23

Hybrid Grids

23.1 Introduction

Recent years have witnessed much conjecture over the relative merits of the various methodologies that have emerged as candidates for providing a robust, effective, high-quality mesh generation capability for gridding complex three-dimensional domains. These methods are generally classified into one of two categories, namely structured or unstructured approaches, with strong advocates of each still existing amongst both the method development and user communities. Promoters of structured schemes highlight the efficiency and accuracy that is attained through the employment of regularly arranged hexahedral volumes. Supporters of unstructured schemes emphasize the geometric flexibility and suitability for adaptation inherent to the use of irregularly connected tetrahedral volumes.

This Handbook will serve to further the debate on the absolute superiority of one of these approaches over the other without, one suspects, enabling a definitive conclusion to be reached.

However, a review of this handbook in conjunction with the proceedings of the now firmly established series of conferences devoted to numerical grid generation indicates that there is an underlying trend within the field of grid generation. This trend is toward an increasing cross-fertilization of ideas and techniques between the two camps. Practitioners of the unstructured approach are having to use directional information to achieve elements of suitable quality near boundaries, while structured grid generators are devising increasingly irregular schemes to attain appropriate geometric flexibility.

The limit of this trend is to replace the sole use of one mesh type by the use of combined meshes composed of both structured and unstructured grids — hybrid grids. This combination of grid types not only allows the benefits of structured and unstructured grids to be attained simultaneously, but also allows high grid quality to be achieved throughout the domain due to the appropriate use of each element type.

In this chapter, the prime interest is the generation of grids containing more than one element type. This will be termed hybrid grid generation. However, reference will also be made to the generation of single element type meshes where it is felt that the work particularly demonstrates the movement of ideas between the two main fields of mesh generation. This will be termed hybrid grid technology.
This chapter has three main sections and a summary. Section 23.2 is devoted to “Underlying Principles” and contains a general description of both work in the field of hybrid grid generation and the use of hybrid grid technology. It begins by tracing the roots of hybrid grid generation back to two quite distinct sources. The move to hybrid grid generation/technology from the purely unstructured approach is then reviewed, followed by observations on the progression of the structured community to hybrid grids. The section ends with a discussion on the potential savings in execution times and memory requirements that can be made through using hybrid grids instead of solely unstructured grids.

In Section 23.3, entitled “Best Practices,” the discussion becomes more focused around the author’s own experience in generating hybrid grids. This is because, in spite of the very real potential benefits that are to be gained through the use of hybrid grids, there is at present a dearth of evidence that other capabilities exist that are able to form general three-dimensional hybrid meshes. The section begins with a brief overview of the evolution of a mesh generation system that can be used to form either solely structured (hexahedral cells), semistructured (prismatic cells), unstructured (tetrahedral cells) or a hybrid combination of any of these grid types. Attention is then focused briefly on the key elements of the capabilities that are used to form the different types of elements, with the details left either to references or study of other chapters in the book. The very important area of interfacing the different grid types is then covered, and this is followed by a discussion on data structures for describing a hybrid grid. Finally, three examples of hybrid grids for aero- and hydrodynamic applications are presented along with a description of the main considerations that have been borne in mind while forming these grids.

Section 23.4 covers some of the open research issues that will need to be addressed within the field of hybrid grid generation for the approach to realize its potential. The discussion also focuses on some of the practical implications that lie behind the adoption of a hybrid grid strategy, which possibly indicate why there are currently so few general, three-dimensional hybrid capabilities.

### 23.2 Underlying Principles

#### 23.2.1 Historical Review

As with many other ideas, the origins of the concept of hybrid grid generation can be traced back to two unrelated workers, namely Nakahashi from Japan and Weatherill from the U.K.

Nakahashi advocated the use of hybrid grids in conjunction with a zonal finite difference (FD) and finite element (FE) flow solution methodology [Nakahashi and Obayashi, 1987a, b]. An implicit finite difference method was applied on structured grids to viscous flow modeling near geometric surfaces. The remaining regions were modeled by an explicit, node-based finite element solution of the Euler equations formulated on unstructured grids. Communication between the FD and FE zones was achieved by allowing the grids to overlap by one cell, with the grids sharing common nodes in these regions. Hence, information required at the zonal boundary of one region could be taken from the interior of the adjacent grid.

*The observation that the approach combined both the computational efficiency of the FD method and the geometric flexibility of the FE method was central to Nakahashi’s promotion of the use of hybrid grids.*

In his early work, Nakahashi does not present sufficient detail about the techniques used to generate the grids for it to be possible to judge the generality of the mesh generation tools he used. Nevertheless, the fact that he was able to demonstrate that three-dimensional zonal flow solutions could be achieved on hybrid grids composed of tetrahedra and hexahedra is indeed worthy of note.

