

INTERACTIVE HIGH QUALITY GRID GENERATION

Ning Lu and Peter R. Eiseman
Program Development Corporation
300 Hamilton Avenue, Suite 409
White Plains, New York 10601

ABSTRACT

We generate high quality grids interactively on a computer workstation through optimization for some simple yet commonly encountered geometric shapes.

1. INTRODUCTION

Algebraic and elliptic grid generation are the two most widely used methods for generating grids. Algebraic grid generation is known for fast interactive grid generation. The grid smoothness and lack of skewness, however, are usually not assured. On the other hand, elliptic grid generation can generate smooth grids but the time it takes to generate them is usually too long for interactive response. It is naturally desirable to combine the advantages of both methods and to generate high quality grids (ref. 1) fast and efficiently on a computer workstation. This requires people to strike a proper balance between the grid generation speed and the grid quality. We have pursued this idea for some simple yet commonly encountered geometric shapes. Many of them have a symmetry that can be used. We have been able to obtain high quality grids fast and interactively. Our method is to analyze each case carefully and to take full advantage of the region symmetry (if any) so that optimization is needed only for a part of geometric region under consideration. We have combined various grid generation methods, such as a fast elliptic grid generation technique (combining the control point form of algebraic grid generation with elliptic grid generation, ref. 2) and a transfinite interpolation with orthogonal boundary terms, to achieve an optimization for each individual case. After such an optimization, we have been able to generate excellent single- and multi-block grids quickly. This achievement enables us to generate certain single- and multi-block grids interactively with reasonable response even on modestly powered workstations.

2. GRID WITHIN A RHIOMBUS

A simple transfinite interpolation (TFI) will not generate an orthogonal grid in general. A more general TFI with boundary derivative terms can improve grid orthogonality. In certain cases, grid orthogonality can be achieved along edges of a face grid or on surfaces of a volume grid. This, in general, requires a careful analysis for each geometry region encountered. In this section, we present an example of carrying out such a detailed analysis.

We consider a surface grid bounded within a quadrilateral. Since each of four boundary edges is a straight line, a bicubically blended interpolation formula (i.e., with derivative information) can be used,

$$\mathbf{r}(u, v) = \underline{B}^T(u) \underline{S} \underline{B}(v), \quad (1)$$

where \underline{S} is a 4×4 matrix

$$\underline{S} = \begin{pmatrix} \mathbf{r}(0,0) & \mathbf{r}(0,1) & \mathbf{r}_v(0,0) & \mathbf{r}_v(0,1) \\ \mathbf{r}(1,0) & \mathbf{r}(1,1) & \mathbf{r}_v(1,0) & \mathbf{r}_v(1,1) \\ \mathbf{r}_u(0,0) & \mathbf{r}_u(0,1) & \mathbf{r}_{uv}(0,0) & \mathbf{r}_{uv}(0,1) \\ \mathbf{r}_u(1,0) & \mathbf{r}_u(1,1) & \mathbf{r}_{uv}(1,0) & \mathbf{r}_{uv}(1,1) \end{pmatrix}, \quad (2)$$

and \underline{B} is a 4-component vector

$$\underline{B}(u) = \begin{pmatrix} B_1(u) \\ B_2(u) \\ B_3(u) \\ B_4(u) \end{pmatrix} = \begin{pmatrix} 1 - 3u^2 + 2u^3 \\ 3u^2 - 2u^3 \\ u - 2u^2 + u^3 \\ -u^2 + u^3 \end{pmatrix}. \quad (3)$$

In Eq. (1), u and v are parametric variables. The parametric region of $0 \leq u \leq 1$, $0 \leq v \leq 1$ corresponds to the area within the quadrilateral. In Eq. (2), $\mathbf{r}(0,0)$, $\mathbf{r}(1,0)$, $\mathbf{r}(0,1)$, and $\mathbf{r}(1,1)$ are four corners of the quadrilateral (i.e., the surface grid), four \mathbf{r}_u 's are partial derivatives with respect to the variable u at four corners, four \mathbf{r}_v 's are partial derivatives with respect to the variable v at four corners, and \mathbf{r}_{uv} 's are second-order mixed derivatives (also called twists) at four corners. In Eq. (3), the blending functions $B_j(u)$ have the properties: $B_1(u) + B_2(u) = 1$, $B_j(0) = \delta_{j1}$, $B_j(1) = \delta_{j2}$, $dB_j(0)/du = \delta_{j3}$, $dB_j(1)/du = \delta_{j4}$.

