

Geometry Modeling and Grid Generation for Design and Optimization

Jamshid A. Samareh
Multidisciplinary Optimization Branch
Mail Stop 159
NASA Langley Research Center
Hampton, VA 23681-2199
j.a.samareh@larc.nasa.gov
<http://fmad-www.larc.nasa.gov/mdob>

ICASE/LaRC/NSF/ARO WORKSHOP ON
COMPUTATIONAL AEROSCIENCES IN THE 21st CENTURY
Hampton, Virginia
April 22-24, 1998

GEOMETRY MODELING AND GRID GENERATION FOR DESIGN AND OPTIMIZATION

JAMSHID A. SAMAREH

Multidisciplinary Optimization Branch

NASA Langley Research Center

Mail Stop 159

Hampton, VA 23681-2199

J.A.SAMAREH@LaRC.NASA.GOV

Abstract. Geometry modeling and grid generation (GMGG) have played and will continue to play an important role in computational aerosciences. During the past two decades, tremendous progress has occurred in GMGG; however, GMGG is still the biggest bottleneck to routine applications for complicated Computational Fluid Dynamics (CFD) and Computational Structures Mechanics (CSM) models for analysis, design, and optimization. We are still far from incorporating GMGG tools in a design and optimization environment for complicated configurations. It is still a challenging task to parameterize an existing model in today's Computer-Aided Design (CAD) systems, and the models created are not always good enough for automatic grid generation tools. Designers may believe their models are complete and accurate, but unseen imperfections (e.g., gaps, unwanted wiggles, free edges, slivers, and transition cracks) often cause problems in gridding for CSM and CFD. Despite many advances in grid generation, the process is still the most labor-intensive and time-consuming part of the computational aerosciences for analysis, design, and optimization. In an ideal design environment, a design engineer would use a parametric model to evaluate alternative designs effortlessly and optimize an existing design for a new set of design objectives and constraints. For this ideal environment to be realized, the GMGG tools must have the following characteristics: (1) be automated, (2) provide consistent geometry across all disciplines, (3) be parametric, and (4) provide sensitivity derivatives.

This paper will review the status of GMGG for analysis, design, and optimization processes, and it will focus on some emerging ideas that will advance the GMGG toward the ideal design environment.

1. Introduction

In 1975 Dean Chapman (Chapman et al. 1975) made the following prediction, “to displace wind tunnels as the principal source of flow simulations for aircraft design, computers must reach about ten thousand times the speed of the Illiac IV.” By some accounts we have already reached this goal with today’s supercomputers. But still the wind tunnels play a major role in aircraft design, which may require over 30 thousand hours of wind tunnel testing (Roskam 1990). The airplane design process resembles a jigsaw puzzle, requiring MultiDisciplinary Analysis (MDA) and Multidisciplinary Design and Optimization (MDO). GMGG has an important role in both areas. Complexity of the geometry models is increasing; in today’s preliminary design environment it is not unusual for a CAD model to use over 20 thousand curves and surfaces to represent an aircraft. This level of complexity underlines the importance of automation. An ignored consideration in most existing GMGG tools is the sensitivity analysis which is required for the gradient-based optimization. Sensitivity is defined as the partial derivative of the geometry model or grid point with respect to a design variable. They can be calculated either analytically or by finite differences. To streamline and automate the MDA and MDO processes, the following GMGG tools and capabilities are required:

- design oriented CAD systems
- creation of a complete and accurate CAD model
- easy and rapid model parameterization
- automatic and accurate tools to transfer geometry from CAD to grid generators
- robust and fully automatic (push button) grid generators
- easy and accurate grid sensitivity computation with respect to design variables
- tools to handle multidisciplinary interactions
- consistent CAD models for all disciplines

Geometry is a common data set that must be manipulated and shared among various disciplines. In traditional design processes, MDA and MDO are performed in an ad hoc manner, with data “thrown over the wall” from one discipline to another with no considerations for consistency and accuracy. This cultural habit not only affects the consistency and accuracy of the processes, it also increases the design cycle time and cost. Robust and automated GMGG tools could reduce the design cycle time and cost.

There is a large volume of published research in GMGG areas; however, there are few robust tools that are ready for incorporation into the MDA and MDO process. It takes many years to implement a published research into a robust tool. For example, research in Solid Modeling (SM) became

visible in the mid-1960s, and by the mid-1970s, the first generation of experimental systems had appeared (Requicha and Voelcker 1982); these systems were based on simple, analytical solids. After three decades of development, these commercial solid modelers can handle relatively complex models; however, they are not robust yet. Similarly, research in Feature-Based Solid Modeling (FBSM) technology has been conducted since the 1980s, but the FBSM has only become available in commercial CAD systems within the past five years. Also, it took more than a decade to implement the automated algorithm for CSM tetrahedral grid generation (Shepard and Yerry 1984) for use with solid models.

This paper will review the essential elements of GMGG. These are: CAD, SM, FBSM, standards for geometry exchange, grid generation, and geometry parameterization.

