Memoirs of the Faculty of Engineering and Design
Kyoto Institute of Technology 52 (March, 2004)

HYPERBOLIC-ELLIPTIC ZONAL METHOD OF GRID GENERATION FOR INTERNAL FLOW REGIONS

Kenichi MATSUNO

(Received September 30, 2003; Accepted October 31, 2003)

Abstract

A method to generate a grid for an enclosed or internal flow region with guaranteeing grid orthogonality at the wall boundary region as well as smooth grid-line connections is presented in this paper. The computational region is divided into three types of zones to which the hyperbolic, elliptic, or hyperbolic-elliptic blended methods of grid generation are applied. The hyperbolic grid method is applied to a zone next to wall boundary in order to get the excellent grid orthogonality. While the elliptic grid method is applied to an inner zone where the grid smoothness is important. These zones are connected through a transitional zone whose grid is generated by the hyperbolic-elliptic blended method in order to avoid sudden change of grid property. In order to obtain superior grid orthogonality at the hyperbolic grid as well as superior robustness at the elliptic grid, an upwind-biased finite-difference formulation is introduced. A new pseudo-time iteration method using the stable rational Runge-Kutta scheme is introduced to solve resultant discretized equations for further robustness. The generated grids for complex geometries demonstrate the performance of the method.

Key Words: *Grid generation, hyperbolic grid, elliptic grid, structured grid.*

1. Introduction

Developments of computers and numerical methods have made it possible to simulate practical flow problems with reasonable accuracy. However, since the computer memory and processor speed are still limited we cannot use enough grid points for complex flow problems. It is also well known that the accuracy and resolution of the numerical solutions are strongly dependent on quality of the grid in the flowfield. Thus it is very important in Computational Fluid Dynamics to generate the grid of high quality. In generating the grid of high quality, there exist several important issues. The one of them is how to efficiently fit the grid normal to the body boundaries of complex geometry. Grid orthogonality near the body boundary is especially important to obtain an accurate solution of a viscous flow.

A hyperbolic grid method is a grid generation method which uses a partial differential equation of hyperbolic type as a basic system of equations and generates a grid by marching from an initial body surface grid towards outer far-field boundary.[1] The hyperbolic system of grid generation is based on conditions of orthogonality of grid lines and grid cell volume specification. Thus the hyperbolic grid method generates a grid of especially excellent grid orthogonality as well as grid-spacing control. The outer far-field boundary, however, cannot be specified in advance because of its inherent hyperbolic property, that is, the outer boundary is inevitably determined by the equations. Thus, the hyperbolic grid method has been developed mainly for applications to outer flow problems. From the limitation of impossible explicit specification of outer boundary grid, the hyperbolic grid method is hardly applied to enclosed regions, where the internal flow problems are usually written.

An elliptic grid method, on the contrary, generates a grid of regions surrounded by specified boundaries.[2),3)] The elliptic grid method uses a system of partial differential equations of elliptic type, and the grid of all of the region is generated simultaneously. The elliptic system for grid generation is usually a system of the Poisson equations. The elliptic equation is a kind of "averaging" or "smoothing" operation on grid points between boundaries. Therefore the generated grid by the elliptic grid method is extremely smooth. The grid control is done by source terms of the Poisson equation. The grid orthogonality is difficult to be imposed on the generated grid by the source terms of the Poisson equations.

Internal flow region is usually surrounded by the wall, thus the computational region is enclosed with fixed boundary. To generate a grid for an enclosed region, a method using the elliptic equations is usually applied. The grid orthogonality has been unwillingly accepted with rough approximation imposed on the near-boundary grid.[3)]

It is desirable to develop a method which includes both orthogonality property of the hyperbolic grid and smoothness with explicitly specified boundaries of the elliptic grid. Especially, the grid for internal flow simulation is required these properties, grid orthogonality and smoothness, simultaneously.

As a consequence we naturally arrive at an idea of combination of the hyperbolic grid near boundary region for getting proper othogonality and the elliptic grid or elliptic-like averaging (smoothing) operation at inner region. This line of approach was presented by Spradling *et al.*.[4)] They proposed a method which uses the elliptic equations blended with the hyperbolic equation near body-boundary regions. The blending of two different types of equations is done by simple linear summation. Their method applied to grid generation for an outer flow around an automobile with specifying outer far-field boundary. The method of Spradling *et al.* has overcome the shortcoming of the hyperbolic grid method that the outer boundary condition cannot be imposed. However, their method is for outer flow problems and not suitable to internal flow regions because the orthogonal property is applicable to only one side of the flow region. To apply their method to internal flow problem, some modification is necessary. There still remains a problem of blending two grid methods of different types. The other development is further necessary for their method to get converged grid solution when initial guess is improperly made, as stated in their paper.[4)] The method of Spradling *et al.* requires to use an appropriate initial guess. It means the robustness of the method depends on the initial guess (condition). Generally the grid method should not be affected by initial guess. It is important for the grid method to have robustness and to generate the grid as we desire, even if the initial guess is improper for the finally generated grid.

Another sophisticated approach of grid generation for enclosed regions using the hyperbolic grid method was proposed by Jeng *et al.*.[5)] They first generate four grids using the hyperbolic grid method starting from the north, south, west and east boundaries independently. Next the four independently generated hyperbolic grids were averaged with weight so as not to influence each other at the boundaries. This approach achieves excellent orthogonality near the boundary region and easy to implement. The smoothness of the grid is also dependent on the weight for averaging. However, estimation of the weights for averaging procedure is difficult and many of trial-and-error are necessary.

The objective of the present paper is to propose a new hyperbolic-elliptic zonal method of grid generation for internal flow problems or enclosed regions. The present research effort is to combine two different types of partial differential equations for appropriate regions with necessary grid orthogonality and smoothness and to construct robust grid generation procedure. The idea of blending the hyperbolic and elliptic equations is close to the method of Spradling *et al.*,[4] but the approach is definitely different from Spradling *et al.* The present method firstly divides a region into three types of grid zone and applies appropriate grid generation method to them. The hyperbolic grid method is applied to the zones around the boundaries and the elliptic grid method is applied to the interior zone. These hyperbolic and elliptic grid zones are connected by a transitional grid zone to which the hyperbolic and elliptic blended method is applied. To guarantee the grid orthogonality and smoothness, the *upwind* finite-difference formulations are introduced for all types of equations: the hyperbolic, elliptic, and hyperbolic-elliptic blended types of equations.