In grid generation, the positions of four corners $\mathbf{r}(0,0)$, $\mathbf{r}(1,0)$, $\mathbf{r}(0,1)$, and $\mathbf{r}(1,1)$ are known. However, the values of 8 first-order derivatives and 4 second-order derivatives are not given. With a proper choice of these derivatives, one can generate a better grid than those generated by the simple TFI or, for a quadrilateral, by the simple bilinear interpolation. The idea is to require that the grid direction at the boundary edges point to a desired direction. For example, on the $v = 0$ edge, the grid direction going inside the quadrilateral can be easily found from Eqs. (1)–(3) to be

$$\mathbf{r}_v(u, 0) = B_1(u)\mathbf{r}_v(0,0) + B_2(u)\mathbf{r}_v(1,0) + B_3(u)\mathbf{r}_{uv}(0,0) + B_4(u)\mathbf{r}_{uv}(1,0). \quad (4)$$

By requiring it to be perpendicular to the $v = 0$ edge (a straight line here), it is possible to achieve boundary orthogonality at certain places on the $v = 0$ boundary edge.

In the following, we take a rhombus as an example to illustrate how a detailed analysis can dramatically improve the grid orthogonality. A rhombus is a special parallelogram with opposite sides and opposite angles equal but having no right angle. For a rhombus centered at origin (on the $z = 0$ plane), its four corner positions can be written as (see fig. 1)

$$\mathbf{r}(1,1) = -\mathbf{r}(0,0) = h \mathbf{j} , \quad \mathbf{r}(1,0) = -\mathbf{r}(0,1) = w \mathbf{i} . \quad (5)$$

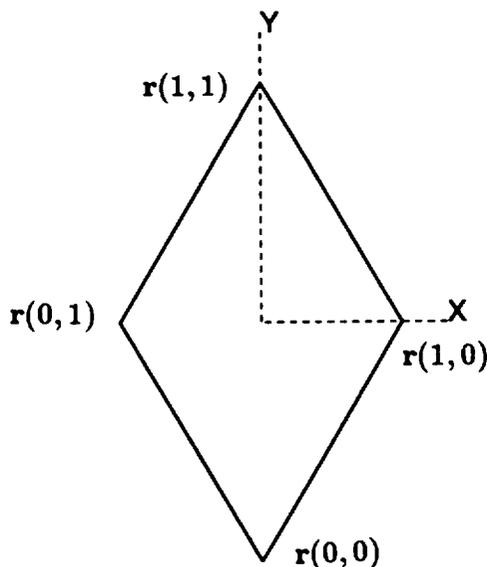


Figure 1.—A rhombus centered at origin.

From symmetry considerations, we know (i) that the magnitudes of $\mathbf{r}_u(0,0)$, $\mathbf{r}_v(0,0)$, $\mathbf{r}_u(1,1)$, and $\mathbf{r}_v(1,1)$ must be equal, and (ii) that the magnitudes of $\mathbf{r}_u(1,0)$, $\mathbf{r}_v(1,0)$, $\mathbf{r}_u(0,1)$, and $\mathbf{r}_v(0,1)$ must be equal too. In other words, we have the following relations:

$$\mathbf{r}_u(0,0) = \mathbf{r}_u(1,1) = \alpha \hat{\mathbf{u}} , \quad (6a)$$

$$\mathbf{r}_v(0,0) = \mathbf{r}_v(1,1) = \alpha \hat{\mathbf{v}} , \quad (6b)$$

$$\mathbf{r}_u(1,0) = \mathbf{r}_u(0,1) = \beta \hat{\mathbf{u}} , \quad (6c)$$

$$\mathbf{r}_v(1,0) = \mathbf{r}_v(0,1) = \beta \hat{\mathbf{v}} , \quad (6d)$$

where

$$\hat{\mathbf{u}} = (w\mathbf{i} + h\mathbf{j})/\sqrt{w^2 + h^2}, \quad \hat{\mathbf{v}} = (-w\mathbf{i} + h\mathbf{j})/\sqrt{w^2 + h^2} \quad (7)$$