2. CAD Systems

Use of CAD systems for geometry modeling potentially could save development time in an MDO environment. However, there are two drawbacks: (1) initial investment and (2) inability to calculate analytical sensitivity.

In the past decade, CAD systems have gone through a series of revolutionary changes; for a more detailed account on these changes readers are referred to the handbook by Machover (1996). CAD systems have evolved from a two-dimensional modeling paradigm to a three-dimensional, solid, parametric and feature-based modeling paradigm. Among today's major CAD systems, sets of functionalities are similar but sets of flavors are different. As a result, the selection of a CAD system is more a business decision than a technical one. Three major U.S. car companies demonstrated this in that each company selected a single but different CAD system for the entire company.

Computer-aided design tools have arrived at this present state through three major advances over the past several decades: the incorporation of (1) NonUniform Rational B-Splines (NURBS), (2) SM, and FBSM. In a traditional CAD system, the geometry is represented as one of many possible mathematical forms, such as Bezier, Coons patches, B-spline curves and surfaces. However, one can use NURBS equations to represent most spline and implicit curves and surfaces without loss of accuracy (Farin 1990). NURBS can represent quadric primitives (e.g., cylinders, and cones), as well as free form geometry (Farin 1990). Although some surfaces [e.g., helix and helicoidal (Letcher and Shook 1995)] cannot be directly converted to a NURBS representation, these surfaces are not common in most aerospace applications. The SM and FBSM systems are discussed in the next two sections.

3. Solid Modeling (SM)

Most SM CAD systems use either a Boundary Representation (B-Rep) or Constructive Solid Geometry (CSG) method to represent a physical solid object (LaCourse 1995). The B-Rep and CSG representations provide a complete mathematical definition of a solid object. In contrast to traditional surface modeling software, solid modeling software have automated the process of creating solid model topology. Users need neither to trim surfaces nor to keep track of relevant parts. Solid modeling CAD systems keep track of surfaces, intersection curves and appropriate trim sections. Also, they keep track of the space that lies outside and inside the closed volume of the part, so that the described shapes can unambiguously be physically realized as solids. Most SM software hide the tedious topology information from users. This approach enables designers to create and modify shapes much faster than is possible with explicit surface modeling software. Solid modeling helps to avoid design errors, and it allows designers to better understand how their products will look and function before physical models are made. The following is a list of SM capabilities:

- create a complete geometry that is suitable for detailed CFD and CSM analyses
- clearly define mating conditions between parts
- detect interference automatically
- create a computer model for rapid prototyping (e.g., through stereolithography)
- allow reuse of solids in design

Solid modeling technology has a great potential for automating the GMGG process, but it is not yet mature. Building accurate, complicated geometry is still the Achilles heel of SM systems. Often, designers believe their models are complete and accurate, but unseen imperfections cause problems in applications such as grid generation, data exchange, numerical control programming, and rapid prototyping. The following is a list of problems that affect topology, accuracy, and grid generation:

- free edges
- bad loops (inconsistent face or surface normals)
- unacceptable vertex-edge gaps
- unacceptable edge face-gaps
- unacceptable loop closure gaps
- minute edges
- sliver faces
- transition cracks

The first two on the list are topology errors. Free edge is an edge of a face that is not shared by any other face. Bad loop occurs when the edge

of a face has a wrong orientation; as a result the face normal points in the wrong direction. Another source of error is inaccuracy in computing deviations allowed among different topological entities, such as faces, intersection curves, and vertices (Ferguson et al. 1996). For example, there is no precise solution for calculating the curve of intersection between two arbitrary B-spline surfaces. Consequently, SM software must use an approximate intersection curve, which does not lie on either surface. These deviations are usually so small that they cannot be detected by rendering the solid model. Yet, in the presence of these deviations, automatic grid generation and translation tools often fail. This problem can be avoided for a simple geometric design by using simple analytical surfaces (e.g., conics) which have exact analytical intersection curves. However, using simple analytical surfaces is not possible for an aircraft design process which relies on complex free-form surfaces.

Often the data translation failure is mistakenly blamed on the data exchange standard [e.g., Initial Graphics Exchange Specifications (IGES) and STEP (an acronym derived from the French title)]. In reality, lack of a consistent tolerance between sending and receiving systems is the source of the problem. To avoid this problem the CAD systems must store an intersection tolerance with each entity that defines the solid. Few CAD systems follow this approach.

With a complete and accurate solid model, the grid generation software may still fail. The problem is usually the sliver faces that result from patching between larger surfaces in a model. In order to create almost equilateral triangles, automatic grid generation tools will create an unnecessarily fine grid near these sliver faces. The resulting analysis will require large amounts of computer resources, and the analysis result will not be accurate due to excessive grid skewness.

