Geometry-Adaptive Surface Grid Generation Using a Parametric Projection

K. D. Lee* and J. M. Loellbach†
University of Illinois at Urbana-Champaign, Urbana, Illinois

A general surface grid generation technique is presented that produces grids on arbitrary three-dimensional surface patches, with geometry-adaptive grid control. The method uses a two-dimensional parametric representation of the surface patch obtained by projecting the surface onto a locally defined screen. Arbitrary surfaces, including those with geometric singularities, are transformed into simple domains in the parametric space. A grid is generated on the parametric domain using a two-dimensional grid generation technique and the grid points are mapped back to the original surface. The method is applied to a variety of configurations with different geometrical difficulties.

Introduction

RECENT advances in computational technology are making it possible to obtain numerical solutions to problems with increasing degrees of physical and geometrical complexity. As the geometry becomes more complicated and the desired level of confidence in numerical results increases, the demand for suitable grids also increases. The accuracy and efficiency of numerical solutions depend on certain grid properties, such as grid topology, grid density, grid skewness, grid smoothness, cell aspect ratio, and variations of these properties throughout the grid. Also, the total number of grid points should be kept to a minimum for computational economy. The optimum grid qualities suggested by these factors are often conflicting and compromises must be made. Improved grid generation techniques are needed as physical and geometrical complexities increase.

After choosing a grid topology, surface grid generation is the next step in providing field grids for three-dimensional flow calculations. However, few techniques have been developed specifically for surface grid generation. Algebraic methods are often used, but most of these methods are based on assumed configurations or require excessive amounts of input. Other approaches include a method discussed by Thompson et al.[1] using a three-dimensional curvilinear coordinate system with one coordinate constant on the surface. Also, Thomas[2] proposed a surface grid generation scheme that uses a quasi-two-dimensional elliptic system for analytically defined, smooth surfaces. The method was later extended by Takagi and Miki[3] for arbitrary curved surfaces using a parametric surface representation. Eiseman[4] has demonstrated an adaptive surface grid generation scheme that distributes grid points to optimal positions based on flow solutions.

The purpose of this paper is to describe and demonstrate a surface grid generation technique that can be applied to arbitrary surface patches. The method uses a projection concept to define a parametric representation of the surface patch and a two-dimensional grid generation technique on the parametric domain. Geometry-adaptive grid control is achieved through the definition of the parametric coordinates. First, an overall description of the method is presented, followed by discussions of specific items used in the implementation of the method. Then, several example applications are presented to demonstrate the capabilities of the method.

Presented as Paper 88-0522 at the AIAA 26th Aerospace Sciences Meeting, Reno, NV, Jan. 11–14, 1988; received April 20, 1988; revision received July 26, 1988.

*Associate Professor, Department of Aeronautical and Astronautical Engineering. Member AIAA.

†Graduate Research Assistant, Department of Aeronautical and Astronautical Engineering. Student Member AIAA.

Method

The approach consists of a set of numerical conformal mappings of a surface patch in the physical space onto a domain in a two-dimensional parametric space. An elliptic grid generation technique is applied on this domain and the resulting grid points are mapped back onto the original surface. The use of the projection reduces the three-dimensional surface grid generation problem to a two-dimensional field grid generation problem and provides for a significant degree of grid control.

The surface grids consist of quadrilateral cells with rectangular topologies. However, the method could be applied individually to a set of adjacent subdomains of a more complicated topology. In this case, the overall smoothness of the grid would be ensured by proper communication across the subdomain boundaries.

Surface Definition

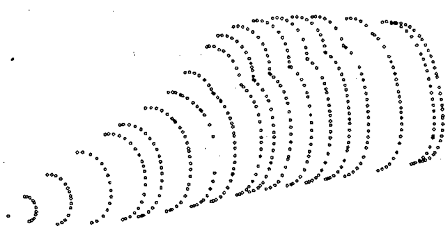
The input format defining the surface patch is a series of cuts, each consisting of a string of points on the surface (Fig. 1a). The cuts do not have to be planar or parallel as long as they do not cross each other. The number of points on each cut is arbitrary and generally different for each cut.