The author has proposed the idea of making use of an upwind finite-difference scheme in the elliptic grid generations.[6),7),8] The use of the upwind finite-difference scheme with combination of a central finite-difference scheme was successfully applied to an adaptive grid method for the elliptic grid generation.[7),8] For the robustness for the elliptic grid generation, the upwind scheme with combination of central scheme was demonstrated to be sufficient. The upwind formulation for the hyperbolic grid generations was also demonstrated to be effective for constructing robust grid generation with sufficient grid orthogonality.[9] The present research effort forms a sequel to the papers.[6),7),8),9] The hyperbolic-elliptic zonal method of grid generation for internal flow or enclosed regions with use of the upwind formulation is proposed in this paper.

One of the difficulty in the grid generation is an iteration procedure to get the grid solution. As stated by Spradling *et al.*,[4] an inappropriate initial guess for the iteration causes negative Jacobian and results in the termination of the iterative cycle. Generally speaking, the iteration procedure should not be affected by the initial guess, otherwise the sufficient robustness of the method cannot be obtained. The present upwind formulation automatically keeps numerical dissipation at proper one and is therefore robust. We further introduce a procedure which makes gridding more robust in this paper. That is, a new pseudo-time iteration approach for grid generation is introduced in this paper. Thus the combination of the hyperbolic-elliptic zonal grid method using upwind formulation with the pseudo-time iteration is proposed and developed in this paper.

Sections 2 and 3 present upwind formulation for hyperbolic and elliptic grid generations, respectively. It is emphasized that the use of upwind scheme for discretizing partial differential equation is essential for robustness of the grid method. Section 4 presents an approach of the present hyperbolic-elliptic zonal method of grid generation.

Section 5 discusses about the iteration strategy and presents the new pseudo-time iteration method. Section 6 shows some illustrative grids to demonstrate capability of the method and quality of the generated grids. Finally section 7 gives concluding remarks of the paper.

2. Upwind Formulation for Hyperbolic Grid Generation

Let $\boldsymbol{r}=(x,y)$ be a position vector of the grid point, and (ξ,η) be a body-fitted general coordinate system. Then partial derivatives of the position vector, $\boldsymbol{r}_\xi$ and $\boldsymbol{r}_\eta$, represent tangential vectors along the grid lines at that point and their absolute values are according to corresponding grid spacing. The orthogonality relation between ξ and η and the equation of specifying a grid cell area A are represented as,

$$\boldsymbol{r}_\xi \cdot \boldsymbol{r}_\eta = 0 \tag{1}$$

$$|\boldsymbol{r}_\xi \times \boldsymbol{r}_\eta| = A(\xi,\eta). \tag{2}$$

Applying the Newton linearization to Eqs.(1) and (2), we get a system of the grid generation equations in a vector form,

$$\mathbf{r}_\eta^{\nu+1} + \tilde{Q}\mathbf{r}_\xi^{\nu+1} - P^{-1}\mathbf{g} = 0, \tag{3}$$

where, superscript ν is an iteration counter, and

$$\mathbf{r}=\begin{bmatrix} x \\ y \end{bmatrix}=\boldsymbol{r}^T,\quad P=\begin{bmatrix} x_\xi & y_\xi \\ -y_\xi & x_\xi \end{bmatrix}^\nu,\quad Q=\begin{bmatrix} x_\eta & y_\eta \\ y_\eta & -x_\eta \end{bmatrix}^\nu,$$

$$\mathbf{g}=\begin{bmatrix} (x_\xi x_\eta + y_\xi y_\eta)^\nu \\ A+(x_\xi y_\eta - y_\xi x_\eta)^\nu \end{bmatrix},\quad \text{and}\quad \tilde{Q}=P^{-1}Q \tag{4}$$

Since $|P|=x_\xi^2+y_\xi^2 \neq 0$ (non zero grid spacing), $\tilde{Q}=P^{-1}Q$ is exist and proved to be symmetric. Thus Eq.(3) is hyperbolic with respect to the time-like direction η. The matrix $\tilde{Q}$ has two real distinct eigenvalues, $\pm\lambda$:

$$\pm\lambda,\quad \text{where}\quad \lambda=\frac{|\mathbf{r}_\eta|}{|\mathbf{r}_\xi|}=\sqrt{\frac{x_\eta^2+y_\eta^2}{x_\xi^2+y_\xi^2}}. \tag{5}$$

These two distinct real eigenvalues constitute characteristics. The information of initial data at $\eta=0$ propagates along these two characteristic lines. Thus, there exist *windward* directions and we can apply the upwind scheme to discretize Eq.(3).

Since the matrix $\tilde{Q}$ is symmetric and has two distinct eigenvalues, the matrix $\tilde{Q}$ can be diagonalized. Now we split the matrix $\tilde{Q}$ into the sum of two matrices $\tilde{Q}^+$ and $\tilde{Q}^-$ such that each has only non-negative and non-positive eigenvalues respectively.

$$\tilde{Q}=\tilde{Q}^+ + \tilde{Q}^-, \tag{6}$$

where,

$$\tilde{Q}^+ = \begin{bmatrix} \frac{\alpha+\lambda}{2} & \frac{\beta}{2} \\ \frac{\beta}{2} & -\frac{\alpha-\lambda}{2} \end{bmatrix}, \quad \tilde{Q}^- = \begin{bmatrix} \frac{\alpha-\lambda}{2} & \frac{\beta}{2} \\ \frac{\beta}{2} & -\frac{\alpha+\lambda}{2} \end{bmatrix},$$

$$\alpha = \frac{x_\xi x_\eta - y_\xi y_\eta}{x_\xi^2 + y_\xi^2}, \quad \beta = \frac{x_\xi y_\eta + y_\xi x_\eta}{x_\xi^2 + y_\xi^2}. \tag{7}$$

Substituting Eq.(6) into Eq.(3), we obtain

$$\mathbf{r}_\eta^{\nu+1} + \tilde{Q}^+ \mathbf{r}_\xi^{\nu+1} + \tilde{Q}^- \mathbf{r}_\xi^{\nu+1} - P^{-1}\mathbf{g} = 0 . \tag{8}$$

To discretize Eq.(8) we apply at least second order accurate scheme. Here, a three-point backward difference scheme to the η-derivative term and the MUSCL upwind scheme with minmod limiter[10)] to the ξ-derivatives are applied. The use of the MUSCL upwind scheme with minmod limiter is to change the order of accuracy of the upwind scheme according to the grid smoothness. Thus the hyperbolic grid generation equation Eq.(8) can be discretized as,

$$\nabla_\eta^{(2)} \mathbf{r}^\nu_{i,j+1} + (\tilde{Q}^+)^\nu \nabla_\xi^{(M)} \mathbf{r}^\nu_{i,j+1} + (\tilde{Q}^-)^\nu \Delta_\xi^{(M)} \mathbf{r}^\nu_{i,j+1} - (P^{-1})^\nu \mathbf{g}_{i,j+1} = 0 . \tag{9}$$