Weatherill proposed the use of hybrid grids by considering the apparent advantages and disadvantages of both the structured and unstructured approaches [Weatherill, 1988a] (at this time, he was well placed to give a pragmatic view on both approaches, having been involved in pioneering work in both block-structured [Weatherill and Forsey, 1985] and unstructured [Jameson, Baker, and Weatherill, 1986] mesh generation for complete aircraft.) He observed that the structured grid approach provides high-quality meshes at a relatively low cost and, because of inherent directional qualities, also provides an ideal environment for accurate and efficient flow algorithm techniques. However, structured meshes can be somewhat restrictive when applied to complex geometries and do not readily admit mesh point enrichment. In contrast,
the unstructured mesh generation techniques have almost total flexibility for complex shapes and readily accept mesh enrichment. These advantages are counterbalanced, however, by their relatively high computational costs and lack of directional properties.

*The observation that lies at the heart of Weatherill's proposition of hybrid grids is that the real advantages of one approach are the disadvantages of the other. The combination of the approaches is an attempt to capitalize on the merits of both approaches.* This was demonstrated in two dimensions by embedding unstructured regions of triangular grid in a background structured quadrilateral grid to

1. Form grids for multielement aerofoils.
2. Perform mesh adaptation to the flow over an aerofoil.
3. Improve mesh quality locally.

In this work, the structured regions were formed using the block-structured approach and the unstructured regions were created using the Delaunay connectivity algorithm [Weatherill, 1988b].

In contrast to Nakahashi, Weatherill [1988a] developed a single finite-volume flow algorithm for use with hybrid grids, as an extension of the scheme of Jameson, Baker and Weatherill [1986]. In this cell-vertex scheme, the control volume for a node was viewed as being the sum of elements containing the node, thereby creating overlapping control volumes. Hence, the flux balancing for nodes at the interface of the two mesh types was achieved by operating over both triangular and quadrilateral elements.

### 23.2.2 The Trend from Unstructured to Hybrid Grids

The unstructured grid approach, based primarily around the Delaunay [Weatherill, 1988b] (see Chapter 16) and moving-front (advancing front) [Morgan, Peraire and Peiro, 1992] (see Chapter 17) algorithms, has been shown to provide a highly effective basis for simulating inviscid flows over complex configurations, particularly when coupled with solution adaptive point enrichment and removal algorithms. However, considerable obstacles have been encountered in attempting to extend these algorithms to the generation of the highly compressed tetrahedra that are necessary for the efficient computation of viscous flows. These difficulties arise principally because both techniques use the properties of a sphere to determine the suitability of point connectivities, which works very well for the generation of isotropic grids, but not for the highly anisotropic grids required to allow shear layers to be resolved.

These problems have motivated workers [Pirzadeh, 1992; Kallinderis, 1996; Marchant and Weatherill, 1994] to investigate employing structured grid generation techniques locally to march triangulations of the geometric surfaces a distance sufficient to cover the expected extent of the shear layer (see Chapter 25). The conventional unstructured mesh generation techniques are then employed to yield a triangulation of the remainder of the domain. This approach allows most of the flexibility of the unstructured approach to be maintained through the use of triangles to cover the surface of the geometry, while also enabling the required point density close to solid surfaces to be achieved.

In some cases, the semistructured layers of prismatic elements that are formed by this surface inflation approach are retained for the flow simulation for reasons of efficiency [Kallinderis, 1996]. In others, each prism is subdivided into three tetrahedra [Marchant and Weatherill, 1994] to avoid the need to have a flow algorithm that operates over more than one element type. Whichever the case, it is possible to make use of the structured nature of the grid normal to the surface to enhance the sophistication of the subsequent modeling, as demonstrated by Weatherill, et al., [1987] in their use of locally structured triangular grids for multielement aerofoil flows.

This approach has met with a considerable degree of success. However, it is prone either to lack geometric flexibility, require excessive user intervention, or produce grids whose quality is not sufficient to support an accurate flow simulation. These limitations are observed in junction regions at discontinuities in surface slope and where a geometry has a high degree of surface curvature in one direction only. Combinations of these features exacerbate matters. Furthermore, the polar-like topology of the semistructured region of grid is such that the point distribution normal to the wake center-line is not of sufficient density for the wake to be adequately resolved.
An alternative approach to the generation of unstructured grids for viscous flows, which uses hybrid grid technology, centers on the use of directional refinement, as proposed in two dimensions by Barth [1994] and extended to three dimensions by Peraire and Morgan [1996]. Initially, an isotropic grid is formed, which is subsequently enriched until each point in the grid satisfies user specified stretching distributions that have been defined for curves and surfaces within the domain.

The scheme appears to offer significant potential savings in that, in addition to the stretching of the grid normal to geometric and wake surfaces, it allows anisotropic surface triangulations to be established in regions where the surface curvature is only high in one direction. However, it is not clear how well point density and element quality can be controlled in junction regions and at the edge of the refined regions. Furthermore, there remains the question of how well viscous flows can be simulated on highly stretched tetrahedral elements. The analysis of Baker [1996] adds significantly to this particular debate.

### 23.2.3 The Trend from Structured to Hybrid Grids

Within the class of mesh generation schemes that have been proposed to extend the application of structured grids to geometrically complex domains, the block-structured [Weatherill and Forsey, 1985] (see Chapter 13) and overlying [Benek, Steger, and Dougherty, 1983] (see Chapter 11) approaches have met with most success. However, neither approach has yet matured sufficiently for novel configurations to be treated accurately in a routine manner.

