JOURNAL OF COMPUTATIONAL PHYSICS **123,** 394–401 (1996)
ARTICLE NO. 0032

Computational Conformal Mapping for Surface Grid Generation

AHMED KHAMAYSEH* AND C. WAYNE MASTIN†

Los Alamos National Laboratory, Los Alamos, New Mexico 87545 and †NASA Langley Research Center, Hampton, Virginia 23681

Received January 17, 1995; revised June 20, 1995

The paper describes the development and application of a new approach for formulating an elliptic generation system on parametrically defined surfaces. The present derivation of the surface equations proceeds in two steps: First, conformal mapping of smooth surfaces onto rectangular regions is utilized to derive a first-order system of partial differential equations analogous to Beltrami's system for quasi-conformal mapping of planar regions. Second, a general elliptic generation system for three-dimensional surfaces, including forcing functions, is formulated based on Beltrami's system and quasi-conformal mapping. The resulting elliptic system is solved using an iterative method on arbitrary surfaces represented analytically by rational B-splines. The overall effect of this approach is a reliable and versatile elliptic method for generating and improving surface grids. Examples will be presented to demonstrate the application of the method in constructing practical grids. © 1996 Academic Press, Inc.

1. INTRODUCTION

For the past decade or so, there has been a great deal of interest in the field of numerical grid generation as an essential element of the numerical solution of partial differential equations on arbitrary regions. Not only the grid must be generated for the region of interest, but grid quality in terms of smoothness, skewness, and orthogonality must be achieved which can maintain the same level of accuracy as the accuracy of the method used in solving the physical problem (e.g., the Navier–Stokes equations).

Conformal mappings have been used to generate orthogonal boundary-fitted coordinates in two-dimensions for solving various problems in simply connected regions (see, e.g., Ryskin and Leal [13] and Tamamidis and Assanis [15]). In particular, Mastin [10], Thompson *et al.* [17], and Winslow [21] have developed their elliptic generation systems based on harmonic mapping borrowed from the properties of conformal transformation between the physical and the computational regions to produce smooth grids.

Indeed, an efficient method for constructing curvilinear coordinates is to let the coordinates be the solutions of an elliptic system of partial differential equations with Dirichlet or Neumann boundary conditions on all boundaries. The choice of elliptic models to generate curvilinear coordinates is well known; see Thompson *et al.* [18]. Since elliptic partial differential equations determine a function in terms of its values on the entire closed boundary of a region, such a system can be used to generate the interior values of a surface grid from the values on the sides. An important property is the inherent smoothness in the solutions of elliptic systems. As a consequence of smoothing, slope discontinuities on the boundaries are not propagated into the field. Variational methods can also be used in the construction of curvilinear coordinate systems; see Knupp and Steinberg [7]. These methods have been extended to surface grid generation by Knupp [8] and Saltzman [14].

Early progress on the generation of surface grids using elliptic methods has been made by Warsi [19, 20] and Thomas [16]. The proposed generation system and the surface equations obtained by Warsi were based on the *fundamental theory of surfaces* from differential geometry, which says that if there exists a surface, then the surface coordinates must satisfy the formulas of Gauss and Weingarton. On the other hand, the same generation system was derived by Thomas based on the three-dimensional Poisson's partial differential equations. However, implementation of these systems to construct surface grids on arbitrary surfaces in a general purpose grid generation code remains a major issue due to the lack of common mathematical representation for both standard analytical shapes and free-form surfaces.

The main focus of the present work is to employ conformal mapping to develop a new methodology for generating curvilinear coordinates on arbitrary surfaces. We begin our study by exploring the basic properties of conformal mapping from two-dimensional regions to three-dimensional surfaces. After the basic concepts have been introduced, we begin our discussion of the relationship and the connection between the derived equations based on conformal mapping and those derived for planar regions based on quasi-conformal mapping to formulate the generation equations for the surface case. In later sections the numerical procedure of mapping and constructing the physical, as well as the parametric coordinates, is described.

The effectiveness of the method is demonstrated via the generation of smoothed surface grids on real-world geometries.

2. CONFORMAL MAPPING ON SURFACES

A conformal mapping of a smooth bounded surface onto a rectangular region can be constructed by establishing a mapping from a square region of the computational plane onto the surface which is orthogonal and has a constant aspect ratio. The advantage of conformal coordinates are well known; for example, problems involving heat conduction, ideal fluid flow, and electric fields can be solved as easily on the surface as they can on a rectangular region in the cartesian plane. Also this approach brings out the intrinsic orthogonality and uniformity properties that are inherent in a grid generated by such a mapping. Another advantage of using conformal coordinates on parametric surfaces for solving elliptic partial differential equations is the fact they permit solutions of differential equations on surfaces with the same ease as they can be solved on a rectangle in the cartesian plane.

In classical differential geometry, a surface $\mathscr{S}$ is viewed as a mapping from $\mathbb{R}^2$ to $\mathbb{R}^3$. Consequently, parametric surfaces (physical variables) are defined in terms of parametric variables. In grid generation, parametric variables are defined in terms of computational variables, i.e.,

$$\begin{gathered}\mathbf{s} = (x, y, z) = (x(u, v), y(u, v), z(u, v)),\\ (u, v) = (u(\xi, \eta), v(\xi, \eta)).\end{gathered} \tag{2.1}$$

A rectilinear grid in the computational square generates a curvilinear grid in the parametric square which maps to a curvilinear grid on the surface. Thus, a uniform grid in the computational space generates a curvilinear grid on the surface. The elliptic system of partial differential equations which defines the transformation between computational variables and parametric variables is related to conformal mappings on surfaces.