These errors could prevent the SM technology from being used in an automated GMGG environment. Some CAD systems are finding solutions in using tolerance modeling and healing to bridge precision issues. Tolerance modeling allows the receiving system to relax its default precision requirements, but these exceptions may not be supported by all integrated CAD and Computer-Aided Engineering (CAE) applications. Healing software runs the CAD model through automatic cleaning or tightening algorithms, which may make adjustments that may be unacceptable to the designer. Some SM software allow users to control tolerances, and these software can be used to correct accuracy problems. However, selecting extremely low tolerances may prohibit models from regenerating. Cleaning up these anomalies impedes the automation of grid generation and can potentially add 50 percent to the time it takes to go from a CAD model to a CFD or CSM grid.

For a detailed airplane design, working with a solid model requires at least an order of magnitude more computer resources than working with a surface model. For example, aircraft designers will have a hard time finding resources to create and assemble the large number of components for a wing or fuselage using current SM tools. As a result, the airplane cannot be modeled completely with currently available computer hardware and SM CAD systems. Despite these problems, SM software produce much higher quality data than a user can create with a traditional surface modeling software.

Another basic problem with the solid model representation is that the design intent is not captured: the final design is not made up of features that capture the design intent. The design process is bottom-up, and the design changes are very time consuming.

4. Feature-Based Solid Modeling (FBSM)

Adding features to SM CAD has resolved the design intent problem. Features are dimension-driven objects that are the basis of the FBSM construction techniques (Shah and Mantyla 1995). They use Boolean operations such as intersection and union of simple features. Examples of simple features include holes, slots (or cuts), bosses (or protrusions), fillets, chamfers, sweep, and shell. Although research in FBSM technology has been conducted for more than fifteen years, FBSM has only become available in commercial CAD systems in the past five years. Today's CAD systems allow designers to work in 3D using topologically complete geometry (solids) that could be modified by altering the dimensions of the features from which it was created. The FBSM has made design modification much easier and faster. The developers of FBSM CAD systems have put the "D" back in CAD. Today's design engineers can create a new, complete, parametric model for a configuration, and FBSM CAD can be incorporated into a design environment.

With FBSM tools, the designers must define the relationship and constraints among geometric entities for each feature in the model. This requires some additional time, thought, and planning, but it will pay off when the designer needs to change the model. As a result, the design changes are not time consuming, and it is easier to change the model in order to develop new variants of existing designs. For example, it is much easier and faster to model holes in a design using solid cut operations than it is to do them with traditional surface modeling tools.

The FBSM process relies on simple top-down and high-level geometric constructions. The most important capability of FBSM is the ability to capture the design intent. Embedding this intelligence in a model allows

workers who are not thoroughly familiar with a product to make changes to existing designs. Another benefit is the capability of suppressing the small features for analysis purposes.

Feature-based solid modeling facilitates the implementation of object-oriented design in CAD systems. For example, a screw within a product can be a unique object. In a design for a new engine, that screw may need to be replicated hundreds of times. Simply copying a single part like a screw is fairly easy with FBSM CAD systems. However, taking this example one step further, the screw has a property that describes its diameter. Suppose the overall engine design changes during a project review, and the screws need to be thicker to support more weight. Then, it should be possible to change just one copy of the screw used in the design. Through links managed by the system, all of the identical screws would reflect the new diameter. Feature-based solid modeling CAD systems treat components as objects, and they can do this task fairly simply. The result may be the automation of design methods used consistently within one organization, or the result may be an engineer's design changes reflected through dynamically linked objects used across multiple designs.

Object-oriented CAD makes it easier to share data between applications because it introduces a layer of abstraction between the data and the user. Instead of requiring every application to translate data between different formats, objects hold property information that describes how the data should be handled by an application. Current FBSM CAD systems offer a library of completed, or partially completed, parameterized objects, enabling one to capture a design process and knowledge and to document it as a set of objects.

There is another object model in use today. Within the software community, Microsoft's Object Linking and Embedding (OLE) specification provides one of the most broadly used implementations. This specification has been approved by the Design and Modeling Application Council (see DMAC web site) and many vendors. Object linking and embedding provides a variety of services enabling data to be shared easily across different applications. For example, the most common way to incorporate data from other applications into a single data file is to embed it as an object within an OLE document. This could facilitate a closer integration between FBSM CAD systems and CAE.

Because today's FBSM systems rely on SM techniques, created models are not always good enough for automatic grid generation tools. The CAD process may create a solid with unacceptable accuracy (e.g., cracks), with sliver faces, or with unacceptable and excessive geometric details. Feature-based solid modeling is not yet a mature and robust technology for complicated aerospace geometric modeling. Even though use of parametric mod-

eling in design would make the FBSM tools ideal for optimization, existing FBSM tools do not have the capability to calculate the analytical sensitivity of a CAD model with respect to the design variables. So it is far from trivial to incorporate FBSM CAD systems into a design optimization process, and it is even more difficult to incorporate them into an MDO environment. Also, it is still a challenging task to parameterize an existing model that is not parametric. It took over thirty years for SM tools to reach today's maturity, and FBSM has been around for less than ten years. The FBSM approach is a sound approach, but it will take another decade to mature enough for complicated aerospace MDO applications.