The block-structured approach has the potential to be the ultimate demonstration of hybrid grid technology. Within each block the grid is formed of regularly arranged hexahedra that can be generated by either of the established structured methods, namely the solution of elliptic partial differential equations or transfinite interpolation. The blocks have an irregular connectivity, however, which for all but the simplest of domains is not amenable to efficient manual specification, as discussed in Shaw and Weatherill [1992]. This motivates a requirement to be able to decompose a domain automatically into a suitable block structure, which can be cast as the need to generate a coarse unstructured grid of hexahedra.

Schonfeld and Weinerfelt [1991] proposed a scheme for this in two dimensions based on the use of the moving front technique to form quadrilateral cells. However, while the scheme was demonstrated for multielement aerofoil configurations, the block structures created did not form the most natural topology for each component, which is a key feature in the successful application of block-structured grids. The objective of forming effective block structures, which can be readily controlled, remains an open problem which if ever realized may be so irregular as to negate most of the advantages of structured grids. The semiautomatic approaches of Shaw and Weatherill [1992], Eiseman, Cheng and Hauser [1994], and Dannenhoffer [1996] represent the most advanced solutions to the problem to date (see Chapter 10).

The proposition that there is a limited range of problems that can be efficiently resolved using the block-structured approach has led Shaw, et al., [1991] to discuss situations where the use of hybrid grids would be favored. This is discussed further in Section 23.3.

Overlying grids do not have some of the restrictions of block-structured grids. However, the time taken to establish the meshes can be significant because of the need to ensure that sudden changes in mesh size are not encountered in overlapping regions. The FAME (feature associated mesh embedding) scheme of Albone [1988; 1992], which adopts a unified treatment to both geometric and flow features, appears to overcome this particular problem. For each feature (whose topology is either a corner, line, or surface), the approach forms individual meshes that are ordered hierarchically for the flow modeling based on the degrees of constraint possessed by the feature. An octree grid, formed by the repetitive subdivision of Cartesian hexahedral cells into eight, is then used to cover the remainder of the domain, with the refinement driven by the mesh spacing of the feature associated grids. This use of very many overlapping regular grids, coupled to the employment in the background of multiple levels of unstructured, embedded hexahedra, appears very flexible. However, it suffers along with other overlying methods with conservation and nonuniqueness problems when transferring solutions between meshes.
Increasingly, overlying mesh generators are migrating toward the use of hybrid grids. Liou and Kao [1994] demonstrate an approach in two dimensions whereby an initial set of regular, overlying grids is formed. The quadrilateral cells in the regions of overlap are then identified and removed from the grids, leaving a void which is subsequently filled with triangles. The approach adopted allows much of the technology developed for overlapping grids to be retained while overcoming the problems of conservation.

Noack, Steinbrenner, and Bishop [1996] have pursued a similar approach in three dimensions. In their work, the background mesh is an octree grid. Structured body-conforming meshes are formed adjacent to solid surfaces, and tetrahedra are used to fill the void between the background and local sets of hexahedra. In contrast to the methodology described in Section 23.3, the triangular faces of the tetrahedra are allowed to abut the quadrilateral faces of the hexahedra directly. The flow algorithm that operates on these meshes is described in Bishop and Noack [1995].

### 23.2.4 Potential Computational Benefits of Using Hybrid Meshes

In this section, observations are made on the computational benefits that structured grids offer in comparison with unstructured grids. If the majority of a hybrid mesh is composed of structured grid, then it is apparent that these benefits will also extend to the hybrid environment.

Shaw, et al., [1993] undertook a study to model inviscid flow over a wing-foreplane-fuselage configuration with both a solely block-structured grid and with a hybrid grid, with the unstructured region containing the foreplane. While this study cannot be viewed as totally rigorous, their findings were that to achieve a similar accuracy in the flow simulation, the surface mesh density of the unstructured grid had to be nearly an order of magnitude more dense than the structured grid. This was due primarily to the isotropic nature of the surface mesh, which meant that to resolve the streamwise curvature of the surface, the spanwise density of the mesh needed to be very high.

Also, it is apparent that independent of the strategy that is pursued to create a mesh for modeling shear layer flow, the required point density normal to the surface will be the same. These factors lead to the conclusion that in the viscous region of the flow domain the point density in an unstructured grid will be approximately ten times greater than in a structured grid. Furthermore, the rate at which time marching can be performed on the unstructured grid will be about one half (for a cell-vertex scheme) of that on a structured grid of the same point density. In addition to this, the amount of work done per time step will be greater due to the larger number of faces and edges in the unstructured grid, and there will also be increased processing time due to the amount of indirect addressing that needs to be done.

This short discussion suggests conservatively that to achieve the same level of convergence and accuracy, computations of viscous flow on an unstructured grid will be more than 20 times as expensive as on a structured grid.

Turning to memory requirements, the findings reported in Shaw, Peace and Weatherill [1994] indicate that the storage requirements per point for a flow solution on an unstructured grid are about four times those of a structured grid. Even if the unstructured grid is coarser in the farfield, the storage of the unstructured grid, with its greater total number of points, would typically be 40 times greater than for a structured grid.

There are thus very clear incentives to use structured grids whenever possible. For hybrid grids the implication is that the extent of unstructured grid employed should be as minimal as possible.