A surface grid generated by the conformal mapping of a rectangle onto the surface $\mathscr{S}$ is orthogonal and has a constant aspect ratio. These two conditions can be expressed mathematically as the system of equations

$$\mathbf{s}_\xi \cdot \mathbf{s}_\eta = 0 \tag{2.2}$$

$$\mathscr{F}|\mathbf{s}_\xi| = |\mathbf{s}_\eta|, \tag{2.3}$$

where $\mathscr{F}$ is the grid aspect ratio. These two equations can be rewritten as

$$x_\xi x_\eta + y_\xi y_\eta + z_\xi z_\eta = 0 \tag{2.4}$$

$$\mathscr{F}^2(x_\xi^2 + y_\xi^2 + z_\xi^2) = x_\eta^2 + y_\eta^2 + z_\eta^2. \tag{2.5}$$

Using the chain rule for differentiation, the physical derivatives are expanded as

$$\begin{aligned}
\mathbf{s}_\xi &= \mathbf{s}_u u_\xi + \mathbf{s}_v v_\xi,\\
\mathbf{s}_\eta &= \mathbf{s}_u u_\eta + \mathbf{s}_v v_\eta,\\
\mathbf{s}_{\xi\xi} &= \mathbf{s}_{uu} u_\xi^2 + 2\mathbf{s}_{uv} u_\xi v_\xi + \mathbf{s}_{vv} v_\xi^2 + \mathbf{s}_u u_{\xi\xi} + \mathbf{s}_v v_{\xi\xi},\\
\mathbf{s}_{\xi\eta} &= \mathbf{s}_{uu} u_\xi u_\eta + \mathbf{s}_{uv}(u_\xi v_\eta + u_\eta v_\xi) + \mathbf{s}_{vv} v_\xi v_\eta + \mathbf{s}_u u_{\xi\eta} + \mathbf{s}_{vv\xi\eta},\\
\mathbf{s}_{\eta\eta} &= \mathbf{s}_{uu} u_\eta^2 + 2\mathbf{s}_{uv} u_\eta v_\eta + \mathbf{s}_{vv} v_\eta^2 + \mathbf{s}_u u_{\eta\eta} + \mathbf{s}_v v_{\eta\eta}.
\end{aligned} \tag{2.6}$$

Thus, the system of equations in (2.4) and (2.5) is equivalent to

$$\begin{aligned}
&x_u^2 u_\xi u_\eta + x_u x_v(u_\xi v_\eta + u_\eta v_\xi) + x_v^2 v_\xi v_\eta\\
&\quad + y_u^2 u_\xi u_\eta + y_u y_v(u_\xi v_\eta + u_\eta v_\xi) + y_v^2 v_\xi v_\eta\\
&\quad + z_u^2 u_\xi u_\eta + z_u z_v(u_\xi v_\eta + u_\eta v_\xi) + z_v^2 v_\xi v_\eta = 0
\end{aligned} \tag{2.7}$$

and

$$\begin{aligned}
&\mathscr{F}^2(x_u^2 u_\xi^2 + 2x_u x_v u_\xi v_\xi + x_v^2 v_\xi^2 + y_u^2 u_\xi^2 + 2y_u y_v u_\xi v_\xi + y_v^2 v_\xi^2\\
&\quad + z_u^2 u_\xi^2 + 2z_u z_v u_\xi v_\xi + z_v^2 v_\xi^2)\\
&\quad = x_u^2 u_\eta^2 + 2x_u x_v u_\eta v_\eta + x_v^2 v_\eta^2 + y_u^2 u_\eta^2\\
&\quad\quad + 2y_u y_v u_\eta v_\eta + y_v^2 v_\eta^2 + z_u^2 u_\eta^2 + 2z_u z_v u_\eta v_\eta + z_v^2 v_\eta^2.
\end{aligned} \tag{2.8}$$

The above equations are combined to give the complex equation

$$\begin{aligned}
&(x_u^2 + y_u^2 + z_u^2)(\mathscr{F} u_\xi + i u_\eta)^2\\
&\quad + 2(x_u x_v + y_u y_v + z_u z_v)(\mathscr{F} u_\xi + i u_\eta)(\mathscr{F} v_\xi + i v_\eta)\\
&\quad + (x_v^2 + y_v^2 + z_v^2)(\mathscr{F} v_\xi + i v_\eta)^2 = 0.
\end{aligned} \tag{2.9}$$

This equation can be put into a compact form

$$\bar{\mathbf{g}}_{11} Z^2 + 2\bar{\mathbf{g}}_{12} Z W + \bar{\mathbf{g}}_{22} W^2 = 0, \tag{2.10}$$

where

$$\begin{aligned}
\bar{\mathbf{g}}_{11} &= \mathbf{s}_u \cdot \mathbf{s}_u = x_u^2 + y_u^2 + z_u^2,\\
\bar{\mathbf{g}}_{12} &= \mathbf{s}_u \cdot \mathbf{s}_v = x_u x_v + y_u y_v + z_u z_v,\\
\bar{\mathbf{g}}_{22} &= \mathbf{s}_v \cdot \mathbf{s}_v = x_v^2 + y_v^2 + z_v^2,\\
Z &= \mathscr{F} u_\xi + i u_\eta, \quad W = \mathscr{F} v_\xi + i v_\eta.
\end{aligned} \tag{2.11}$$