The operators are described as follows, with $\Delta\xi = \Delta\eta = 1$,

$$\begin{aligned}
&\nabla_\eta^{(2)} \mathbf{r}^\nu_{i,j+1} = \frac{3}{2}\mathbf{r}^\nu_{i,j+1} - 2\mathbf{r}_{i,j} + \frac{1}{2}\mathbf{r}_{i,j-1}, \\
&\nabla_\xi^{(M)} \mathbf{r}_{i,j+1} = \mathbf{r}^-_{i+1/2,j+1} - \mathbf{r}^-_{i-1/2,j+1}, \\
&\Delta_\xi^{(M)} \mathbf{r}_{i,j+1} = \mathbf{r}^+_{i+1/2,j+1} - \mathbf{r}^+_{i-1/2,j+1}, \\
&\mathbf{r}^-_{i+1/2,j+1} = \mathbf{r}_{i,j+1} + \frac{1}{4}\left[(1-\kappa)\overline{\nabla \mathbf{r}_{i,j+1}} + (1+\kappa)\overline{\Delta \mathbf{r}_{i,j+1}}\right], \\
&\mathbf{r}^+_{i+1/2,j+1} = \mathbf{r}_{i+1,j+1} - \frac{1}{4}\left[(1+\kappa)\overline{\nabla \mathbf{r}_{i+1,j+1}} + (1-\kappa)\overline{\Delta \mathbf{r}_{i+1,j+1}}\right], \\
&\overline{\nabla \mathbf{r}_{i,j+1}} = \mathrm{minmod}(\nabla \mathbf{r}_i, b\Delta \mathbf{r}_i), \\
&\overline{\Delta \mathbf{r}_{i,j+1}} = \mathrm{minmod}(\Delta \mathbf{r}_i, b\nabla \mathbf{r}_i), \\
&\mathrm{minmod}(x,y) = \mathrm{sgn}(x)\cdot \max\left[0, \min\left(|x|, \mathrm{sgn}(x)\cdot y\right)\right], \\
&\nabla \mathbf{r}_i = \mathbf{r}_{i,j+1} - \mathbf{r}_{i-1,j+1}, \quad \Delta \mathbf{r}_i = \mathbf{r}_{i+1,j+1} - \mathbf{r}_{i,j+1}, \\
&b = \frac{3-\kappa}{1-\kappa}.
\end{aligned} \tag{10}$$

In this paper, the order of the upwind scheme in Eq.(9) is second-order accurate($\kappa = -1$) or third-order accurate($\kappa = 1/3$). The matrices $\tilde{Q}^{\pm}$ and P^{-1} in Eq.(9) are all evaluated at $(i,j+1)$ using known values at previous iteration step ν. These matrices include the ξ- and η-derivatives. We evaluate them using central differencing for the ξ-derivatives and three-point backward differencing for the η-derivatives.

Specification of cell area A : The cell area $A_{i,j+1}$ directly controls the grid spacing between (i,j) and $(i,j+1)$. The specification of $A_{i,j+1}$ is very important, but more complex and empirical. The method proposed in Ref.9) is used in this paper. First we specify the desired grid spacing in η-direction, $\Delta s_{\eta_{i,j+1}}$. We, then, multiply a *scaled* ξ-grid spacing, $\widetilde{\Delta s}_{\xi_{i,j+1}}$, to it to estimate the cell area $A_{i,j+1}$,

$$A_{i,j+1}=\widetilde{\Delta s}_{\xi_{i,j+1}}\cdot\Delta s_{\eta_{i,j+1}}. \tag{11}$$

Here, the scaled ξ-grid spacing is estimated as,

$$\widetilde{\Delta s}_{\xi_{i,j+1}}=\max(\Delta s_{\xi_{i,j}},C\overline{\Delta s_{\xi_j}}). \tag{12}$$

where,

$$\Delta s_{\xi_{i,j}}=\left\{\left(\frac{x_{i+1,j}-x_{i-1,j}}{2}\right)^2+\left(\frac{y_{i+1,j}-y_{i-1,j}}{2}\right)^2\right\}^{1/2},$$

$$\overline{\Delta s_{\xi_j}}=\frac{1}{i_{\max}-1}\sum_{i=1}^{i_{\max}-1}\Delta s_{\xi_{i,j}}, \tag{13}$$

Here C is a constant of a value between 0.8 and 1.5. The area averaging is also employed in this paper, which is effective for smooth change of grid spacing:

$$A_{i,j+1}=(1-w)A_{i,j+1}+w\frac{A_{i+1,j+1}+A_{i-1,j+1}}{2}, \tag{14}$$

where w is a weight of averaging, and usually given a value 0.4 in the present examples shown later. This area averaging is usually operated twice or three times.

3. Upwind Formulation for Elliptic Grid Generation

The two-dimensional forms of the Poisson equations for a grid generation due to Thomas and Middlecoff[11) can be written as

$$\begin{aligned}\xi_{xx}+\xi_{yy}&=\left(\xi_x^2+\xi_y^2\right)p\\ \eta_{xx}+\eta_{yy}&=\left(\eta_x^2+\eta_y^2\right)q\end{aligned} \tag{15}$$

where $p=p(\xi,\eta)$ and $q=q(\xi,\eta)$ are control functions which control grid spacing along ξ and η grid lines. Equation(15) can be transformed to ξ-η coordinate by interchanging the role of dependent and independent variables. This yields the following a system of the elliptic equations for two-dimensional grid generation,

$$\alpha(\mathbf{r}_{\xi\xi}+p\mathbf{r}_\xi)-2\beta\mathbf{r}_{\xi\eta}+\gamma(\mathbf{r}_{\eta\eta}+q\mathbf{r}_\eta)=0, \tag{16}$$

where, $\mathbf{r}=[x,y]^T$ and

$$\alpha=\frac{|\mathbf{r}_\eta|^2}{|\mathbf{r}_\xi|^2+|\mathbf{r}_\eta|^2},\quad\beta=\frac{\mathbf{r}_\xi\cdot\mathbf{r}_\eta}{|\mathbf{r}_\xi|^2+|\mathbf{r}_\eta|^2},\quad\gamma=\frac{|\mathbf{r}_\xi|^2}{|\mathbf{r}_\xi|^2+|\mathbf{r}_\eta|^2}, \tag{17}$$

$$|\mathbf{r}_\eta|^2=x_\eta^2+y_\eta^2,\quad|\mathbf{r}_\xi|^2=x_\xi^2+y_\xi^2,\quad\mathbf{r}_\xi\cdot\mathbf{r}_\eta=x_\xi x_\eta+y_\xi y_\eta \tag{18}$$

Here, the coefficients α, β, and γ are scaled in order to balance with a hyperbolic part when the elliptic equation is combined with the hyperbolic equation. It will be discussed later.