### 23.3 Best Practices

There is general agreement on what is needed in a mesh. It must conform to boundaries, contain points that are distributed effectively, be defined in a manner amenable to efficient computations, and have connections between points that form elements whose geometric properties satisfy certain criteria. The relative importance attached to these requirements will depend upon the problem being addressed.
In this chapter, the application is the modeling of the high Reynolds number turbulent flows associated with the complex bodies studied by aero- and hydrodynamicists, for which reviews of the demands on a mesh are given by Albone [1988], Albone and Swift [1996] and Patis and Bull [1996]. The task of simulating the flows over these geometries presents the severest of challenges to the traditional use of a single element type in a mesh and represents the principal industrial need that has motivated the current interest in hybrid grid generation/technology.

An example of this, which will be the focus of the remainder of this chapter, is the work of the Research Group at ARA in hybrid grid generation that has led to the development of the SAUNA (Structured and Unstructured Numerical Analysis) CFD system.

23.3.1 Mesh Generation Techniques Employed in the SAUNA System

In this section the basic mesh generation techniques that have been developed for the SAUNA system, which has been used to generate the grids described later, are reviewed. The system is capable of forming either solely block-structured, semistructured, or unstructured grids. In addition, it is capable of forming a hybrid combination of any of these mesh types. Hence, the same system can be used to form meshes efficiently for problems as diverse as the steady, viscous flow over a civil aircraft or the unsteady inviscid flow over a store released from a carriage bay.

23.3.1.1 Overview of Development

The initial approach to grid generation pursued within SAUNA was coined “Multi-block” [Weatherill and Forsey, 1985]. It centered around the formation of a global grid through the patching together of many structured, nonoverlapping grid systems, each of which covered a region that was topologically equivalent to a cuboidal block. This block-structured approach was applied to increasingly complex problems through the 1980s with a considerable degree of success [Shaw, Georgala and Weatherill, 1988]. However, as more and more complex configurations were attempted, so an appreciation of the limits to the range of problems the approach can handle developed.

Following a study of the hybrid approach in two dimensions [Weatherill, 1988a], work began on the initial development of a three-dimensional hybrid capability [Shaw, Peace and Weatherill, 1994]. The objectives were to explore ideas and gain an appreciation of the major issues that would need to be addressed to create a CFD system based on the hybrid philosophy.

The full development of a hybrid capability then began in earnest. The grid generation strategy for inviscid flow modeling centered around the use of hexahedral volumes combined into blocks, wherever they are readily attained, with pockets of tetrahedral grid embedded as appropriate to model local regions of high geometric complexity [Shaw, Georgala, Peace and Childs, 1991].

In the extension of this hybrid approach to the creation of grids for viscous flow modeling, the use of prismatic grid regions has been addressed, this additional grid type fitting in naturally to the hybrid grid framework [Chappell, Shaw, and Leatham, 1996; Peace, Chappell, and Shaw, 1996]. For geometric regions that are sufficiently complex to require an unstructured surface grid, the structured extension of the grid away from the surface allows layers of semistructured prismatic elements to be created. The regular nature of the grid normal to the surface is seen as being preferable to a fully unstructured approach in terms of both accuracy and efficiency. However, to achieve high mesh quality in junction regions, the approach requires to be augmented by a capability to create local block-structured regions between two intersecting surfaces from which prisms are grown. This avoids the need to generate prismatic elements in the regions highlighted as being difficult in Section 23.2.2.

A natural hierarchy of mesh elements for viscous flows can be drawn from the discussion to date and indeed this is the order in which the elements are created:

1. Block-structured hexahedral grid.
2. Semistructured prismatic grid.
3. Unstructured tetrahedral grid.

The generation of these grids is now considered in turn.
23.3.1.2 Structured Grid Generation

The multi-block approach [Weatherill and Forsey, 1985] is employed for the generation of structured grids. The domain is decomposed into an assemblage of topologically cuboidal blocks, each of which possesses its own curvilinear coordinate system. Grid lines are constrained to pass between block interfaces with continuity of position, slope and curvature. The technique allows the embedding of appropriate mesh structures local to components. The connectivity arrangement of blocks, known as the block topology, is determined via a semiautomatic approach, based on an input schematic representation of the configuration [Shaw and Weatherill, 1992].

23.3.1.2.1 Surface Grid Generation.

The generation of the surface grids is accomplished via the solution of elliptic PDEs [Thompson, Thames, and Mastin, 1974], with the initial boundary point distribution established automatically using an algorithm that is sensitive to local grid topology and geometry [Shaw and Weatherill, 1992] (see Chapter 9). If the default grids are found to be of insufficient quality, a graphics-based module is employed to modify boundary point distributions and add constraints to the mesh. The meshes are subsequently regenerated until satisfactory quality is achieved.

23.3.1.2.2 Field Grid Generation

The field mesh for inviscid flows is also generated by solution of elliptic PDEs with the source terms calculated using the method proposed by Thomas and Middlecoff [1980] (see Chapter 4). Algebraic techniques are employed to enrich the mesh for viscous flow modeling to allow exact control of the first cell height away from the surface (see Chapter 3). A capability to regenerate the mesh automatically in response to a perturbation of the geometry allows the system to be embedded within a design optimization strategy [Lovell and Doherty, 1994].