are the unit vectors along the directions of $\mathbf{r}(1,0) - \mathbf{r}(0,0)$ and of $\mathbf{r}(0,1) - \mathbf{r}(0,0)$, respectively. It is clear that both α and β are positive: $\alpha > 0$ and $\beta > 0$. Also, because of symmetry, (i) twists $\mathbf{r}(0,0)$ and $\mathbf{r}(1,1)$ do not have x component whereas twists $\mathbf{r}(1,0)$ and $\mathbf{r}(0,1)$ do not have y component, and (ii) the magnitudes of the twists at each pair of opposite corners are equal but point to opposite directions. Namely, twists satisfy the relations

$$\mathbf{r}_{uv}(1,1) = -\mathbf{r}_{uv}(0,0) = a \mathbf{j}, \quad (8a)$$

$$\mathbf{r}_{uv}(1,0) = -\mathbf{r}_{uv}(0,1) = b \mathbf{i}. \quad (8b)$$

Expressions (6) and (8) are the most general form allowed (by the symmetry requirements) for a rhombus. In other words, the symmetry consideration reduces the number of independent first-order derivatives from 8 to 2, and also reduces the number of independent second-order derivatives from 4 to 2. Now we only need to decide 4 scalar quantities: α , β , a , and b . For the rhombus shown in fig. 1, Eq. (4) reduces to

$$\mathbf{r}_v(u,0) = [\alpha B_1(u) + \beta B_2(u)]\hat{\mathbf{v}} - aB_3(u) \mathbf{j} + bB_4(u) \mathbf{i}. \quad (9)$$

Without loss of generality, we assume in the following discussion that $h > w$. When $h > w$, the angle between the two unit vectors $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$, which is the internal angle at corner $\mathbf{r}(0,0)$, is less than 90° , since $\hat{\mathbf{v}} \cdot \hat{\mathbf{u}} = (h^2 - w^2)/(h^2 + w^2) > 0$. To determine the 4 quantities α , β , a , and b , we first consider an (unnormalized) orthogonality function $f(u) \equiv \mathbf{r}_v(u,0) \cdot \hat{\mathbf{u}}$, which is a function (third-order polynomial) of the parametric variable u . Its value at the two ends $\mathbf{r}(0,0)$ and $\mathbf{r}(1,0)$ of the $v = 0$ edge is positive: $f(0) = \alpha \hat{\mathbf{v}} \cdot \hat{\mathbf{u}} > 0$ and $f(1) = \beta \hat{\mathbf{v}} \cdot \hat{\mathbf{u}} > 0$. We want to force the grid to be orthogonal at the middle of the $v = 0$ edge (but not overly doing this enforcement). Specifically, we require that the orthogonality function $f(u)$ reaches its minimum value at $u = \frac{1}{2}$ and that the minimum value of the orthogonality function is zero (for the interval $0 \leq u \leq 1$). These two requirements lead to two conditions for the orthogonality function $f(u)$,

$$\frac{df(\frac{1}{2})}{du} = 0, \quad f(\frac{1}{2}) = 0. \quad (10)$$

From these two conditions, we can express the twists a and b in terms of the first-order derivatives α and β ,

$$a = \frac{(5\alpha - \beta)(h^2 - w^2)}{h\sqrt{w^2 + h^2}}, \quad (11a)$$

$$b = \frac{(5\beta - \alpha)(h^2 - w^2)}{w\sqrt{w^2 + h^2}}. \quad (11b)$$

Eliminating the twists a and b in the orthogonality function $f(u)$ with the help of Eqs. (11), $f(u)$ becomes

$$f(u) = \frac{1}{2} \hat{\mathbf{v}} \cdot \hat{\mathbf{u}} [(\alpha + \beta)(2u - 1)^2 - (\alpha - \beta)(2u - 1)^3], \quad (12)$$

which indeed reaches its minimum value of 0 at $u = \frac{1}{2}$ (for the interval $0 \leq u \leq 1$) for any positive α and β .

