



**ROYAL INSTITUTE
OF TECHNOLOGY**

Hybrid grid generation for viscous flow
computations around complex geometries

by

Lars Tysell

January 2010
Technical Reports from
Royal Institute of Technology
Department of Mechanics
SE-100 44 Stockholm, Sweden

Typsatt i $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\text{\LaTeX}$.

Akademisk avhandling som med tillstånd av Kungliga Tekniska Högskolan i Stockholm framlägges till offentlig granskning för avläggande av teknologie doktorsexamen fredagen den 19:e februari 2010 kl 10:15 i sal F3, Kungliga Tekniska Högskolan, Lindstedtsvägen 26, Stockholm.

© Lars Tysell 2010

Tryck: Universitetsservice US-AB, Stockholm 2010

How to place points and connect them with straight lines.

Hybrid grid generation for viscous flow computations around complex geometries

Lars Tysell

Department of Mechanics, Royal Institute of Technology
SE-100 44 Stockholm, Sweden

Abstract

A set of algorithms building a program package for the generation of two- and three-dimensional unstructured/hybrid grids around complex geometries has been developed.

The unstructured part of the grid generator is based on the advancing front algorithm. Tetrahedra of variable size, as well as directionally stretched tetrahedra can be generated by specification of a proper background grid, initially generated by a Delaunay algorithm.

A marching layer prismatic grid generation algorithm has been developed for the generation of grids for viscous flows. The algorithm is able to handle regions of narrow gaps, as well as concave regions. The body surface is described by a triangular unstructured surface grid. The subsequent grid layers in the prismatic grid are marched away from the body by an algebraic procedure combined with an optimization procedure, resulting in a semi-structured grid of prismatic cells.

Adaptive computations using remeshing have been done with use of a gradient sensor. Several key-variables can be monitored simultaneously. The sensor indicates that only the key-variables with the largest gradients give a substantial contribution to the sensor. The sensor gives directionally stretched grids.

An algorithm for the surface definition of curved surfaces using a biharmonic equation has been developed. This representation of the surface can be used both for projection of the new surface nodes in h-refinement, and the initial generation of the surface grid.

For unsteady flows an algorithm has been developed for the deformation of hybrid grids, based on the solution of the biharmonic equation for the deformation field. The main advantage of the grid deformation algorithm is that it can handle large deformations. It also produces a smooth deformation distribution for cells which are very skewed or stretched. This is necessary in order to handle the very thin cells in the prismatic layers.

The algorithms have been applied to complex three-dimensional geometries, and the influence of the grid quality on the accuracy for a finite volume flow solver has been studied for some simpler generic geometries.

Descriptors: grid generation, unstructured grids, hybrid grids, surface grids, three-dimensional grids, adaptive grids, moving grids, grid quality, accuracy.

Preface

This thesis deals with algorithms for three-dimensional hybrid grid generation. The papers in the thesis have been written concurrently with the development of a complete grid generation system. This development spans over two decades, and the papers, including the appendix, may be viewed as a survey of the advances in grid generation during this period, from structured grid generation to hybrid grid generation. Each paper covers a separate subject of grid generation.

The thesis is divided into three parts. The first part is an introduction and background to the work covered by the papers. The contribution to the field of grid generation by this thesis is also included. A guide to the papers is included as the last chapter of this part. The second part contains the papers, and the last part is an appendix with the supplementary papers.

The papers in part two and three are adjusted to comply with the present thesis format for consistency, but their content have not been altered compared to published versions except for minor corrections.

Stockholm, December 2009

Lars Tysell

Contents

Preface	vi
Part 1. Overview and summary of papers	1
Chapter 1. Overview	3
1.1. Introduction	3
1.2. The Navier-Stokes equation from a grid generation point of view	4
1.2.1. Spatial discretization	5
1.2.2. Temporal discretization	6
1.3. Grid generation - a survey	7
1.3.1. Structured grids	8
1.3.2. Unstructured grids	10
1.3.3. Hybrid grids	11
1.3.4. Adaptive grids	12
1.3.5. Hybrid grid generation systems	13
1.4. Grid quality and accuracy	15
1.5. The grid generation environment	21
1.6. Major grid generation algorithms in use	22
1.7. Applications of the grid generation algorithms	27
1.8. Recent advances in hybrid grid generation	28
1.9. Summary and conclusion	34
1.10. Outlook	35
Chapter 2. Summary of papers and authors contributions	36
Acknowledgements	40
Bibliography	41
Part 2. Papers	49
Paper 1. Adaptive grid generation for 3D unstructured grids	53

Paper 2.	Hybrid grid generation for complex 3D geometries	67
Paper 3.	Grid deformation of 3D hybrid grids	79
Paper 4.	A higher order representation of unstructured surface grids	91
Paper 5.	Accuracy evaluation of the unstructured node-centered finite volume method in aerodynamic computations	105
Paper 6.	Experiences of grid generation and steady/unsteady viscous computations for complex geometries	121
Part 3.	Appendix	143
Paper 1.	Towards a general three-dimensional grid generation system	147
Paper 2.	An advancing front grid generation system for 3D unstructured grids	169
Paper 3.	The TRITET grid generation system	197
Paper 4.	CAD geometry import for grid generation	211

Part 1

Overview and summary of papers

CHAPTER 1

Overview

1.1. Introduction

Continuum mechanic problems, like fluid dynamics, can be represented by a set of partial differential equations. These equations can normally be solved analytically only for very simple cases. For general cases the equations need to be solved numerically. The numerical methods for solving partial differential equations require a discrete set of points covering the physical domain. The solution of the system of partial differential equations can be greatly simplified by a well-constructed grid. Thus, one of the central problems in computing numerical solutions to partial differential equations is that of grid generation. The most important requirements for grids are smooth distribution of grid points in the entire field and the concentration of grid points in regions of large gradients of the flow quantities. The efficiency of the computation can be greatly improved if the points are placed where they are most needed.

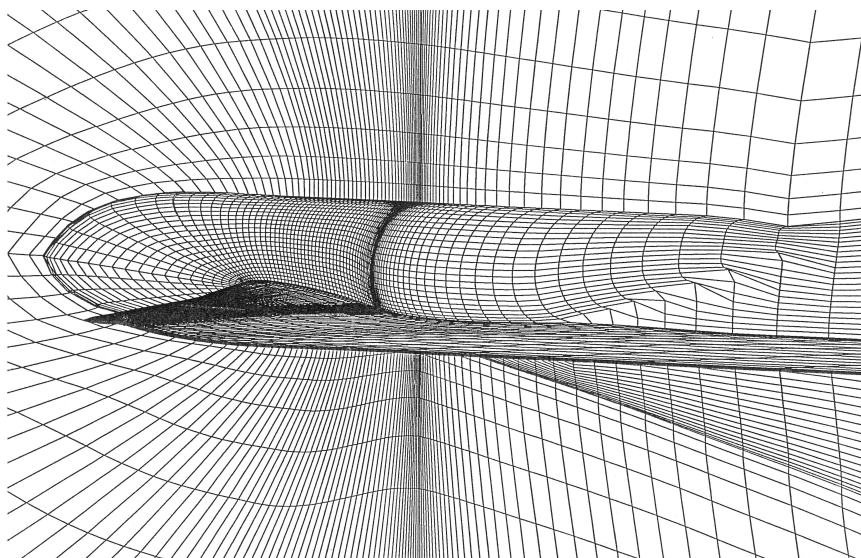


FIGURE 1.1. Single block grid for a transport aircraft - fuselage, wing, wake and symmetry plane.

From the point of view of practical engineering applications, it is desirable for the grid generation method to be fast and of great generality. The application of computational fluid dynamics in the solution of real world problems has given rise to many difficult grid generation problems. Although remarkable progress has been made in numerical grid generation, there is still room for the development of numerical procedures, especially for complex three-dimensional geometries.

Finally, grid generation is an applied science, and a set of grid generation algorithms has a value only when they have been packaged together into a well-written, robust, fast and user friendly environment.

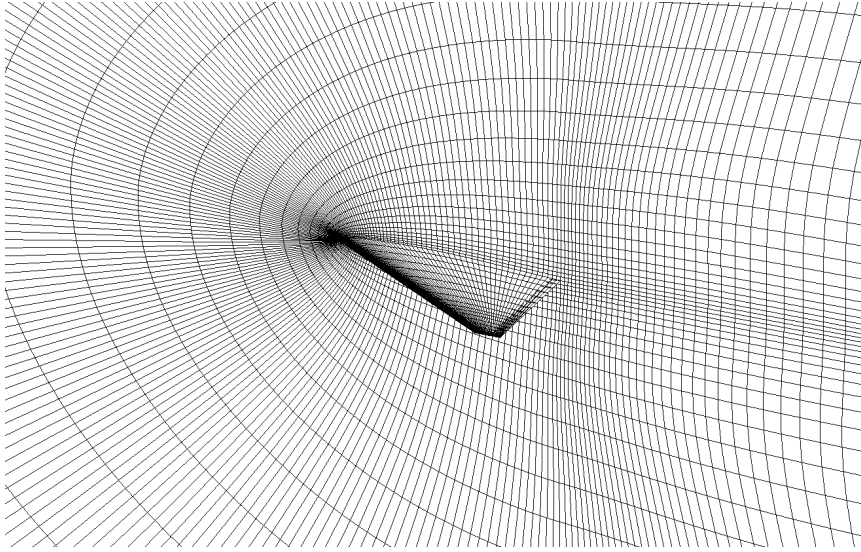


FIGURE 1.2. Single block grid for a delta wing.

1.2. The Navier-Stokes equation from a grid generation point of view

The set of partial differential equations governing steady viscous flow is the Reynolds-averaged Navier-Stokes equation. This equation expresses the conservation of mass, momentum and energy. The turbulent viscosity introduced in these equations is calculated by adding equations for a turbulence model, thus closing the set of equations. To solve the Navier-Stokes equations for a real complex three-dimensional configuration is a challenging task. The space must be properly resolved by the grid. In many regions, like at shocks, wing leading and trailing edges, regions of separation, and in boundary layers having very steep gradients, the grid cells must be very fine. In order to resolve the boundary layer close to boundaries, without using wall functions, the grid must

have extremely fine resolution in the direction normal to the boundary. In this region the grid cells are extremely stretched. Due to computational requirements it is not possible to have these fine grid cells throughout the entire flow field. Thus, this is also a challenging task for grid generation.

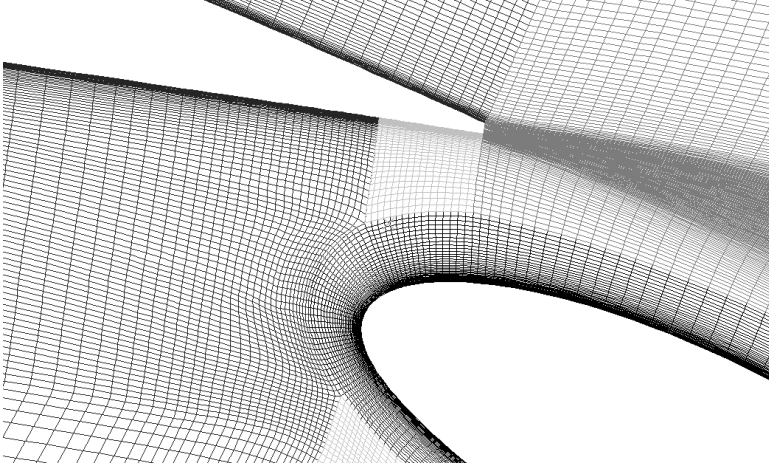


FIGURE 1.3. Close-up of a two-dimensional multiblock grid for a high-lift configuration.

1.2.1. *Spatial discretization*

The most common methods for space discretization in continuum mechanics are the finite difference method, the finite element method and the finite volume method. The last two methods can be applied on grid cells of general type, like hexahedra, pyramids, prisms or tetrahedra. The main reason for using methods capable of handling different types of grid cells is the major simplification in grid generation for complex geometries. Unstructured grids have for a long time been used in finite element methods but have later become popular also for finite volume methods. For the flow computations in this thesis the edge-based finite volume solver in Eliasson (2001) and Eliasson (2002) has been used. The accuracy of the solution is dependent on the grid resolution and is improved as the grid is refined. However, the number of grid points used is limited in practice by the available computer. Thus, the number of grid points should be distributed in a manner which minimizes the error in the solution. It is well known that the errors can be reduced by clustering the grid points in regions where the solution is changing rapidly.

Finite volume methods have an inherent requirement of high grid quality. The size and shape of the grid cells should not vary too much between two adjacent cells. Tetrahedra having large dihedral angles should also be avoided. Theoretically the order of accuracy should be second order, but for real grids

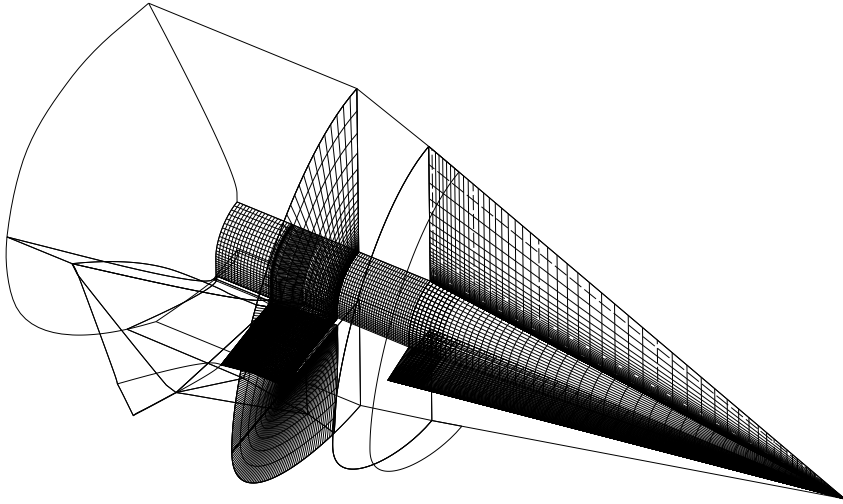


FIGURE 1.4. Three-dimensional multiblock grid for a space vehicle.

of normal quality it is often lower. Finite element methods are more forgiving in the respect of grid cell quality, even if it has been shown that large dihedral angles should be avoided, see Babuska & Aziz (1976).

In finite volume methods the governing equations are written in integral form. This integral equation represents the conservation of mass, momentum and energy. The conservation holds for any volume, in particular for the control volumes. In a cell-centered method the control volumes are taken to be the volumes defined by the grid itself, whereas in a node-centered method the control volumes are defined by the union of the volumes around each grid point in the dual grid. These control volumes defined by the dual grid are normally not convex and they may have very odd shapes. Only for a completely tetrahedral grid generated by a Delaunay method the control volumes will always be convex. The Navier-Stokes equations are solved by computing the fluxes for the control volumes. For the node-centered method the conserved flow variables are stored at the grid points.

1.2.2. *Temporal discretization*

Either implicit or explicit techniques may be used for solution in time of the Navier-Stokes equation. In the finite volume solver used for the flow computation in this thesis an explicit technique is applied. Convergence to steady-state can be accelerated by sacrificing the time accuracy and advancing the solution at each grid point in time by the maximum possible time step at this grid point. For further convergence acceleration the multigrid techniques must be used.