Mesh adaptation to either viscous or inviscid flow phenomena is performed using the LPE method of Catherall [1996]. This involves the numerical solution of equations for node positions that are formed as a linear combination of an inverted Laplace equation, an inverted Poisson equation, and an equidistribution equation. The Laplace term promotes smoothness and orthogonality, the Poisson term enables the retention of favorable features of the initial mesh, and the equidistribution term controls the redistribution of nodes according to a measure of solution activity. Mesh adaptation is covered in Part IV of this Handbook.

Prior to performing a flow simulation, the grid is decomposed into microblocks containing four cells in each coordinate direction. This micro-block structure is then recombined into macro-blocks based on either the requirement to distribute the grid effectively over a number of processors or to allow long loops to be achieved on vector machines. This recombination capability is also used in the generation of hybrid grids to redefine the grid into blocks when part of the initial block-structured grid has been removed to be replaced by tetrahedra and/or prisms.

23.3.1.3 Semistructured Grid Generation

The technique employed [Chappell, 1996] for generating prismatic elements is a marching method, and as such starts from a defined surface and propagates outwards to an outerboundary, the exact shape or location of which cannot be predetermined. The prismatic grid is built up one layer at a time. At each stage, the positions of points in the next layer are determined as a function of the current outer grid surface, which will initially be the input unstructured surface grid.

The generation of a prismatic layer can be separated into two distinct processes: the evaluation of normal vectors and the determination of marching distances along these vectors.

23.3.1.3.1 Evaluation of Normal Vectors

The first stage of the prismatic grid generation process is the determination of marching direction vectors at all points on the unstructured surface. This is achieved by evaluating the normals to all surface triangles and sending contributions to the forming nodes weighted according to the angle subtended at the node. All nodal vectors are then normalized to unit magnitude.
This yields an approximately normal marching vector for every point on the current grid surface. If these vectors are used in this form, however, the normal grid lines will converge from concave surface regions, leading to grid crossover. This undesirable feature can be overcome by an iterative smoothing of the vectors using a Laplacian filter, with the amount required being surface-topology-dependent. The trade-off is a reduction in grid orthogonality.

23.3.1.3.2 Marching Distances along Normal Vectors
In the development of prismatic grid generation methods, workers have given much attention to the determination of appropriate marching distances along each grid line. If the initial surface features any concave regions, then the maximum distance away from the body to which the grid can extend will be limited, unless some form of marching distance variation is employed.

The goal of marching distance variation within a layer is to compensate for regions of high concave and convex curvature, increasing marching distances in the former case and reducing them in the latter. The overall effect is that the grid tends toward a spherical effect as it moves away from the geometric surface. Several approaches to this problem have been investigated with a spring analogy approach found to be the most successful [Chappell, 1996].

By treating the normal vectors connecting a point to its neighbors as springs and summing their effects, an overall “spring force” vector for the point can be calculated. The scalar product of this vector with the nodal normal vector gives a measure of the local surface curvature. In convex regions, where the net effect of the adjoining points will act in opposition to the marching direction, a negative measure will be returned, and vice-versa in concave regions. This measure can form the basis of a marching distance modification function which, with appropriate use of unit vectors, is independent of the distance between a node and its neighbors. The modification function is subject to two constraints. The first checks that the value lies within an appropriate range, the second ensures stability as the grid propagates radially. An average distance for the layer is calculated based on user-defined parameters, which is then multiplied by the modification function to give the nodal marching distance.

23.3.1.4 Unstructured Grid Generation
The optimal properties of the Delaunay connection algorithm, and efficient algorithms that exist for its implementation, led to its adoption within the SAUNA system for forming the regions of tetrahedral grid [Childs, et al., 1992; Childs and Shaw, 1993]. The mesh generation is performed in two stages: surface grids, followed by volume grids. For the former, the generation of grids that are independent of the geometry definition has been a particular focus of effort. For the latter, the problem of boundary integrity requires careful attention. See Chapters 19 and 16 for a discussion of unstructured surface and volume grid generation.

23.3.1.4.1 Unstructured Surface Grid Generation
Separate meshes are formed for each surface of the configuration and for the boundary of the domain. For each, boundary point distributions are defined in a graphics-based working environment, with boundary lines delimited into segments to facilitate precise control over distributions. These point distributions can be augmented by fixed internal lines either to exercise precise control of the local grid or ensure that a feature (i.e., a slope discontinuity) is resolved accurately.

To be consistent with the creation of a high quality Delaunay field mesh, it is required that the surface meshes consist of triangles that are approximately equilateral in physical space. To this end a pseudo-Delaunay surface triangulation procedure has been developed [Childs and Shaw, 1993], which is coupled to a grid point location algorithm. Control of grid density in regions of high surface curvature is assured through the solution of an optimization problem based on determining a desired edge length distribution. Each surface grid is generated independently, and they are then unified to form the bounding grid for the field grid.

