COMPUTER METHODS IN APPLIED MECHANICS AND ENGINEERING 57 (1986) 279–295
NORTH-HOLLAND

GENERATION OF SOLUTION-ADAPTIVE COMPUTATIONAL GRIDS USING OPTIMIZATION

Richard CARCAILLET, George S. DULIKRAVICH and Stephen R. KENNON

Texas Institute for Computational Mechanics, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, TX 78712-1085, U.S.A.

Received 24 July 1985
Revised manuscript received 28 October 1985

A new method for generating solution-adaptive computational grids is presented that builds on the requirement that the adapted grid retains maximum possible smoothness and local orthogonality. The approach taken is one of nonlinear optimization, where an objective function combining measures of grid smoothness, local orthogonality, and cell volume control is minimized using a fast iterative scheme. The method is multidimensional by construction, and accepts any arbitrary grid as input, even a grid that is initially overlapped. Several applications to published test problems allow comparisons with some of the existing adaptive grid generation methods. An example of dynamic adaptation to the solution of finite difference equations is also given that demonstrates the error reduction capabilities of the new adaptive grid generation and optimization method, and suggests further applications to practical engineering problems.

1. Introduction

The generation of solution-adaptive computational grids for finite difference or finite element computation is a research topic currently receiving a great deal of interest [1–3]. The accuracy of a numerical scheme can be greatly improved when a fixed number of grid points are dynamically redistributed so as to better resolve the regions where large variations occur in the solution. Alternatively, the benefits of grid adaptation may be understood in terms of increased computational efficiency when using a single solution-adaptive grid as opposed to straightforward static grid refinement. Since the stability of most numerical schemes depends on the grid quality—smoothness in particular—the conclusion is that the grid adaptation process should not result in excessive and uncontrolled distortion of the initial computational grid.

Solution-adaptive grid generation methods fall into two broad categories, according to the principle underlying the grid point motion algorithm: equidistribution of some error measure over the domain on one hand, and explicit use of variational principles on the other hand. Methods belonging to the first category are often based on one-dimensional point motion algorithms [4–6]. Methods taking the variational approach are not subject to this constraint and therefore seem most promising for multidimensional solution-adaptive grid generation. They are based on the minimization of a volume-weighted measure of the variations of some solution quantity or of its numerical error over the computational domain [7–10].

The method described in this paper takes the latter approach with the goal of achieving

effective control over the conflicting requirements of grid smoothness, local orthogonality, and adaptation. In this sense, good quality solution-adaptive grids are sought that avoid pitfalls such as: (a) excessive skewness of the adapted grid; (b) lack of convergence of the point motion algorithm that frequently results in oscillatory behavior of the adapting grid; (c) excessive clustering that leads to collapsing of grid lines in regions of large solution variations; (d) excessive depletion of points in regions of small solution variations. Such drawbacks have been reported in the past [2, 3] for certain adaptive grid methods applied to two-dimensional problems, thus raising serious questions about the reliability of solution-adaptive grid generation.

2. Background

The present method is a natural extension of a recently published static grid generation and optimization concept that has been successfully applied in two and three dimensions [11, 12]. The variational method of Brackbill and Saltzman [7] gave the impetus for its development, and analogies between the method of Kennon and Dulikravich and the method of Brackbill and Saltzman will be outlined first. The following discussions address two-dimensional problems, but all the concepts presented here are easy to formulate in three dimensions. Brackbill and Saltzman achieved control of grid quality and grid adaptation by quantifying the mapping (Fig. 1) between the physical space (x, y) and the uniformly discretized computational space (ξ, η) by the following integrals:

$$\begin{aligned} I_s &= \int\int [(\nabla\xi)^2 + (\nabla\eta)^2]\, d\xi\, d\eta \quad \text{(smoothness)}, \\ I_o &= \int\int [\nabla\xi \cdot \nabla\eta]^2\, d\xi\, d\eta \quad \text{(orthogonality)}, \\ I_v &= \int\int w(\xi, \eta) J^2\, d\xi\, d\eta \quad \text{(volume control)}. \end{aligned} \tag{1}$$

Here, J is the determinant of the local Jacobian transformation matrix,

$$J = \xi_x \eta_y - \eta_x \xi_y ,$$

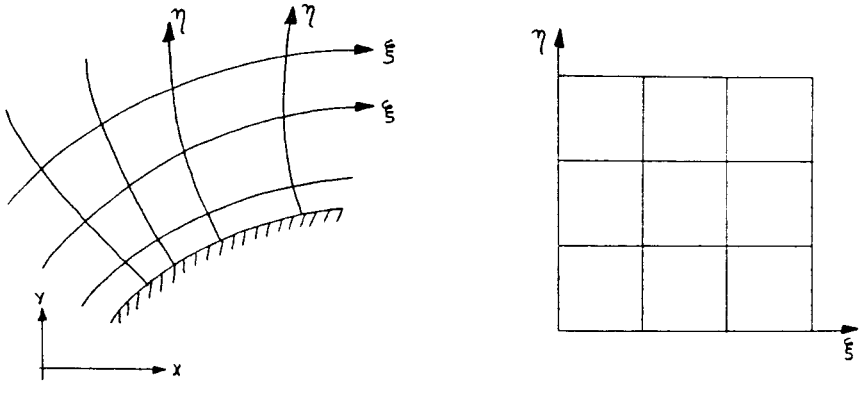


Fig. 1. Physical plane and computational plane.