Next, we ask \mathbf{r}_v and \mathbf{r}_u to be orthogonal at the center of the rhombus $\mathbf{r}(\frac{1}{2}, \frac{1}{2}) = 0$. Due to symmetry, the angle between $\mathbf{r}_v(\frac{1}{2}, \frac{1}{2})$ and \mathbf{j} is always equal to the angle between $\mathbf{r}_u(\frac{1}{2}, \frac{1}{2})$ and \mathbf{j} . Thus, asking \mathbf{r}_v and \mathbf{r}_u to be orthogonal at the center of the rhombus is equivalent to requiring that $\mathbf{r}_v(\frac{1}{2}, \frac{1}{2}) \cdot \mathbf{j} = \mathbf{r}_u(\frac{1}{2}, \frac{1}{2}) \cdot (-\mathbf{i})$ and

$$\mathbf{r}_u(\frac{1}{2}, \frac{1}{2}) \cdot \mathbf{i} = \mathbf{r}_v(\frac{1}{2}, \frac{1}{2}) \cdot \mathbf{j}. \quad (13)$$

Equation (13) leads to a third relation among the derivatives α , β , a , and b ,

$$\mathbf{r}_u(\frac{1}{2}, \frac{1}{2}) \cdot (\mathbf{j} - \mathbf{i}) = \frac{h - w}{4\sqrt{w^2 + h^2}} (6\sqrt{w^2 + h^2} - \alpha - \beta) + \frac{a + b}{16} + \frac{3(\beta - \alpha)(w + h)}{8\sqrt{w^2 + h^2}}. \quad (14)$$

Substituting Eqs. (11) into Eq. (14), we can eliminate a and b and obtain an expression relating α to β ,

$$\alpha = \frac{(w + 5h)\beta + 24wh(h - w)(w^2 + h^2)^{-1/2}}{5w + h}. \quad (15)$$

Now we can still choose the value of β as it fits. There is, however, a restriction. In the case of $h > w$ under discussion, the twists a and b , as determined by Eqs. (11), should not be negative. Otherwise, the resulting grid will not look nice.

In practice, we find that a vanishing twist at the pair of obtuse angles of the rhombus, i.e.,

$$b = 0, \quad (16)$$

is a good choice. This leads to

$$\alpha = 5\beta \quad (17)$$

according to Eq. (11b). It follows from Eqs. (15) and (17) that

$$\beta = \frac{h(h - w)}{\sqrt{w^2 + h^2}}, \quad h > w. \quad (18)$$

Substituting Eqs. (17) and (18) into Eq. (11a), we find that correspondingly

$$a = \frac{24(h - w)(h^2 - w^2)}{w^2 + h^2}. \quad (19)$$

Equations (16)–(19) thus uniquely determine all 12 derivatives in Eq. (1).

Figure 2 shows three rhombuses inside a regular hexagon of side length $2w$. Two rhombuses with a pair of edges parallel to the x -axis can be obtained from the third rhombus (oriented as in fig. 1) with a simple rotation of $\pm 120^\circ$ around the corner $r(1,0)$ on $z = 0$ plane. For each rhombus in fig. 2, $h = \sqrt{3}w$ and Eqs. (18) and (19) give $\beta = 0.634w$ and $a = 8.785w$.