Solving the quadratic equation (2.10) either for Z or W, say Z, we have

$$Z = \frac{-\bar{g}_{12} \pm \sqrt{\bar{g}_{12}^2 - \bar{g}_{11}\bar{g}_{22}}}{\bar{g}_{11}} W \tag{2.12}$$

or in terms of u and v

$$\mathscr{F}u_\xi + iu_\eta = \frac{-\bar{g}_{12} \pm i\bar{J}}{\bar{g}_{11}}(\mathscr{F}v_\xi + iv_\eta), \tag{2.13}$$

where $\bar{J} = \sqrt{\bar{g}}$ and $\bar{g} = \bar{g}_{11}\bar{g}_{22} - \bar{g}_{12}^2$ is the Jacobian of the mapping from the parametric space to the surface. Multiply the right-hand side of the above equation and equate the real and the imaginary parts to get the two real equations

$$\mathscr{F}u_\xi = -\frac{\bar{g}_{12}}{\bar{g}_{11}}\mathscr{F}v_\xi \pm \frac{\bar{J}}{\bar{g}_{11}}v_\eta \tag{2.14}$$

$$u_\eta = \pm\frac{\bar{J}}{\bar{g}_{11}}\mathscr{F}v_\xi - \frac{\bar{g}_{12}}{\bar{g}_{11}}v_\eta. \tag{2.15}$$

The above system of equations can be expressed in the form of the first-order elliptic system

$$\mathscr{F}u_\xi = av_\eta - bu_\eta \tag{2.16}$$

$$\mathscr{F}v_\xi = bv_\eta - cu_\eta, \tag{2.17}$$

where

$$\begin{aligned} a &= -\frac{\bar{g}_{22}}{\pm\bar{J}}, \\ b &= \frac{\bar{g}_{12}}{\pm\bar{J}}, \\ c &= -\frac{\bar{g}_{11}}{\pm\bar{J}}. \end{aligned} \tag{2.18}$$

Note that $ac - b^2 = 1$ which is sufficient for ellipticity. The sign ± needs to be chosen such that the Jacobian

$$J = u_\xi v_\eta - u_\eta v_\xi > 0. \tag{2.19}$$

The quantities $\bar{g}_{11} \geq 0$ and $\bar{g}_{22} \geq 0$ by definition, so choosing the negative sign will make $a \geq 0$ and $c \geq 0$. From Eqs. (2.15) and (2.16), we see that $\mathscr{F}J = \mathscr{F}(u_\xi v_\eta - u_\eta v_\xi) = av_\eta^2 - 2bu_\eta v_\eta + cu_\eta^2$ and $b^2 = ac - 1$, implies that $b = \sqrt{ac-1} < \sqrt{ac}$ and $\mathscr{F}J > av_\eta^2 - 2\sqrt{ac}u_\eta v_\eta + cu_\eta^2 = (\sqrt{a}v_\eta - \sqrt{c}u_\eta)^2 \geq 0$ and, hence, $J > 0$.

3. FORMULATION OF THE ELLIPTIC GENERATOR

In this section, a complete formulation of a second-order quasi-linear elliptic system will be established, based on conformal mapping. First, we start with a brief overview of quasi-conformal mapping of planar regions. In two dimensions there is a close relationship between grid generation and quasi-conformal mapping. This can be seen by establishing a homeomorphism

$$\psi(u, v) = \xi(u, v) + i\eta(u, v) \tag{3.1}$$

which maps the (u, v) in the parameter space onto (ξ, η) in the computational space and the real and the imaginary parts of ψ satisfy Beltrami's system of equations

$$\mathscr{M}\eta_v = d\xi_u + e\xi_v \tag{3.2}$$

$$-\mathscr{M}\eta_u = e\xi_u + f\xi_v, \tag{3.3}$$

where d, e, and f are functions of u and v with $a, c > 0$ and satisfy the equation $df - e^2 = 1$. The quasi-conformal quantity $\mathscr{M}$ is invariant and often referred to as the module or the aspect ratio of the region of consideration. For further study of the theory and application of quasi-conformal mappings, we refer to Ahlfors [1] and Renetl [12].

It is the system of Eqs. (3.2) and (3.3) which forms the basis of general elliptic grid generation for the two-dimensional case; see Mastin and Thompson [9]. An earlier approach was proposed by Belinskii *et al.* [3] and Godunov and Prokopov [5] to handle the problem of quasi-conformal mappings to construct curvilinear grids.

The procedure for the surface case proceeds in a similar fashion. The first task is to invert the system of Eqs. (2.16) and (2.17); that is, to develop an equivalent system so that the computational variables (ξ, η) are dependent variables and the parametric variables (u, v) become the independent variables. Let the computational variables be

$$\begin{aligned} \xi &= \xi(u, v), \\ \eta &= \eta(u, v). \end{aligned} \tag{3.4}$$

Assuming that ξ and η are twice continuously differentiable and the Jacobian of the inverse transformation $J = u_\xi v_\eta - u_\eta v_\xi$ is nonvanishing in the region under consideration. Then the metrics ("$u_\xi, u_\eta, v_\xi, v_\eta$") and ("$\xi_u, \xi_v, \eta_u, \eta_v$") are uniquely related by

$$\begin{aligned} \xi_u &= \frac{v_\eta}{J}, \quad \xi_v = -\frac{u_\eta}{J}, \\ \eta_u &= -\frac{v_\xi}{J}, \quad \eta_v = \frac{u_\xi}{J}. \end{aligned} \tag{3.5}$$