23.3.1.4.2 Boundary Integrity
The Delaunay approach is beset by its inability to ensure that the resulting triangulation conforms to the boundaries of the flow domain–boundary integrity. Therefore, if the scheme is to be applied routinely, the basic methodology must be supplemented by a procedure that overcomes this limitation.
To this end, an automatic boundary integrity algorithm has been developed that consists of local modifications to the datum bounding surface grids so that they more closely match the Delaunay triangulation of the boundary points. Such modifications are limited only by topological considerations and the need to keep a faithful geometric description of curved surfaces. The procedure is an iterative one that is deemed to have converged when all edges and faces of the boundary triangulation are contained in the tetrahedrization. The full implications of this approach to boundary integrity are discussed later, in the section covering the interfacing of different grid types.

23.3.1.4.3 Unstructured Field Grid Generation

The three-dimensional grid is determined automatically from the bounding surface grids with grid points positioned according to boundary grid density, curvature and desired rate of change of grid density. The procedure commences with the creation of an initial octree model of the flow domain (see Chapters 14 and 15). Each octant is subdivided as necessary until the density of the terminal octants cutting boundary surfaces is comparable with that of the boundary grid. Further levels of refinement of the octree are then performed based on surface curvature. Finally, the octree is graded so that adjacent octants do not differ by more than one level. Grid points are then located within the empty octants that lie interior to the unstructured domain and connected together to form a coarse tetrahedrization of the domain. This grid is used as the basis for solving a coupled set of PDEs which yield a desired edge length in the field.

A denser set of points is then formed by selective addition of suitable points to the Delaunay grid, via an automatic edge refinement procedure, until the optimal edge lengths for the tetrahedra are attained. Throughout, it is found essential to employ the generalized Delaunay algorithm wherein the grid is allowed to become non-Delaunay, due to boundary influences, but only if grid quality is enhanced.

Mesh smoothing techniques, coupled with point addition and removal algorithms are used to regenerate the grid in response to a change to the shape of the boundary of the domain. This technology can be used to achieve meshes rapidly either as a result of a design modification or in response to the motion of a body, as in a store release [Leatham, 1996].

23.3.2 Interfacing Different Grid Types

The interfacing of the different elements of a hybrid grid represents a major component in the development of a hybrid grid generation system, which must be performed in an automatic manner. In this section, the interfacing of block-structured, unstructured, and semistructured grids is discussed.

23.3.2.1 Interfacing Structured and Semistructured Grids

At the interface of block-structured and prismatic grid regions, the quadrilateral faces of the elements must abut. This means that all points on the interface will be fixed points to which the prismatic grid generator must conform as the layers are formed.

To make the transition from block-structured to prismatic grid as smooth as possible, the vectors resulting from the fixed boundary points are used in the smoothing process for the normals in the prismatic region [Chappell, 1996]. This has the effect of preventing any sharp changes of direction near the interface.

To obtain a representative marching distance for the prismatic grid, a Laplacian equation is solved for each layer, with the multi-block mesh spacing providing the necessary boundary data.

23.3.2.2 Interfacing Structured and Unstructured Grids

Clearly, some form of special treatment is required at the interface between regions of structured hexahedral grid and unstructured tetrahedral grid. One strategy could be to allow a number of tetrahedral faces to abut the face of a hexahedra. However, while this would simplify the grid generation process, a significant burden would be placed on the flow solver, which would not only have to perform well on different types of elements but would also have to be insensitive to hanging nodes, edges, and faces.

Alternatively, an additional element, the pyramid can be used. For if the quadrilateral base of this element adjoins a hexahedron, the remaining triangular faces can abut to this tetrahedra, thereby maintaining a one-to-one connectivity of all faces within the mesh. This is the approach that has been followed.
However, due to the point addition and edge swapping techniques adopted to ensure that the Delaunay algorithm conforms to the boundary of the domain, the interface of the pyramid elements is augmented locally by a buffer “layer” of tetrahedra, prior to the generation of the unstructured grid. These tetrahedra are formed in two stages, the first of which protects the faces, and the second the edges of the pyramids from the unstructured grid generator. The pyramids and initial layer of tetrahedra are both formed in an automatic manner [Shaw, et al., 1991].

Following the generation of the unstructured region of grid, the initial layer of tetrahedra at the interface needs to be adjusted as a result of the steps taken to ensure boundary integrity. An automatic module has been developed to accomplish this task in response to knowledge of the edges that have been swapped on, and nodes that have been added to, the boundary of the unstructured domain.

23.3.2.3 Interfacing Semistructured and Unstructured Grids

There are three principal factors governing the ideal extent of the prismatic region, the first two of which place a lower limit and the third an upper limit on the extent of the prismatic region:

1. The grid should extend to a distance where viscous effects become negligible.
2. The cell aspect ratio (height/average side length) should be as close to unity as possible to promote a smooth transition to the tetrahedral region.
3. The quality of the triangulation on the outer layer should be as good as possible in order to achieve a good quality tetrahedral mesh.