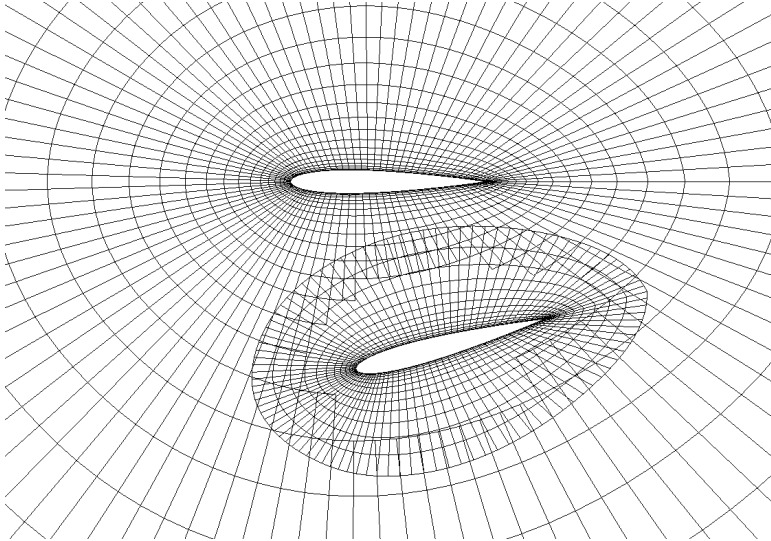


FIGURE 1.5. Two-dimensional Chimera grid.

For structured multiblock grids multigrid can easily be applied by just excluding every second point in each direction for the grid on one coarser level. This is not possible to do in the same natural way for unstructured grids. To generate a coarser grid from a fine one is a difficult task. The main characteristics of the configuration should preferably be kept and the grid cell distribution should be smooth. Some algorithms have been presented by Waltz & Löhner (2000) and Ollivier-Gooch (2003). The opposite approach is to first generate a coarse grid and then successively refine it, see Connel & Holmes (1994). A disadvantage with this approach is the difficulty to generate very coarse grids for complex three-dimensional configurations, e.g. in narrow gaps between a wing and a flap. Another approach is to generate a set of completely unnested grids, see Mavripilis (1990) and Berglind (1997). This means the nodes in one grid do not coincide with the grid in the next finer grid. This approach share the same disadvantage as the previous one above. Another disadvantage is the burden on the user generating a set of grids, not only one grid. The most promising approach is the agglomeration of fine grid volumes to coarse control volumes, see Smith (1990). In this way multigrid can naturally be applied. This can be done automatically resulting in large polyhedra. This method has been used in Eliasson (2001) and Eliasson (2002).

1.3. Grid generation - a survey

Grids can generally be said to be of two types, boundary conforming grids or non-conforming grids. Today the most common type of grid is the boundary conforming grid. A very coarse single block grid partly conforming is shown

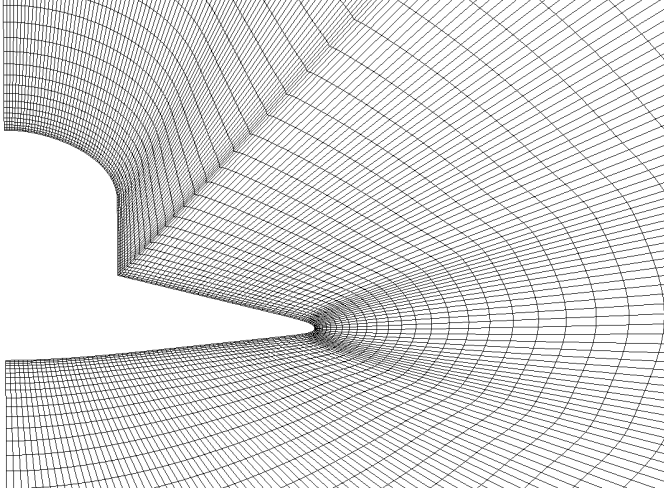


FIGURE 1.6. Two-dimensional cut of a structured grid generated by transfinite interpolation.

in Figure 1.1. The grid is conforming to the wing but not entirely to the fuselage-symmetry intersection.

Methods for boundary conforming grid generation can be divided into structured grid methods and unstructured grid methods. The first developed methods were those for structured grids, due to the simplicity both for the flow solver as well as for the grid generator.

Structured as well as unstructured grids may be used for all kind of continuum mechanic problems. The application of the grid generation in this work has been computational fluid dynamics. For this application the prismatic grid close to the boundary is of particular interest.

1.3.1. *Structured grids*

Different approaches for the generation of structured grids can be seen in the early review of grid generation methods by Thompson (1984). The first methods were only able to generate single block grids. A three-dimensional single block grid around a delta wing is shown in Figure 1.2. To make good grids for finite volume calculations about realistic three-dimensional geometries is a very difficult task. Therefore, the tendency in grid generation two decades ago was to develop a new grid generation program for each new type of application. But this was a very expensive way to generate grids. In the long run it would be better with grid generation programs of a more general nature. Some more general codes were developed by Thompson (1987), Holcomb (1987) and Woan (1987). A general code should be interactive so the user can build up the grid step by step, inspect it, and gradually improve it. The program should also be able to handle grids composed of a number of blocks,

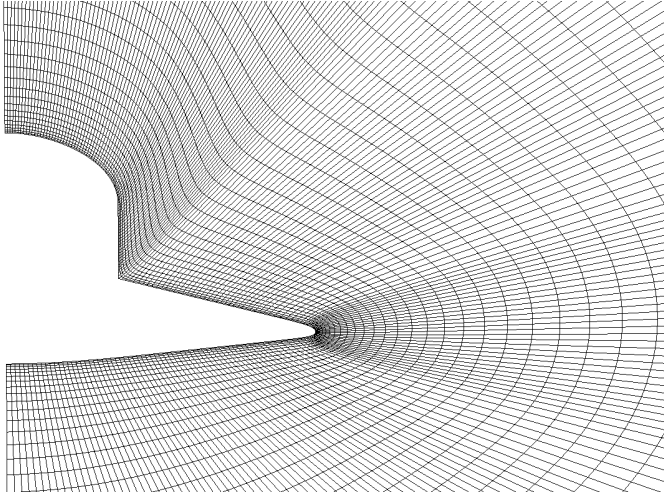
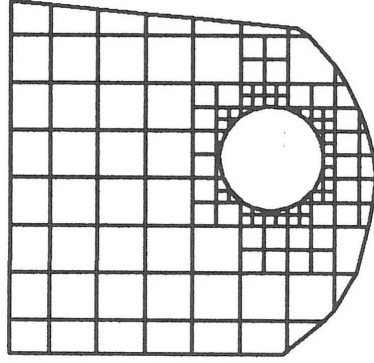


FIGURE 1.7. Two-dimensional cut of a structured grid smoothed by elliptic smoothing.

where some early references are Rubbert & Lee (1982), Shaw *et al.* (1986), Eriksson (1987), Sorensen & Mccann (1991) and Soni & Thompson (1992). A two-dimensional multiblock grid is shown in Figure 1.3 and the complex block structure of a three-dimensional multi block grid is shown in Figure 1.4. To write an interactive program for two-dimensional grid generation is relatively easy, since the user in such a case can treat all areas of the grid where difficulties arise one at a time. There is only one grid plane to inspect, and the user can run the program without too much work, even if such a program can do only very basic operations. On the other hand, to write a general purpose grid generation program for three-dimensional grid generation is a much more difficult task, since it is impossible for the user to inspect all grid planes and treat them one at a time. This implies that the operations for such a program must be performed on large regions of the grid each time they are used. Thus, the operations must be very robust, reliable and function within a range of geometry variations.

An alternative to the multiblock method for complex geometries is the so called Chimera method. An application for a very complex geometry is given in Pearce *et al.* (1993). A simple two-dimensional example is shown in Figure 1.5. In this method separate grids are generated around each component. This makes the grid generation much simpler. The disadvantage is the interpolation between different grids that must be done in the solver.

There are two dominating types of methods for structured grid generation: algebraic methods and elliptic methods. The dominating algebraic method is transfinite interpolation, see Eriksson (1982), Smith (1983), Eriksson (1985)

FIGURE 1.8. Quadtree grid, from Cheng *et al.* (1988).

and Thompson *et al.* (1985). Good references for elliptic methods are Thompson *et al.* (1985) and Sorenson (1986). A simple grid generated by transfinite interpolation is shown in Figure 1.6. Transfinite interpolation is much faster than elliptic methods. The main disadvantage with transfinite interpolation, compared to elliptic methods using a Laplace system, is that there is no inherent guarantee against grid inversion. This guarantee against grid inversion is based on the maximum principle for the Laplace equation. Unfortunately, the Laplace grid generation system is not so attractive, since it does not give the desired grid point distribution. Usually a Poisson system is used instead, see Thompson *et al.* (1985). Then the maximum principle is lost. Nevertheless, these elliptic methods are probably safer than transfinite interpolation. A grid generated by transfinite interpolation and smoothed by an elliptic algorithm is shown in Figure 1.7.

When using transfinite interpolation the grid points in the interior of a region are computed by interpolation from the grid points on the boundaries. This means, if the boundaries are non-smooth the grid will not be smooth either. Thus the grid must be postprocessed after the transfinite interpolation in order to smooth the grid and remove cross-overs. This is usually done by an optimization algorithm, see e.g. Carcailett (1986) and Jacquotte (1991), or by use of elliptic methods, as in Figure 1.7.

1.3.2. Unstructured grids

Generating a structured multiblock grid may be very time consuming. The time to generate a grid for a complex configuration with many blocks can take up to a month. Thus there is a need for some more automated methods.

Grid generation programs for unstructured grids are normally based on the quadtree/octree-method, see Shephard *et al.* (1988), the advancing front method, see Löhner & Parikh (1988) and Peraire *et al.* (1990), or Delaunay triangulation, see Baker (1988) and Weatherill *et al.* (1993). The principle of

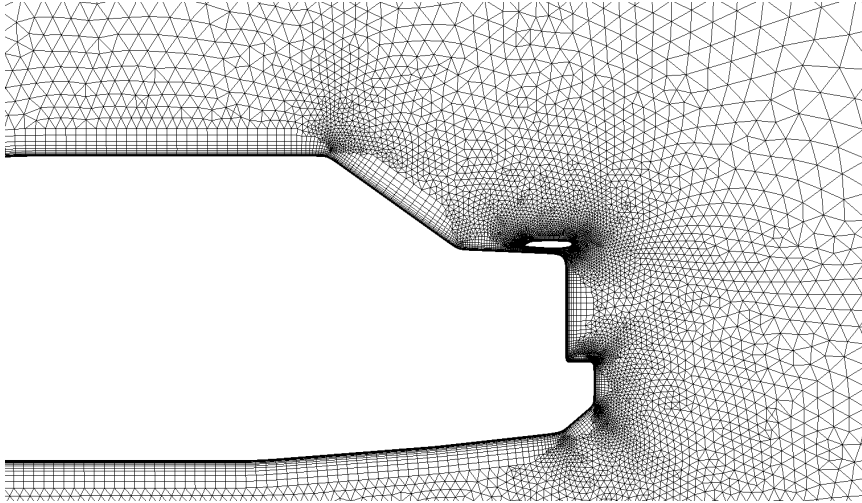


FIGURE 1.9. Two-dimensional hybrid grid for a car configuration.

the quadtree method is shown in Figure 1.8. A triangular grid can then easily be constructed by drawing the diagonals in each quad. A grid generated by the advancing front method is shown in Figure 1.15

The major advantage of using unstructured grids is that the grid generation process can be automated to a much higher extent than for multiblock grids. The grid generation time can be reduced from months to days. Another important advantage of using unstructured grids is the possibility to efficiently implement adaptive grid refinement. In this manner it is possible to improve the resolution of flow gradients during the flow computation. The disadvantage of using unstructured grids is that the computational time will be longer than for computations using a multiblock grid. The disadvantage of a slower flow solver is of much less importance than the advantage of a much more simple and user friendly grid generation procedure.

1.3.3. *Hybrid grids*

Structured hexahedral grids have successfully been used for aerodynamic computations for a long time, for inviscid as well as viscous computations. Unstructured tetrahedral grids have successfully been used for inviscid computations, but they are not well suited for viscous computations where there is a need for very stretched grid cells in the boundary layer. The solution is to use what is called a hybrid grid. That is, an unstructured tetrahedral grid in the inviscid part of the flow domain and a prismatic grid in the viscous part. A two-dimensional hybrid grid is shown in Figure 1.9 and a three-dimensional grid is shown in Figure 1.10. Another three-dimensional prismatic grid is shown in Figure 1.12 for the surface grid in Figure 1.11.

A prismatic grid can be said to be a compromise between a structured and an unstructured grid, having the best properties of both. A prismatic grid utilizes an unstructured surface grid, while it is structured in the direction normal to the body surface. Thus, the advantage of using an unstructured surface grid for complex three-dimensional configurations is retained. The main technique for generating prismatic grids is the advancing layer method, described in Pirzadeh (1996) and Kallinderis *et al.* (1996). Problem areas are concave regions and regions of narrow gaps.

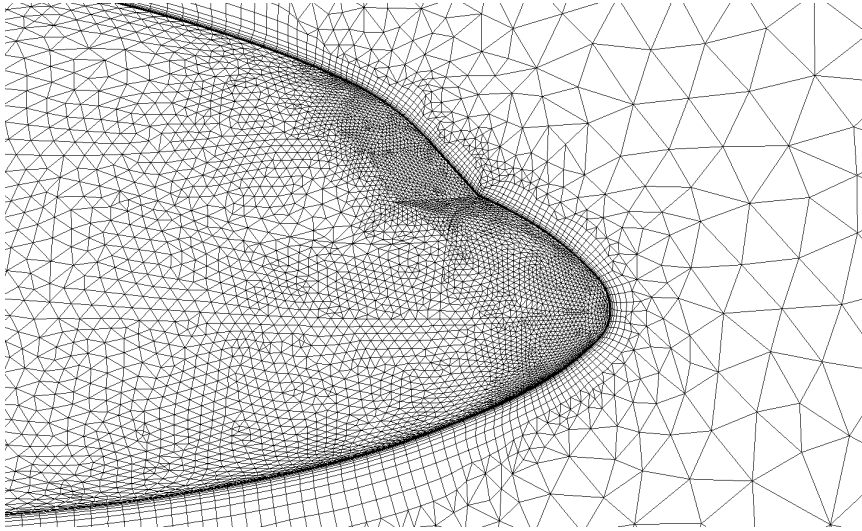


FIGURE 1.10. Hybrid grid around cockpit for an aircraft configuration.

1.3.4. Adaptive grids

For flows about complex geometries the locations of rapid variations in the flow are not known a priori. Generation of a fine grid throughout the entire flow field will give unacceptable execution times. To be able to solve this type of problem one has to use solution adaptive grid refinement. During the last decades an extensive number of papers have been written about adaptive computations.

Four different strategies can be used to generate adaptive grids. *Mesh node movement*: The number of grid nodes is constant, but the grid nodes are redistributed, see Catherall (1988). This method is natural to use for structured grids but it can give rise to very distorted grid cells. An example is shown in Figures 1.13. *Node insertion*: New nodes are inserted into the existing grid. see Weatherill *et al.* (1993). This method is very well suited for Delaunay grid generation. *Mesh enrichment*: Cells which are too large are divided into smaller ones, see Figure 1.14. A similar technique can also be used for unstructured

grids, see Löhner (1995). This method is fast and well suited for transient flows where the grid is updated often. A drawback is that stretched grid cells can not be generated easily. An example is shown in Figure 1.16. *Remeshing*: The entire grid is regenerated, see Peraire *et al.* (1990). This method is natural to implement into an advancing front grid generator. It is well suited for stationary problems, but not for transient flows, since it will take some time to regenerate the entire grid. It is also easy to generate grids with moderate stretching. An example is given in Figure 1.17

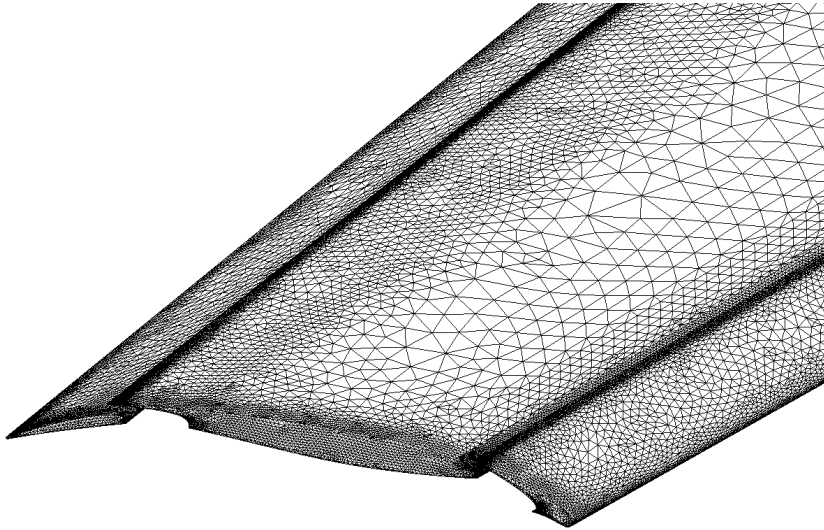


FIGURE 1.11. Surface grid for a high-lift configuration.