In this paper, the elliptic system is used for the inner region for the purpose of connecting the grid smoothly between regions generated by the hyperbolic method. Thus the control functions p and q can be omitted for the case of simple geometry. We, however, often have to control the grid density according to the flow physics like a solution adaptive grid. When we like to control the grid for some purpose, the functions p and q are used to control the grid location. When we use these control function for grid generation, one of the problems using the control functions is that the grid crossing or negative Jacobian might occur if we use a large value of p or q.[8),9)] This is a kind of *wiggles*.[8)] In order to avoid such a grid line crossing or negative Jacobian and to make the method robust, we apply a central-upwind hybrid finite-difference scheme[6),8)] to the first derivative terms whose coefficients are the control functions in Eq.(16). The central-upwind hybrid finite difference scheme is the one that difference approximation stencil is shifted towards *upwind*-ward according to the scale of its coefficient with minimal truncation error. See Refs.6),7),8) for details. When we apply the central-upwind hybrid approximation[6)] to the first derivative terms and central difference approximation to the second derivative terms and cross derivative term, the finite difference approximation to Eq.(16) becomes

$$\begin{aligned}\alpha_{i,j}\Big[(\mathbf{r}_{i+1,j}-2\mathbf{r}_{i,j}+\mathbf{r}_{i-1,j})+p_{i,j}\Big\{\phi^{(\xi)}_{i,j}(\mathbf{r}_{i+1,j}-\mathbf{r}_{i,j})+\left(1-\phi^{(\xi)}_{i,j}\right)(\mathbf{r}_{i,j}-\mathbf{r}_{i-1,j})\Big\}\Big]\\ -2\beta_{i,j}\frac{1}{4}\left[\mathbf{r}_{i+1,j+1}-\mathbf{r}_{i-1,j+1}-\mathbf{r}_{i+1,j-1}+\mathbf{r}_{i-1,j-1}\right]+\gamma_{i,j}\Big[(\mathbf{r}_{i,j+1}-2\mathbf{r}_{i,j}+\mathbf{r}_{i,j-1})\\ +q_{i,j}\Big\{\phi^{(\eta)}_{i,j}(\mathbf{r}_{i,j+1}-\mathbf{r}_{i,j})+\left(1-\phi^{(\eta)}_{i,j}\right)(\mathbf{r}_{i,j}-\mathbf{r}_{i,j-1})\Big\}\Big]=0,\end{aligned} \tag{19}$$

where, for example, $\phi^{(\xi)}_{i,j}$ is a parameter which shifts the finite-difference approximation from central to upwind-biased according to the value $p_{i,j}$ at the grid (i,j). It is evaluated as

$$\phi^{(\xi)}_{i,j}=\begin{cases}\frac{1}{2} & \text{if}\,|p_{i,j}|\le 2-\epsilon\\ -\frac{1}{p_{i,j}}+\frac{1}{1-\exp(-p_{i,j})} & \text{otherwise}\end{cases} \tag{20}$$

Here, ϵ is a small positive number and given by 0.01. Note that $\phi^{(\xi)}_{i,j}=\phi^{(\xi)}(p_{i,j})$ and $\phi^{(\xi)}(-p)=1-\phi^{(\xi)}(p)$. The parameter $\phi^{(\eta)}_{i,j}$ is estimated similarly using $q_{i,j}$. The present central-upwind hybrid scheme approximates the advection-diffusion equation so as to minimize the numerical error without the wiggles.[6)]

4. Hyperbolic-Elliptic Zonal Grid Generation

The hyperbolic and elliptic systems for grid generation described above are combined in order to be applied to an enclosed or internal flow region.

To explain the formulation, we give one-dimensional case in detail as an introduction. For one dimension, the hyperbolic and elliptic equations of grid generation are written respectively as

$$x_\xi-A=0, \tag{21}$$

$$x_{\xi\xi}+p x_{\xi}=0. \tag{22}$$

Here x is a physical coordinate, and ξ is a computational coordinate. The marching step width in the hyperbolic equation (21) corresponds to so called cell volume A (grid spacing in case of one dimension). The control function is represented by p in the elliptic equation (22).

Suppose that the both end of the coordinate, $x=0$ and $x=1$, are the body surface boundaries. We explicitly specify the grid spacings from boundary to some extent. For these regions, the hyperbolic grid generation method is suitable because the exact grid spacings (and orthogonality in 2 or 3 dimension) can be specified. For the inner region apart from the boundaries, the grid spacing (and grid orthogonality) is not important, and the smoothness is the highest priority. Thus the inner region is suitable for the elliptic grid generation method. It is desirable to connect hyperbolic grid and elliptic grid gradually. We put a transitional grid zone between the hyperbolic grid and the elliptic grid, to which a system of hyperbolic-elliptic hybrid equations is applied. The change from the hyperbolic type to the elliptic type through the hyperbolic-elliptic hybrid type is accomplished by use of a parameter θ, which is a function of position in computational coordinate. Thus the parameter θ divides the region into three types of zones. The present hyperbolic-elliptic zonal grid generation equation is written as follows.

$$\theta_h^L\left[x_\xi-A\right]+\theta_h^U\left[-x_\xi+A\right]=\theta_e\left[x_{\xi\xi}+p x_\xi\right] \tag{23}$$

where θ_h^L, θ_h^U, and θ_e are functions of ξ and switching or weighting parameters with the following constraints,

$$0\leq\theta_h^L,\theta_h^U,\theta_e\leq 1,\qquad \theta_h^L+\theta_h^U+\theta_e=1,\ \text{ and }\ \theta_h^L\cdot\theta_h^U=0. \tag{24}$$

Superscripts L and U indicate *Lower* (the left) and *Upper* (the right) side respectively. The last constraint, $\theta_h^L\cdot\theta_h^U=0$, is forced on the hyperbolic grids in order not to overlap each other.