Using these quantities in Eqs. (2.15) and (2.16) so that the parametric variables become the independent variables, the system can be expressed either in the form

$$\mathscr{F}\eta_v = a\xi_u + b\xi_v \tag{3.6}$$

$$-\mathscr{F}\eta_u = b\xi_u + c\xi_v \tag{3.7}$$

or

$$\xi_u = \mathscr{F}(c\eta_v + b\eta_u) \tag{3.8}$$

$$-\xi_v = \mathscr{F}(a\eta_u + b\eta_v). \tag{3.9}$$

The first-order elliptic systems are analogous to Beltrami's system of equations for quasi-conformal mapping of planar regions. The relation between quasi-conformal transformation and elliptic grid generation can be seen by utilizing the system of Eqs. (3.6)–(3.9). Indeed, the above systems can be easily uncoupled to the second-order elliptic system,

$$\begin{aligned} &a\xi_{uu} + 2b\xi_{uv} + c\xi_{vv} + (a_u + b_v)\xi_u \\ &\quad + (b_u + c_v)\xi_u = 0 \end{aligned} \tag{3.10}$$

$$\begin{aligned} &a\eta_{uu} + 2b\eta_{uv} + c\eta_{vv} + (a_u + b_v)\eta_u \\ &\quad + (b_u + c_v)\eta_v = 0. \end{aligned} \tag{3.11}$$

From these equations, it follows that the computational variables ξ and η are solutions of a second-order elliptic system (3.10) and (3.11). In fact, ξ and η are solutions of the following second-order linear elliptic system with $\Phi = \Psi = 0$,

$$\bar{\mathbf{g}}_{22}\xi_{uu} - 2\bar{\mathbf{g}}_{12}\xi_{uv} + \bar{\mathbf{g}}_{11}\xi_{vv} + (\Delta_2 u)\xi_u + (\Delta_2 v)\xi_v = \Phi \tag{3.12}$$

$$\bar{\mathbf{g}}_{22}\eta_{uu} - 2\bar{\mathbf{g}}_{12}\eta_{uv} + \bar{\mathbf{g}}_{11}\eta_{vv} + (\Delta_2 u)\eta_u + (\Delta_2 v)\eta_v = \Psi, \tag{3.13}$$

the Beltramians $\Delta_2 u$ and $\Delta_2 v$ have been multiplied by $\bar{J}^2$ as follows:

$$\Delta_2 u = \bar{J}(a_u + b_v) = \bar{J}\left[\frac{\partial}{\partial u}\left(\frac{\bar{\mathbf{g}}_{22}}{\bar{J}}\right) - \frac{\partial}{\partial v}\left(\frac{\bar{\mathbf{g}}_{12}}{\bar{J}}\right)\right] \tag{3.14}$$

$$\Delta_2 v = \bar{J}(b_u + c_v) = \bar{J}\left[\frac{\partial}{\partial v}\left(\frac{\bar{\mathbf{g}}_{11}}{\bar{J}}\right) - \frac{\partial}{\partial u}\left(\frac{\bar{\mathbf{g}}_{12}}{\bar{J}}\right)\right]. \tag{3.15}$$

It is this system which forms the basis of the elliptic methods for generating surface grids. The source terms (or control functions), Φ and Ψ, are added to allow control over the distribution of grid points on the surface. In the computation of a surface grid, the points in the computational space are given and the points in the parametric space must be computed. Therefore, in implementation of a numerical grid generation scheme, it is convenient to interchange variables again so that the computational variable ξ and η are the independent variables. Introducing (3.5) in (3.12) and (3.13), the transformation is reduced to the system of equations

$$-Au_\xi + Bv_\xi = \Psi, \tag{3.16}$$

$$Au_\eta - Bv_\eta = \Phi, \tag{3.17}$$

where

$$\begin{aligned} A &= \frac{\mathbf{g}_{22}v_{\xi\xi} - 2\mathbf{g}_{12}v_{\xi\eta} + \mathbf{g}_{11}v_{\eta\eta}}{\bar{J}J^3} - \frac{\Delta_2 v}{\bar{J}J}, \\ B &= \frac{\mathbf{g}_{22}u_{\xi\xi} - 2\mathbf{g}_{12}u_{\xi\eta} + \mathbf{g}_{11}u_{\eta\eta}}{\bar{J}J^3} - \frac{\Delta_2 u}{\bar{J}J}, \\ \mathbf{g}_{11} &= \mathbf{s}_\xi \cdot \mathbf{s}_\xi = \bar{\mathbf{g}}_{11}u_\xi^2 + 2\bar{\mathbf{g}}_{12}u_\xi v_\xi + \bar{\mathbf{g}}_{22}v_\xi^2, \\ \mathbf{g}_{12} &= \mathbf{s}_\xi \cdot \mathbf{s}_\eta = \bar{\mathbf{g}}_{11}u_\xi u_\eta + \bar{\mathbf{g}}_{12}(u_\xi v_\eta + u_\eta v_\xi) + \bar{\mathbf{g}}_{22}v_\xi v_\eta, \\ \mathbf{g}_{22} &= \mathbf{s}_\eta \cdot \mathbf{s}_\eta = \bar{\mathbf{g}}_{11}u_\eta^2 + 2\bar{\mathbf{g}}_{12}u_\eta v_\eta + \bar{\mathbf{g}}_{22}v_\eta^2. \end{aligned} \tag{3.18}$$