and ∇ is the gradient operator in physical coordinates. The integrals above are transformed in the computational space and a linear combination, I, of the transformed integrals is minimized. This variational problem is solved by applying the Euler–Lagrange equations of variational calculus to the integrand of

$$I = I_s + \lambda_o I_o + \lambda_v I_v , \quad \lambda_0, \lambda_v \geq 0 . \tag{2}$$

A system of coupled second-order partial differential equations results that is solved by standard Gauss–Jacobi iteration after discretization by central differences in the computational space.

3. Analysis

In order to improve computational efficiency and reliability of this solution-adaptive grid generation method, a more heuristic formulation is adopted here by considering local optimal adaptation problems for a given arbitrary computational grid. This approach is discussed in detail elsewhere [13]. We will limit ourselves here to a brief description of the principles and unique features of our method.

Consider that a computational grid is given by the set of the physical coordinates of all its vertices. A master cell is centered at the grid point $P_{ij} \equiv P(x_{ij}, y_{ij})$. Each master cell is made up of the four neighboring grid cells that share point P_{ij} (Fig. 2). The grid points are assumed to be connected by straight line segments. Four position vectors connecting P_{ij} to its immediate neighbors are defined as:

$$\begin{aligned}
\boldsymbol{r}_{i+1,j} &= (x_{i+1,j} - x_{i,j})\hat{e}_x + (y_{i+1,j} - y_{i,j})\hat{e}_y , \\
\boldsymbol{r}_{i,j+1} &= (x_{i,j+1} - x_{i,j})\hat{e}_x + (y_{i,j+1} - y_{i,j})\hat{e}_y , \\
\boldsymbol{r}_{i-1,j} &= (x_{i-1,j} - x_{i,j})\hat{e}_x + (y_{i-1,j} - y_{i,j})\hat{e}_y , \\
\boldsymbol{r}_{i,j-1} &= (x_{i,j-1} - x_{i,j})\hat{e}_x + (y_{i,j-1} - y_{i,j})\hat{e}_y ,
\end{aligned} \tag{3}$$

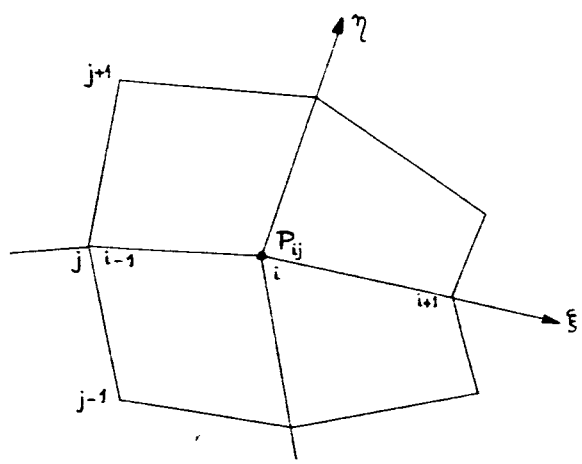


Fig. 2. Two-dimensional master cell.

w e $\hat{e}_x$ and $\hat{e}_y$ are unit vectors in the Cartesian coordinate directions x and y, respectively. The master cell is smooth if it has minimal change in cell area from one elementary cell to the next. Therefore, a quantitative measure of the departure from smoothness of each master cell can be formulated as:

$$\mathrm{SM}_{ij} = (A_1 - A_2)^2 + (A_2 - A_3)^2 + (A_3 - A_4)^2 + (A_4 - A_1)^2 , \tag{4a}$$

where the A_k are approximate area measures for the areas of the elementary cells, e.g.,

$$A_1 = \|(\boldsymbol{r}_{i+1,j} \times \boldsymbol{r}_{i,j+1})\| . \tag{4b}$$

The master cell is orthogonal if the curvilinear coordinate lines $\xi = i = \text{const}$ and $\eta = j = \text{const}$ intersect at P_{ij} at right angles (Fig. 2). Hence, a quantitative measure of the departure from local orthogonality of the master cell is:

$$\mathrm{ORT}_{ij} = (\boldsymbol{r}_{i+1,j} \cdot \boldsymbol{r}_{i,j+1})^2 + (\boldsymbol{r}_{i,j-1} \cdot \boldsymbol{r}_{i+1,j})^2 + (\boldsymbol{r}_{i-1,j} \cdot \boldsymbol{r}_{i,j-1})^2 + (\boldsymbol{r}_{i,j+1} \cdot \boldsymbol{r}_{i-1,j})^2 . \tag{5}$$