The concept of a buffer layer is also used to interface prisms and tetrahedra [Chappell, Shaw, and Leatham, 1996]. In this case the buffer is not needed to ensure compatibility of element faces but rather to eliminate the need to modify the prismatic grid after the generation of the unstructured grid. The prismatic region can be insulated from the effects of the procedure followed to achieve boundary integrity by breaking down the outer layer of prismatic cells, with each prism becoming three tetrahedra. This operation must be performed in such a way that the diagonal introduced by splitting a quadrilateral face matches for both prisms abutting that face. An iterative algorithm for achieving this type of decomposition of the outer-most layer of semistructured grid was originally proposed by Lohner [1993]. The set of face splits derived from this are used to determine the initial make-up of the tetrahedral buffer layer.

Following the generation of the unstructured grid, the same procedures that are used to modify the definition of the tetrahedra in the structured/unstructured interface region are used to modify the tetrahedra in the semistructured/unstructured interface.

On completion of the generation of the unstructured grid, all grid types are passed to a separate module that forms the complete data structure describing the grid.

23.3.3 Data Structures for Describing Hybrid Grids

The data structure that describes the hybrid grid to the flow solver is central to the success of the approach. The description that has been adopted for a cell-vertex scheme is detailed in Peace and Shaw [1992].

The nodes are all uniquely numbered, with all nodes at which a given boundary condition is applied stored contiguously. Nodes that either lie inside each block or solely within the unstructured or semistructured field grids are also stored contiguously.

Connectivity matrices are used to describe the joining of faces of tetrahedra to the triangular faces of either other tetrahedra, or prisms, or pyramids. Similarly, for the prisms, the unstructured surface grids are stored in edge-based connectivity matrices, with surface node-based pointers used to define the nodes lying along the lines of structure in the grid. The block-structured region is stored in a block-based structured manner for points that do not lie on block faces. A pointer system, based on the faces of each block, is used to access nodes that lie on block faces; these nodes might be either part of more than one block or be part of other elements. All edges in the unstructured grid and on the boundaries of both blocks and regions of semistructured grid are stored also.
In Section 23.3.1, the main emphasis of the discussion is on the techniques that have been developed to position nodes for each of the grid types. It is worth noting that a significant part of the total work undertaken to develop the individual mesh generation modules has focused on creating and communicating data that allows the data structure described above for the complete hybrid grid to be defined automatically.

23.3.4 Examples of Hybrid Meshes

The creation of grids for three different aerodynamic and hydrodynamic configurations is discussed in order to highlight what are considered to be “best practices” in the generation of hybrid grids. The configurations are chosen because they demonstrate the three possible types of mesh that can be formed with the SAUNA grid generation system. The examples begin with the combination of block-structured and unstructured grids, followed by the use of semistructured and unstructured grids and end with an example which utilizes all mesh types.

Due to the commercial sensitivity of some of the configurations shown, there are no results presented in this section from flow calculations performed on these meshes. However, details of the flow algorithm that operates on these meshes can be found in Peace and Shaw [1992], with results from both inviscid and viscous flow calculations given in Shaw, et al., [1993], Shaw, et al., [1994b], Peace, et al., [1994], and Peace, et al., [1996]. Further discussions on the generation of hybrid meshes can be found in these references and in Shaw, et al., [1994b].

23.3.4.1 Creation of a Block-Structured/Unstructured Grid for a Civil Aircraft*

To illustrate the creation of a block-structured/unstructured grid, a civil wing-fuselage-pylon-nacelle configuration has been chosen. While block-structured grids have been formed for this type of layout, significant time was taken to establish these meshes, which are of a questionable quality around the pylon. Furthermore, apparently minor changes to the pylon geometry can lead to a major requirement to modify the local block topology.

However, if the configuration is considered without the pylon, then the remaining components, both individually and collectively, are well-suited to the generation of a block-structured grid. The highly three-dimensional pylon, with its complex shaping and intersection with both the wing and nacelle surface, is readily modeled by an unstructured grid. The complete configuration can therefore be addressed efficiently by the hybrid approach without having to incur the overhead associated with the completely unstructured approach.

The creation of the hybrid grid commenced with the decomposition of the domain around the wing-fuselage and nacelle into blocks. A polar topology was embedded around the fuselage, with a spherical polar topology chosen to model the nose region. A “C” topology conformed to the wing leading-edge geometry. Finally, a polar topology internal and external to the nacelle, with a “C” topology around the intake lip, yielded a total of 642 blocks.

The next stage in the creation of the mesh was to identify the extent of the structured grid that should be removed and replaced by unstructured grid. In this case, the pylon geometry was introduced into the structured grid and all micro-blocks that either contained the pylon or lay within a user specified distance of the pylon were removed. The remainder of the structured grid was combined into 34 macro-blocks and the initial structured/unstructured interface formed as depicted in Figure 23.1.

An unstructured grid was formed on the pylon and part of the wing and nacelle surfaces. In conjunction with the inner triangulation of the structured/unstructured interface this formed the boundary data for the generation of the unstructured field mesh. The meshes were then fused together as shown in Figure 23.2.

*Grid generated by A. Shires, DERA, Bedford, UK and C.M. Newbold, DERA, Farnborough, UK.
FIGURE 23.1 Boundary for unstructured part of hybrid grid.

FIGURE 23.2 Hybrid structured/unstructured grid configuration 1.
23.3.4.2 Creation of a Semistructured/Unstructured Grid for a Submarine*

To demonstrate the use of semistructured/unstructured grids, a fully appended submarine is chosen. The configuration features several regions of high geometric difficulty for the generation of prismatic grids, i.e., where the trailing-edges of the control surfaces intersect the hull, and a large range of length scales.