Solving (3.16) and (3.17) for A and B, we have

$$\begin{aligned} A &= -\frac{1}{J}[\Phi v_\xi + \Psi v_\eta], \\ B &= -\frac{1}{J}[\Phi u_\xi + \Psi u_\eta]. \end{aligned} \tag{3.19}$$

From the above equations, we see that u and v are solutions of the quasi-linear elliptic system

$$\mathbf{g}_{22}(u_{\xi\xi} + \mathscr{P}u_\xi) - 2\mathbf{g}_{12}u_{\xi\eta} + \mathbf{g}_{11}(u_{\eta\eta} + \mathscr{Q}u_\eta) = J^2\Delta_2 u, \tag{3.20}$$

$$\mathbf{g}_{22}(v_{\xi\xi} + \mathscr{P}v_\xi) - 2\mathbf{g}_{12}v_{\xi\eta} + \mathbf{g}_{11}(v_{\eta\eta} + \mathscr{Q}v_\eta) = J^2\Delta_2 v, \tag{3.21}$$

where

$$\begin{aligned} \mathscr{P} &= \frac{\bar{J}J^2}{g_{22}}\Phi, \\ \mathscr{Q} &= \frac{\bar{J}J^2}{g_{11}}\Psi. \end{aligned} \tag{3.22}$$

To this end, We have demonstrated the use of conformal mapping onto surfaces to derive the basic elliptic generation system given by (3.20) and (3.21). As mentioned previously, the same equations to generate surface coordinates have been derived by Warsi [19, 20] from the formulas of Gauss and Weingarten from differential geometry and by Thomas [16] from three-dimensional Poisson's partial difference equations. It should be noted that the same system is used to generate the surface (face) grids for a volume grid in three dimensions such that the computational coordinates (ξ, η, ζ) are taken in an orthogonal cyclic order.

A few remarks on this system are in order. It is evident that the system is applied to the parametric variables u and v. However, the metric coefficients are solved simultaneously on both the parametric and physical regions, where the right-hand side terms are solved on the physical region. Also, the solution of the system requires an initial grid to be constructed.

4. IMPLEMENTATION AND NUMERICAL RESULTS

One of the challenging issues in the area of elliptic surface grid generation is not only that the grid needs to be smoothed, but to ensure that the resulting grid points must stay on the surface. With this objective in mind, the efficient approach of constructing a smooth grid is to work in the parametric space rather than in the physical surface. However, there are some disadvantages associated with this approach. The differential equations become more complicated and contain two sets of derivatives, the derivatives of the physical variables with respect to the parametric variables ("$x_u, x_v, y_u, y_v, z_u, z_v, x_{uu}, x_{uv}, x_{vv}, ...$") and the derivatives of the parametric variables with respect to the computational variables ("$u_\xi, u_\eta, v_\xi, v_\eta, u_{\xi\xi}, u_{\xi\eta}, u_{\eta\eta}, ...$").

Consequently, the application of the elliptic surface grid generator has been limited to a small spectrum of geometries due to the lack of a robust mathematical representation of arbitrary shapes. To overcome this problem, there has been a move towards non-uniform rational B-splines (NURBS) in grid generation systems. This move follows current trends in geometric modeling and computer-aided geometric design (CAGD). NURBS are becoming the *de facto* industry standard for geometry representation but are still a rather new "tool" to the grid generation community. NURBS allow the representation of nearly all geometries relevant for aircraft and automobile design. They are particularly advantageous for elliptic grid generation due to their numerical stability and efficient (derivative) evaluation (see Bartels *et al.* [2] and Farin [4]).

The proposed grid generation method assumes that the surface for which the grid is to be constructed is given as a NURBS surface,

$$s(u, v) = \frac{\sum_{j=0}^{n}\sum_{i=0}^{m} \omega_{i,j}\mathbf{d}_{i,j}N_i^k(u)N_j^l(v)}{\sum_{j=0}^{n}\sum_{i=0}^{m} \omega_{i,j}N_i^k(u)N_j^l(v)}, \tag{4.1}$$

defined by

- two orders k and l,
- control points $\mathbf{d}_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j})$, $i = 0, ..., m$; $j = 0, ..., n$,
- real weights $\omega_{i,j}$, $i = 0, ..., m$; $j = 0, ..., n$,
- a set of real u-knots, $\{u_0, ..., u_{m+k} \mid u_i \leq u_{i+1}, i = 0, ..., (m + k - 1)\}$,
- a set of real v − knots, $\{v_0, ..., v_{n+l} \mid v_j \leq v_{j+1}, j = 0, ..., (n + l - 1)\}$,
- B-spline basis functions $N_i^k(u)$, $u \in [u_i, u_{i+k}]$, $i = 0, ..., m$,
- B-spline basis functions $N_j^l(v)$, $v \in [v_j, v_{j+l}]$, $j = 0, ..., n$, and
- surface segments $s_{i,j}(u, v)$, $u \in [u_i, u_{i+1}]$, $i = (k - 1), ..., m$, $v \in [v_j, v_{j+1}]$, $j = (l - 1), ..., n$.

The advantage of using a NURBS-based geometry definition is the ability to represent both standard analytic shapes (e.g., conics, quadrics, surfaces of revolution, etc.) and free-form curves and surfaces. Therefore, both analytic and free-form shapes are represented precisely, and a unified database can store both. Another potential advantage of using NURBS, is the fact that positional, as well as derivative, information of surfaces, required for the elliptic grid generator, can be evaluated analytically. For a detailed discussion of NURBS curves and surfaces and their relevant derivatives (e.g., $\mathbf{s}_u, \mathbf{s}_v, \mathbf{s}_{uu}, \mathbf{s}_{uv}, \mathbf{s}_{vv}$) required for the generation of smooth grids we refer to Piegl [11].