1.3.5. Hybrid grid generation systems

Generation of hybrid grids, consisting of prismatic layers close to the bodies and tetrahedral cells outside of these layers, is a much more automatic process than the generation of block structured grids. Nevertheless, much work must be spent to put all hybrid grid generation algorithms into a high quality grid generation system. Such a system must have a user friendly graphical interface, taking care of the entire process from import of the CAD file, through geometry clean-up and cell size setting to export of the final grid. There exist several such systems. An overview of the most well known hybrid grid generation systems is given below.

The two hybrid grid generation systems, CENTAUR, showing some applications in Khawaja & Kallinderis (2000) and VGRIDns, given in e.g. Pirzadeh (1999), with two different user interfaces described in Jones (2003) and Forrester & Bogue (2008) are similar to the grid generator described in this theses.

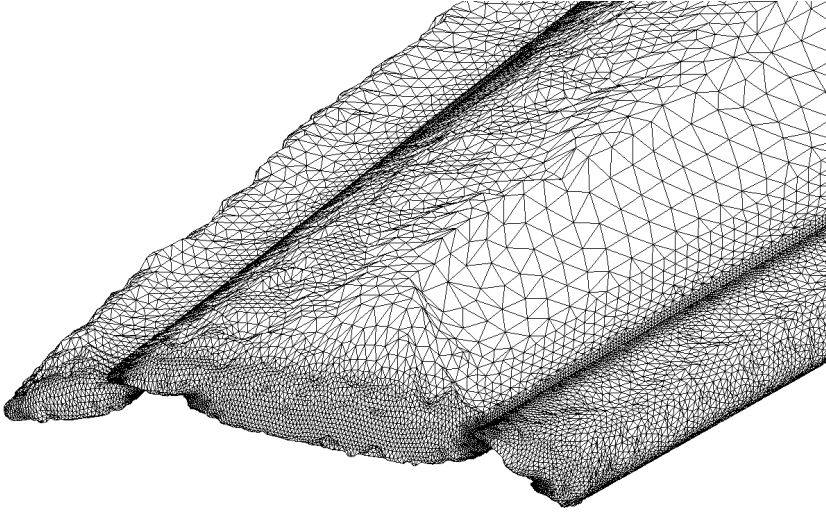


FIGURE 1.12. Outer layer of prismatic grid for a high-lift configuration.

SolidMesh, see Gaither *et al.* (2000), is another example of a similar hybrid grid generator, but where the tetrahedral part of the grid is generated by the Delaunay algorithm instead of the advancing front algorithm. A large set of grid generation routines has been developed at the University of Wales, see e.g. Weatherill *et al.* (1998) and Weatherill *et al.* (2001), which also use the Delaunay method. The graphic user interface for this grid generation system is called PSUE, which is described in Zheng *et al.* (2002). Gridgen, see e.g. Chawner *et al.* (2000) and Steinbrenner & Abelanet (2007), can be used for the generation of structured as well as hybrid grids. The program SOLAR, see Leatham *et al.* (2000) and Martineau *et al.* (2006), can generate hybrid grids with a mix of different cell types. The toolkit GGTK, see Soni *et al.* (2001), is a set of general modules which can be put together to grid generation programs for different purposes. The set of programs from Löhner, see e.g. Löhner (2000) and Aubry & Löhner (2007) defines an extensive set of grid generation methods and algorithms. The grid generator Gmsh described in Geuzaine & Remacle (2009) is a grid generator for unstructured grids. Finally, the software ICEM CFD from ANSYS has several methods to generate hybrid grids.

An overview of the TRITET hybrid grid generation system is given in Tysell (2007) together with some new extensions in Tysell (2009). All the different modules in this unstructured/hybrid grid generation system (in total around 85,000 active lines) are put together into one system by use of a C-shell script. The user access the system by an interactive user interface and can do all operations in the chain of steps from importing the geometry definition to exporting the final grid through the interactive user interface.

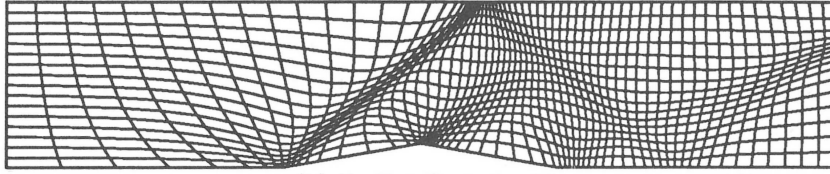


FIGURE 1.13. Adaption by node movement, from Dannenhoffer (1988).

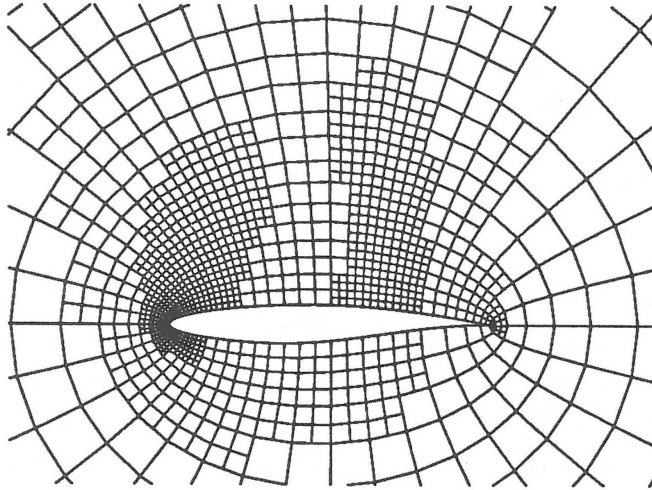


FIGURE 1.14. Adaption by node insertion, from Dannenhoffer (1988).

1.4. Grid quality and accuracy

A grid of good quality is a grid that for a certain number of grid cells, or nodes, gives a solution having a small error. Thus, the question of grid quality is coupled to the specific solver being used. What is a good grid for some solvers may not be this for other type of solvers. The question of grid quality can be divided into two separate parts. The first is the global distribution of the grid cells and the second is the local distribution and shape of the individual cells. The first part is handled by grid adaptation, which has already been discussed, and will also be covered in more detail later. The second part is about generating the grid with as smooth node distribution and well-shaped grid cells as possible. These two aspects are not completely uncoupled, since a grid that at a first glance may look very irregular may be well fitted to the solution. Thus, the shape of the grid cells should not be measured in the physical space, but in the transformed space given by the adaptation. In this

transformed space all grid cells should have equal size and all cells should be isotropic.

Many quality indicators for unstructured grids have been proposed in the literature, see e.g. Woodard *et al.* (1994), Weatherill *et al.* (1994), Lewis *et al.* (1996) and Kallinderis & Kwong (2001). However, the relationship between quality indicators and the overall solution accuracy is difficult to establish. Some of them only detect specific types of degenerated grid cells. Many indicators are also very similar. A common quality indicator for structured grids is orthogonality, but this can not be used for unstructured grids. Quality indicators for individual tetrahedra can only be computed during the grid generation process, since the transformed space is not known afterwards. In Tysell & Nordström (2007) local quality indicators, originally given in Tysell (2000:1), are proposed, where tetrahedra connected to each other are compared. These indicators are very similar to those given in Kallinderis & Kwong (2001). Connected tetrahedra should have about the same volume. If one tetrahedra have a non-optimal shape or volume it will differ from its neighbours. In Kallinderis & Kontzialis (2009) quality indicators for two-dimensional hybrid grids with a direct relation to the truncation error for the gradient are given.

The question if a grid cell is valid or not is dependent of the specific solver in use. In this work the volumetric coefficient for hexahedra, prisms and pyramids are computed by decomposing these grid cells into all possible combination of tetrahedra. If all combinations give a negative volume for some tetrahedron the grid cell is considered invalid. If only some combinations have a tetrahedron of negative volume the grid cell is considered skewed, but still valid.

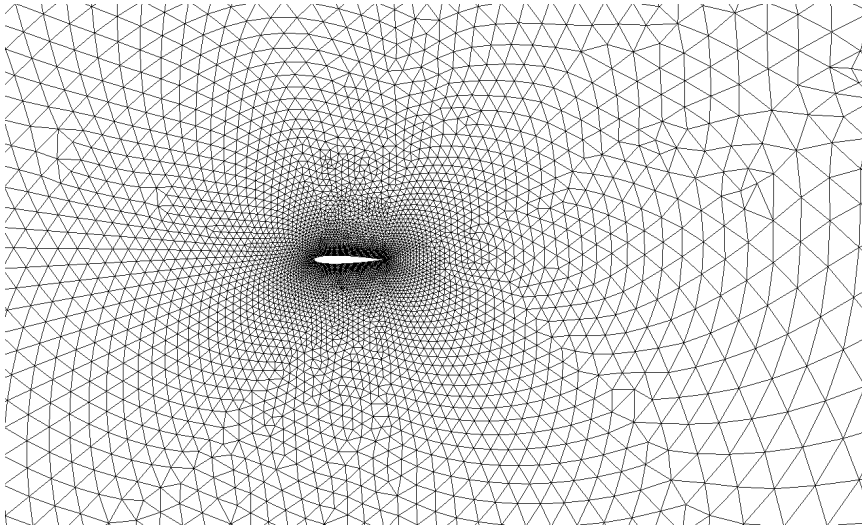


FIGURE 1.15. Initial grid for NACA 0012 airfoil.

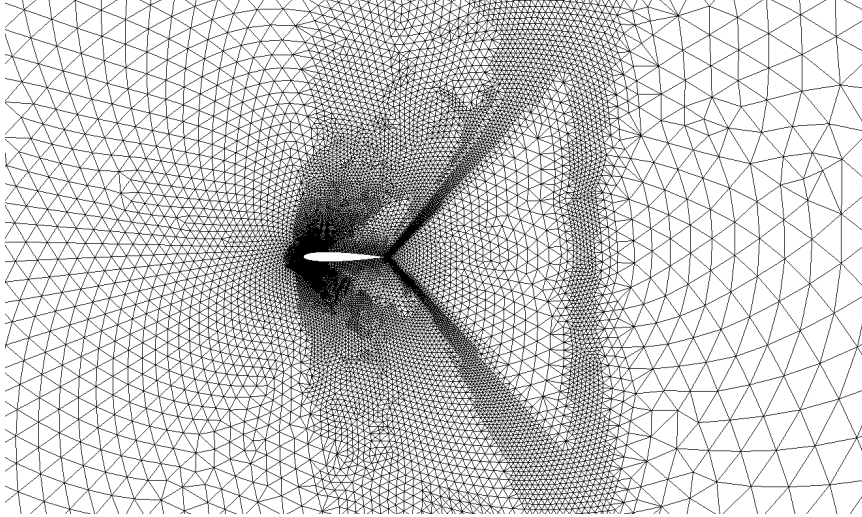


FIGURE 1.16. H-refined grid for NACA 0012 airfoil, $M_\infty = 0.95$, $\alpha = 0.0^\circ$.

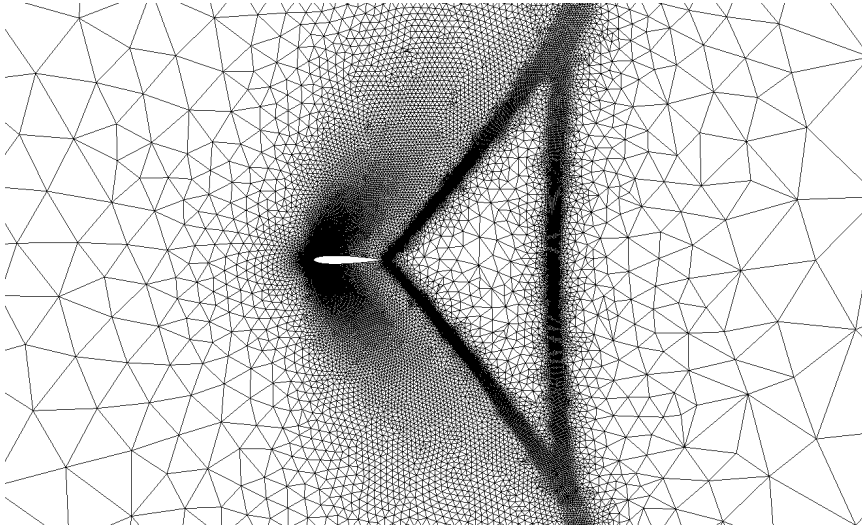


FIGURE 1.17. Adapted grid by remeshing for NACA 0012 airfoil, $M_\infty = 0.95$, $\alpha = 0.0^\circ$.

For stretched grid cells it is important to have the orientation aligned with the solution. In Luke *et al.* (2008) it is shown that for tetrahedra having a stretching of 1:10 the error is low for a misalignment below 10° , and over this the error increases successively. For tetrahedra having a stretching of 1:100 the error rises sharply only after a few degrees of misalignment. It is also

well known that it is important to have as low maximum angle as possible for triangular cells. In Tysell (2007) and Tysell (2008) stretched triangular surface grids aligned to the principal curvature directions are shown. The maximum angle is about 90° (not 180°) for most triangles, since special care has been taken in the advancing front algorithm to achieve this, but still there will be a lot of edges not completely aligned with the curvature, opposed to what is the case for a well fitted structured grid, where these "diagonal" edges are not present. This may affect the accuracy. Nevertheless, the computations show the stretching effects the result very little, see Tysell (2008). The dual grid can be constructed in a modified way for these stretched cells which improves the accuracy, see Mavripilis (2008). A better way is to allow a combination of quadrilateral and triangular surface cells, as in Leatham *et al.* (2000), which also will reduce the number of cells. The drawback is the complex data structure in the grid generator.

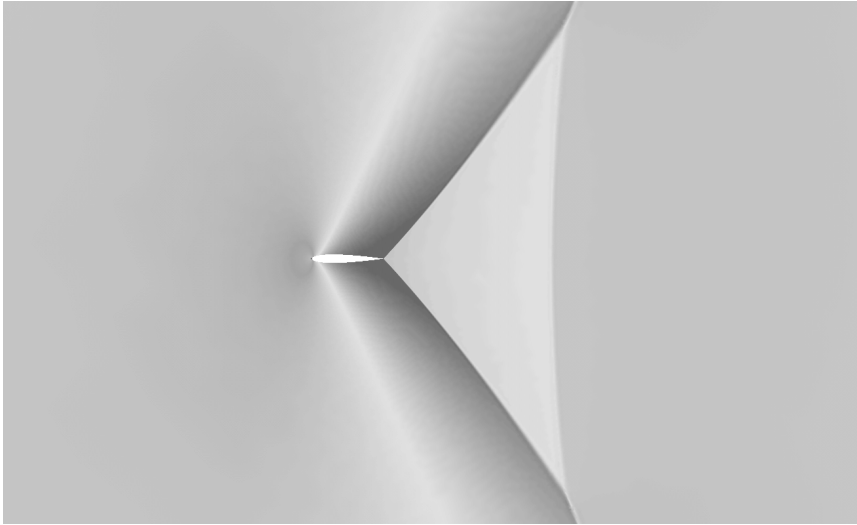


FIGURE 1.18. Adapted solution by remeshing showing Mach number, scale $[0.0 - 1.5]$, for NACA 0012 airfoil, $M_\infty = 0.95$, $\alpha = 0.0^\circ$.