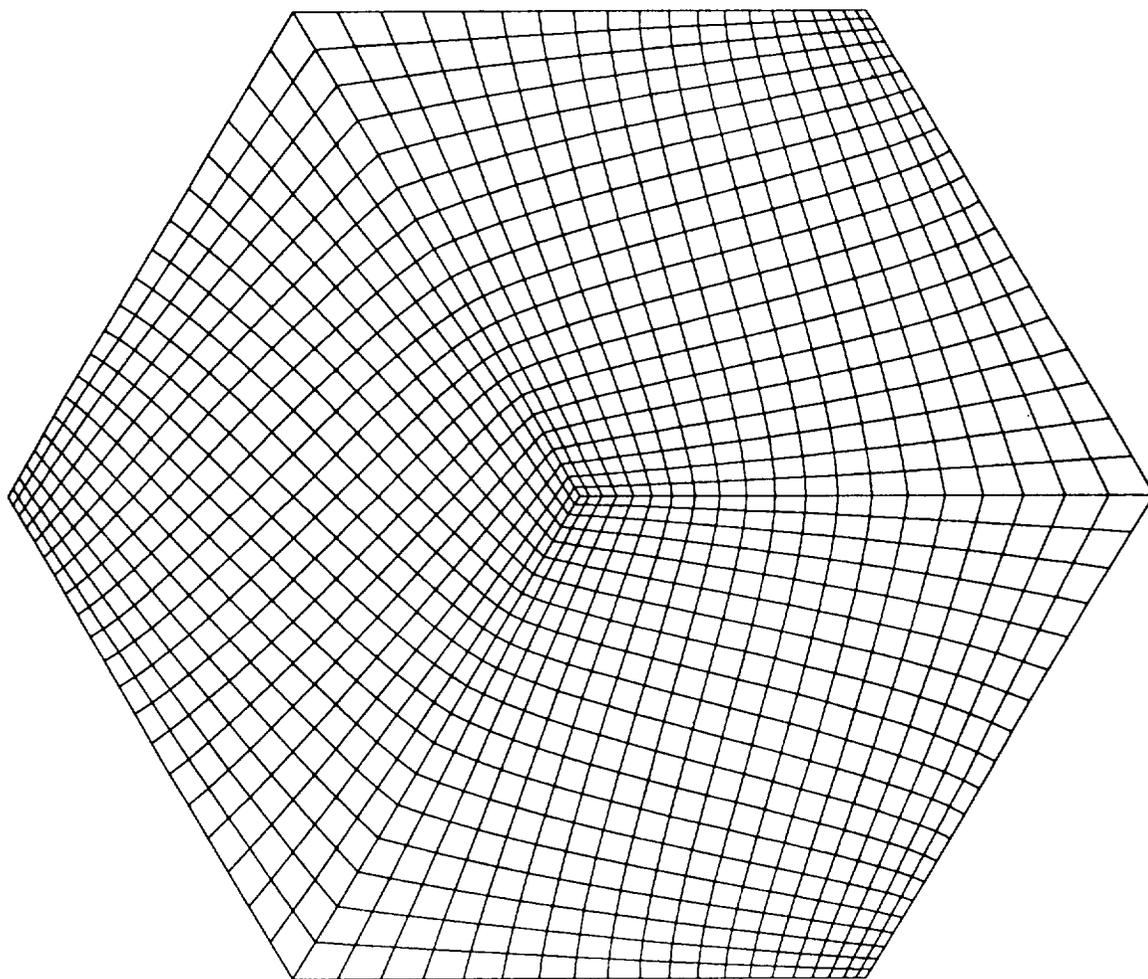


Figure 2.—A three-block grid inside a regular hexagon.

3. OTHER EXAMPLES

When the geometric region for a surface grid is no longer a quadrilateral (as in most cases), a TFI formula with boundary information terms, i.e., a bicubically blended TFI (see, for example, ref. 3), should be used to replace Eq. (1). By a careful analysis for each geometric region or shape under study, one can properly choose the directions and

magnitudes of boundary derivative terms and thus dramatically improve the grid quality and smoothness. In this way, a surface grid with much improved grid orthogonality can be generated. For a volume grid, one can also include boundary derivative terms into the often-used volume grid FTI formula and thus generate a volume grid with improved grid orthogonality. We have carried out this approach for some simple yet commonly encountered geometric shapes besides the rhombus case discussed above in section 2. Besides this pure algebraic grid generation approach, we have also combined the control point form of algebraic grid generation with elliptic grid generation (ref. 2). Here we first elliptically generate a sparse net of control points and then use the control point form of algebraic grid generation to generate a much denser grid. As a result, we generate a smooth grid with less computational time and storage space. This approach also enables us to obtain a high quality grid in an interactive environment on a computer workstation. We are incorporating these new grid generation techniques into a software product **GridPro**TM/IGgen from Program Development Corporation, so that a user will be able to readily generate high quality grids interactively on a computer workstation. Currently, a collection of boundary geometries are provided and a user can choose among them. In addition, the user can also specify the grid type wanted, such as O-type, C-type, H-type, and/or multi-block grids. The user can also specify the grid density and clustering pattern. Once the user makes a choice, our software immediately generates high quality grids. In this section we show some of the grids generated interactively using **GridPro**/IGgen.