The input geometry is given to a preprocessor that interpolates out a constant number of points on each cut. This gives the input the form of a structured grid with quadrilateral elements. In order to preserve the given information about the surface, all of the input points are retained. New points are interpolated where needed to make the total number on each cut the same. The result of the preprocessor is a network of elements on the surface, with nodes denoted by $(x,y,z)_{mn}$. The index m specifies a cutting line and n denotes a point on the cut (Fig. 1b). A damped cubic spline is used for this interpolation to avoid oscillations while maintaining smoothness. The details of the damped spline are discussed later in a separate section.

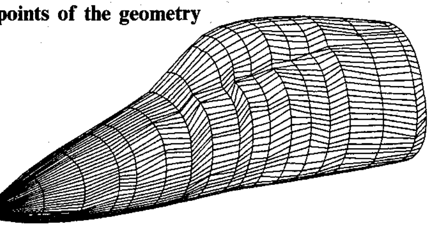
Parametric Projection

The central element of the method is the parametric projection. Its primary function is to provide a bridge between the three-dimensional physical space and the two-dimensional parametric space in which a grid will be generated. The process can be visualized as a physical projection of the surface patch onto a screen by a light source.[5] The image of the surface on the screen can be described by two families of parametric coordinates. This simple projection is conformal and produces no singularities or regions of foldover in the parametric domain as long as the screen and light source are properly tailored to the geometry.

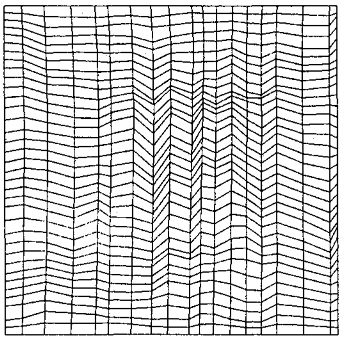
In order to generalize the projection, a geometry-adaptive projection concept was developed in which the screen and the light source are determined locally and automatically. Para-



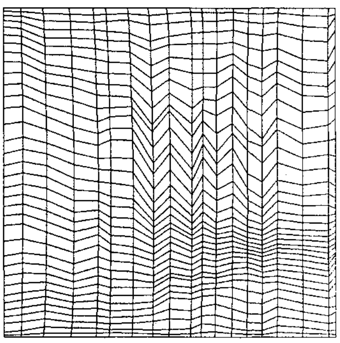
a) Input points of the geometry



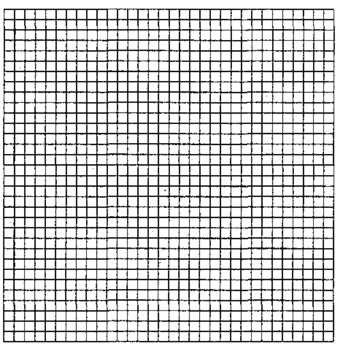
b) Surface definition with quadrilateral elements



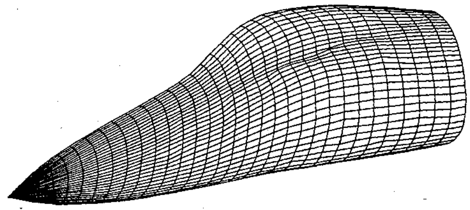
c) Parametric projection onto a unit square



d) Geometry-adaptive control in the parametric space



e) Two-dimensional grid generation



f) Surface grid generated from inverse mapping

Fig. 1 Surface grid generation procedure for an aircraft forebody.

metric coordinates (s',t') of the image of the surface on the screen are found by numerically solving a set of elliptic partial differential equations. This process is described in detail in a later section.

The result from the projection is a map between the surface points $(x,y,z)_{mn}$ in the physical space (Fig. 1b) and their images $(s',t')_{mn}$ in the parametric space (Fig. 1c). This correlation is required for the inverse mapping procedure. As seen in the figure, the parametric domain is normalized to a unit square.

Geometry-Adaptive Control

Grid control is provided by a second mapping that adjusts the grid density according to local surface properties. Two control sources are defined at the center of each element of the surface definition for separate control in each of the parametric directions.[5] The influences of the sources are superposed at each point (s',t') in the parametric space, defining new parametric coordinates (s,t),

$$s_{mn} = s'_{mn} + \sum_{k,l} K^s_{mnkl}\sigma^s_{kl}$$

$$t_{mn} = t'_{mn} + \sum_{k,t} K^t_{mnkl}\sigma^t_{kl} \tag{1}$$

where the σ_{kl} are the source strengths for the cell (k,l) and the K_{mnkl} the influence coefficients of the cell (k,l) at the point (m,n). In order to provide grid control independently in both parametric directions, the source strengths and influence coefficients are defined separately in each coordinate.