Discretization of Eq.(23) based on the upwind-biased scheme described above can be written as,

$$\begin{aligned}&\theta_{hi}^L\left[\nabla_\xi x_i-V_i\right]+\theta_{hi}^U\left[-\Delta_\xi x_i+V_i\right]\\&\quad=\theta_{ei}\left[\Delta_\xi\nabla_\xi x_i+p_i\left\{\phi_i\Delta_\xi x_i+(1-\phi_i)\nabla_\xi x_i\right\}\right]\end{aligned} \tag{25}$$

where

$$\nabla_\xi x_i=x_i-x_{i-1},\ \text{ and }\ \Delta_\xi x_i=x_{i+1}-x_i.$$

Equation (25) is tridiagonal system and is solved at once.

Figure 1 shows an example of one-dimensional grid. The grid is generated on the region between $x=0(i=1)$ and $x=1(i=i_{\max}=41)$. Here, the first eight grid points adjacent boundaries are generated by the pure hyperbolic system ($i=2$ to 9 and 33 to 40), while the central 13 grid points are generated by the pure elliptic system (i=15 to 27). The rest (i=10 to 14 and 28 to 32) is owing to avoid sudden change from the hyperbolic to the elliptic system and is generated by the hyperbolic-elliptic linearly blended system. In this case as described above, the hyperbolic equation controls the grid spacing in the regions attached to the boundaries. While the elliptic equation generates a grid smoothly in the region between hyperbolic regions of both sides. The variations of the parameters, A, p, and θ's are shown in Fig.1. The variation of θ at the hyperbolic-elliptic blended system is a linear function of ξ, or i.

For two and three dimensions, the method is described by a straight extension of the one-dimensional case described above. In this paper, we consider two types of geometries: a channel and a closed region covered with a grid of O-topology. For the channel illustrated in Fig.2, wall boundaries are the south and north boundaries, corresponding to $\eta = 0$ and $\eta = \eta_{\max}$ respectively. The west and east boundaries, corresponding to $\xi = 0$ and $\xi = \xi_{\max}$, are inflow and outflow boundaries, and therefore, not necessarily enforced the grid orthogonality condition on, and usually put a condition of constant position ($x = const.$). The grid spacing and strict orthogonality condition are both enforced on the north and south boundaries. A typical C-grid around airfoil belongs to this type; the south boundary is at the airfoil surface and wake-cult line, the north boundary is far-field boundary, and the west and east boundaries are outflow boundary. For the enclosed O-grid, the north boundary is coincide with a body surface or a specified boundary, while the south boundary is the other specified boundary or a singular point where all the grid points coincide with this point like a pole. In this situation, the west and east boundaries are periodic each other. We put the hyperbolic grid at the wall boundary region. For the boundary on which the grid orthogonality is not necessary enforced, the elliptic grid is used.

To describe equations concisely, we define two operators: $\mathcal{H}$ and $\mathcal{E}$. Let $\mathcal{H}(\mathbf{r})$ and $\mathcal{E}(\mathbf{r})$ be the hyperbolic and elliptic grid generators respectively:

$$\mathcal{H}(\mathbf{r}) \equiv \mathbf{r}_\eta + \tilde{Q}\mathbf{r}_\xi - P^{-1}\mathbf{g}, \tag{26}$$

$$\mathcal{E}(\mathbf{r}) \equiv \alpha(\mathbf{r}_{\xi\xi} + p\mathbf{r}_\xi) - 2\beta\mathbf{r}_{\xi\eta} + \gamma(\mathbf{r}_{\eta\eta} + q\mathbf{r}_\eta). \tag{27}$$

Then the present hyperbolic-elliptic zonal grid generation equation is represented as:

$$\theta_h^L \mathcal{H}(\mathbf{r}) + \theta_h^U\{-\mathcal{H}(\mathbf{r})\} = \theta_e \mathcal{E}(\mathbf{r}), \tag{28}$$

where θs are assumed to be a function of coordinate η and supposed to be subjected to constraints:

$$0 \le \theta_h^L, \theta_h^U, \theta_e \le 1, \qquad \theta_h^L + \theta_h^U + \theta_e = 1, \qquad \theta_h^L \cdot \theta_h^U = 0. \tag{29}$$

The parameters, θ_h^L, θ_h^U and θ_e are weights which change the property of the grid generation equation from the hyperbolic type to the elliptic type via transitional zone. The variation of θ in the hyperbolic-elliptic transitional zone is linear with respect to η. Equation (28) represents the situation that the $\eta = 0$ and $\eta = \eta_{\max}$ correspond to the body surface and $\xi = 0$ and $\xi_{\max}$ are respectively inflow and outflow boundaries for the channel geometry.

When the south boundary is singular, the pole, it is not necessary to use the hyperbolic system. Thus the elliptic system is applied to the region around the pole. Since the grid points at the south boundary is the pole and singular, a floating boundary condition is suitable for this situation, and estimated as:

$$\mathbf{r}_{i,j=1} = \frac{1}{i_{\max}} \sum_{k=1}^{i_{\max}} \mathbf{r}_{k,j=2}, \qquad (i = 1, 2, \ldots, i_{\max}). \tag{30}$$

Applying upwind formulations described at the previous two sections to Eq.(28), we finally have a system of algebraic equations. The system of the algebraic equations is solved by two kinds of iteration methods: a line relaxation method and a pseudo-time iteration, which are described in next section.

5. Iteration Methods

5.1 Line relaxation method.

The line relaxation method is very popular to solve a large system of algebraic equations, and is used for the present method. The line of the relaxation method is taken along the η-coordinate in the present grid equations. A relaxation parameter (denoted by ω in most of available text books) is set to 1.0. In some cases, under relaxation with $\omega = 0.2$ is preferred to solve the resultant algebraic equations for superior robustness.

The line relaxation method is efficient to solve the algebraic equation for the moderate complexity of the geometry. However, for the case with deep (high) or steep concave (convex) geometry, the line-relaxation does not converge or, in the worst case, results in divergence, even if the under-relaxation parameter is used. The hyperbolic-elliptic blended grid zone is apt to diverge. To overcome such situation, we propose the use of the pseudo-time iteration approach in the next sub-section.

5.2 Pseudo-time iteration method.

The pseudo-time subiteration (inner iteration) method has been recently successfully applied to computations of unsteady fluid dynamics problems.[13),14)] The efficiency and robustness of the pseudo-time subiteration method are discussed in detail for unsteady turbulent flows in Ref.14), and very promising properties of the method have been demonstrated.

As for the use of the pseudo-time subiteration approach in the grid generation, we have successfully applied the pseudo-time subiteration method to the hyperbolic grid generation method for two and three dimensions[9)]. Thus the pseudo-time iteration methods is expected to work well for the present hyperbolic-elliptic zonal grid method as well. The pseudo-time iteration approach is used as a global iteration procedure in the present grid method.