For viscous flow there is a need for an extremely fine grid resolution normal to the surface in the viscous layer. This can be achieved by the use of prismatic cells. It is important to keep the marching direction for these prismatic layers as orthogonal to the surface as possible. Of course this restriction must be somewhat relaxed in convex and concave corners in order to get a smooth non overlapping grid, see Tysell (2000:2). It is also important not to have a too rapid expansion of the cell sizes from one layer to another. The expansion ratio should not be more than 1.2 for good accuracy. Some quality indicators for prismatic cells are given in Tysell & Nordström (2007). In order to get a

smooth transition between the outermost cells in the prismatic layers and the tetrahedral grid it is also better to have a variable number of prismatic layers, than just reducing the thickness of the prismatic layers, keeping a constant number of layers.

The question of accuracy has been studied in the AIAA Drag Prediction Workshop I-IV, see Mavripilis (2005), Eliasson & Peng (2007), Mavripilis *et al.* (2008) and Vassberg *et al.* (2008). Due to the increase in computer capacity the last decades it has become possible to do flow computations on more complex cases with an ever increasing number grid points. To compute viscous flow on grids with more than 10 million nodes is now a daily routine. For some special studies grids with more than 30-50 million nodes are being used. For grids with this resolution on a simple wing/fuselage configuration it should be possible to do a series of computations on grids of different resolution and extrapolate the result to infinite grid resolution. This has been done in the AIAA Drag Prediction Work Shop series. Surprisingly these studies have shown that many solvers do not show the predicted asymptotic behaviour, and there is a big discrepancy between different solvers. Even the same solver may show different results for slightly different type of grids, see Mavripilis *et al.* (2008), suggesting that the solutions are not grid converged and the grid resolution is not in the asymptotic region for these grids. The scatter in the results is about 5-10%. A conclusion that may be drawn from these studies is that the usual resolution of 10 million nodes used today for complex wing-body configurations does not give grid converged solutions.

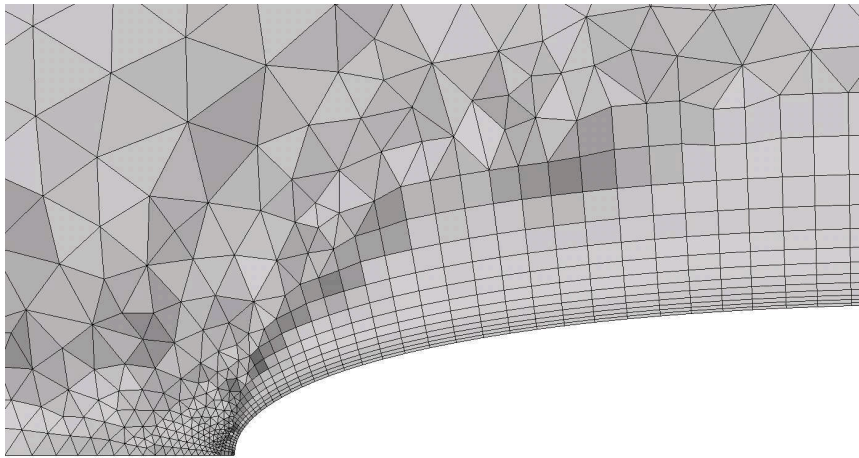


FIGURE 1.19. Change in total pressure within each cell for NACA 0012 airfoil. Dark colour indicates more change: regular grid.

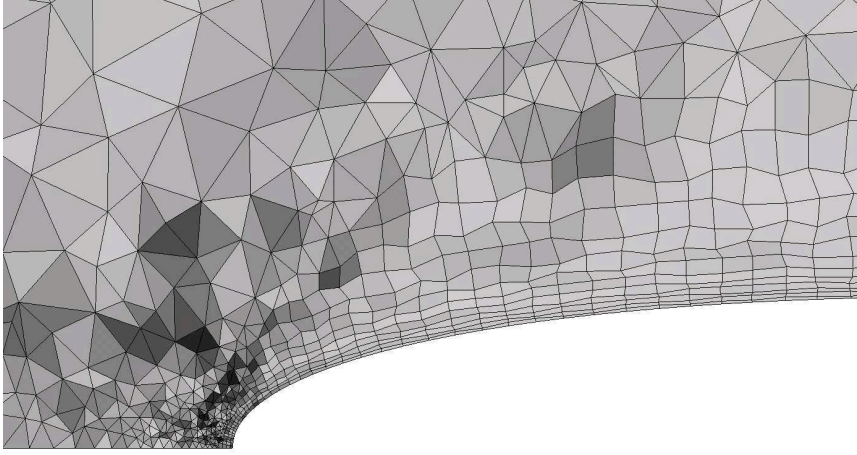


FIGURE 1.20. Change in total pressure within each cell for NACA 0012 airfoil: randomly disturbed grid with a distortion of 15%.

In Svärd *et al.* (2005) the accuracy questions were investigated for a number of model problems related to the Euler and Navier-Stokes equations for a node-centered edge-based finite volume method. This scheme should be second order accurate for very regular unstructured grids, but it is shown that for non-smooth grids the order of accuracy is lower. For two-dimensional hybrid grids with a non-smooth structured region or a non-smooth interface between the structured region and the unstructured region the order of accuracy may be much lower. In Tysell & Nordström (2007) the accuracy of the Euler equation has been studied. There it has been shown that the accuracy for real life configurations is lower than second order, especially for two-dimensional configurations. The interface between the prismatic region and the unstructured region seems to be smooth enough not reducing the accuracy globally. The order of accuracy is significantly reduced if the prismatic region is non-smooth for two-dimensional configurations, but seems not to be affected for three-dimensional configurations. This is shown by Figure 1.19 and Figure 1.20 which show the change in total pressure within each cell for the double NACA 0012 inviscid computations at $M_\infty = 0.3$ in Tysell & Nordström (2007). If there is no error the total pressure should be constant throughout the entire flow field. Here it can be seen that the error is higher at the interface between the structured region and the unstructured region at locations where the number of prismatic layers change, as predicted in Svärd *et al.* (2005). Figure 1.20 also shows that the error is increased for a non-smooth grid.

Computations in Eliasson *et al.* (2008) show that the accuracy is lower than second order also for viscous flow around a simple NACA 0012 airfoil for very fine grids. This is the same result as for the inviscid computations in Tysell & Nordström (2007). A study of the change in total pressure for the Euler

equations for the single NACA 0012 triangular grid computation at $M_\infty = 0.1$ in Tysell & Nordström (2007) shows that there is a significant error at the trailing edge, see Table 1.1. Here it can be seen that the error at the leading edge is reduced, and the accuracy order is 1.6 in average, but the error at the trailing edge is constant, when the grid size is reduced. This error is introduced by the lack of proper boundary condition at the sharp trailing edge. A closer look into the result shows in Figure 1.21 that the local error at the trailing edge introduce an error in the circulation around the airfoil. This error is very small but might cause an accuracy loss in studies using very fine grids.

$1/\Delta$	ΔP_{tot} LE %	ΔP_{tot} TE %	u_y/U_∞ TE %
0.0625	3.36	0.709	2.1478
0.125	1.80	1.045	-2.5365
0.25	8.58E-01	0.939	-0.0050
0.5	3.40E-01	1.042	-0.0205
1.0	4.02E-02	1.063	0.2581
2.0	5.54E-03	0.959	1.1696
4.0	1.55E-03	0.990	-0.5865
8.0	6.65E-04	0.886	0.8471
16.0	3.88E-04	0.888	0.5623

TABLE 1.1. Change in total pressure within each cell for NACA 0012 airfoil at $\alpha = 0.0^\circ$ at leading and trailing edges.

1.5. The grid generation environment

The structured grid generation algorithms described in Tysell & Hedman (1988) together form one of the first general purpose codes, where well known algorithms were put together to an interactive code. This grid generator has for many years been used for the generation of structured multiblock grids, see Figures 1.1, 1.2, 1.3 and 1.4. The grid can be generated by use of transfinite interpolation, see Figure 1.6, or elliptic methods, see Figure 1.7. Transfinite interpolation is a fast algebraic method, but for complex geometries it may give grid inversion. The grid generated by transfinite interpolation can be used as input to the slower elliptic grid generation method in order to improve grid quality. Modules for projection of grids on other grids, and computation of the intersection between grids have also been implemented.

It was decided not to import CAD files directly into the unstructured/hybrid grid generator, since the grid generator then has to handle a large number of different surface entities. The surface input to the grid generator is simply a higher order representation defined by a structured network of points and the trim curves for each surface patch. These surface patches has for several cases

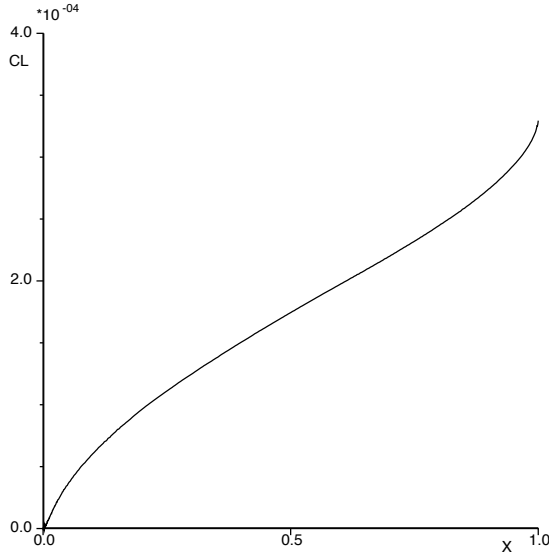


FIGURE 1.21. $\int_0^x (Cp_{upper}(x) - Cp_{lower}(x)) dx$ for NACA 0012 airfoil at $\alpha = 0.0^\circ$.

shown in the papers been created by importing the CAD file into a CAD interface program called Spider, developed by Petter Eneroth at FOI. The CAD file can now also be imported into the grid generator using the newly developed CAD interface described in Tysell (2009).

Today the structured grid generator described in Tysell & Hedman (1988) has been extended to be used as a geometry editor, like a simple CAD program for surface patches defined by bicubic splines through a structured network of points. This grid generator has now been integrated with the unstructured grid generation system for use as a geometry manipulation tool. It can do almost any geometry editing on geometry patches given by a structured network of points and a set of trim curves. Surfaces can be trimmed, exchanged, deleted, created or modified in almost any way. Figure 1.22 shows a surface patch structure for a wing-body combination inside a wind tunnel, and Figure 1.23 shows a surface patch description completely generated, from blue prints, within the structured grid generator.

The complete interactive system for grid generation in two and three dimensions is described in more detail in Tysell (2007). The input to the grid generator is given by using a graphical interface, also described there.

1.6. Major grid generation algorithms in use

To be able to generate grids for very complex configurations within one or a few days, the unstructured grid approach was chosen. The advancing front method

was selected, see Tysell (1994). This method can naturally handle anisotropic cells better than the Delaunay method. The octree method can not handle anisotropic cells at all. Another advantage with the advancing front method, compared to the Delaunay method and the octree method, is the easy way to handle the boundaries of the computational region. The disadvantage with the advancing front method is longer execution times, because of the heavy use of three-dimensional search algorithms. Important search algorithms are described in Löhner (1988) and Bonet & Peraire (1991). The unstructured grid generation algorithms described in this work includes some new features at the time the paper was written. The use of extra ideal points for the advancing front algorithm as well as the recognition of the typical cavity (the so called Schönhart prism) build up by eight faces are described. See Figure 1.15 for a simple two-dimensional grid and Figure 1.11 for a three-dimensional surface grid. The method to generate a background grid by use of a Delaunay method is also described in Tysell (1994) and further extended in Tysell (2009). An important feature in the advancing front algorithm is the possibility to generate stretched surface and volume grids. This significantly reduces the number of cells needed for an unstructured grid. This has previously been a major drawback compared to structured grids. Results of stretched grid generation and flow solution are presented in Tysell (2008).

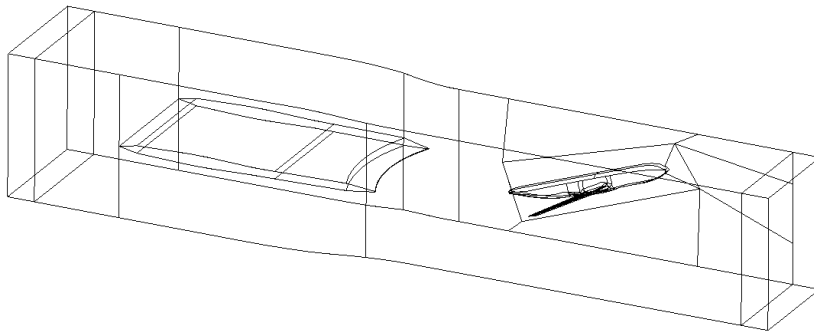


FIGURE 1.22. Structure of surface patches for a wing-body configuration in a wind tunnel.

A completely unstructured grid can not be used for viscous computations, since the boundary layer will not be resolved. The region close to the boundaries should be covered by a prismatic grid, having a very fine resolution in the direction normal to the boundary. A marching layer algorithm was selected to generate the prismatic grid. The implemented algorithm, see Tysell (2000:2), is tuned to give a prismatic grid of good quality. In this method also a new algorithm to compute the most visible normal vector to a surface is given. This algorithm is straightforward in opposite to the algorithms proposed in Kallinderis & Ward (1993) and Pirzadeh (1994), which most likely

do not give the same solution. An optimization algorithm used in order to give prismatic cells of good quality is also described. This also increases the distance the prismatic layers can advance before the process may break down due to grid inversion. The marching layer process can stop locally, depending of different criteria, see Tysell (2007). Figure 1.9 shows the hybrid grid around a two-dimensional car configuration, whereas Figure 1.10 shows the hybrid grid around a three-dimensional aircraft configuration. Figure 1.12 shows the outer layer of a prismatic grid around the tip of a three-dimensional wing, with the surface grid shown in Figure 1.11.

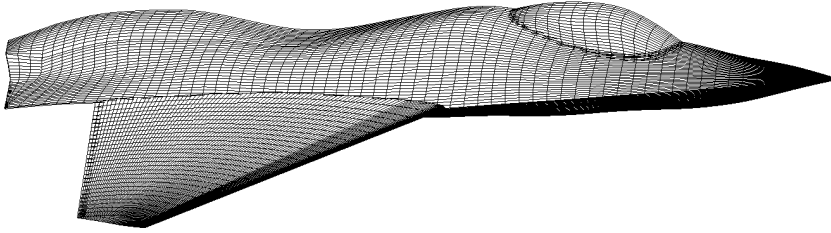


FIGURE 1.23. Surface patches for a fighter configuration.

The adaptive remeshing makes use of a gradient sensor. Several key-variables can be monitored simultaneously. Only the key-variables with the largest gradients will give a substantial contribution to the sensor, since the square of the gradients are added to form the sensor. The sensor gives directionally stretched grids, by computing a new background grid, which is used in order to completely regenerate a new grid, see Tysell *et al.* (1998). Strictly a gradient sensor should be used only for first order accurate flow solver schemes. This implies that the second order derivatives should be used for second order schemes, see Löhner (1995). Nevertheless, gradient sensors have been used with success even for formally second order accurate schemes. One explanation is that the estimation of the error from the second order derivatives inherently assumes that the solution is smooth and sufficiently close to the exact solution. This is not the case at e.g. shocks and if the grid resolution is too coarse. An adaptation procedure for a formally second order accurate scheme often starts with a very coarse grid where second order accuracy is not achieved. Furthermore, it has been shown by Delanaye & Essers (1997) that for grids of good quality the accuracy order might be only about 1.7, and for grids of lower quality as low as just above 1.1. The order of accuracy has also been studied in Tysell & Nordström (2007), where it is shown that for typical aeronautical problems the order of accuracy is less than second order. Figures 1.17 and 1.18 show the result of remeshing of the initial grid in Figure 1.15.

