excellent agreement of pressure coefficient plot with that of experimental data, points to accuracy and robustness of the present grid generation and flow solution method.

12. Conclusions

High aspect ratio grids are required for accurate solution of viscous boundary layer and wake flows. This paper presents a novel approach for robust and accurate generation of unstructured grids suitable for viscous flow computations.

The grid generation algorithm starts with a very coarse initial grid. In far field regions, isotropic cells of excellent quality are produced using a combination of point insertion and cell subdivision techniques. Simultaneously, a directional grid refinement strategy is used to construct highly stretched triangular cells in the highly viscous boundary layer and wake regions.
Unstructured grid data are stored in two different data structures to form a connectivity matrix. An edge-base data structure stores the required edge connectivity data, while a cell-base structure stores the required cell connectivity data. This novel feature eliminates long searches through stacks of data that are inherent of most unstructured grid generation algorithms.

Grid refinement in isotropic regions is achieved by splitting the longest edge of each triangular cell. This, inherently, tends to avoid generation of unwanted skewed triangles in that region. Qualities of badly shaped triangles are then improved through the use of an edge swapping algorithm. In addition, use of a weighted grid point smoothing procedure further ensures generation of isotropic cells in the far field.

A directional grid refinement strategy is developed and used to construct highly stretched triangular cells in the boundary layer and wake regions. First, anisotropic unstructured grids are produced in the stream-wise direction. Then, cells close to the solid surface are refined to highly stretched layer of triangles suitable for boundary layer regions. Smooth transition between the boundary layer grid and the outer isotropic grid is obtained with a user specified cell size and stretch parameter.

An analogous algorithm to the two-dimensional approach has been developed to generate three dimensional grids. Tetrahedral cells, as opposed to triangular cells, are generated for this purpose. Grid refinement is based on a combination of point insertion and cell-subdivision methods. Tetrahedral cells are refined by selecting the largest edge of each cell and inserting a new grid point at the center of that edge [1]. To show the possibility of using the proposed algorithm in three dimensions, sample tetrahedral grids are given for a circular cylinder attached to a plane-of-symmetry within a cubical domain.

Laminar and turbulent compressible viscous flow solutions around NACA0012 and RAE2822 airfoils are obtained to show grid accuracy and integrity. Computational unstructured grids are generated around an airfoil with NACA0012 cross section as the first test case. The normal spacing of the first viscous layer from the airfoil surface was set to 0.001% of the airfoil’s chord. This grid contains 14,223 isotropic and viscous anisotropic directional stretched cells and was employed to obtain laminar viscous flow solutions. Flow solution results indicate an excellent agreement for this laminar test case.

The second test case involved computational unstructured grid generation around an airfoil with RAE2822 cross section. The normal spacing of the first viscous layer from the airfoil surface was set to 0.00004% of the airfoil’s chord. This small viscous layer thickness was selected to enable accurate prediction of shock location on the airfoil’s surface. Transonic flow solutions are obtained at a free stream Mach number of 0.73, Reynolds number of $6.5 \times 10^6$, and attack angle of 3.19°. The results of transonic flow simulations are compared with published experimental data. Comparisons indicate that the proposed unstructured boundary layer and wake grid generation approach have enabled the viscous flow solver to accurately capture shock locations on a RAE2822 airfoil.

The third and last test case presented here involved turbulent flow solution around a 3-element NHLP airfoil at $\text{Ma} = 0.197$, $\text{Re} = 3.52 \times 10^6$ and $\alpha = 4.01^\circ$. The computational grid for this turbulent test case employed 68,722 cells / 34,881 vertices. Mach number contours and flow streamlines clearly show viscous flow behavior in the boundary layer and wake regions. Comparison of surface pressure coefficient results with experimental data points to the accuracy of the current grid generation and flow solution approach.

References