Unstructured grids were generated on all surfaces, as shown in Figure 23.3, and the surface inflation technique employed to generate the semistructured grid away from the submarine. In practice, several attempts had to be made to achieve a valid extent of prismatic grid, with user input parameters that control the amount of smoothing and stretching of the grid adjusted to achieve the desired result.

Unstructured surface grids were then formed for the farfield boundary and the remaining extent of the symmetry plane that had not yet been covered. These were used in conjunction with the outer triangulation of the semistructured/unstructured buffer to provide boundary data for the generation of the unstructured field grid.

Part of the unified hybrid mesh is shown in Figure 23.4. Note in particular how the marching distance of the grid adjusts to the size of the local surface triangulation to achieve a smooth transition of cell sizes at the prismatic/tetrahedral interface.

While the example is an impressive demonstration of prismatic mesh generation, it does illustrate some of the weaknesses of a strategy based on the sole use of prisms and tetrahedra that were alluded to in Section 23.2.2.

23.3.4.3 Creation of a General Hybrid Grid for a Store Below a Research Aircraft**

The final configuration examined is a wing/fuselage/foreplane research model below the wing of which is a finned store. To model the full trajectory of the store as it is released from the aircraft, possibly pitching, yawing, and rolling, is beyond the efficient application of block-structured grids.

* Grid generated by J.A. Chappell, ARA, Bedford, UK.
**Grid generated by J.A. Shaw and J.A. Chappell, ARA, Bedford, UK.
However, the parent aircraft is amenable to the generation of a block-structured grid, which was readily attained. A region of this grid below the wing was then removed and the block-structured/unstructured interface constructed.

Layers of prismatic grid were grown from a surface triangulation of the store and fins. The field mesh was completed by forming tetrahedral grid in the region between the block-structured/unstructured and semistructured/unstructured buffers.

Figures 23.5 and 23.6 illustrate the full hybrid surface grid and a section through the field grid, respectively. The case amply demonstrates the building block route to forming efficient, high-quality grids for configurations of great complexity that is possible with hybrid grids.

### 23.4 Research Issues and Summary

It is generally accepted that the techniques for generating meshes are well established and that the main challenge lies in developing highly usable systems around these techniques. This is particularly the case for hybrid grids.

For hybrid grids to become acceptable in the user environment, clear indication of where a given mesh type should be used for the problem of interest needs to be available. A user’s own knowledge base, coupled with good training, on-line support, and documentation will go some way toward meeting this objective. However, what is and is not a difficult region of geometry to mesh with a structured grid is
not always readily appreciated, even for experienced practitioners. Some form of artificial intelligence that interrogates the local geometric properties of the boundaries of a domain would appear to be required, but the level of sophistication that would be needed should not be underestimated.

It is apparent that the simulation of aerodynamics and hydrodynamic flows will be performed increasingly on parallel processors. For effective computations to be achieved on these platforms, the algorithms used to decompose the domain need to be capable of providing a good load balance across all processors. This becomes an increasingly significant issue in a hybrid grid where the topology of the structured regions imposes significant constraints on the decomposition and the different elements require different processing times per time step.

It was expected initially that significant problems may be encountered in the flow simulations at the interface between the different element types. To date this has not been observed, which may be testament to the care taken to join the grids together. However, it would be naive to suggest that this region of mesh, which inevitably contains significant changes in element size, could not lead to difficulties. Further validation of the flow solution in these regions is needed.

Furthermore, each of the tools within the hybrid system must be of a similarly high quality and easy to use since the number of modules that need to be executed to produce the complete grid is inevitably significant compared to single element systems.

The inevitable impact of this is the expense and long term commitment to the philosophy that is required to develop a usable capability. When many groups have already invested heavily in either structured or unstructured grid technology, the decision to move to hybrid grids is not taken easily.

While numerous papers are now appearing on the approach in two dimensions, the evidence of work in three dimensions is sparse. The formation of strategic alignments between major industrial companies and/or government bodies, which allow specialists in the two main fields of grid generation to collaborate, could arguably have the greatest impact on changing this situation.

Hybrid grid generation offers the potential of combining the advantages of structured and unstructured grids, enabling high quality, efficient meshes to be formed for a wide range of problems. The meshes will inevitably take longer to form and require greater expertise than totally unstructured grids. However, the potential efficiency and modeling gains that hybrid grids offer are such that the total elapsed time and cost to achieve the end result the engineer needs justifies this required investment.
Acknowledgments

This work has been undertaken with the support of the Procurement Executive, United Kingdom Ministry of Defence. I am grateful to my past and present colleagues at ARA who have contributed their ideas and effort to the work described here and am indebted to those who have so willingly provided the figures for the examples discussed.

Finally, I would like to dedicate this article to Dr. David Catherall, who has acted as technical monitor for the work described here throughout its development and who is shortly to retire from full-time work at DERA Farnborough, U.K. Dave's consistent support over many years for the work described has been, and still is, greatly appreciated.

References


17. Leatham, M., Private communication, 1996.