5. Geometry Exchange

Once the CAD model has been completed, the next step is to transfer the data to a CAE application such as a CFD code or a CSM code. Geometry exchange is always the biggest issue for going from CAD to CAD or from CAD to CAE, and it could be impeding the development of automatic CAE applications. There is very little incentive for CAD companies to provide a robust tool for geometry exchange. They fear that if they provide a robust tool, then they will loosen their hold on customers.

Obviously, the best way to share data is to use the same CAD system. Generally, exchanging data among different CAD systems is an unreliable process, so it makes sense to limit the number of CAD systems used in a process. For example, major U.S. car companies have reduced the number of data translations dramatically by selecting a single CAD system for the entire company. To exchange data between a small number of CAD systems, a direct translation is the most efficient and accurate way. These direct translation tools are expensive, but they are cost-effective for a large volume of data exchange.

If exchanging data is necessary, understanding what the data will be used for is the key ingredient for success. If a structured CFD grid must be created using imported data, then only the surface model is required. However, if an unstructured CFD grid must be generated, then transfer of solid models is the only way to satisfy this requirement without having to rebuild the geometry.

There are a number of different file formats for exchanging data among CAD and CAE systems. The most popular formats are IGES in the U.S., SET (an acronym derived from the French title "Standard d'Echange et de Transfer") in France, VDA (an acronym derived from the German title "Verband der Automobilindustrie") in Germany, and STEP worldwide. Table 1 shows a list of CAD representations and associated U.S. file formats to support them. For wireframe and solid data exchange, IGES or STEP

TABLE 1. Geometry Standards

Representation	Standards
Feature-Based Solid Models	Not supported
Solid Models	STEP and IGES
Surface Models	STEP, IGES, DXF
Tessellated Models	VRML, STL

can do the job, but to bring data from another system into an FBSM system, the only choice now is to rebuild the model manually. There is no standard file format to support the transfer of parametric data contained in an FBSM. Thus, if a parametric solid model is translated to a data exchange format and then read directly back into an FBSM, all parametric information is lost.

Initial Graphics Exchange Specifications (IGES 1996) was designed in 1979. It is the most popular format in North America, and it has become reliable for production work. A survey in 1993 found that 66 percent of firms used IGES for data transfer (PDES 1993). The format has gone through several major revisions. It has one big flaw: the data are stored in two sections of the file: a directory section and a parameter section. Many IGES bugs have to do with mismatches between directory and parameter sections. Also, IGES uses fixed-length records, which consume a lot of space even when nothing is in them, and therefore IGES files are very bulky. The development of IGES started in an era when punch cards were popular for putting data into computers. Even when the physical punch cards are not used, data are stored in the form of 80-character records.

STEP (STEP 1994) is an international standard for the exchange of product model data (ISO 10303). The Product Data Exchange using STEP is an American National Standard. STEP is a better geometry standard than IGES in several areas. It is international, is more compact, stores data for each entity in only one place, and uses a more modern data architecture. STEP is defined in terms of a new language, EXPRESS. The STEP Application Protocol number 203 (AP203), entitled "Configuration Controlled Design," encompasses the relationship between product parts, assemblies, bills of material, change authorizations, change requests, and model release information. The Part 42 of AP203 provides methods for describing three-dimensional CAD geometry. Boeing and its primary contractors have been using STEP successfully to check for interference between engine parts and airframe structures.

Part 42 includes most elements found in the IGES standard, including

2D and 3D points, lines, arcs, B-splines, conic sections, and planar, spherical, cylindrical, ruled, NURBS, trimmed, and offset surfaces. It also contains topology information for creating solids and their boundary representation. Also, primitive solids such as blocks and spheres may define shapes. AP203 data exchange is not yet highly reliable for analysis of aerodynamics solid models. Most of these solids are created with free-form surfaces, where curves of intersection cannot be defined precisely (Ferguson et al. 1996).

The most significant omission is that STEP does not have a way to describe the geometry constraints employed by FBSM CAD systems. It also lacks methods for rule-based geometry construction. STEP does not contain a history tree relating parts, and this prevents changes to individual parts. Consequently, both the feature descriptions and the parametric relationships that allow CAD models to be changed quickly will be lost in any STEP translation. There are some efforts to bring STEP into line with today's FBSM systems capabilities. The enhanced STEP could add means for capturing and exchanging parametric, constraint-based, and feature-based product models. This addition to STEP will not be a trivial exercise, since parameterization is often associated with a history-based approach to modeling, while STEP is currently oriented firmly towards the exchange of the explicit, or 'snapshot', type of product model.