The current approach which we have adopted utilizes a three-stage grid generation methodology. The first stage is the representation of the initial geometry as a computer-aided design (CAD) surface, where CAD systems typically represent the surfaces of a certain geometry with a set of structured points or patches. The second stage is the analytical representation of the CAD surface as a NURBS surface. The third stage is the generation of structured surface grid.

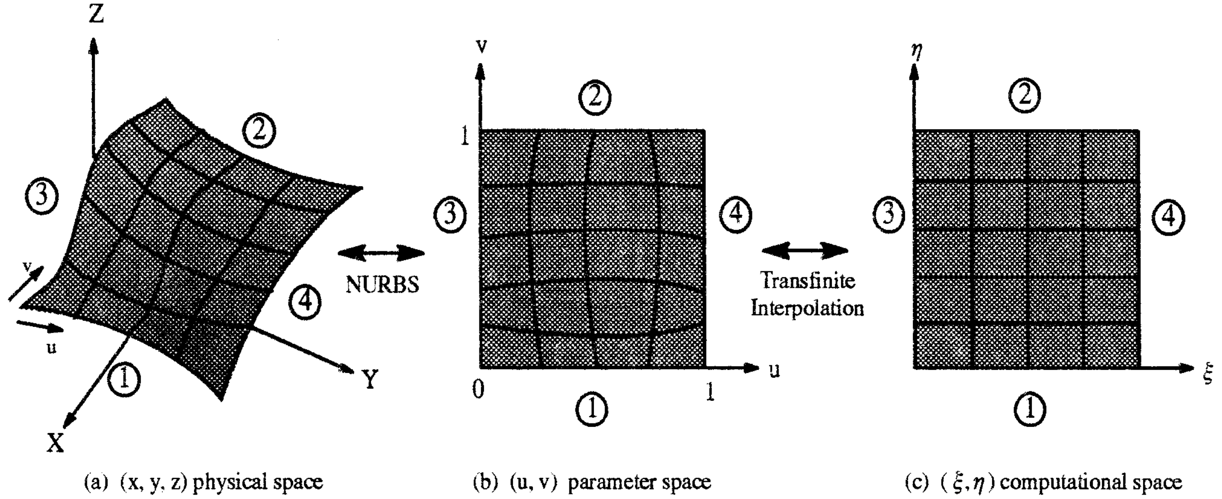
It is, of course, the purpose of the present work to construct the desired high quality surface grid on a given geometry. To accomplish this task, the grid generation process proceeds in two steps. First, an algebraic grid is constructed by interpolation from the boundaries of the surface, and the grid is then enhanced, and possibly modified in other ways, by employing the elliptic system (3.20) and (3.21) on the initial grid with prescribed boundary constraints.

The process of algebraic grid generation for a parametrically defined surface is divided into three steps: forward mapping, grid generation, and backward mapping. The forward mapping is the mapping of the three-dimensional physical surface (i.e., a NURBS surface) to a two-dimensional parametric rectangle. Once the forward mapping is completed, the grid is generated in the parametric space, using transfinite interpolation (TFI) and then mapped back into physical space (see Fig. 1.). It is important to understand that the type of parametrization does not change the shape of the surface, but it does change the distribution of the points on it. A "poor" parametrization may cause the surface grid to be highly skewed.

The methodology of constructing an $m \times n$ algebraic grid on a physical surface starts either with the specification of the boundary distribution along the physical boundaries (then mapped to the parametric boundaries), or with the distribution of grid points directly on the parametric boundaries. The parametric values used are denoted as

$$\mathscr{M}_u : \{u_{1,j} < u_{2,j} < \cdots < u_{m,j} \mid j = 1, n\}, \tag{4.2}$$

$$\mathscr{M}_v : \{v_{i,1} < v_{i,2} < \cdots < v_{i,n} \mid i = 1, m\}. \tag{4.3}$$



(a) (x, y, z) physical space

(b) (u, v) parameter space

(c) (ξ, η) computational space

FIG. 1. Mapping from physical ("x, y, z") to computational ("ξ, η") space via parametric ("u, v") space.

The first step in constructing a parametric grid requires a correspondence to be established between the parametric space (u, v) and the computational space (ξ, η) via a distribution space (s, t). A uniform spacing in computational space is used, i.e., $\xi = 1, 2, ..., m$ and $\eta = 1, 2, ..., n$. Therefore, the computational grid $\mathcal{G}_{\xi\eta}$ is defined by integer coordinates (i, j) where $i = \xi$ and $j = \eta$. In the next step, the boundary distribution of the (u, v) space is mapped to the boundaries of the (s, t) space using for example arc length mapping.

Let

$$\mathcal{M}_s\!:\{0 = s_{1,j} < s_{2,j} < \cdots < s_{m,j} = 1 \mid j = 1, n\} \quad (4.4)$$

be a partitioning of the s-domain [0, 1] and let

$$\mathcal{M}_t\!:\{0 = t_{i,1} < t_{i,2} < \cdots < t_{i,n} = 1 \mid i = 1, m\} \quad (4.5)$$

be a partitioning of the t-domain [0, 1]. The distribution grid $\mathcal{G}_{st} = \mathcal{M}_s \otimes \mathcal{M}_t$ partitioning $[0, 1] \times [0, 1]$ is generated by solving for the interior points assuming the distribution points on opposite boundaries are joined by straight lines.