Figure 3 shows a single-block 2D grid for a passage which has a circular extra space. Due to symmetry, we only need to carry out grid orthogonality optimization for a half, a quarter, or even one eighth of the grid.

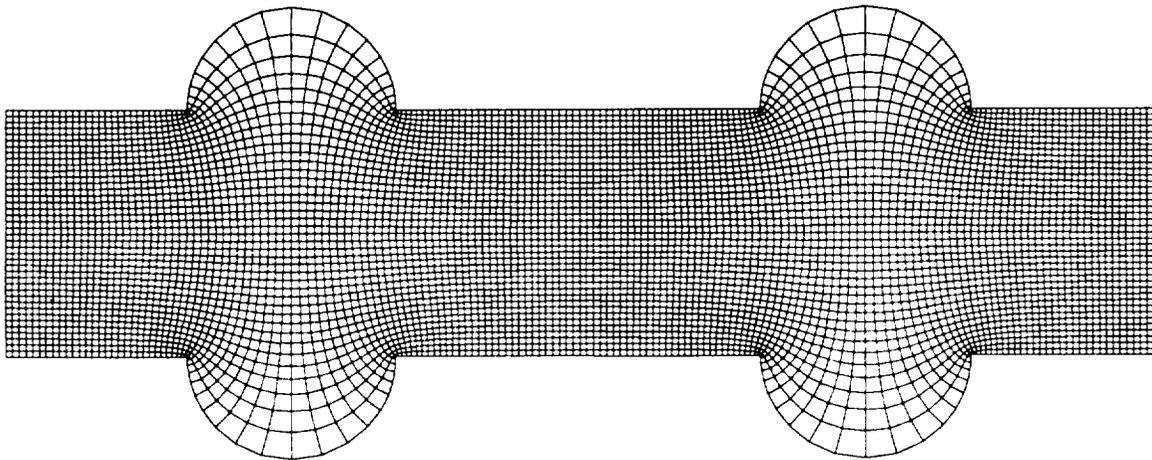


Figure 3.—A single-block 2D grid inside a 2D pipe which has symmetric circular extra space.

Figure 4 displays an H-type 2D grid passing around a cylinder whose cross section looks like a super circle. There are two blocks. One block is the mirror image of the other block about the horizontal symmetry line. For each block, there exists a vertical symmetry line. So we only need to do optimization for one quarter of the whole grid.

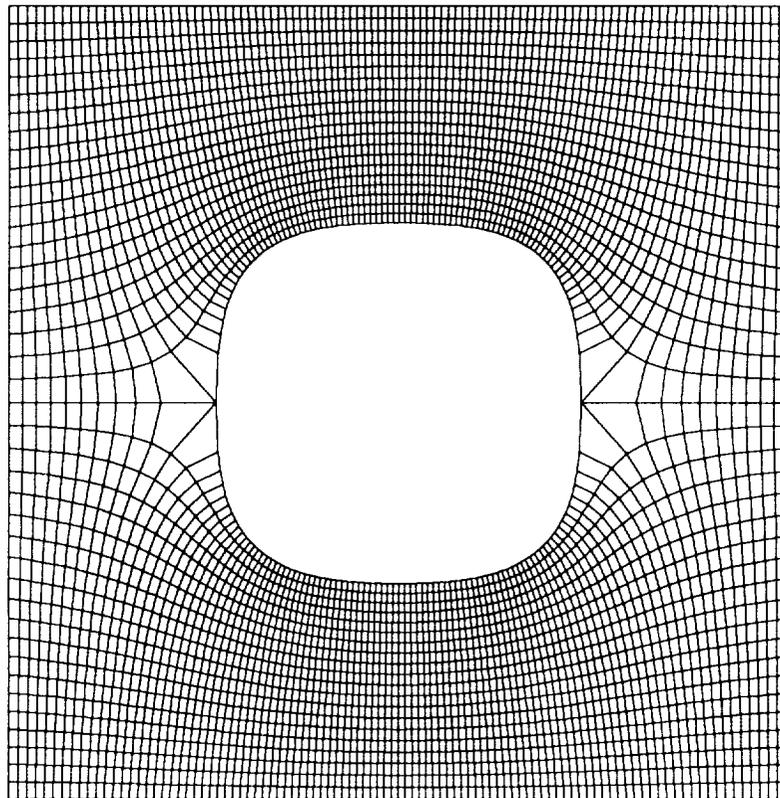


Figure 4.—A 2-block grid passing around a supercircle-like object.