The source strength is proportional to the local surface curvature along the parametric coordinate. The influence coefficient is defined as a function of the relative position between a source point and a grid point. When the influences of the sources are added, the parametric coordinates are redistributed, as shown in Fig. 1d. As a result, the parametric coordinates are stretched more near sharp corners and in regions of high curvature. When the grid is generated on the modified parametric domain and mapped back to the surface in the physical space, the grid becomes finer in these regions. With this definition of control sources, the method is geometry adaptive. The source strengths could also be defined based on the distribution of a flow quantity, making the method solution adaptive.

Two-Dimensional Grid Generation

Once a suitable parametric representation of the surface has been obtained, the grid can be generated on the parametric domain. Any two-dimensional grid generation technique can be used for this step. An elliptic method was chosen that solves the equation

$$(\nabla\xi)^2(\boldsymbol{R}_{\xi\xi} + P\boldsymbol{R}_\xi) + 2(\nabla\xi\cdot\nabla\eta)\boldsymbol{R}_{\xi,\eta}$$
$$+ (\nabla\eta)^2(\boldsymbol{R}_{\eta\eta} + Q\boldsymbol{R}_\eta) = 0 \tag{2}$$

where $\boldsymbol{R}$ is the position vector (s,t) of the generated points in the parametric space and indices (ξ,η) are used as the independent variables. The coefficients in the equation are defined as

$$\nabla\xi = (\xi_s,\xi_t) = (t_\eta, -s_\eta)/J$$
$$\nabla\eta = (\eta_s,\eta_t) = (-t_\xi,s_\xi)/J \tag{3}$$

where J is the transformation Jacobian

$$J = s_\xi t_\eta - s_\eta t_\xi \tag{4}$$

Equation (2) is solved using a finite-difference method with Dirichlet boundary conditions. The terms P and Q are grid control terms extracted from the boundary discretization.

While any grid generation technique can be used, this particular method is chosen to maintain generality. The numerical grid generation technique allows for arbitrary shapes in defining the parametric domain, although a unit square is chosen in the present example. It also provides an additional method of grid control, through the discretization of the parametric domain boundaries. In the example of Fig. 1e, the boundaries are discretized uniformly so that no control is exercised in this step.

Inverse Mapping

The grid generation procedure gives values of the parametric coordinates at the generated grid points. Physical coordinates of these points are found with an inverse mapping that projects the generated grid back onto the original surface. This is an inverse operation of the parametric projection discussed earlier. The procedure first identifies the surface element in which a generated grid point lies. Then the physical coordinate are interpolated from the known values at the nodes of the surface element. The interpolation uses a bivariate form of the damped cubic spline mentioned in the discussion of surface definition. Sharp edges and corners are resolved and the grid points are located on the surface without overshoot. Figure 1f shows the generated grid for the aircraft forebody configuration defined in Fig. 1a.

Damped Bicubic Spline

In the present method, a spline-fitting technique is needed for both the surface definition stage and the inverse mapping stage. For the surface definition, the input geometry is preprocessed to define the surface as a network of quadrilateral elements. In the inverse mapping, the physical coordinates of the grid points are extracted from the map between parametric and physical coordinates defined in the parametric mapping procedure. These processes require both curve- and surface-fitting techniques.

There are two major difficulties in surface fitting. One comes from the two-dimensional nature of a surface. In polynomial spline fitting, an appropriate set of basis functions should be chosen in number equal to the number of data points in each piecewise segment. For curve fitting, this requirement can be easily satisfied. For surface fitting, however, the choice of basis functions is not simple nor apparent in matching with the number of data points. Furthermore, there is no guarantee for the existence and uniqueness of polynomial interpolants in surface fitting as there is in curve fitting.