Now, we add a *pseudo-time* term to Eq.(28), then we have

$$\frac{\partial \mathbf{r}}{\partial \tau} = -\mathcal{R}(\mathbf{r}), \tag{31}$$

where

$$\mathcal{R}(\mathbf{r}) = \theta_h^L \mathcal{H}(\mathbf{r}) + \theta_h^U (-\mathcal{H}(\mathbf{r})) - \theta_e \mathcal{E}(\mathbf{r}). \tag{32}$$

Here, τ represents the pseudo-time. As shown, the presence of the pseudo-time term alters the basic grid generation equations, but the $\partial\mathbf{r}/\partial\tau$ term is treated in such a way that it vanishes as $\mathbf{r}$ approaches its correct value after the sufficient iterations. The pseudo-time τ has neither meaning of physical time nor geometrical interpretation.

An explicit schemes of the Runge-Kutta type is preferable to Eq.(31). We consider the efficiency and robustness to be most important, and adopt the Rational Runge-Kutta (RRK) scheme[15] to iterate Eq.(31) in the pseudo-time. The RRK scheme is computationally explicit and yet unconditionally stable for some class of parabolic partial differential equations.

The RRK scheme is written for Eq.(31) as:

$$\mathbf{r}^{(1)}=-\Delta\tau\mathcal{R}\left(\mathbf{r}^{\nu}\right), \tag{33}$$

$$\mathbf{r}^{(2)}=-\Delta\tau\mathcal{R}\left(\mathbf{r}^{\nu}+c_2\mathbf{r}^{(1)}\right), \tag{34}$$

$$\mathbf{r}^{(3)}=b_1\mathbf{r}^{(1)}+b_2\mathbf{r}^{(2)}, \tag{35}$$

$$\mathbf{r}^{\nu+1}=\mathbf{r}^{\nu}+\frac{2\mathbf{r}^{(1)}\left\langle\vec{\mathbf{r}}^{(1)},\vec{\mathbf{r}}^{(3)}\right\rangle-\mathbf{r}^{(3)}\left\langle\vec{\mathbf{r}}^{(1)},\vec{\mathbf{r}}^{(1)}\right\rangle}{\left\langle\vec{\mathbf{r}}^{(3)},\vec{\mathbf{r}}^{(3)}\right\rangle}, \tag{36}$$

where $\vec{\mathbf{r}}^{(1)}$, for example, is a vector whose elements are the whole of the vectors $\mathbf{r}^{(1)}$ and $\left\langle\vec{\mathbf{r}}^{(1)},\vec{\mathbf{r}}^{(3)}\right\rangle$ is a scalar product of two vectors $\vec{\mathbf{r}}^{(1)}$ and $\vec{\mathbf{r}}^{(3)}$. The coefficients, b_1, b_2, and c_2, in the above equations are

$$b_1=1-b_2,\quad b_2=-1/c_2,\quad c_2=1/16. \tag{37}$$

Superscript ν is an iteration counter and $\mathbf{r}^{\nu+1}\rightarrow\mathbf{r}$ after the sufficient iterations. The order of $\Delta\tau$ is usually comparable to the order of grid size $\Delta\xi$ or $\Delta\eta$ in computational plane. (Note that $\Delta\xi=\Delta\eta=1.0$ in this paper). The value of $\Delta\tau$ is not necessarily constant. The constant value of 1.0, however, is used throughout the paper.

The required number of iteration depends on the number of grid points, especially on the number of the elliptic grid points. For the case that includes the large number of the elliptic grids, the number of iteration is large, say several hundred to a thousand. On the contrary, there included few grid points of the elliptic grid zone, then the number of the iteration is roughly equal to the number of the grid points along the η-line, usually less than one hundred. Fortunately, it is not always necessary to converge completely as for the grid generation. The important issues for the grid generation are orthogonality, smoothness, efficiency and robustness. Thus the tolerance of convergence criterion may be relatively loose.

Although the convergence rate of the pseudo-time iteration approach is worse than the line-relaxation method, the pseudo-time iteration approach is very robust even if the initial grid is unfavorable as a initial condition. Even if grid-line crossings happen in the midst of the iteration, the pseudo-time iteration method accomplishes the iteration without divergence in most cases and results a grid of sufficient quality. Thus the recommended choice of the iteration method will be; the first choice is the line-relaxation method otherwise the pseudo-time iteration method.

6. Results and Discussion

To demonstrate a performance of the present method, we first show a grid generated for a channel with bent corners. Figure 3 shows the result. In the figure, along the η-grid line, the first 4 grid points from upper and lower walls are generated by the hyperbolic grid method, while the next 10 grid points are by the hyperbolic-elliptic blended method. The rest 13 grid points at the center are generated by the elliptic grid method. As can be seen from the figure, the grid orthogonality at the region next to the wall boundary is excellent, while the grid at the central region of the channel are smoothly connected each other.

Next result is a closed region of a ring-like geometry, which is often seen at a cross-section of a cooling pipe. The hyperbolic grid method is again used for the regions next to inner and outer wall boundaries. The elliptic grid method is used for inside region with connecting hyperbolic regions of both sides through the hyperbolic-elliptic blended region. Figure 4 shows the generated grid. Figure 5 shows a grid of more complex geometry than Fig.4. Both results show excellent grid orthognality near the wall boundaries and smooth connection between them. In these cases, no control function in the elliptic system is used.

Next results are the application to the single connected enclosed regions. Figures 6 and 7 demonstrate the generated grids. The hyperbolic method is used for the first 7 grid points next to the boundary for both cases. The region inside except the singular south boundary is generated by the elliptic system. As stated previously, the inner south boundary point, pole, is treated as a floating boundary. Again the hyperbolic-elliptic blended grid (3 grid points in Fig.6 and 6 grid points in Fig.7 along the η-direction) is put to connect them. These two grids shown in Figs. 6 and 7 require the grid control in the elliptic grid by the control functions. In the case of Fig.6, the grid control towards η-direction (radial direction) was done with constant q $(=Q_c=-2(K-1)/(K+1)$ with $K=0.8)$. Note that the control function q estimeted by this equation controls grid point distribution in η-direction such that the grid spacing ratio of adjoining grid points becomes locally approximately K.[8] In the case shown in Fig.7, the grid control towards ξ-direction as well as η-direction is made simultaneously. For η-direction, the same value Q_c as in the case of Fig.6 was given to the control function q where grid control is necessary. For the grid control along the ξ-direction, we gave the following values to p:

$$p=\begin{cases} P_c & \text{for upper right,} \\ -P_c & \text{for lower right,} \\ P_c & \text{for lower left,} \\ -P_c & \text{for upper left,} \end{cases} \qquad P_c=-2\frac{K-1}{K+1}, \quad K=0.8.$$

When further subtle grid control in the elliptic grid region is necessary, it is possible using control functions p and q.