There are two other standards that can be used to exchange data between CAD and CAE. These standards are STereoLithography (STL) and Virtual Reality Markup Language (VRML)—both simple in nature. The STL format is the de facto standard for rapid prototyping. The STL file format allows for the representation of a CAD model as a set of triangles and their normals. The specification of the STL format states that the model must represent a tessellated solid, and STL may be used for automatic grid generation instead of the full CAD geometry. A limitation of STL, which is a tessellated representation, relates to accuracy. For STL models, certain features such as rounds are converted to triangles, and radius information is not accessible. The VRML standard is very similar to STL, and most CAD systems support both. In addition to tessellated data, the VRML standard supports quadrilaterals, cones, cubes, and circular cylinders. Dimensions can be queried from these models, but accuracy becomes an issue due to the approximation of the actual model. The VRML models can be used for grid generation, but they too lack accuracy.

Another important element for design and optimization that has been left out of all standards for data exchange is the sensitivity of CAD models with respect to design variables. And presently there is no plan to include sensitivity in the future standards.

6. Grid Generation

Grid generation is the first step in CAE analysis. There is a tremendous amount of published research on the mathematics of grid generation and its algorithms [(Smith 1980), (Thompson 1982), (Hauser and Taylor 1986), (Sengupta et al. 1988), (Arcilla et al. 1991), (Weatherill et al. 1994), (Mitchell 1996), (Soni et al. 1996)]. But, there are few good grid generation codes. The reason is that writing a good code is significantly more time consuming than writing a good paper.

Fortunately the CAE software companies have realized that the need for stand-alone grid generation products is diminishing in favor of more integrated tools. These tools have a direct connection to CAD systems either through a tight integration with CAD or through the data exchange standards (e.g., IGES and STEP). CAD is traditionally seen as the carrier of information about design. However, grid generation usually requires simplification and idealization of the design model. This requirement is the most cumbersome aspect of grid generation process. Therefore, the analysis model is often rebuilt from scratch, relying upon the judgment of skilled analysts in removing details from the design, and duplicating much of the work in creating the geometry. Often, integrated tools are interactive and require the design engineer to provide complex input. As a result, the grid generation process is not yet a “push button” process; it is the most labor-intensive and time-consuming aspect of the computational aerosciences. It takes too many man-hours and calendar days, and it requires a grid specialist. This limits the use of analysis codes in the preliminary design. To incorporate grid generation tools into a design and optimization system, the tools must

- use CAD generated geometry
- handle solid models with many surfaces [$O(10,000)$]
- handle surfaces with bad parameterization
- handle complex geometry
- be fully automatic (“push button”)
- be designed for non-specialists
- be robust and have a short design cycle time
- calculate grid sensitivity
- be able to create boundary layer/stretched grids
- have some level of grid quality control
- operate within an integrated system

This paper focuses on CFD and CSM grid generation methods. Even though both have the same goal of model discretization for analysis, they have different requirements. Generally CSM requires a relatively coarse grid, but it must handle very complex internal and external geometries.

In contrast, the CFD grid is very fine, but it must model the external geometry only. Both classes of grid generation techniques will be discussed in subsequent sections.

The feature-based approach has not been used in grid generation yet. This approach, Feature-Based Grid Generation (FBGG), could automate and simplify the grid generation process for very complicated designs based on FBSM. With this technique, the grid is generated for each base feature. As each CAD feature is combined with other features using a Boolean operation to form the model, the individual feature grids could be combined using the same Boolean operation to form a new grid. As with FBSM, FBGG could be based on Boolean operations such as intersection and union of simple grids. As a result, design changes would have little or no effect on the grid generation process, and it would be easy to generate a new grid for a variant of an existing design. Also as with FBSM, FBGG relies on a simple, top-down, high-level grid generation construction. It is also possible to create a grid for an idealized model by suppressing the features unnecessary for analysis purposes.

It is important to note that, with respect to design optimization, very few grid generation tools can provide the grid point sensitivity required for gradient-based optimization process (Jones and Samareh 1995).

6.1. CFD GRID GENERATION

CFD grid generation techniques have been developed around the formulations of spatial discretization of flow equations, such as multiblock structured, unstructured tetrahedral, unstructured mixed elements, and Cartesian grids. With the exception of Cartesian grid generation methods (Melton et al. 1995), all produce body-fitted grids. The Cartesian method is based on decomposing the domain into cells (Melton et al. 1995) that are oriented along the three Cartesian directions (x , y , and z). This approach can fully automate the CFD grid generation process. However, there are some questions regarding the accuracy of these methods for complicated physics. Structured and unstructured techniques have three distinct steps: topology creation, surface grid generation, and volume grid generation. With multiblock structured grid methods comes the problem of block topology creation, which has not been adequately automated. The unstructured tetrahedral, unstructured mixed-elements, and Cartesian grid generation techniques require the same surface geometry topology as the solid B-Rep model. Once the topology has been created, most grid generation techniques could be fully automated. There are some integrated CAD, structured, unstructured, and hybrid grid generation tools for CFD analysis, but they lack automation. Efforts in unstructured grid generation now appear to

concentrate on automation and grid quality. Readers are referred to three articles on CFD grid generation in this proceeding for further discussions.