As an example, consider Fig. 2a where a surface is subdivided into quadrilateral elements, each with four data points. A natural choice for the polynomial representation of each element would be the use of bilinear basis functions. That is,

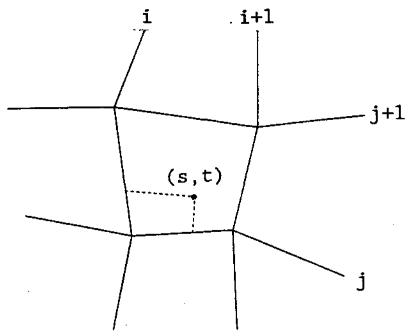
$$c(s,t) = a_1 + a_2 s + a_3 t + a_4 st \tag{5}$$

However, this choice may contain the problem of existence and uniqueness as mentioned above. In the present study, therefore, the bilinear expression is chosen in the transformed space, as shown in Fig. 2b. The quadrilateral element is mapped into a square in the transformed space, representing an interior point (s,t) with a point (ξ,η). The surface interpolation then becomes

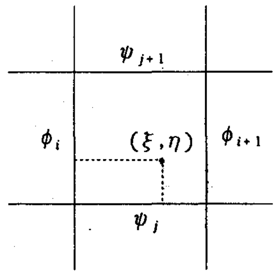
$$c(s,t) = c(\xi,\eta) = [(1-\xi)\phi_i(\eta) + \xi\phi_{i+1}(\eta)] \times [(1-\eta)\psi_j(\xi) + \eta\psi_{j+1}(\xi)] \tag{6}$$

where ϕ_i and ψ_j are defined along the boundaries where one-dimensional polynomials are used for curve fitting. This approach can eliminate the weaknesses in surface fitting by replacing the surface fitting with a bilinear interpolation of curve fittings.

Another issue with surface fitting is accuracy. Sometimes linear splines are acceptable, but often a higher-order approximation is needed for smoothness. Higher-order polynomials, however, can lead to unwanted oscillations, especially near sharp edges. Neither form is acceptable nor versatile in practice.



a) Quadrilateral element in the parametric space



b) Corresponding cell in the transformed space

Fig. 2 Bivariate interpolation for surface fitting.

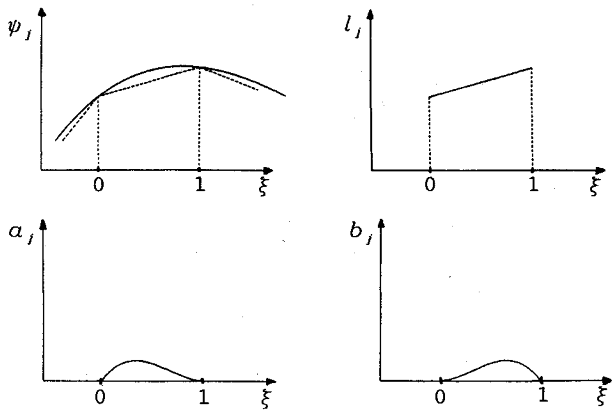


Fig. 3 Damped cubic spline.

In order to lessen this problem, a damped cubic spline is developed that controls oscillation while still giving a smooth fit to the data. Between any two data points, the spline is constructed from a linear term connecting the points and two cubic terms multiplied by damping factors (Fig. 3). For example, the spline ψ_j with the independent variable ξ in Eq. (6) is

$$\psi_j(\xi) = l_j(\xi) + \alpha_j a_j(\xi) + \beta_j b_j(\xi) \tag{7}$$
$$0 \leqslant \xi \leqslant 1$$

where l_j is the linear term, a_j and b_j the cubics, and α_j and β_j the damping factors. As seen in Fig. 3, the linear term connects the data between the endpoints of the interval to be fitted. The cubics are defined with vanishing values at the endpoints. Each cubic has zero slope at one endpoint and a finite slope at the other. The finite slopes are matched with those of a natural spline so that the spline in Eq. (7) becomes a natural spline when there is no damping.

The damping functions are defined as inverse exponential functions of the distance from the endpoints. The functions α_j and β_j are of unit value at the points $\xi = 0$ and $\xi = 1$, respectively, decaying with distance from the points. This maintains slope continuity at the data points, while suppressing the overshoot between them. The rate of decay is proportional to the local curvature of the data. Near sharp corners, the spline is highly damped and becomes more linear. Where the data are gradually changing, the spline behaves like a cubic without damping. The overall degree of damping can be selected through user input as desired. Figure 4 shows some examples of the damped spline compared to a natural cubic spline for different test cases. The damped cubic resolves the sharp corners well and gives results very similar to those of a natural cubic in smooth regions.