The disadvantage with adaptive remeshing is the computational time for the regeneration of the entire grid. The advantage with adaptive h-refinement

is the computational speed, whereas the disadvantage is that it is not possible to generate larger or anisotropic cells. For h-refinement the technique in Löhner (1995) has been implemented. The new nodes on the boundary are in the first stage located on the triangular surface grid. Thus, they are not located on the true surface. In a later stage the new boundary nodes are projected onto the surface by use of the higher order surface grid representation reported in Tysell (2005). The interior nodes are relocated to match the new location of the boundary nodes. This is done by use of the technique for moving grid, see Tysell (2002). The nodes in the prismatic layer are redistributed to match the flow gradients for the first cell. This is done by computing the y^+ value of each first node connected to the surface and then redistribute the nodes so the first cell will have the y^+ value specified by the user. Figure 1.16 shows the result of h-refinement of the initial grid in Figure 1.15.

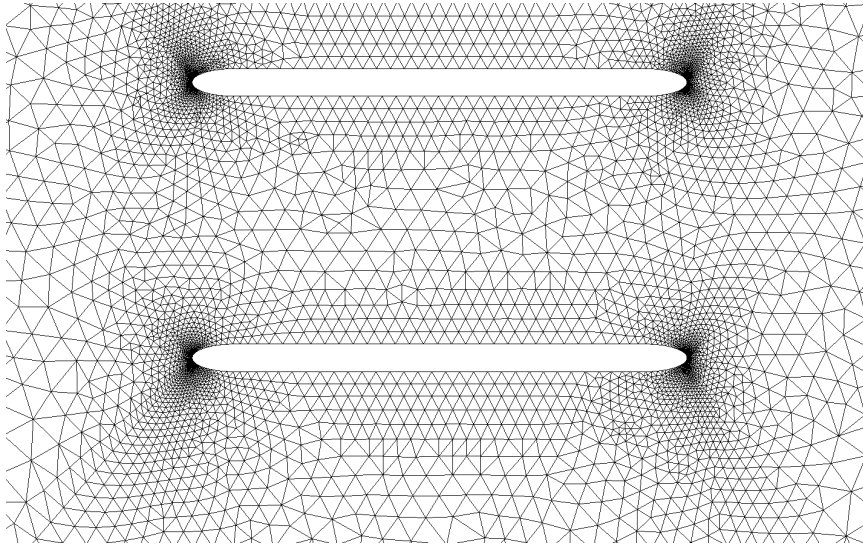


FIGURE 1.24. Original grid before deformation.

There are two common methods used for deforming unstructured grids. They are the spring analogy method, represented by the work in Batina (1991) and Singh *et al.* (1995) and the solution of the Laplace equation represented by Löhner & Yang (1996) and Baum *et al.* (1997). Both work sufficiently well for small deformations, but lack in quality for larger deformations, especially the spring analogy method. The algorithm developed in this work for the deformation of hybrid grids in two and three dimensions is based on the solution of the biharmonic equation for the deformation field, see Tysell (2002). The use of the biharmonic equation has also been reported by Helenbrook (2003). The

main advantage of this grid deformation algorithm is that it can handle large deformations, especially close to the boundaries. It also produces a smooth deformation distribution for cells which are very skewed or stretched. This is necessary in order to handle the very thin cells in a prismatic layer. The algorithm can handle skewed cells since the finite element method is used for the solution of the biharmonic equation. The disadvantage is that the method is slow, compared to algebraic methods, since a set of partial differential equations needs to be solved. Figure 1.25 shows the deformation of the initial twin airfoil grid in Figure 1.24.

The biharmonic surface grid projection algorithm used for h-refinement may also be used for the generation of the initial surface grid. This algorithm is better to handle surface patches with poor parameterization and internal surface discontinuities than bicubic splines. The algorithm has later been modified in Tysell (2009). The latest improvements are the use of more edge swapping in order to get a more regular mixed grid and also the setting of a fix position of some nodes close to or on the curves defining the surface patch. The initial position of the nodes in the mixed grid can be computed using a tensor-product patch definition. The algorithm is then used to adjust the position of the nodes in order to get a smooth surface grid.

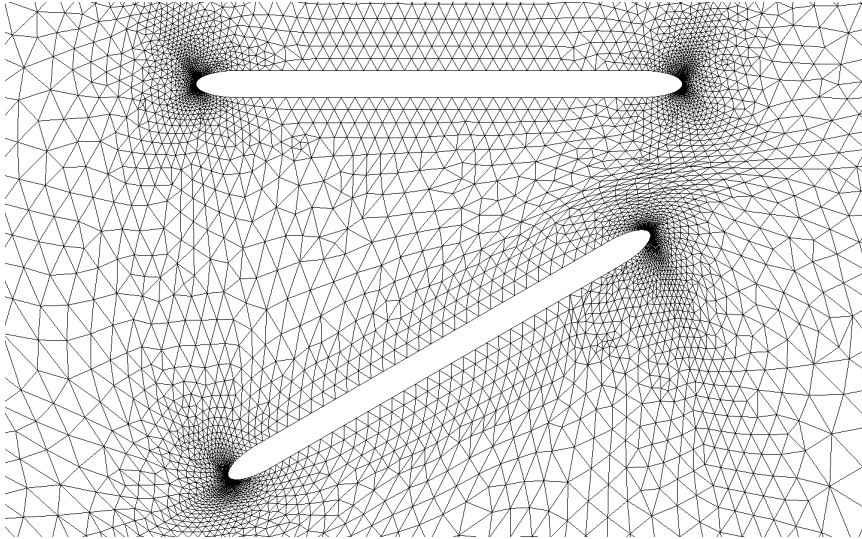


FIGURE 1.25. Grid after deformation.

Since the internal geometry format for the surface grid generation routines is the higher order representation through a structured network of points the structured multiblock grid generator described in Tysell & Hedman (1988) can

be used as a geometry editor. The inclusion of this interactive module in the hybrid grid generation system is described in Tysell (2007).

To summarize - the main aspects of grid generation introduced in the papers are:

- The use of a directional adaptivity sensor consisting of an arbitrary number of flow quantities in a consistent way.
- The algorithm to compute the most visible normal in the marching layer generation method.
- The use of an optimization method to smooth the layers in the marching layer generation method.
- The biharmonic grid deformation algorithm.
- The use of the biharmonic grid algorithm for higher order representation of triangular surface grids.

In addition, in the appendix are also included:

- The introduction of extra ideal points at the corners of the triangles in the front when using the advancing front method. These points are also essential when generating highly stretched grid cells.
- The generation of the background grid by a Delaunay algorithm.
- The observation of the special kind of cavity in the front formed by eight triangles ordered in a particular way often preventing closure of the front.
- The use of the multiblock grid generation module as an interactive geometry editor.

1.7. Applications of the grid generation algorithms

In Tysell & Hedman (1988) - Tysell (2005) the given grid generation algorithms have mainly been demonstrated for some simple test cases. The compilation of all grid generation modules into a grid generation system was reported in Tysell (2007). In Tysell (2008) the use of these algorithms on real complex geometries are shown together with usage of the edge-based finite volume flow solver for viscous flow computations described in Eliasson (2001) and Eliasson (2002). In that paper hybrid grid generation and flow computations for six different three-dimensional geometries are presented, showing the versatility of the grid generator. The built-in capability for flow visualization is also used. The most complex geometry studied is the wing-fuselage-slat-flap configuration in a wind tunnel. For this case several features of the grid generator are used, among them curvature stretched surface grid along the leading edges, flow adaptation and grid deformation. Another study shows that stretched cells can be used as a way to reduce the number of grid points with only minor changes of the result. A third case studied in that paper is the rear-fuselage and empennage configuration. This case supports the conclusion that finer grids than are usually used for production runs are needed for grid converged results. Several other applications have also been carried out. Figure 1.26 shows the

surface grid and the number of prismatic layers at the trailing edge of a high-lift configuration. This configuration is the same as the MLA configuration studied within the HiReTT project, see Rolston & Elsholz (2002). Despite the very thin gap between the wing and the aileron there is at minimum twenty prismatic layers in the grid. Figure 1.27 shows the surface grid and solution for a configuration studied in an aircraft accident investigation, see Winzell *et al.* (2004).

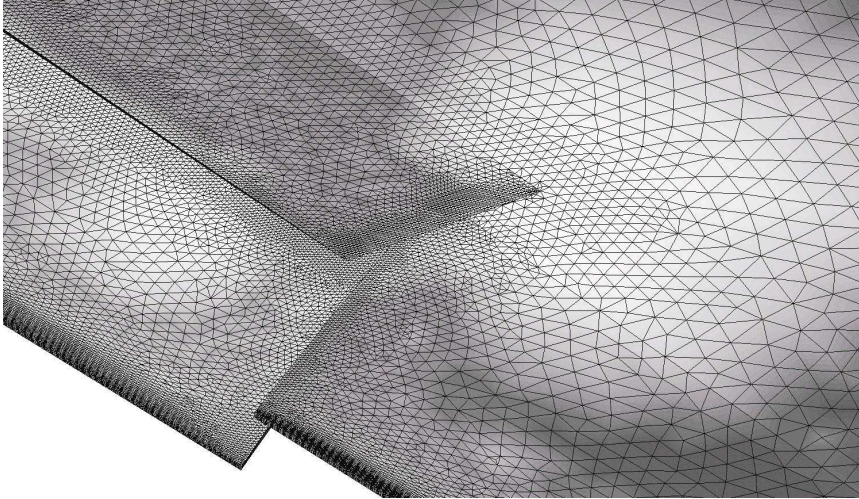


FIGURE 1.26. Close-up of surface grid for a wing-aileron-spoiler configuration. Minimum/maximum number of prismatic layers is 20/50. Dark colour represents more prismatic layers.

1.8. Recent advances in hybrid grid generation

There is a need for an ever increasing resolution of flow field computations, requiring grids with more grid points. Thus there is a need for faster grid generation algorithms. Fornasier *et al.* (2003) show a method where surface grid generation is done in parallel. Each processor generates the surface grids for a number of surface patches. Löhner (2000) reports on the use of a parallel advancing front algorithm for volume grid generation. The front is dynamically divided into boxes by use of the octree algorithm. For each processor the grid is generated for a number of boxes. The number of boxes is much larger than the number of processors. Another alternative is to first generate a surface grid in parallel and then divide the surface grid and volume into smaller regions, as in Larwood *et al.* (2003). The volume grid in these regions is generated in parallel. Parallel grid generation by the Delaunay algorithm has been presented in

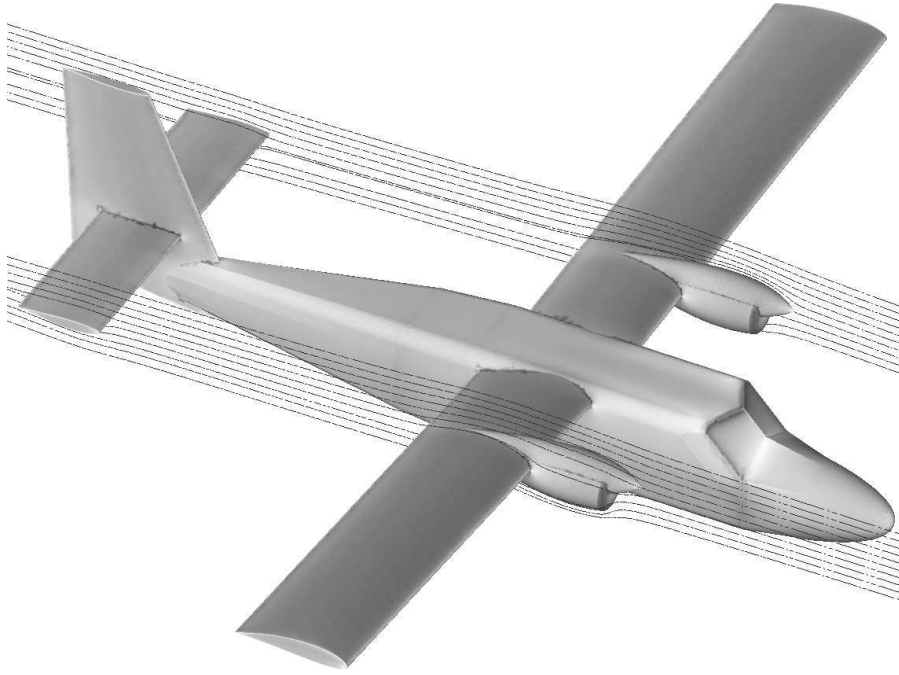


FIGURE 1.27. Streamlines and y^+ for first prismatic layer for a high mounted wing configuration at $M_\infty = 0.4$, $\alpha = 0.0^\circ$. Dark colour represents higher y^+ value. Average y^+ is < 2.0

Weatherill *et al.* (2001). Ito *et al.* (2008) describe a method where it is possible to introduce new components into an existing grid without regenerating the entire grid. The grid needs only to be regenerated in the region where the new component has been introduced. This speed up the time for the grid generation significantly. A similar extension of the grid generator TRITET with routines for the generation of locally remeshed grids has been done. This local remeshing works also for hybrid grids, but the original prismatic layers must be kept. Figure 1.28 shows the cavity in a grid where the grid around a rotated airfoil is inserted into a grid around another airfoil. The cavity is formed by a set of inner and outer boxes. Inside the cavity a grid is generated by the advancing front algorithm. The merged grid is shown in Figure 1.29. An application of this technique together with the flow solver Edge, see Eliasson (2002), for store separation computations are reported in Berglind (2009) and Berglind *et al.* (2009).

The use of stretched grids in the advancing front method, which was reported in Tysell (2008), has also been reported in Ghidoni *et al.* (2006). The methods are similar but the specification of the background grid differs from the smoothing technique used in Ghidoni *et al.* (2006), since the algorithm used

here only modify the background grid cell size for a node if it is larger than allowed by the prescribed expansion from the nodes in the background grid connected to the node. The technique is similar to the one used for isotrop cell sizes reported in Kania & Pirzadeh (2005), where surface curvature dependent grids also are generated.

A modification of the advancing front algorithm for surface grid generation, in which the algorithm prefers a newly generated edge as base face for the generation of the next triangle, has been given in Fornasier *et al.* (2003). This gives a front with less cavities, which will reduce the time spent for searching, since there will be less edges in the front.

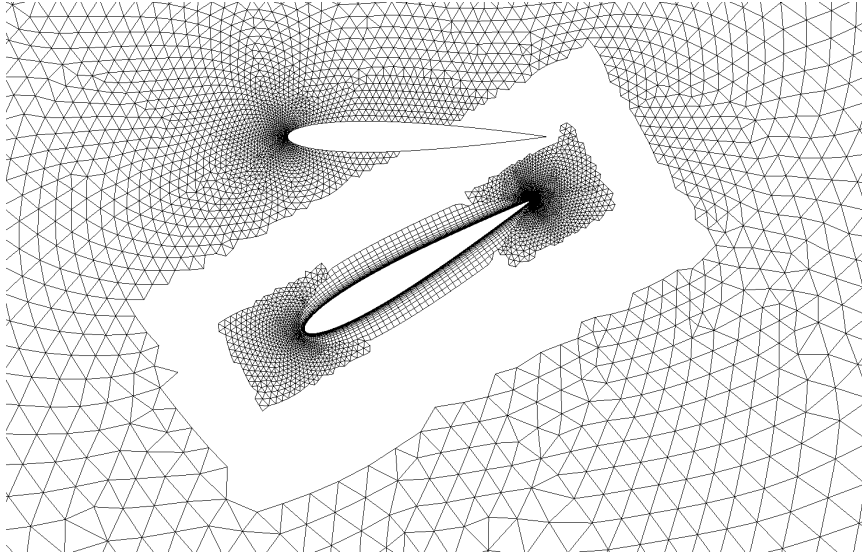


FIGURE 1.28. Local remeshing - mix of two grids. Cavity formed by a set of boxes.