7. Concluding Remarks

The method to generate a grid for internal flow problems with guaranteeing grid orthogonality at the wall boundary region has been presented in this paper. The method first devides a region into three types of zones and applies appropriate grid methods to each zones. The hyperbolic grid method is applied to the zone next to the boundary to guarantee grid orthogonality and make precise grid-spacing control. While the elliptic grid method is applied to the zone at the central region to get smooth grid-point distribution. To the zone connectiong these hyperbolic grid and elliptic grid zones, the hyperbolic-elliptic linearly blended grid method is applied. The upwind-type formulations play an important role in both the hyperbolic and elliptic grid method in this paper. The combination of the upwind formulations and pseudo-time iteration method has made the present hyperbolic-elliptic zonal method robust and allows to use a relatively rough initial grid at the start of the iteration. The generated grids demonstrated the performance of the method.

In this paper, the two-dimensional method has been described and demonstrated. The extension to three dimension is easy and straightforward.

Department of Mechanical and System Engineering
Faculty of Engineering and Design
Kyoto Institute of Technology,
Matsugasaki, Sakyo-ku, Kyoto 606-8585 JAPAN

References

1) J. L. Steger and D. S. Chaussee, *SIAM J. Sci. Stat. Comput.*, **1**, 431 (1980).

2) J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin, *Numerical Grid Generation*, Elsevier, NewYork (1985).

3) R. L. Sorenson, *NASA TM-81193* (1980).

4) M. L. Spradling, S. Nakamura and K. Kuwahara, *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, edited by A. S.-Arcilla, J. Hauser, P. R. Eiseman, J. F. Thompson, p.237, Elsevier (1991).

5) Y.-N. Jeng and Y.-L. Shu, *AIAA J.*, **33**, 1152 (1995).

6) K. Matsuno, *Jour. of Japan Society for Aeronautics ans Space Sciences*, **38**, 386 (1990) (in Japanese).

7) K. Matsuno, *Jour. of Japan Society for Aeronautics ans Space Sciences*, **43**, 26 (1995)(in Japanese).

8) K. Matsuno, *CFD Review 1998*, p.127, World Scientific Publishers (1998).

9) K. Matsuno, *Computers and Fluids*, **28**, 825 (1999).

10) C. Hirsch, *Numerical Computation of Internal and External Flows*, Vol.2, John Wiley & Sons (1988).

11) P. D. Thomas and J. F. Middlecoff, *AIAA J.*, **18**, 652 (1980).

12) K. Matsuno and H. A. Dwyer, *J. Comp. Phys.*, **77**, 40 (1988).

13) A. Arnone, M.-S. Liou and L. A. Povinelli, *AIAA Paper*-93-3361 (1993).

14) C. L. Rumsey, M. D. Sanetrik, R. T. Biedron, N. D. Melson, and E. B. Paralette, *Computers and Fluids*, **25**, 217 (1996).

15) A. Wanbecq, *Computing*, **20**, 333 (1978).

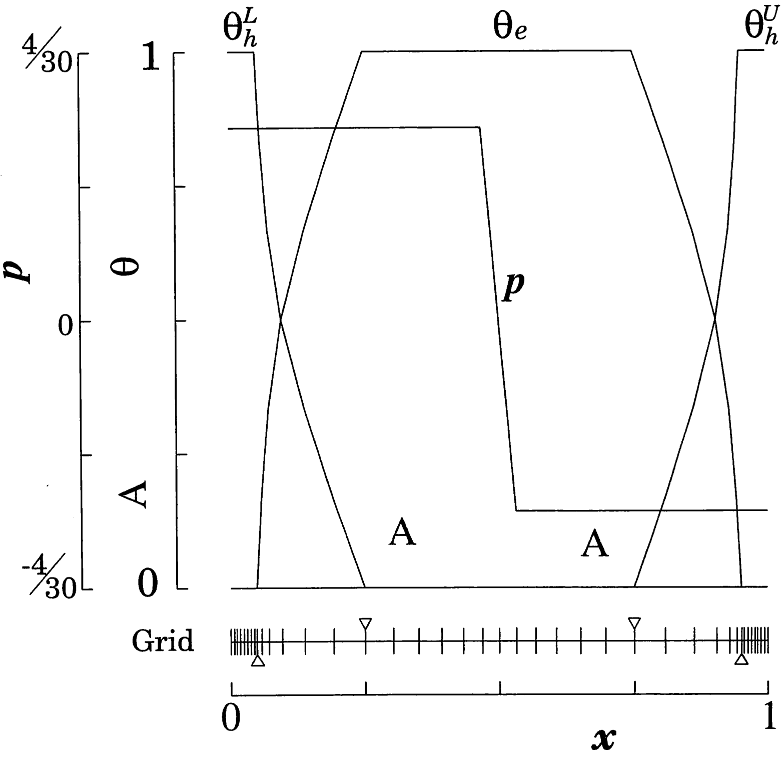

Figure 1. One dimensional hyperbolic-elliptic zonal grid.

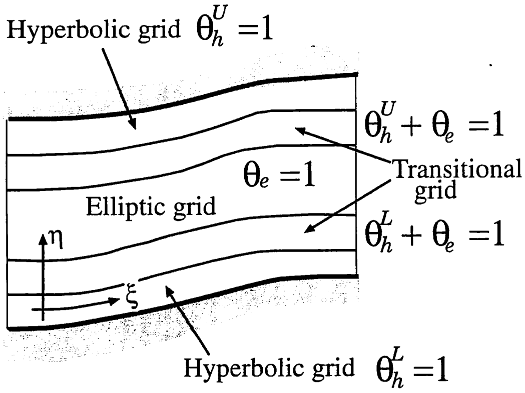
Hyperbolic grid $\theta_h^U = 1$
$\theta_h^U + \theta_e = 1$
$\theta_e = 1$
Transitional grid
Elliptic grid
$\theta_h^L + \theta_e = 1$
η
ξ
Hyperbolic grid $\theta_h^L = 1$

Figure 2. Schematic hyperbolic and elliptic zones for a channel.