Numerical Projection Using Local Screen

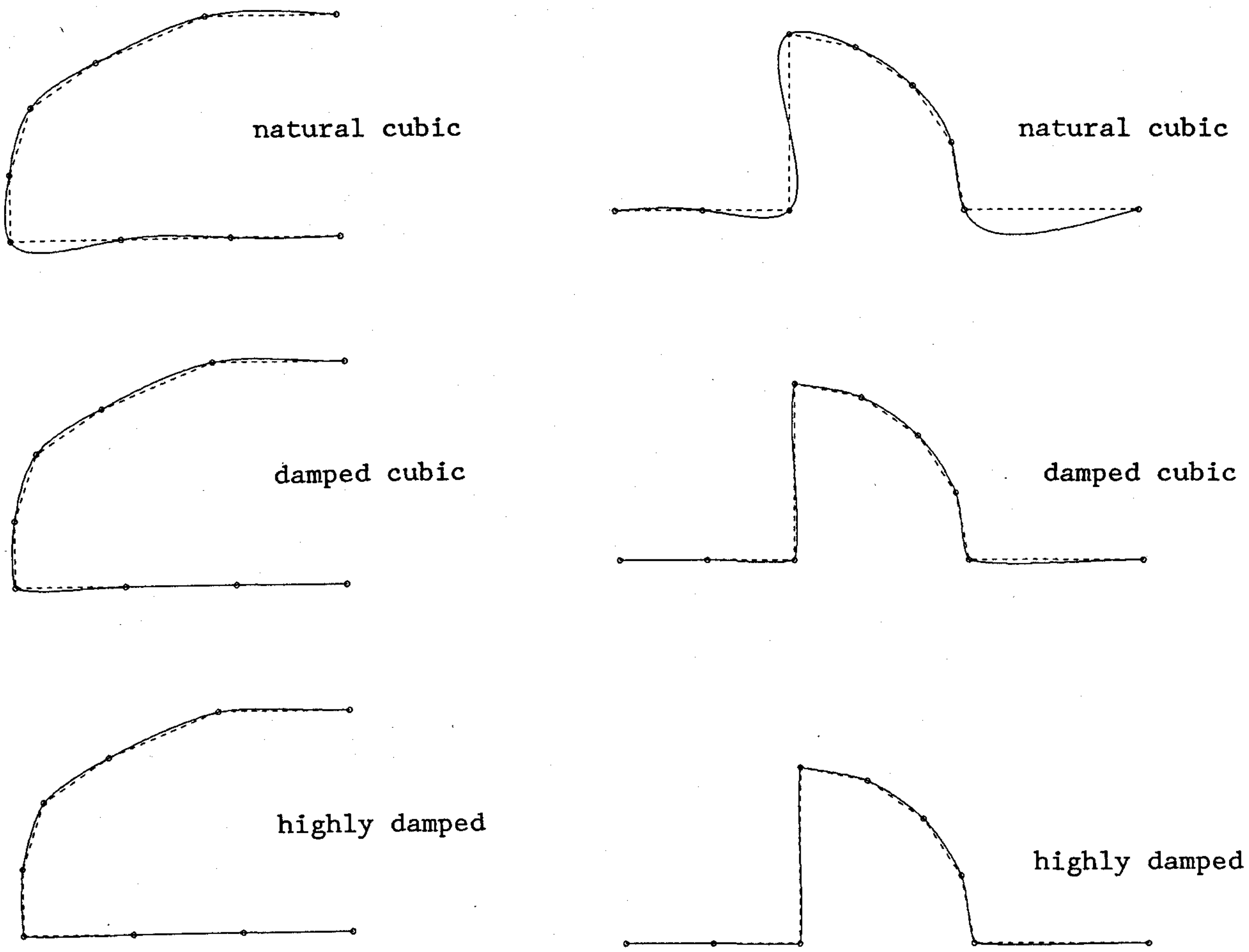
In the parametric projection process, a surface patch is mapped into a unit square in the parametric space. Three-dimensional coordinates (x,y,z) of the surface points are mapped into two-dimensional coordinates, say (s,t). The mapping should be conformal and free of singularities and should include the effects of surface characteristics such as slope and curvature. The parametric projection process is divided into three steps: 1) definition of a screen that reflects the surface characteristics well, 2) projection of the surface onto the screen, and 3) extraction of the parametric coordinates from the image of the surface.