A way to improve the initial background grid by introducing extra background grid nodes using background boxes is given in Tysell (2009). This plays somewhat the same role as the concept of bounding boxes and volume sources introduced in Pirzadeh (2008). A method to achieve an even better background grid by first generating a coarse volume grid and then interpolating the surface cell size specification into this grid and finally using this grid as a new background grid is given in Tysell (2009). This is an alternative to the octree background grid generation method given in McMorris & Kallinderis (1997).

In Aubry & Löhner (2008) the same prismatic grid generation algorithm as was given in Tysell (2000:2) to compute the most visible normal vector to a surface is described. A problem in prismatic grid generation is the quality of the grid in corners of the surface. Sharow *et al.* (2001) suggest this can be solved by generating extremely fine surface grids at corners, having a cell size

of about the height of the first cells in the prismatic layer. A drawback of this approach is of course that the number of cells will increase considerably. Soni *et al.* (2001) use the concept of a semistructured topology in the near surface regions, generated by a parabolic grid generation algorithm. In this concept corners can be handled by excluding or introducing nodes from one layer to the next layer. Khawaja *et al.* (1999) introduce the concept of varying number of prismatic layers, where different surface nodes can have different number of layers. This concept has also been used in Tysell (2007). For some surface nodes there may not be possible to define a visible normal vector. Thus a prismatic layer can not be generated at these nodes. A remedy is to use multiple normal vectors at these nodes. This has been shown in Ito *et al.* (2006) and Aubry & Löhner (2007). The use of multiple normals also improve the grid quality at sharp convex corners like wing trailing edges. In Steinbrenner & Abelanet (2007) difficult regions, especially concave regions, is handled by collapsing cells. Thus, the number of nodes in one layer may be less than in the previous layer. This kind of technique can only be used for layers consisting of stretched tetrahedra instead of prisms.

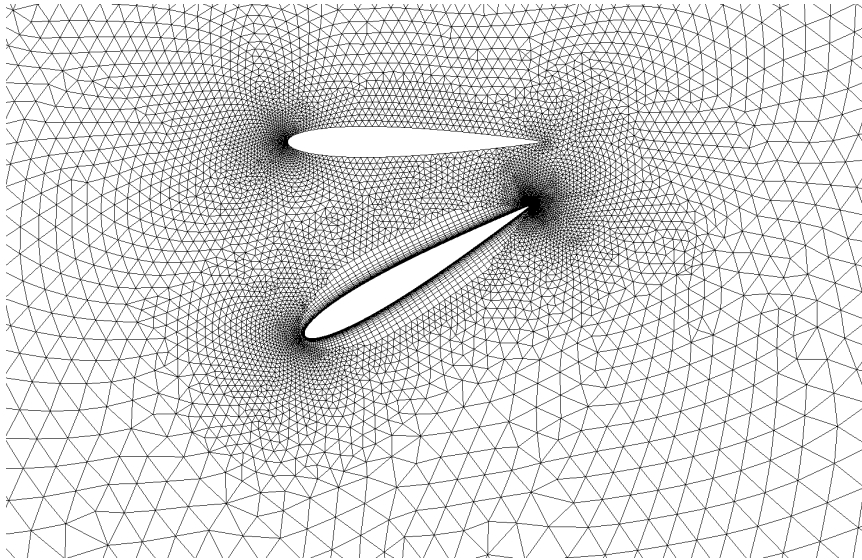


FIGURE 1.29. Local remeshing - merged inner and outer grid.

In all the methods above the stretched cells close to the boundary have been generated first and the isotrop tetrahedra have been generated in a second step. Ito & Nakahashi (2004) and Karman (2007) use a different strategy where a grid consisting of isotrop tetrahedra has been generated first. In a second step this grid is pushed away from the boundary and the gap is filled with prismatic cells. A drawback with this method is that it will likely be a jump in cell size

at the interface between the prismatic layer and the isotrop tetrahedra, since it is difficult to push the initial tetrahedra far enough without inversion to get a sufficient height of the prismatic layer. Löhner & Cebal (2000) present a method where an isotropic tetrahedral grid is generated first and then refined with stretched tetrahedra close to the boundary. The use of prismatic layers along imaginary surfaces in the flow field in order to catch shocks is presented in Shih *et al.* (2007).

The spring analogy method in Batina (1991) for grid deformation easily gives grid inversion for larger deformations. This has been improved by Farhat *et al.* (1998) by the introduction of torsion springs in addition to the tensions springs. In Acikgoz & Bottasso (2006) springs has been introduced also between nodes and edges and not only between nodes as in the previous methods. This greatly improves the capability to prevent the generation of inverted cells. Still there can be cells of bad quality, especially close to the boundaries, since only translation of the boundary has been taken into account. In Martineau & Georgala (2004) the rotation has been taken into account by rigid movement of the nodes close to the boundary and use of the spring analogy away from the boundary. In Samareh (2002) the use of quaternions has been introduced. Quaternions is an extension of complex analysis to three-dimensional space, which are ideal for modeling rotations. Since the method in Tysell (2002) solves a 4:th order partial differential equation it is possible to have two boundary conditions on the boundary. In the original paper the derivative of the displacement was set to zero on the boundary. Later an option has been introduced to take the local rotation around each boundary node into account, see Figure 1.25 and compare with the result without local rotation in Tysell (2007). Another modification of the algorithm is that in the discrete minimization problem the volume of the tetrahedra is retained when solving the biharmonic equation, otherwise the movement will be too rigid for the small tetrahedra close to the boundaries. If the deformation is governed by the Laplace equation only the original formulation is kept.

In Nielsen & Anderson (2001) a structural mechanic analogy is used by solving the equation for isotropic linear elasticity. A similar approach is used in Sheta *et al.* (2006) where the structural Navier equation is applied. Both methods show good results, but they probably suffer by the fact that they both solve a 2:nd order partial equation, thus only one boundary condition can be applied. In Liu *et al.* (2006) a fast algebraic method is presented based on the deformation of a temporary grid generated from the boundary nodes only by a Delaunay algorithm. The deformation of the grid is computed by interpolation in the temporary Delaunay grid. The deformation of the boundary must be done in such small steps that the Delaunay grid is not inverted. After each step the Delaunay grid may be regenerated. For large three-dimensional grids this may take a considerable time, and the method does not account for surface rotation. Another new algebraic method is the use of radial basis functions, see Jakobsson & Amignon (2005) and Allen & Rendall (2007).

In Peraire *et al.* (1992) the use of the Hessian, which measures the second derivatives of the flow quantities, is introduced in order to generate directionally adapted grids. One drawback with the use of the Hessian is that the grid can only be adapted to one selected flow quantity. It is in most cases not possible to find one single flow quantity that works for all cases, or even for all regions of one case. One remedy to this has been presented in Castro-Diaz *et al.* (1997) by computing the Hessian for several flow quantities and then compute the combined Hessian by so called metric intersection. This way to compute the metric intersection is not optimal, since only the combined eigenvalues are computed, while the eigenvectors are arbitrarily chosen from one of the flow quantities. Thus, the way to compute the combined metric from several flow quantities presented in Tysell *et al.* (1998) is done in a more rigorous way, but the drawback may be that only the first derivatives of the flow quantities are used instead of the second derivatives. Another way to compute the metric intersection has been presented in Frey & Alauzet (2005). The Hessians can be represented by a set of ellipsoids and in that paper the intersection is computed by computing the largest ellipsoid inscribed in all intersected ellipsoids. A rigorous way to compute this metric intersection has recently been presented in McKenzie *et al.* (2009), where each Hessian is introduced successively by transforming the Hessian to a space where the current transformation is represented by a sphere. In this space the intersection is easy to compute.

Both for first and second derivative adaptive sensors limits of the cell sizes must be set in regions of flow discontinuities, where the cell size otherwise would become indefinitely small, and in regions where the flow is varying very slowly, where the cell sizes would become too large. One drawback by using the Hessian is also that the cell sizes tends to be indefinitely large where the flow is varying linearly. It has been shown in Venditti & Darmofal (2003), where they compare results for flow around a multiple airfoil configuration, that a Hessian based method gives too large cell sizes in regions of linearly varying flow compared to an adjoint based adaptation method. In the paper they propose a combination of the two methods, where the cell sizes are taken from the adjoint computation, whereas the directional stretching are taken from the Hessian. In Remaki *et al.* (2006) the metric tensor is computed by taking a weighted sum of the Hessian and the gradient tensor, in order to get better grid resolution also in areas of linearly varying flow quantities.

The use of the Hessian and combinations of grid cell split, merge, swapping and node movement is used in Xia *et al.* (2001) and Dompierre *et al.* (2002) for directional h-adaptation. In these papers the method has been applied in two dimensions. The same method has been used to present three-dimensional results in Park & Darmofal (2008). This method is faster than the total remeshing used in Peraire *et al.* (1992) and Tysell *et al.* (1998) but appears to give grids of less good quality. In Pirzadeh (2000) adaption by remeshing is done by only doing local remeshing, where the grid needs to be adapted. In this way the time for remeshing is reduced. In Xia *et al.* (2001) the edge length

proposed by the sensor is optimized for all grid points around each grid point in the transformed space defined by the metric tensor in order to get the optimal grid.

A problem in grid adaptation is that almost all grid points usually tend to be placed in regions of strong phenomena and very few points are left for regions of weak phenomena. A way to overcome this is to estimate the local relative error instead of the absolute error. This has been done in Frey & Alauzet (2005).

In Dompierre *et al.* (2002) grid adaptation has been done in several steps until the process is converged, i.e. the cell sizes and stretching ratios do not change anymore. A finite volume as well as a finite element flow solver were used. For studies on some two-dimensional test cases they could show that on properly adapted grids the two solvers gave the same solution. They also draw the provocative conclusion that much of the advantage of some solver scheme over another are due to not properly adapted grids.

In Tchon *et al.* (2005) a novel method to generate two-dimensional adapted grids by use of the Hessian has been reported. The tensor lines generated by the Hessian are connected to a quad dominated grid.

For all the adaptive methods described above the metric tensor given in Tysell *et al.* (1998) may be used instead of the Hessian.

1.9. Summary and conclusion

The advancing front algorithm, which is able to generate anisotrop cells, has been used for the generation of unstructured grids. A marching layer algorithm has been developed for the generation of prismatic grids. Together these two algorithms form the basis for the generation of hybrid grids. Adaptation is made by remeshing or h-refinement. The adaptive remeshing algorithm is able to do directional adaptation as well as larger cells than in the original grid. The h-refinement algorithm is faster but lacks in this respect. A biharmonic grid deformation algorithm has been developed. A modification of this biharmonic method has also been used for surface grid generation, especially together with the h-refinement algorithm. The total set of algorithms is able to generate tetrahedra and prismatic cells in three dimensions and triangles and quads in two dimensions. The grid quality and the sensitivity to irregular grids has been studied for a node-centered edge-based finite volume solver. This study shows the accuracy is often lower than the theoretically estimated value, especially at surface slope discontinuities, like wing trailing edges.

The emphasis in the development of the algorithms has been on grid quality and reliability, thus sacrificing speed. The advancing front method is able to generate smooth stretch grids. This is in opposite to the octree methods which can not generate stretched grids. Delaunay generated grids also suffers in this respect. On the other hand booth these methods are faster than the advancing front method. The implemented marching layer algorithm is reliable and fast. The biharmonic grid deformation is more reliable but much slower

than algebraic methods, since a set of partial differential equations must be solved.

The included methods produce grids of good quality for real complex configurations, which has been shown by the application to numerous configurations during several years.

1.10. Outlook

The number of grid points used in typical studies has increased dramatically during the last decades due to increase in computer capacity. The complexity of the studied configurations has also increased significantly. These trends can be expected to continue. The interest in aerodynamic applications is also going in the direction of computations of drag and moments accurately, not only for complete configurations but also the contributions from separate parts. Thus, this may renew the interest in grid generation in order to generate high quality grids. Since there is a complex interdependence between grid quality and solution accuracy, and the CFD solutions can only be as good as the underlying grid, this may require increased collaboration between grid generation and flow solver development communities. Furthermore there is a trend to unsteady computations, where the geometry is changing during the computation. This requires a closer coupling between the grid generator and the flow solver, which may be difficult to achieve with commercial software. There is also a need for further parallelization of grid generation routines in order to facilitate the generation of very large grids. The most recent trend is the use of higher order methods for flow computations. These methods do not only require the location of the grid nodes at the boundaries, but also the surface derivatives.

CHAPTER 2

Summary of papers and authors contributions

Paper 1: Adaptive computations have been done both for two- and three-dimensional cases with use of a gradient sensor. Several key-variables can be monitored simultaneously. The sensor gives directionally stretched grids. The grid generation is based on the advancing front method, whereas the adaptation strategy used is remeshing, where the entire grid is regenerated. The flow solver is based on finite volume discretization. Computations are shown for a NACA 0012 airfoil, the Aerospatiale AS28G-wing and the JAS-39 Gripen fighter.

The development of the adaptation sensor and grid adaptation program, as well as the generation of the grids and writing of the paper were done by Tysell. The development of the flow solver was done by Berglind. The development of the tool for translating CAD-files, and even the translation of the Gripen fighter configuration were done by Eneroth. The computations were done by Berglind and Tysell in cooperation.

TYSELL, L., BERGLIND, T. & ENEROTH, P. 1998, Adaptive Grid Generation for 3D Unstructured Grids. Proceedings of the *6th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 391-400, Greenwich, UK. International Society of Grid Generation (ISGG).

Paper 2: A marching layer prismatic grid generation algorithm is described. The algorithm is able to handle regions of narrow gaps, as well as concave regions. The body surface is described by a triangular unstructured surface grid. The subsequent grid layers in the prismatic grid are marched away from the body by an algebraic procedure combined with an optimization procedure, resulting in a semi-structured grid of prismatic cells. The algorithm is demonstrated for a high-lift configuration and a complex wing-body-pylon-nacelle configuration.

TYSELL, L. 2000, Hybrid Grid Generation for Complex 3D Geometries. Proceedings of the *7th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 337-346, Whistler, British Columbia, Canada. International Society of Grid Generation (ISGG),

Paper 3: An algorithm is described for the deformation of hybrid grids in two and three dimensions, based on the solution of the biharmonic equation for the deformation field. The main advantage of the grid deformation algorithm is that it can handle large deformations. It also produces a smooth deformation

distribution for cells which are very skewed or stretched. This is necessary in order to handle the very thin cells in a prismatic layer.

TYSELL, L. 2002, Grid Deformation of 3D Hybrid Grids. Proceedings of the *8th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 265-274, Honolulu, Hawaii, USA. International Society of Grid Generation (ISGG).

Paper 4: An algorithm is described for the generation of unstructured surface grids using a higher order representation of an existing unstructured representation. This algorithm can be used both for the initial generation of the surface grid and the projection of new surface points generated by an h-refinement method.

TYSELL, L. 2005, A Higher Order Representation of Unstructured Surface Grids. Proceedings of the *9th International Conference on Numerical Grid Generation in Computational Field Simulations*, San Jose, California, USA. International Society of Grid Generation (ISGG).