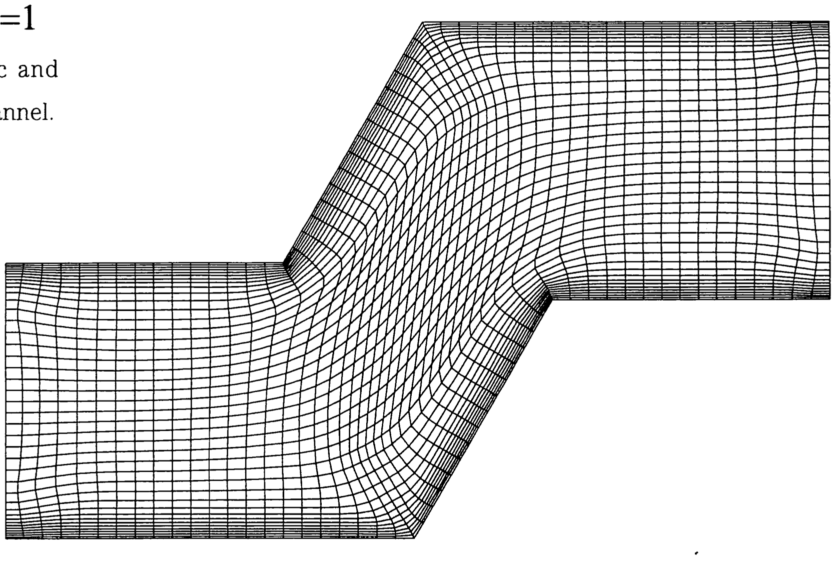

Figure 3. Grid generated for a channel.

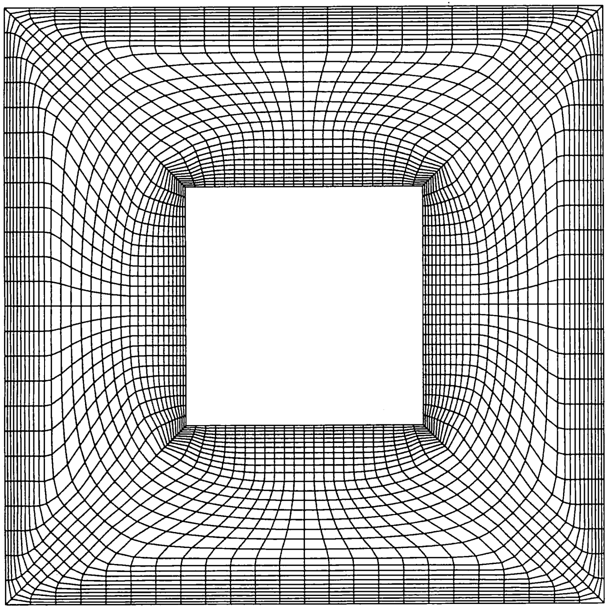

Figure 4. Grid generated for a square with inner boundary.

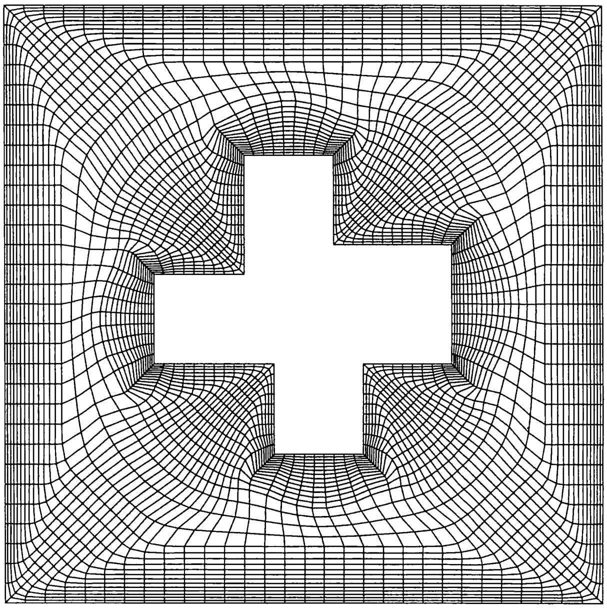

Figure 5. Grid generated for a square including complex inner boundary.

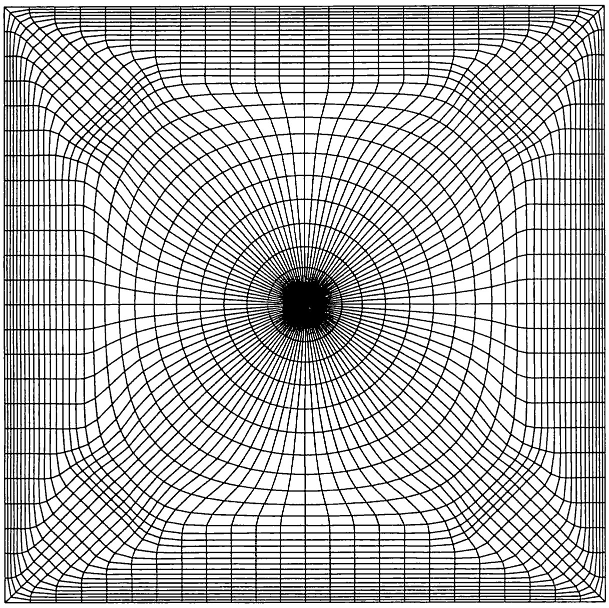

Figure 6. O-grid generated for an enclosed square region.

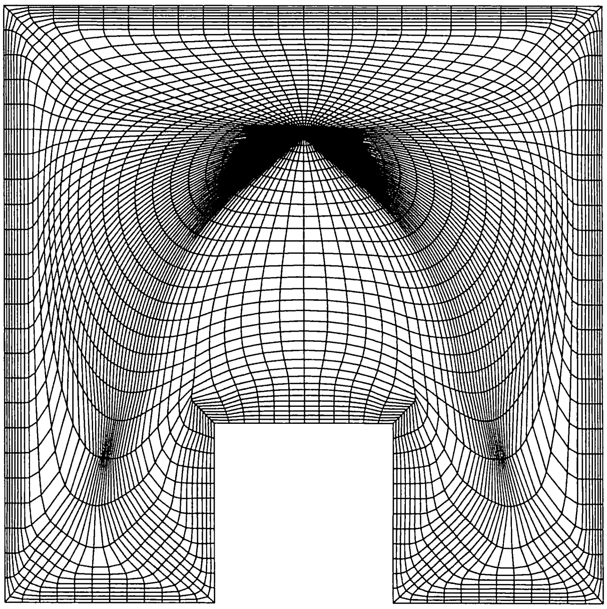

Figure 7. Grid generated for an enclosed region with sharp corners.