Figure 5 illustrates the projection concept in a one-dimensional case. Consider a curved line approximated by only three line segments. The values of the parametric coordinate s corresponding to the two-dimensional physical coordinates (x,y) are to be determined at points 1–4. If the parametric coordinates are assigned based on locations of the image points on a global flat screen (Fig. 5a), the mapping does not properly represent the relationships between the points on the curve and may result in foldover. If a smooth curved screen is defined as in Fig. 5b and the parametric coordinates are assigned based on locations of the images on this screen, the resulting mapping will produce a good distribution of parametric coordinates. However, it is difficult to define a global screen that reflects the geometrical characteristics well. Therefore, the local screen projection concept is adopted as in Fig. 5c. In this case, a screen is defined at each point based on the local geometry. Then, the point and its neighbors are projected to the screen. Relationships between the points are established from their images and an elliptic differential equation,

$$s_{pp} = 0 \tag{8}$$

where p is the local screen coordinate. With Dirichlet boundary conditions given for the values of s at both boundary points 1 and 4, the parametric coordinates at interior points are obtained by solving the boundary value problem.

The surface mapping procedure is an extension of the above curve mapping. At each surface point, a local screen is defined by averaging the four neighboring surface elements containing the point. Then the nine nodes of the elements are projected onto the screen along its normal direction. A local orthogonal coordinate system (p,q) is established on each screen to obtain



a) Case 1 b) Case 2

Fig. 4 Use of damped cubic spline for curve fitting.

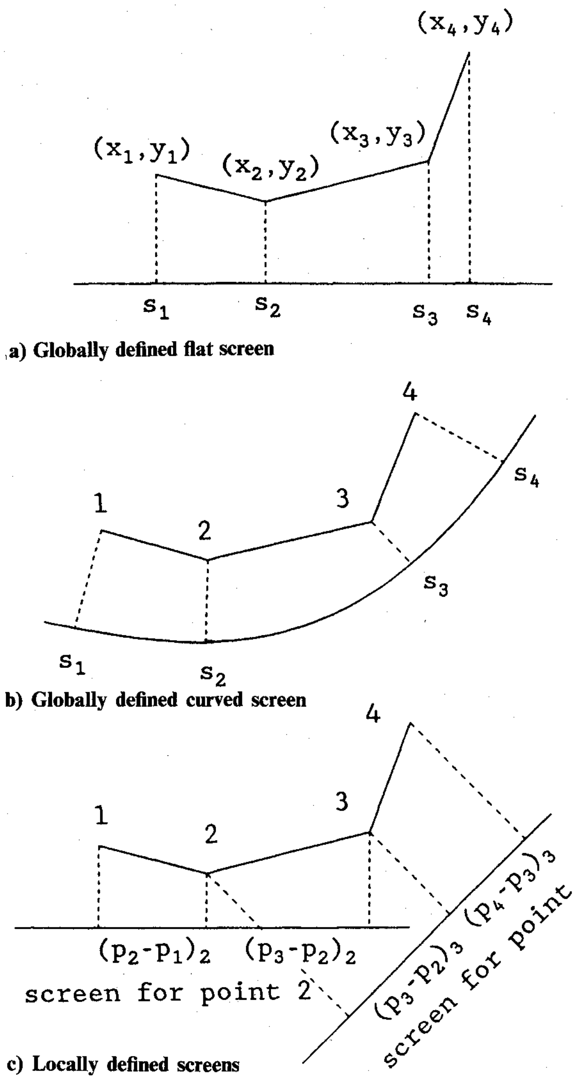


Fig. 5 Parametric projection of a curve.

the relationships of the center point with its neighbors. The parametric coordinates (s,t) are obtained from the solution of a set of elliptic partial differential equations,

$$(s_{pp} + As_p) + (s_{qq} + Bs_q) = 0$$
$$(t_{pp} + At_p) + (t_{qq} + Bt_q) = 0 \qquad (9)$$

with the local screen coordinates (p,q) as independent variables. The above equations are very similar to the two-dimensional elliptic grid generation equations. The coefficients A and B are the control parameters defined along boundaries to provide another control option. Dirichlet boundary conditions are used, with the boundary discretizations obtained from the one-dimensional curve mapping described above. This projection procedure guarantees the conformality of the mapping for any arbitrary configuration.

Examples

In this section, the results of several test cases are discussed. The examples involve different issues in surface grid generation and demonstrate the capability of the developed method. Not all of the available grid control options are used in the demonstration. Some cases include the effects of interactive controls through user-supplied input.

The first example is a hemisphere on a planar surface (Fig. 6). This case involves mixed topologies of rectangular and spherical shapes in one configuration, each of which has its own natural coordinate system. Another difficulty with the configuration comes from the circular edge of intersection. The method resolves these difficulties and produces a well-distributed grid. The grid points are most dense in the intersection region, with the grid lines conforming to the circular intersection edge.