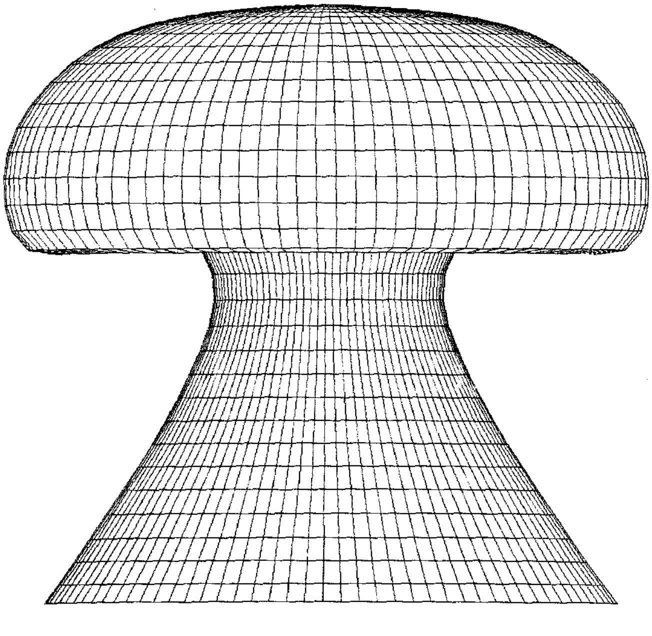


FIG. 2. Conformal surface grid on a singular geometry.

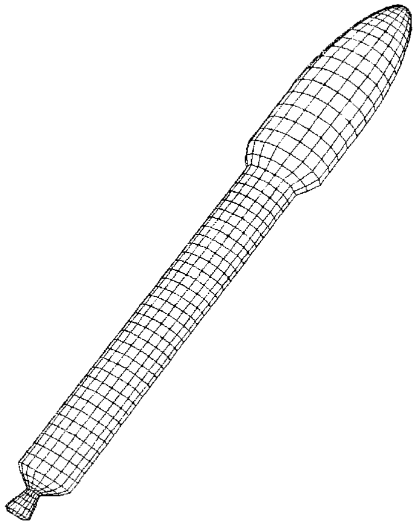


FIG. 3. Conformal surface grid on a rocket geometry.

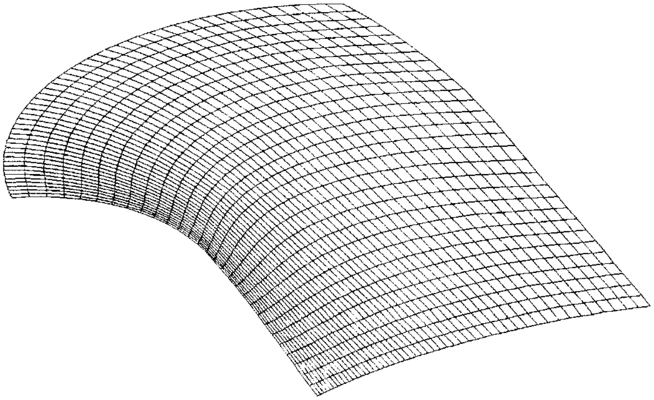


FIG. 4. Conformal surface grid on a concave–convex geometry.

Once the distribution grid is computed, by some standard technique such as tensor product interpolation or TFI, the parametric grid $\mathcal{G}_{uv} = \mathcal{M}_u \otimes \mathcal{M}_v$ is constructed, where the mapping establishes a one to one correspondence between the parametric values $(u_{i,j}, v_{i,j})$ and the computational nodes $(s_{i,j}, t_{i,j})$. Finally, the surface grid $\mathcal{G}_{xyz}$ is constructed using the NURBS representation of the physical coordinates

$$\mathbf{s}_{i,j} = \mathbf{s}(u_{i,j}, v_{i,j}), \quad i = 1, ..., m; j = 1, ..., n. \tag{4.6}$$

The next stage in the process of surface grid generation consists of employing the elliptic grid generator (3.20) and (3.21) using the initial grid as a background grid. The elliptic grid generator is implemented using finite difference discretization and Gauss–Seidel strategy.

The elliptic system may be applied to the interior grids and may preserve the original distribution of grid points or redistribute points based upon the choice of the ''control functions'' $\mathcal{P}$ and $\mathcal{Q}$ which are commonly used in adaptive grid generation. The control functions are evaluated either directly from the initial algebraic grid, or by interpolation from the boundary point distributions and then smoothed. In both cases smoothing is done in the directions of the control function coordinates. This allows the relative spacing of the algebraic grid to be retained but on a smoother grid from the elliptic system. Orthogonality of the grid may be imposed along certain boundary components of the physical region. Boundary orthogonality can be achieved through Neumann boundary conditions which allow the boundary points to float along the boundary of the surface. Alternatively, the control functions can be determined to provide orthogonality at boundaries with specified normal spacing. In case of periodic geometries, reflective boundary conditions are used. These options have been addressed in a detailed manner by Khamayseh [6] and incorporated in the national grid project (NGP) system.