6.2. CSM GRID GENERATION

Grid generation methods for CSM applications are based on either decomposition of a solid model into solid elements or dimensional reduction of a solid model into mixed solid/shell/beam elements. Most commercially available tools belong to the former category. Often these tools are based on P-element technology, where the elements may have curved edges. Consequently, a given part can be modeled to higher geometric accuracy with fewer elements than is possible with H-element (linear-edge) codes. The P-element technology developed at IBM's Almaden Research Center has been incorporated into several major CAD and CAE systems. This method is not good for analysis of anisotropic materials (composites), materials with nonlinear elastic curves, or systems with gaps and large nonlinear deflections. P-element codes require unique grid generation routines that can approximate the geometry with polynomial functions used in Finite Element Methods (FEM). There are commercial CSM tools that have integrated CAD, grid generation, and FEM analysis into a single tool. Generally, the use of these tools requires little or no FEM experience, and they are as easy for engineers as using spelling checkers.

The CSM grid generation tools are generally based on an octree approach proposed more than a decade ago (Shepard and Yerry 1984). The process of all-hexahedral grid generation has been automated for solid models. A simple, grid-based approach (Schneiders 1995) can generate the grid in the interior of the model, and then an isomorphism technique is used to generate the elements in the boundary regions. The plastering algorithm (Blacker and Meyers 1993) is based on an advancing front technique that generates a hexahedral grid starting from quadrilateral elements on the model boundary.

The second category of FEM grid generation tools is based on dimensional reduction of solid models where a solid model is converted to an equivalent mixed solid/shell/beam elements. A procedure has been developed for the automatic dimensional reduction of a two-dimensional geometric model to an equivalent one-dimensional-beam model. This was achieved by using the medial axis transform (Armstrong et al. 1995), an alternative, skeleton-like representation of the geometric model, having properties relevant to the model. Operations also have been defined and implemented [(Rezayat 1996); (Price et al. 1995)] for dimensional reduction of three-dimensional solid models. These operations are interactive, with appropriate physical properties, such as shell thickness, beam section, moment of

areas, and torsion constants, calculated automatically. These tools are not fully automated.

7. Geometry Parameterization

To avoid the GMGG complexity, often an aircraft is represented by a simple model during the conceptual and preliminary designs. Because simple models are neither accurate nor complete, optimization of these models could lead to an impractical design (Aidala et al. 1983; Hutchison et al. 1992). To use complex shapes in an MDO environment, the parameterization and geometry modeling must be compatible with existing CAD systems, and it must be adaptable to CFD and CSM. In order to integrate any GMGG tool into a design and optimization environment, the tool must

- use CAD for geometry creation
- generate grids automatically (black-box grid generation system)
- use a common geometry representation for all disciplines
- calculate analytical grid and geometry sensitivities
- transfer data among disciplines consistently (e.g., aeroelastic deflection)
- operate in an integrated system
- parameterize discipline models consistently

The rest of this section will focus on the parameterization issue. There are three approaches for parameterization: discrete, CAD, and free-form deformation.

7.1. DISCRETE APPROACH

The discrete approach is based on using coordinates of the grid points as design variables. This is easy to implement, and the geometry changes do not have a limited form. But it is difficult to maintain a smooth geometry, and the optimization process could create a problem in that the optimum design may be impractical to manufacture. Also, for a grid with a large number of points, the number of design variables often becomes very large, which leads to high costs and a difficult optimization problem to solve. The following is a list of important characteristics for discrete parameterization:

- complex and existing grids can be parameterized
- there is a strong local control
- analytical sensitivity is available
- there is no shape limitation
- there are too many design variables
- since grid for each discipline is parameterized separately, the parameterization is inconsistent

- discipline interaction is difficult to model
- smoothness is not guaranteed

7.2. CAD APPROACH

The second parameterization technique is based on using an FBSM CAD system. Calculations of the sensitivity of geometry with respect to the design variables could prove to be difficult. For some design variables, it is possible to relate the NURBS control points to the design variables. Then the analytical sensitivity can be calculated outside the CAD system. For some limited cases, the analytical shape sensitivity can be calculated based on a CAD model (Hardee et al. 1996). However, this method will not work under all circumstances. One difficulty is that a dimension may be chosen as a design variable for which the variation of a design surface cannot be assumed to be linearly dependent (Hardee et al. 1996). The second difficulty is that for some perturbation of some dimensions, topology of the part may be changed. Another way to calculate the sensitivity is to use finite difference, as long as the perturbed geometry has the same topology as the unperturbed one. Both methods, the analytical and finite differences, have their pitfalls and limitations. The following is a list of important characteristics for the CAD based parameterization:

- parameterization is consistent
- complex models can be parameterized
- smoothness can be controlled
- models require a few design variables
- the shape is limited by the parameterization
- it is difficult to parameterize existing models
- analytical sensitivity is very difficult to obtain
- there is very little local control
- it is difficult to use CAD for discipline interaction

7.3. FREE-FORM DEFORMATION APPROACH

During the preliminary design phase of an aircraft, when the focus is on the mathematical modeling of the outside skin, the free-form deformation technique could serve as an effective tool with sufficient accuracy. Creation of CFD and CSM grids is time consuming and costly. Therefore, the parameterization of existing grids is necessary for shape optimization.