Figure 5 shows a single-block volume grid inside one half of a torus. Five surfaces (except the top surface) are displayed. Symmetry of the geometry shape enables us to focus on one sheet of the volume grid and do grid orthogonality well. Other sheets can be obtained by rotations of different angles.

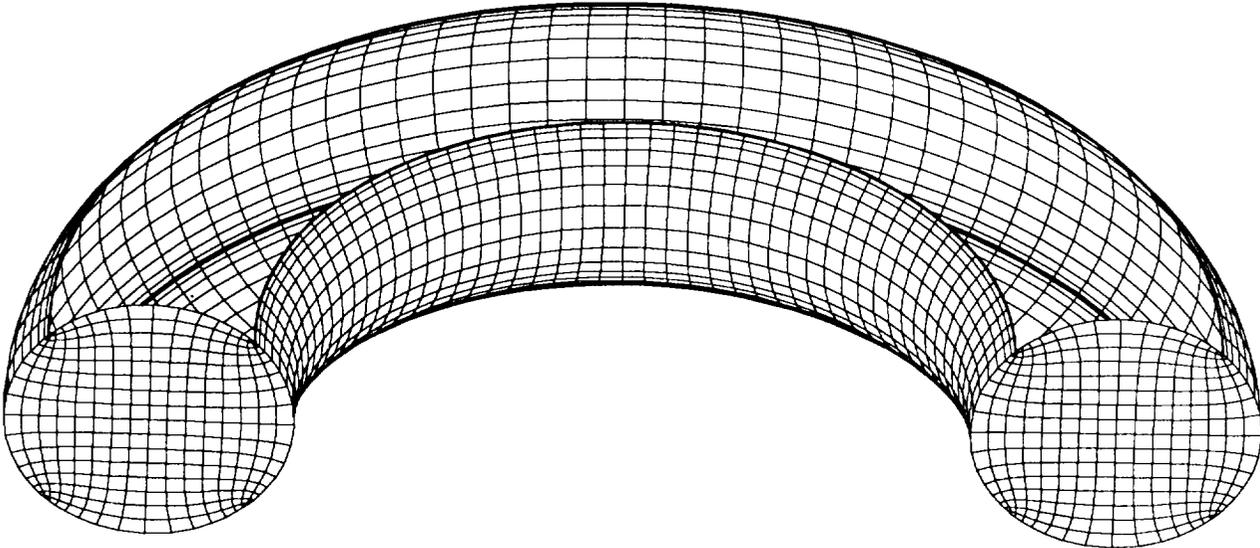


Figure 5.—Half of a single-block volume grid inside a torus.

Figure 6 shows two pictures of a 7-block volume grid inside a sphere. Because of symmetry, we only need to perform optimization for one of six side blocks and the center block.