Figure 7 shows the surface grid around a step with a sharp three-dimensional corner. The grid is most dense in the regions near the corner singularities and the sharp folds of the surface are captured well. The damped spline avoids overshoot in the interpolations of the inverse mapping. The next example, an overhanging platform (Fig. 8), is a more extreme case of the previous one. This case exhibits the versatility of the numerical projection concept using the local screen definition.

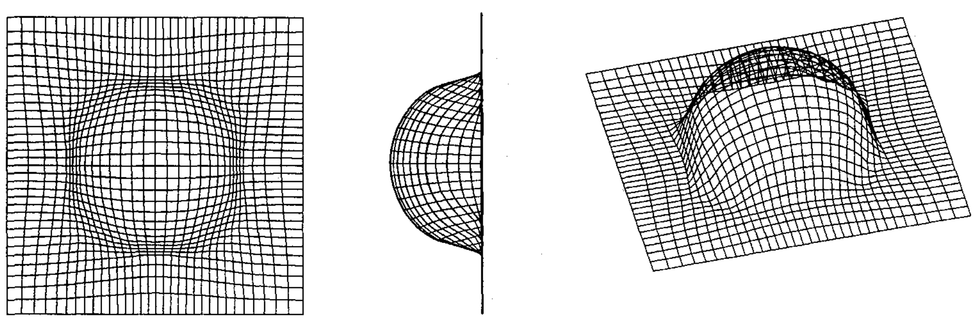


Fig. 6 Surface grid for a hemisphere.

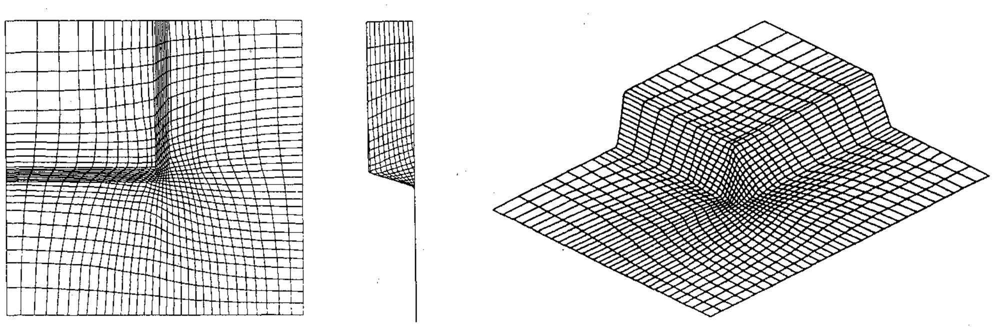


Fig. 7 Surface grid for a two-sided step.

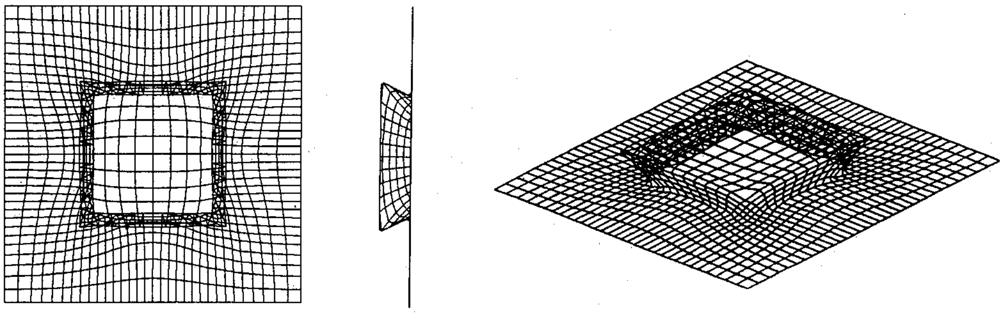


Fig. 8 Surface grid for an overhanging platform.