The free-form deformation is very similar to morphing techniques [(Hall 1993); (Barr 1984)] used in computer animation. It can simulate planform, twist, dihedral, thickness, and camber variations. In a sense, the model is

treated as putty or clay in areas where it can be twisted, bent, tapered, compressed or expanded but retains the same topology.

For example, the planform variations are modeled with a set of quadrilaterals that control the changes. Then the planform design variables are linked to a set of vectors defined at the corners of the quadrilaterals. Any CFD or CSM grid point within a quadrilateral can be mapped from a three-dimensional space (\vec{r}) to two-dimensional parameter space (u, v) of the quadrilateral. The change in grid point location, $d\vec{r}$, is computed based on the parametric value, u, v .

The following is a list of important characteristics for this approach:

- parameterization is consistent
- analytical sensitivity is available
- complex existing analysis models (grids) can be parameterized
- smoothness can be controlled
- it requires few design variables
- shape changes are limited
- there is a strong local control
- discipline interaction is difficult to model

8. Multidisciplinary Interactions

Another important issue is the strong interaction among disciplines common in an MDO environment. All disciplines share the same geometry, and must be able to communicate and share information consistently (e.g., on deflections and loads). Multidisciplinary interactions can reflect physically important phenomena in aircraft, such as those occurring due to aeroelasticity. Correct modeling of these complex aeroelastic phenomena requires direct coupling of CFD and CSM codes. The interactions among various disciplines require the manipulation of the original CAD geometry stored as a set of NURBS. Currently, commercial CAD systems do not support this interaction. It is possible to map scalar fields (e.g., pressure) and vector fields on CAD geometry [(Samareh 1996) and (Samareh 1998)].

9. Summary

The GMGG tools are an enabling technology for traditional design processes of today and even more so for the revolutionary, integrated, multidisciplinary design processes of tomorrow. Geometry modeling and grid generation tools must (1) be automated, (2) provide consistent geometry across all disciplines, (3) be parametric, and (4) provide sensitivity derivatives. Despite the large volume of published research in GMGG areas, there are few robust tools that are ready for incorporation into MDA/MDO pro-

cesses. It usually takes twenty to thirty years from idea to implementation of an algorithm into a robust CAD tool.

Solid modeling tools for aerospace applications are not mature, and to solve their technical problems would require either a new generation of surface mathematics or some sort of tolerance-passing scheme yet to be perfected.

The FBSM technology will help us to automate the design process and perform optimization. However, it will probably take another decade to successfully implement the FBSM techniques in a commercial CAD system capable of handling the detailed design of a complete aircraft. Due to their generality and potential for automation, the unstructured and Cartesian grid generation techniques will become prevalent in future for CFD applications. There are commercial CSM grid generation codes available that are fully automatic; one area of research is the dimensional reduction of solid models into mixed solid/shell/beam elements. An automatic grid generation method has been proposed in this paper. The method, FBGG, is based on features and is compatible with FBSM, but it will take years to implement.

There are still a lot of open issues that need to be resolved. Following is a list of research opportunities for GMGG tools and algorithms.

- tools to automatically heal/mend solid models
- a tolerance-free geometry representation for solid modeling
- fully automatic topology creation for structured grid
- feature-based grid generation using constructive solid geometry
- rule/knowledge-based systems to design CSM topology
- dimensional reduction of solid models to solid/shell/beam elements
- tight CAD, grid generation, and CAE integration for MDO
- automatic tools to idealize geometry models (remove and create geometry)
- CAD-based tools for analytical sensitivity
- object oriented tools for design and optimization
- CAD tools to model the interdisciplinary interactions

References

- Aidala, P.V. Davis, W.H. Mason, W.H. (1983) Smart Aerodynamic Optimization, AIAA Paper 83-1863.
- Arcilla, A.S., Hauser, J., Eiseman, P.R., Thompson, J.F. (1991) *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*. North-Holland, New York.
- Armstrong, C.G., Robinson, D.J., McKeag, R.M., Li, T.S., Bridgett, S.J. (1995) Medials for Meshing and More, *The Proceedings of the 4th International Meshing Roundtable*, Sandia National Laboratories, Albuquerque, New Mexico, pp. 277-288.