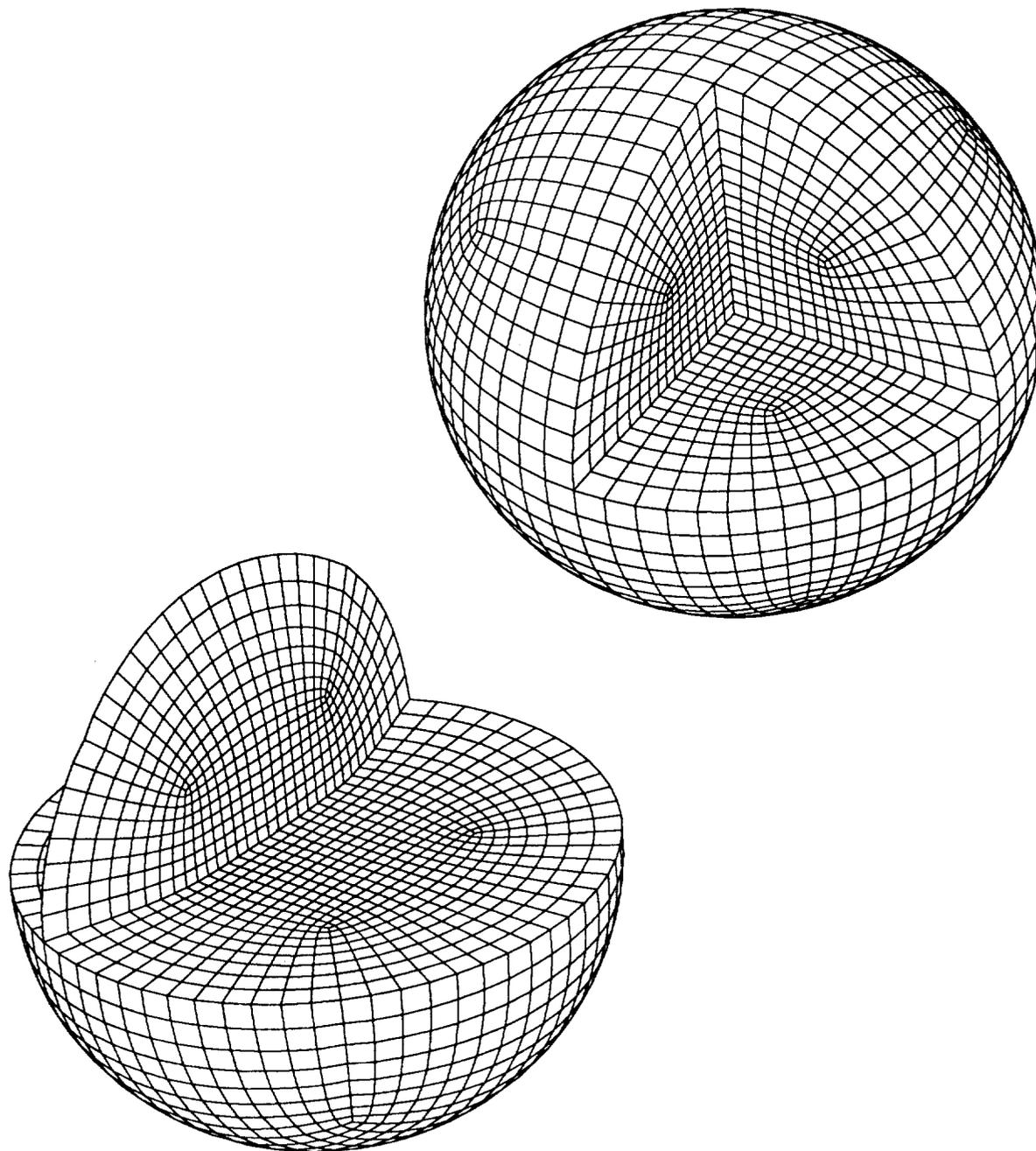


Figure 6.—Two views of a 7-block volume grid inside a sphere.

These examples clearly show the smoothness and lack of grid skewness that can be reliably expected in our new interactive environment.

4. SUMMARY

We have combined several new grid generation methods to obtain high quality grids in an interactive environment. Grids can be generated reliably, and grid quality is high in general. Currently, we are developing a grid generation program called **GridPro**TM/IGgen to take advantage of these new advances in grid generation. Our goal is to generate a range of high-quality single- and/or multi-block grids interactively on a computer workstation.

REFERENCES

1. Lu, N. and Eiseman, P.R., "A grid quality manipulation system," in *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, A. S.-Arcilla, J. Häuser, P. R. Eiseman, and J. F. Thompson, eds. (North-Holland, Amsterdam, 1991), pp. 607–616.
2. Eiseman, P.R., Lu, N., Jiang, M., and Thompson, J.F., "Algebraic-Elliptic Grid Generation," in *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, Weatherill, N.P., Eiseman, P.R., Häuser, J., and Thompson, J.F., eds. (Pineridge Press, Swansea, U.K., 1994), pp. 37–48.
3. G. Farin, "Curves and Surfaces for Computer Aided Geometric Design," 2nd ed. (Academic Press, Boston, 1990), Chapter 20.