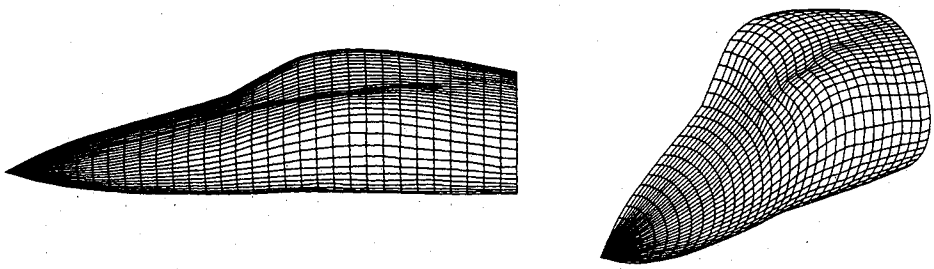


Fig. 9 Surface grid for an aircraft forebody.

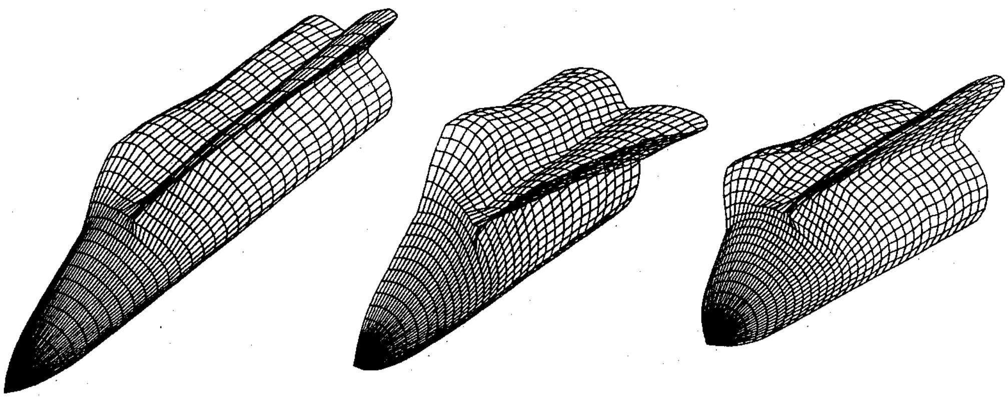


Fig. 10 Surface grid for a forebody with strake.

The method is also applied to two realistic aircraft forebody configurations. Figure 9 shows different views of the body used in Fig. 1 to visualize the steps of the method. Considering the arbitrary nature of the input shown in Fig. 1a, the resulting grid exhibits desirable qualities. The grid point distribution is smooth, with grid lines attracted to the intersection of the canopy and fuselage where the surface slope is discontinuous. Figure 10 shows a more complex configuration containing a strake along the side of the fuselage. The resulting grid lines adapt well to the leading edge of the strake as well as to the junctions of the strake and fuselage.

Conclusions

The surface grid generation technique is shown to be effective for a variety of configurations. The parametric projection allows for complex three-dimensional surfaces to be represented in a two-dimensional parametric space. It also provides the means for adaptive control without complicating the actual grid generation process. The examples demonstrate the range of applicability of the method to geometric test cases as well as to practical engineering configurations.

The implementation of the concepts are aided by two specific developments. The damped cubic spline improves the accuracy of interpolations used in both the input processor and the inverse mapping, capturing sharp corners where they exist while giving a smooth fit to the data elsewhere. The numerical projection using the local screen concept makes the method applicable to arbitrary configurations and less dependent on input surface definition.

Acknowledgments

The authors are indebted to Dr. Raymond R. Cosner from McDonnell Aircraft Company for help with this work and suggestions for test geometries. We are also grateful to Tedd R. Pierce of the Department of Aeronautical and Astronautical Engineering, University of Illinois, for help with graphics.

References

[1]Thompson, J. F., Warsi, Z. U. A., and Mastin, C. W., "Numerical Grid Generation," North-Holland, New York, 1985, pp. 237–250.

[2]Thomas, P. D., "Construction of Composite Three-Dimensional Grids from Subregion Grids Generated by Elliptic Systems," AIAA Paper 81-0996, June 1981.

[3]Takagi, T. and Miki, K., "Numerical Generation of Boundary-Fitted Curvilinear Coordinate Systems for Arbitrarily Curved Surfaces," *Journal of Computational Physics*, Vol. 58, March 1985, pp. 67–79.

[4]Eiseman, P. R., "Alternating Direction Adaptive Grid Generation," AIAA Paper 83-1937, July 1983.

[5]Lee, K. D. and Loellbach, J. M., "Numerical Generation of Surface Grids on Arbitrary Three-Dimensional Surface Patches," *Numerical Methods in Laminar and Turbulent Flow*, Vol. 5, Pt. 2, 1987, pp. 1126–1137.