- Barr, A.H. (1984) Global and Local Deformations of Solid Primitives, *Computer Graphics*, **Vol. 18, No. 3**, pp. 21-30.
- Blacker, T.D., Meyers, R.J. (1993) Seams and Wedges in Plastering, *Engineering with Computers*, **Vol. 9**, pp. 83-93.
- Chapman, D.R., Mark, H., Pirtle, M.W. (1975) Computers VS. Wind Tunnels for Aerodynamic Flow Simulations, *Astronautics & Aeronautics*, April 1975, pp. 22-35.
- Design and Modeling Applications Council. (<http://www.dmac.org>)
- Farin, G. (1990) *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, New York.
- Ferguson, D.R., Lucian, M.L., Seitelman, L. (1996) PDES, Inc., Geometric Accuracy Team Interim Report, ISSTECH-96-013, Boeing Information & Support Services, Seattle.
- Hall, V. (1993) Morphing in 2-D and 3-D, *Dr. Dobb's Journal*, **1993**, pp. 18-26.
- Hardee, E., Chang, K.H., Choi, K.K., Yu, X. Grindeanu, I. (1996) A CAD-Based Design Sensitivity Analysis and Optimization for Structural Shape Optimization Design Applications, AIAA Paper 96-3990-CP.
- Hauser, J., Taylor, C. (1986) *Numerical Grid Generation in Computational Fluid Mechanics*, Pineridge Press Limited, Swansea, UK.
- Hutchison, M.G., Huang, X., Mason, W.H., Haftka, R.T., and Grossman, B. (1992) Variable-Complexity Aerodynamic-Structural Design of a High-Speed Civil Transport Wing, AIAA-92-4695.
- IGES: Initial Graphics Exchange Specification(IGES 5.3)*, U. S. Product Data Association, North Charleston, South Carolina.
- Jones, W. T., Samareh, J.A. (1995) A Grid Generation System for Multi-disciplinary Design Optimization, AIAA Paper 95-1689.
- LaCourse, D. E. (1995) *Handbook of Solid Modeling*, McGraw-Hill, New York.
- Letcher, J.S., Shook, M. (1995) NURBS Considered Harmful for Gridding (Alternative Offered), *4th International Meshing Roundtable*, Sandia National Laboratories, Albuquerque, New Mexico, pp. 253-264.
- Machover, C. (1996) *The CAD/CAM Handbook*, McGraw-Hill, New York.
- Melton, J.E., Berger, M.J., Aftosmis, M.J. 3D Applications of a Cartesian Grid Euler Method, AIAA Paper 95-0853.
- Mitchell, S., (1996) *The 5th International Meshing Roundtable*, Sandia National Laboratories, Albuquerque, New Mexico.
- PDES Progress Report (1993) *Computer-Aided Design Report*, March, pp. 1-6.
- Price, M.A., Sabin, M.A., Armstrong, C.G. (1995) Fully Automatic Quad and Hex Meshing, *The Proceedings of 5th International Conference on Reliability of Finite Element Methods for Engineering Applications*, Amsterdam, pp. 356-367.
- Requicha, A.A.G., Voelcker, H.B. (1982) Solid Modeling: A Historical Summary and Contemporary Assessment, *IEEE Computer Graphics and Applications*, **Vol. 2, No. 2**, pp. 9-24.
- Rezayat, M. (1996) Midsurface abstraction from 3D Solid Models: General Theory and Applications, *CAD*, **Vol. 28, Issue 11**, pp. 917-928.
- Roskam, J., *Airplane Design*, **Vol. 8**, DARcorporation, Lawrence, Kansas.
- STEP: Product Data Exchange using STEP (1994) U. S. Product Data Association, North Charleston, South Carolina.
- Samareh, J.A. (1996) Use of CAD in MDO, *The 6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Bellevue, AIAA-96-3991, Seattle, Washington.
- Samareh, J.A. (1998) Aeroelastic Deflection of NURBS Geometry, *The Sixth International Conference on Numerical Grid Generation in Computational Field Simulation*, To be Published.
- Schneiders, R., (1995) Automatic Generation of Hexahedral Finite Element Meshes, *The Proceedings of the 4th International Meshing Roundtable*, Sandia National Laboratories, Albuquerque, New Mexico, pp. 103-114.

- Sengupta, S., Hauser, J., Eiseman, P.R., Thompson, J. F. (1988) *Numerical Grid Generation in Computational Fluid Mechanics*, Pineridge Press Limited, Swansea, UK.
- Shah, J.J., Mantyla, M. (1995) *Parametric and Feature-Based CAD/CAM*, John Wiley & Sons, New York.
- Shepard, M.S., Yerry, M.A. (1984) Finite Element Mesh Generation for Use with Solid Modeling and Adaptive Analysis, *Solid Modeling by Computers: From Theory to Applications*, Edited by M.S. Pickett and J.W. Boyse, Plenum Press, New York, pp. 53-80.
- Smith, R.E. (1980) *Numerical Grid Generation Techniques*, NASA CP-2166.
- Soni, B.K., Thompson, J.F., Hauser, J., Eiseman, P.R. (1996) *Numerical Grid Generation in Computational Field Simulation*, Mississippi State University, Mississippi State, Mississippi.
- Thompson, J.F. (1982) *Numerical Grid Generation*, North-Holland, New York.
- Weatherill, N.P., Eiseman, P.R., Hauser, J., Thompson, J.F. (1994) *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, Pineridge Press Limited, Swansea, UK.