To illustrate the method, we conclude this section by presenting four numerical examples with different options of control functions and boundary orthogonalities. In all cases the surface grid is generated on a NURBS surface using the elliptic system (3.20) and (3.21) with appropriate control functions and boundary orthogonality. The first example, Fig. 2, shows a surface grid constructed on a singular geometry with the option of zero control functions, i.e., $\mathcal{P} = \mathcal{Q} = 0$, and the boundary points were allowed to move to achieve orthogonality, i.e., Neumann-type orthogonality. Using this option produces a uniform smooth grid, but the initial boundary distribution has not been preserved.

Figure 3 shows results of the method applied to a real-world surface geometry. A grid on the surface of a rocket was constructed using zero control functions in the elliptic smoother. Neumann orthogonality was imposed by adjusting the location of the nodes along the surface boundaries.

Figure 4 demonstrates the use of the method with control functions computed at the boundaries and then projected into the interior of the grid. In particular, this is advantageous in the case of the initial grid contains cells with near zero areas, i.e., $J \approx 0$, where we cannot solve for the control functions from the system (3.20) and (3.21). Initial spacing is preserved and Dirichlet orthogonality is applied by fixing the boundary distribution and allowing the interior grid points to move using control functions along the boundary segments to produce orthogonal grids.

Finally, we present a grid computed on a periodic geometry (see Fig. 5). The elliptic grid was constructed with zero

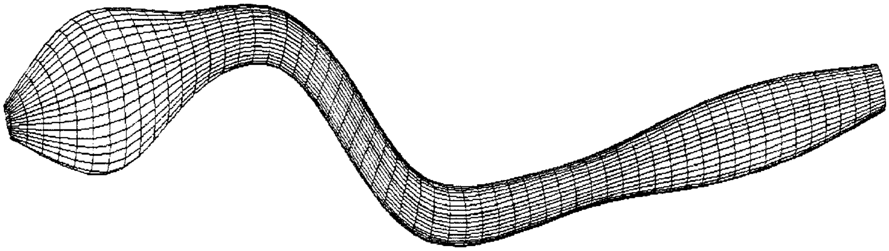


FIG. 5. Conformal surface grid on a periodic geometry.

control functions which results again in a uniform grid; reflective boundary conditions were used to ensure a smooth transition along the branch cut.

CONCLUDING REMARKS

A comprehensive study of surface grid generation has been undertaken with the intention of providing an efficient methodology to construct optimal grids on complex geometries. Basic properties of conformal mapping are utilized to improve methods of generating grids from the solution of elliptic systems of partial differential equations. Current success with NURBS-based geometric representation has proven to be an excellent candidate for generating surface grids on arbitrary geometries. The effectiveness of such strategy would be most useful when the parametrization of the surface is such that interpolation of parametric values does not give a satisfactory grid because of a poorly parametrized surface. Since this situation arises frequently when surfaces are defined by CAD packages, the capability to smooth and improve surface grids is essential in any state-of-the-art grid generation code.

ACKNOWLEDGMENTS

This research was supported by NASA Langley Research Center under Grant No. NSG 1577.

REFERENCES

1. L. V. Ahlfors, *Lectures on Quasi-conformal Mappings* (Van Nostrand, New York, 1966).
2. R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling* (Morgan Kaufmann, Los Altos, CA, 1987).
3. P. P. Belinskii, S. K. Godunov, Yu. B. Ivanov, and I. K. Yanenko, *USSR Comput. Math. Math. Phys.* **15,** 133 (1975).
4. G. Farin, *Curves and Surfaces for Computer-Aided Geometric Design,* 3rd ed. (Academic Press, San Diego, CA, 1993).
5. S. K. Godunov and G. P. Prokopov, *USSR Comput. Math. Math. Phys.* **7,** 89 (1967).
6. A. Khamayseh, Ph.D. dissertation, Mississippi State University, May 1994 (unpublished).
7. P. Knupp and S. Steinberg, *The Fundamentals of Grid Generation* (CRC Press, Boca Raton, FL, 1993).
8. P. Knupp, *Appl. Math. Comput.* **59,** 41 (1993).
9. C. W. Mastin and J. F. Thompson, *SIAM J. Sci. Stat. Comput.* **5,** 305 (1984).
10. C. W. Mastin, ''Elliptic Grid Generation and Conformal Mapping,'' in *Mathematical Aspects of Numerical Grid Generation,* edited by Jose E. Castillo (SIAM, Philadelphia, 1991).
11. L. A. Piegl, ''Rational B-spline Curves and Surfaces for CAD and Graphics'', in *State of the Art in Computer Graphics,* edited by D. F. Rogers and R. A. Earnshaw (Springer-Verlag, New York, 1991).
12. H. Renelt, *Elliptic Systems and Quasi-Conformal Mappings* (Wiley, New York, 1988).
13. G. Ryskin and L. G. Leal, *J. Comput. Phys.* **50,** 71 (1983).
14. J. Saltzman, *J. Comput. Phys.* **63,** 1 (1986).
15. P. Tamamidis and D. N. Assanis, *J. Comput. Phys.* **94,** 437 (1991).
16. P. D. Thomas, *AIAA J.* **81,** 996 (1981).
17. J. F. Thompson, F. C. Thames, and C. W. Mastin, *J. Comput. Phys.* **15,** 299 (1974).
18. J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin, *Numerical Grid Generation: Foundations and Applications* (North-Holland, New York, 1985).
19. Z. U. A. Warsi, *J. Comput. Phys.* **64,** 82 (1986).
20. Z. U. A. Warsi, *AIAA J.* **6,** 1140 (1990).
21. A. M. Winslow, *J. Comput. Phys.* **1,** 149 (1966).