Paper 5: The accuracy of the unstructured node-centered edge-based finite volume method is evaluated for inviscid flow around typical aerodynamic configurations. Both two-dimensional as well as three-dimensional configurations have been studied. The results show the order of accuracy is lower than the assumed order of two.

The selection of geometries, grid topologies, as well as the grid generation and flow computations were done by Tysell. The analysis of the results was done by Tysell in cooperation with Nordström. The paper was mainly written by Tysell, while Nordström was responsible for the chapter about the model equation.

TYSELL, L. & NORDSTRÖM, J. 2007, Accuracy Evaluation of the Unstructured Node-Centered Finite Volume Method in Aerodynamic Computations. Proceedings of the *10th ISGG Conference on Numerical Grid Generation*, Heraklion, Crete, Greece. International Society of Grid Generation (ISGG),

Paper 6: The grid generation algorithms described in the previous papers for the generation of unstructured/hybrid grids have been used for the generation of two-dimensional and three-dimensional grids. An edge based finite volume solver has been used for the flow computations. In this paper the results for several practical studies on complex geometries are summarized.

TYSELL, L. 2008, Experiences of Grid Generation and Steady/Unsteady Viscous Computations for Complex Geometries. Paper ICAS-2008-2.5.2. Proceedings of the *26th ICAS Congress*, Anchorage, Alaska, USA.

Papers in Appendix

Paper 1: The first version of a general purpose three-dimensional grid generation system is described. The system is interactive and user-friendly. It can

be applied to grids in one to three dimensions. The grid can be composed of a number of structured blocks, both patched and overlaid grids, where each block can have its own topology. The patched grid blocks can be either continuous or discontinuous at the block interfaces. The grid is generated by means of transfinite interpolation. A procedure for smoothing of the normal vectors to a surface is presented, as well as a procedure for smoothing of grids. Grid generation routines developed for special applications can be added to the system. It can also be used to postprocess grids from, or preprocess grids to, other grid generation programs. An example with a grid around a wing-body configuration is given.

The development of the grid generation program, generation of the grids and writing of the paper were done by Tysell. The development of the special routine for wing-body configurations was done by Hedman.

TYSELL, L. & HEDMAN, S. 1988, Towards a General Three-Dimensional Grid Generation System. Paper ICAS-88-4.7.4, pp. 1048-1058. Proceedings of the *16th ICAS Congress*, Jerusalem, Israel.

Paper 2: A grid generation system for the generation of three-dimensional unstructured grids around complex geometries is described. The grid generator is based on the advancing front algorithm. Tetrahedra of variable size, as well as directionally stretched tetrahedra can be generated by specification of a proper background grid. Efficient data structures have been implemented. The geometry is defined by a set of surface patches. Each patch is represented by a structured network of points. The surface patch connectivity is computed by the grid generator. The surface triangle grid and volume tetrahedra grid are automatically generated. The grid generator can also be used for the generation of two-dimensional grids.

TYSELL, L. 1994, An Advancing Front Grid Generation System for 3D Unstructured Grids. Paper ICAS-94-2.5.1, pp. 1552-1564. Proceedings of the *19th ICAS Congress*, Anaheim, California, USA.

Paper 3: The grid generator TRITET for generation of unstructured/hybrid grids in two and three dimensions is described. The main algorithm for the grid generation is the advancing front algorithm. This paper is concentrating on user aspects and some miscellaneous algorithms which have not been presented before. Most of the algorithms developed during the years are also summarized here.

TYSELL, L. 2007, The TRITET Grid Generation System. Proceedings of the *10th ISGG Conference on Numerical Grid Generation*, Heraklion, Crete, Greece. International Society of Grid Generation (ISGG).

Paper 4: The use of a CAD geometry import interface to the hybrid grid generator is demonstrated. The interface reads geometry files in a format used by one major commercial grid generation system. In this format all surfaces

are defined by NURBS. This makes it easy to convert the surface patches to the structured network of points used as input to the grid generator. The interface program is explained and an example is given of the use of it. New features related to the surface grid generation algorithm is also explained.

TYSELL, L. 2009, CAD Geometry Import for Grid Generation. Proceedings of the *11th ISGG Conference on Numerical Grid Generation*, Montreal Canada. International Society of Grid Generation (ISGG).

The papers are re-set to the present thesis format.

Acknowledgements

I would like to thank all the people who have encouraged me to finally write this thesis, especially my supervisor Professor Laszlo Fuchs. I also want to thank Petter Eneroth and Staffan Meijer, both formerly at FOI, for transferring CAD-files to the format used by the grid generator and for helping me with the OpenGL graphics programming, respectively.

The supports through all the years making this work possible from FFA - The Aeronautical Research Institute of Sweden and FOI - Swedish Defence Research Agency are greatly appreciated.

Bibliography

- ACIKGOZ, N. & BOTTASSO, C. 2006, A new Mesh Deformation Technique for Simplicial and Non-Simplicial Meshes. Paper AIAA-2006-0885.
- ALLEN, C. & RENDALL, T. 2007, Unified Approach to CFD-CSD Interpolation and Mesh Motion using Radial Basis Functions. Paper AIAA-2007-3804.
- AUBRY, R. & LÖHNER, R. 2007, Geration of Viscous Grids with Ridges and Corners. Paper AIAA-2007-3832.
- AUBRY, R. & LÖHNER, R. 2008, On the "Most Normal" Normal. *Communications in Numerical Methods in Engineering*, **24** (12), pp. 1641-1652.
- BABUSKA, I. & AZIZ, A. 1976, On the Angle Condition in the Finite Element Method. *SIAM Journal on Numerical Analysis*, **13** (2), pp. 214-226.
- BAKER, T. 1988, Generation of Tetrahedral Meshes around Complete Aircraft. Proceedings of the *2nd International Conference on Numerical Grid Generation in Computational Fluid Mechanics*, pp. 675-685, Miami Beach, Florida, USA.
- BATINA, J. 1991, Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aerodynamic Analysis. *AIAA Journal*, **29** (3), pp. 327-333.
- BAUM, J., LUO, H., LÖHNER, R., GOLDBER, E. & FELDHUN, A. 1997, Application of Unstructured Adaptive Moving Body Methodology to the Simulation of Fuel Tank Separation From an F-16 C/D Fighter. Paper AIAA-1997-0166.
- BERGLIND, T. 1997, Multigrid Solution of the Euler Equations on Triangular and Tetrahedral Grids. FFA TN 1997-27, FFA, The Aeronautical Research Institute of Sweden.
- BERGLIND, T. 2009, Numerical Simulation of Store Separation for Quasi-Steady Flow. FOI-R-2761-SE, FOI, Swedish Defence Research Agency.
- BERGLIND, T., PENG, S-H. & TYSELL, L. 2009, FoT25: Studies of Embedded Weapons Bays - Summary Report. FOI-R-2775-SE, FOI, Swedish Defence Research Agency.
- BONET, J. & PERAIRE, J. 1991, An Alternating Digital Tree (ADT) Algorithm for 3D Geometric Searching and Intersection Problems. *International Journal for Numerical Methods in Engineering*, **31** (1), pp. 1-17.
- CARCAILETT, R. 1986, Optimization of Three-Dimensional Computational Grids and Generation of Flow Adaptive Computational Grids. Paper AIAA-1986-0156.
- CASTRO-DIAZ, M., HECHT, F., MOHAMMADI, B. & PIRONNEAU, O. 1997, Anisotropic Unstructured Mesh Adaption for Flow Simulations. *International Journal for Numerical Methods in Fluids*, **25** (4), pp. 475-491.

- CATHERALL, D. 1988, Solution-Adaptive Grids for Transonic Flows. Proceedings of the *2nd International Conference on Numerical Grid Generation in Computational Fluid Mechanics*, pp. 329-338, Miami Beach, Florida, USA.
- CHAWNER, J., STEINBRENNER, J. & WYMAN, N. 2000, Hybrid Grid Generation for Complex Geometries using Gridgen. Proceedings of the *7th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 417-426, International Society of Grid Generation (ISGG), Whistler, British Columbia, Canada.
- CHENG, J., FINNIGAN, P., HATHAWAY, A., KELA, A., SHROEDER, W. 1988, Quadtree/Octree Meshing with Adaptive Analysis. Proceedings of the *2nd International Conference on Numerical Grid Generation in Computational Fluid Mechanics*, pp. 633-642, Miami Beach, Florida, USA.
- CONNEL, S. & HOLMES, D. 1994, A 3D Unstructured Adaptive Multigrid Scheme for the Euler Equations. *AIAA Journal*, **32** (8), pp. 1626-1632.
- DANNENHOFFER III, J. 1988, A Comparison of Two Adaptive Grid Techniques. Proceedings of the *2nd International Conference on Numerical Grid Generation in Computational Fluid Mechanics*, pp. 319-328, Miami Beach, Florida, USA.
- DELANAYE, M. & ESSERS, J. 1997, Quadratic-Reconstruction Finite Volume Scheme for Compressible Flows on Unstructured Adaptive Grids. *AIAA Journal*, **35** (4), pp. 631-639.
- DOMPIERRE, J., VALLET, M., BOURGAULT, Y., FORTIN, M. & HABASHI, W. 2002, Anisotropic Mesh Adaptation: Towards User-independent Mesh-independent and Solver-independent CFD. Part III. Unstructured Meshes. *International Journal for Numerical Methods in Fluids*, **39** (8), pp. 675-702.
- ELIASSON, P. 2001, EDGE - A Navier-Stokes Solver for Unstructured Grids. FOI-R-0298-SE, FOI, Swedish Defence Research Agency.
- ELIASSON, P. 2002, EDGE, a Navier-Stokes Solver for Unstructured Grids. Finite Volumes for Complex Applications III, ISBN 1-9039-9634-1, pp. 527-534.
- ELIASSON, P. & PENG, S-H. 2007, Drag Prediction for the DLR-F6 Wing-Body Configuration Using the Edge Solver. Paper AIAA-2007-0897.
- ELIASSON, P., NORDSTRÖM, J., PENG, S-H. & TYSELL, L. 2008, Effect of Edge-based Discretization Schemes in Computations of the DLR F6 Wing-Body Configuration. Paper AIAA-2008-4153.
- ERIKSSON, L. E. 1982, Generation of Boundary-Conforming Grids Around Wing-Body Configurations Using Transfinite Interpolation. *AIAA Journal*, **20** (10), pp. 1313-1320.
- ERIKSSON, L. E. 1985, Practical Three-Dimensional Mesh Generation Using Transfinite Interpolation. *SIAM Journal on Scientific and Statistical Computing*, **6** (3), pp. 712-741.
- ERIKSSON, L. E. 1987, Flow Solution on a Dual-Block Grid Around an Airplane. *Computer Methods in Applied Mechanics and Engineering*, **64** (1-3), pp. 79-93.
- FARHAT, C., DEGAND, C., KOOBUS, B. & LESOINNE, M. 1998, An Improved Method of Spring Analogy for Dynamic Unstructured Fluid Meshes. Paper AIAA-1998-2070.
- FORNASIER, L., DEISTER, F., HASSAN, O., TREMEL, U. & WEATHERHILL, N. P. 2003, Robust and Efficient Generation of Unstructured Surface Grids about

- Geometrically Complex Configurations Using "Real-Design" CAD Data. Paper AIAA-2003-0805.
- FORRESTER, G. & BOGUE, S. 2008, Production Unstructured Grid Generation using Nemesis powered by Boeings's AGPS. Paper AIAA-2008-0915.
- FREY, P. & ALAUZET, F. 2005, Anisotropic Mesh Adaptation for CFD Computations. *Computer Methods in Applied Mechanics and Engineering*, **194** (48-49), pp. 5068-5082.
- GAITHER, A., MARCUM, D. & MITCHELL, B. 2000, SolidMesh: A Solid Modelling approach to Unstructured Grid Generation. Proceedings of the *7th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 829-838, International Society of Grid Generation (ISGG), Whistler, British Columbia, Canada.
- GEUZAIN, C. & REMACLE J. F. 2009, Gmsh: A 3-D Finite Element Mesh Generator with Built-in Pre- and Post-processing Facilities. *International Journal for Numerical Methods in Engineering*, **79** (11), pp. 1309-1331.
- GHIDONI, A., PELIZZARI, E., REBAY, S. & SELMIN, V. 2006, 3D Anisotrop Unstructured Grid Generation. *International Journal for Numerical Methods in Fluids*, **51**, pp. 1097-1115.
- GUOLIANG, X., QING, P. & CHANDRAJIT, L. 2003, Discrete Surface Modeling using Geometric Flows. ICES Report 03-37, University of Texas, Austin, USA.
- HELENBROOK, B. 2003, Mesh Deformation Using the Biharmonic Operator. *International Journal for Numerical Methods in Engineering*, **56** (7), pp. 1007-1021.
- HOLCOMB, J. 1987, Development of a Grid Generator to Support 3-D Multizone Navier-Stokes Analysis. Paper AIAA-1987-0203.
- ITO, Y. & NAKAHASHI, K. 2004, Improvements in the Reliability and Quality Unstructured Hybrid Mesh Generation. *International Journal for Numerical Methods in Fluids*, **45** (1), pp. 79-108.
- ITO, Y., SHIH, A., SONI, B. & NAKAHASHI, K. 2006, An Approach to Generate High Quality Unstructured Hybrid Meshes. Paper AIAA-2006-0530.
- ITO, Y., MURAYAMA, M., YAMAMOTO, K., SHIH, A. & SONI, B. 2008, Development of a Grid Generator to Support 3-D Multizone Navier-Stokes Analysis. Paper AIAA-2008-7180.
- JACQUOTTE, O. 1991, Recent Progress on Mesh Optimization. Proceedings of the *3rd International Conference on Numerical Grid Generation in Computational Fluid Dynamics and related fields*, pp. 703-715, Barcelona, Spain.
- JAKOBSSON, S. & AMIGNON, O. 2005, Mesh Deformation using Radial Basis Functions for Gradient Based Aerodynamic Shape Otimization. Technical Report FOI-R-1784-SE, FOI, Swedish Defence Research Agency.
- JONES, W. 2003, GRIDEX - An Integrated Grid Generation Package for CFD. Paper AIAA-2003-4129.
- KALLINDERIS, Y. & WARD, S. 1993, Prismatic Grid Generation for Three-Dimensional Complex Geometries. *AIAA Journal*, **31** (10), pp. 1850-1856.
- KALLINDERIS, Y., KHAWAJA, A. & MCMORRIS, H. 1996, Hybrid Prismatic/Tetrahedral Grid Generation for Viscous Flows Around Complex Geometries. *AIAA Journal*, **34** (2), pp. 291-298.
- KALLINDERIS, Y. & KWONG, S. 2001, Viscous Grids and Assessment of their quality. Paper AIAA-2001-2539.

- KALLINDERIS, Y. & KONTZIALIS, C. 2009, A Priori Mesh Quality Estimation via Direct Relation Between Truncation Error and Mesh Distortion. *Journal of Computational Physics*, **228** (3), pp. 881-902.
- KANIA, L. & PIRZADEH, S. 2005, A Geometrically-derived Background Function for Automated Unstructured Mesh Generation. Paper AIAA-2005-5240.
- KARMAN, S. 2007, Unstructured Viscous Layer Insertion Using Linear-Elastic Smoothing. *AIAA Journal*, **45** (1), pp. 168-180.
- KHAWAJA, A., KALLINDERIS, Y., IRMISCH, S., LLOYD, J., WALKER, D. & BENZ, E. 1999, Adaptive Hybrid Grid Generation for Turbomachinery and Aerospace Applications. Paper AIAA-1999-0916.
- KHAWAJA, A. & KALLINDERIS, Y. 2000, Hybrid Grid Generation for Turbomachinery and Aerospace Applications. *International Journal for Numerical Methods in Engineering*, **49** (1-2), pp. 145-166.
- LARWOOD, B., WEATHERILL, N. P., HASSAN, O. & MORGAN, K. 2003, Domain Decomposition Approach for Parallel Unstructured Mesh Generation. *International Journal for Numerical Methods in Engineering*, **58** (2), pp. 177-188.
- LEATHAM, M., STOKES, S., SHAW, J. A., COOPER, J., APPA, J. & BLAYLOCK, T. A. 2000, Automatic Mesh Generation for Rapid-Response Navier-Stokes Calculations. Paper AIAA-2000-2247.
- LEWIS, R. W., ZHENG, Y. & GETHIN, D. T. 1996, Three-Dimensional Unstructured Mesh Generation: Part 3. Volume Meshes. *Computer Methods in Applied Mechanics and Engineering*, **134** (3-4), pp. 285-310.
- LIU, X., QIN, N. & XIA, H. 2006, Fast Dynamic Grid Deformation Based on Delaunay Graph Mapping. *Journal of Computational Physics*, **211** (2), pp. 405-423.
- LUKE, E., HEBERT, S. & THOMPSON, D. 2008, Theoretical and Practical Evaluation of Solver-Specific Mesh Quality. Paper AIAA-2008-0934.
- LÖHNER, R. 1988, Some Useful Data Structures for the Generation of Unstructured Grids. *Communication in Applied Numerical Methods*, **4** (1), pp. 123-135.
- LÖHNER, R. & PARIKH, P. 1988, Generation of Three-Dimensional Unstructured Grids by the Advancing- Front Method. Paper AIAA-1988-0515.
- LÖHNER, R. 1995, Mesh Adaptation in Fluid Mechanics. *Engineering Fracture Mechanics*, **50** (5-6), pp. 819-847.
- LÖHNER, R. 2000, A parallel Advancing Front Grid Generation Scheme. Paper AIAA-2000-1005.
- LÖHNER, R. & YANG, C. 1996, Improved ALE Mesh Velocities for Moving Bodies. *Communications in Numerical Methods in Engineering*, **12** (10), pp. 599-608.
- LÖHNER, R. & CEBRAL, J. 2000, Generation of Non-Isotropic Unstructured Grids via Directional Enrichment. *International Journal for Numerical Methods in Engineering*, **49** (1-2), pp. 219-232.
- MARTINEAU, D. & GEORGALA, J. 2004, A Mesh Movement Algorithm for High Quality Generalised Meshes. Paper AIAA-2004-0614.
- MARTINEAU, D., STOKES, S., MUNDAY, S., JACKSON, A., GRIBBEN, B. & VERHOEVEN, N. 2006, Anisotropic Hybrid Mesh Generation for Industrial RANS applications. Paper AIAA-2006-0534.
- MAVRIPILIS, D. 1990, Accurate Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes. *AIAA Journal*, **28** (2), pp. 213-221.

- MAVRIPILIS, D. 2005, Grid Resolution Study of a Drag Prediction Workshop Configuration Using the NSU3D Unstructured Mesh Solver. Paper AIAA-2005-4729.
- MAVRIPILIS, D. 2008, Unstructured-Mesh Discretizations and Solvers for Computational Aerodynamics. *AIAA Journal*, **46** (6), pp. 1281-1298.
- MAVRIPILIS, D., VASSBERG, J., TINOCO, E., MANI, M., BRODERSEN, O., EISFELD, B., WAHLS, R., MORRISON, J. ZICKUHR, T. LEVY, D. & MURAYAMA, M. 2008, Grid Quality and Resolution Issues from the Drag Prediction Workshop Series. Paper AIAA-2008-0930.
- MCKENZIE, S., DOMPIERRE, J., TURCOTTE, A. & MENG, E. 2009, On Metric Tensor Representation Intersection and Union. Proceedings of the *11th ISGG Conference on Numerical Grid Generation*, International Society of Grid Generation (ISGG), Montreal, Canada.
- MCMORRIS, H. & KALLINDERIS, Y. 1997, Octree-Advancing Front Method for Generation of Unstructured Surface and Volume Meshes. *AIAA Journal*, **35** (6), pp. 976-984.
- NIELSEN, E. & ANDERSON, K. 2001, Recent Improvements in Aerodynamic Design Optimization on Unstructured Meshes. Paper AIAA-2001-0596.
- OLLIVIER-GOOCH, C. 2003, Coarsening Unstructured Meshes by Edge Contraction. *International Journal for Numerical Methods in Engineering*, **57** (3), pp. 391-414.
- PARK, M. & DARMOFAL, D. 2008, Parallel Anisotropic Tetrahedral Adaptation. Paper AIAA-2008-0917.
- PEARCE, D., MARTIN, F., GOMEZ, R., LE BEAU, G., BUNING, P., CHAN, W., CHIU, I., WULF, A. & AKDAG, V. 1993, Development of a Large Scale Chimera Grid System for the Space Shuttle Launch Vehicle. Paper AIAA-1993-0533.
- PERAIRE, J., MORGAN, K. & PEIRO, J. 1990 Unstructured Mesh Methods for CFD. VKI Lecture Series 1990-06.
- PERAIRE, J., PEIRO, J. & MORGAN, K. 1992 Adaptive Remeshing for Three-Dimensional Compressible Flow Computations. *Journal of Computational Physics*, **103** (2), pp. 269-285.
- PIRZADEH, S. 1994, Viscous Unstructured Three-Dimensional Grids by the Advancing-layers method. Paper AIAA-1994-0417.
- PIRZADEH, S. 1996, Three-Dimensional Unstructured Viscous Grids By the Advancing-Layers Method. *AIAA Journal*, **34** (1), pp. 43-49.
- PIRZADEH, S. 1999, Unstructured Grid Generation for Complex 3D High-Lift Configurations. Paper SAE-1999-01-5557.
- PIRZADEH, S. 2000, A solution-Adaptive Unstructured Grid Method by Grid Subdivision and Local Remeshing *Journal of Aircraft*, **37** (5), pp. 818-824.
- PIRZADEH, S. 2008, Advanced Unstructured Grid Generation for Complex Aerodynamic Applications. Paper AIAA-2008-7178.
- REMAKI, L., NADARAJAH, S. & HABASHI, W. 2006, On the a Posteriori Error Estimation in Mesh Adaptation to Improve CFD Solutions. Paper AIAA-2006-0890.
- ROLSTON, S. & ELSHOLZ, E. 2002, Initial Achievements of the European High Reynolds Number Aerodynamic Research Project "HiReTT". Paper AIAA-2002-0421.
- RUBBERT, P. & LEE, K. 1982, Patched Coordinate Systems. Numerical Grid Generation, pp. 235-252, Elsevier Science Publishing Company.

- SAMAREH, J. 2002, Application of Quaternions for Mesh Deformation. Proceedings of the *8th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 47-57, International Society of Grid Generation (ISGG), Honolulu, Hawaii, USA.
- SHAROW, D., LOU, H. & BAUM, J. 2001, Unstructured Navier Stokes Grid Generation at Corners and Ridges. Paper AIAA-2001-2600.
- SHAW, J., FORSEY, C., WEATHERILL, N. P. & ROSE, K. 1986, A Block Structured Mesh Generation Technique for Aircraft Geometries. Numerical Grid Generation in Computational Fluid Dynamics, Pineridge Press Limited.
- SHEPHARD, M., GUERINONI, F., FLAHERTY, J., LUDWIG, R. & BAEHMANN, P. 1988, Finite Octree Mesh Generation for Automated Adaptive Three-Dimensional Flow Analysis. Proceedings of the *2nd International Conference on Numerical Grid Generation in Computational Fluid Mechanics*, pp. 709-718, Miami Beach, Florida, USA.
- SHETA, E., YANG, H. & HABCHI, S. 2006, Solid Brick Analogy For Automatic Grid Deformation For Fluid-Structure Interaction. Paper AIAA-2006-3219.
- SHIH, A., ITO, Y., KOOMULLIL, R., KASMAI, T., JANKUN-KELLY, M., THOMPSSON, D. & BREWER, W. 2007, Solution Adaptive Mesh Generation using Feature-Aligned Embedded Surface Meshes. Paper AIAA-2007-0558.
- SINGH, K., NEWMAN, C. & BAYSAL, O. 1995, Dynamic Unstructured Methods for Flow Past Multiple Objects in Relative Motion. *AIAA Journal*, **33** (4), pp. 641-649.
- SMITH, R. 1983, Three-Dimensional Algebraic Grid Generation. Paper AIAA-1983-1904.
- SMITH, W. 1990, Multigrid Solution of Transonic Flow on Unstructured Grids. *Recent Advances and Applications in Computational Fluid Dynamics*, Proceedings of the ASME Winter Annual Meeting.
- SONI, B. & THOMPSON, J. 1992, GENIE++, EAGLEView and TIGER: General and Special Purpose Graphically Interactive Grid Systems. Paper AIAA-1992-0071.
- SONI, B., THOMPSON, D., KOOMULLIL, R. & THORNBURG, H. 2001, GGTK: A Tool Kit for Static and Dynamic Geometry-Grid Generation and Adaptation. Paper AIAA-2001-1164.
- SORENSEN, R. & MCCANN, K. 1991, A Method for Interactive Specification of Multiple-Block Topologies. Proceedings of the *3rd International Conference on Numerical Grid Generation in Computational Fluid Dynamics and related fields*, pp. 731-742, Barcelona, Spain.
- SORENSEN, R. 1986, Three-Dimensional Elliptic Grid Generation About Fighter Aircraft for Zonal Finite-Difference Computations. Paper AIAA-1986-0429.
- STEINBRENNER, J. & ABELANET, J. 2007, Anisotropic Tetrahedral Meshing Based on Surface Deformation Techniques. Paper AIAA-2007-0554.
- SVÄRD, M., GONG, J. & NORDSTRÖM, J. 2005, An accuracy evaluation of unstructured node-centred finite volume methods. Technical report 2005-04, National Institute of Aerospace, NIA, Virginia, USA.
- TCHON, K-F., GUIBAULT, F., DOMPIERRE, J. & CAMARERO, R. 2005, Adaptive Hybrid Meshing Using Metric Tensor Line Networks. Paper AIAA-2005-5332.
- THOMPSON, J. 1984, Grid Generation Techniques in Computational Fluid dynamics. *AIAA Journal*, **22** (11), pp. 1505-1523.

- THOMPSON, J., WARSI, Z. & MASTIN, C. 1985, Numerical Grid Generation: Foundations and Applications, North-Holland.
- THOMPSON, J. 1987, A Composite Grid Generation Code for General 3-D Regions. Paper AIAA-1987-0275.
- TYSELL, L. & HEDMAN, S. 1988, Towards a General Three-Dimensional Grid Generation System. Paper ICAS-88-4.7.4, pp. 1048-1058, Proceedings of the *16th ICAS Congress*, Jerusalem, Israel.
- TYSELL, L. 1994, An Advancing Front Grid Generation System for 3D Unstructured Grids. Paper ICAS-94-2.5.1, pp. 1552-1564, Proceedings of the *19th ICAS Congress*, Anaheim, California, USA.
- TYSELL, L., BERGLIND, T. & ENEROTH, P. 1998, Adaptive Grid Generation for 3D Unstructured Grids. Proceedings of the *6th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 391-400, International Society of Grid Generation (ISGG), Greenwich, UK.
- TYSELL, L. 2000:1, Grid Quality Indicators. FFAP-B-138, FFA, The Aeronautical Research Institute of Sweden.
- TYSELL, L. 2000:2, Hybrid Grid Generation for Complex 3D Geometries. Proceedings of the *7th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 337-346, International Society of Grid Generation (ISGG), Whistler, British Columbia, Canada.
- TYSELL, L. 2002, Grid Deformation of 3D Hybrid Grids. Proceedings of the *8th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 265-274, International Society of Grid Generation (ISGG), Honolulu, Hawaii, USA.
- TYSELL, L. 2005, A Higher Order Representation of Unstructured Surface Grids. Proceedings of the *9th International Conference on Numerical Grid Generation in Computational Field Simulations*, International Society of Grid Generation (ISGG), San Jose, California, USA.
- TYSELL, L. 2007, The TRITET Grid Generation System. Proceedings of the *10th ISGG Conference on Numerical Grid Generation*, International Society of Grid Generation (ISGG), Heraklion, Crete, Greece.
- TYSELL, L. & NORDSTRÖM, J. 2007, Accuracy Evaluation of the Unstructured Node-Centered Finite Volume Method in Aerodynamic Computations. Proceedings of the *10th ISGG Conference on Numerical Grid Generation*, International Society of Grid Generation (ISGG), Heraklion, Crete, Greece.
- TYSELL, L. 2008, Experiences of Grid Generation and Steady/Unsteady Viscous Computations for Complex Geometries. Paper ICAS-2008-2.5.2, Proceedings of the *26th ICAS Congress*, Anchorage, Alaska, USA.
- TYSELL, L. 2009, CAD Geometry Import for Grid Generation. Proceedings of the *11th ISGG Conference on Numerical Grid Generation*, International Society of Grid Generation (ISGG), Montreal, Canada.
- VASSBERG, J., TINOCO, E., MANI, M., BRODERSEN, O., EISFELD, B., WAHLS, R., MORRISON, J., ZICKUR, T., LAFLIN, K. & MAVRIPILIS, D. 2008, Abridged Summary of the Third AIAA Computational Fluid Dynamics Drag Prediction Workshop. *AIAA Journal*, **45** (3), pp. 781-797.
- VENDITTI, D. & DARMOFAL, D. 2003, Anisotropic Grid Adaptation for Functional Outputs: Application to Two-Dimensional Viscous Flows. *Journal of Computational Physics*, **187** (1), pp. 22-46.

- WALTZ, J. & LÖHNER, R. 2000, A Grid Coarsening Algorithm for Unstructured Multigrid Applications. Paper AIAA-2000-0925.
- WEATHERILL, N. P., HASSAN, O. & MARCUM, D. 1993, Calculation of Steady Compressible Flowfields with the Finite Element Method. Paper AIAA-1993-0341.
- WEATHERILL, N. P., HASSAN O., MARCUM D. & MARCHANT, M. 1994, Grid Generation by the Delaunay Triangulation. VKI Lecture Series 1994-02.
- WEATHERILL, N. P., SAID, R. & MORGAN, K. 1998, The Construction of Large Unstructured Grids by Parallel Delaunay Grid Generation. Proceedings of the *6th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 53-73, International Society of Grid Generation (ISGG), Greenwich, UK.
- WEATHERILL, N. P., HASSAN, O., MORGAN, K., JONES, J. & LARWOOD, B. 2001, Towards Fully Parallel Aerospace Simulations on Unstructured Meshes. *International Journal for Numerical Methods in Engineering*, **18** (3-4), pp. 347-376.
- WINZELL, B., WANG, G. S. & RABIA, H. 2004, Loads on the Horizontal Stabilizer of a Twin-Otter and Vertical Gust Response Related to the Mehamn Accident Investigation. FOI-RH-0345-SE, FOI, Swedish Defence Research Agency.
- WOAN, C. 1987, Three-Dimensional Elliptic Grid Generations Using a Multi-Block Method. Paper AIAA-1987-0278.
- WOODARD, P., BATINA, J. & YANG, H. 1994, Unstructured Mesh Quality Assessment and Upwind Euler Solution Algorithm Validation. *Journal of Aircraft*, **31**(3), pp. 644-650.
- XIA, G., LI, D. & MERKLE, C. 2001, Anisotropic Grid Adaptation on Unstructured Meshes. Paper AIAA-2001-0443.
- ZHENG, Y., WEATHERILL, N. P. & TURNER-SMITH, E. 2002, An Interactive Geometry Utility Environment for Multi-Disciplinary Computational Engineering. *International Journal for Numerical Methods in Engineering*, **53** (6), pp. 1277-1299.