At the same time, a cell volume control functional for the master cell is sought in the general form

$$\mathrm{VOC}_{ij} = A_{ij} \times W_{i,j} . \tag{6}$$

Here, A_{ij} is the area of the master cell (or volume in three dimensions) and $W_{i,j}$ is the value of a suitably chosen positive weight function evaluated at P_{ij}. This weight function will be discussed later. It is easily seen from (6) that minimizing the sum over all the master cells of the volume control functionals will cause the master cells to shrink where W is large and expand where W is small. A simplified, nonexact, but computationally more efficient formulation for VOC_{ij} is obtained, for example, by considering the master cell as a system of tension springs connecting the grid points and the adaptation procedure as the minimization of the energy of this system. It reads:

$$\mathrm{VOC}_{ij} \simeq w_1\|\boldsymbol{r}_{i+1,j}\|^2 + w_2\|\boldsymbol{r}_{i,j+1}\|^2 + w_3\|\boldsymbol{r}_{i-1,j}\|^2 + w_4\|\boldsymbol{r}_{i,j-1}\|^2 , \tag{7}$$

where the 'spring constants' are

$$w_1 = \tfrac{1}{2}(W_{i,j} + W_{i+1,j}) , \quad w_2 = \tfrac{1}{2}(W_{i,j} + W_{i,j+1}) ,$$

$$w_3 = \tfrac{1}{2}(W_{i,j} + W_{i-1,j}) , \quad w_4 = \tfrac{1}{2}(W_{i,j} + W_{i,j-1}) .$$

This simple formulation presents some significant advantages. Notice that only function values of the weight function at the grid points are needed in our method, thus avoiding the numerical computation of its derivatives that are required by most adaptive grid methods. If higher-order derivatives are employed, this computation may yield very 'noisy' data, causing the adapted grid to oscillate [14]. Notice also that the simple averaging of function values in the definition of the spring stiffness constants strongly couples adjacent grid points within a

master cell, effectively preventing the grid points from responding excessively to abrupt variations of W. Consequently, severe distortion of the adapted grid is avoided. The fact that the volume control functional is perfectly symmetric with respect to P_{ij} justifies the expectation that effective volume control should be obtained regardless of the direction of ∇W with respect to the curvilinear coordinate directions of the initial grid. This tension springs analogy for the volume control functional has also been used for the adaptive refinement of triangular finite elements [15]. Nevertheless, it has been done without grid quality control, and the point motion algorithm rests on a static equilibrium statement rather than on minimizing the energy of the analog system as done here.

The global objective function, F, is obtained by forming a weighted linear combination of the local grid quality measures and the local volume control functional for each master cell and summing them up over all the master cells:

$$F = \sum_i \sum_j [\alpha \, \mathrm{ORT}_{ij} + (1-\alpha)\mathrm{SM}_{ij} + \beta \, \mathrm{VOC}_{ij}] , \tag{8}$$

where α is a scalar weight parameter enabling a trade-off between grid smoothness and local orthogonality to be achieved in the static optimization ($\beta = 0$) or optimal solution-adaptation procedure ($\beta \neq 0$). The ranges of variation for these parameters are:

$$0 \leq \alpha \leq 1 , \qquad 0 \leq \beta \leq 1 . \tag{9}$$

However, the adaptation procedure has been shown to converge successfully, even for $\beta \sim \mathrm{O}(10)$ (see Example 4.2).

The global objective function is then rewritten as a function of the vector V containing the physical coordinates of all the grid points in a natural ordering:

$$V = \{(x_{ij}, y_{ij}) : 1 \leq i \leq I, 1 \leq j \leq J\} . \tag{10}$$

Unconstrained minimization of the function F of $2 \times I \times J$ variables is performed next, using the Fletcher–Reeves conjugate gradient method [16]. The minimization of F reduces to a succession of one-dimensional minimization problems in the variable ω_n (11), called the line search parameter. This iterative procedure yields corrections to the physical coordinates of the grid points,

$$V^{n+1} = V^n + \omega_n \delta V^n , \tag{11}$$

where

$$\delta V^0 = -\nabla F(V^0) , \qquad \delta V^n = -\nabla F(V^n) + \beta_n \delta V^{n-1} ,$$

$$\beta_n = \|\nabla F(V^n)\|^2 / \|\nabla F(V^{n-1})\|^2 .$$

At the nth iteration level, this one-dimensional problem can therefore be formulated as:

$$\text{Minimize} \quad \psi(\omega_n) = F(V^{n+1}) = F(V^n + \omega_n \delta V^n) . \tag{12}$$

From (4), (5), (7), (8), and (12) it follows that ψ is a fourth-order polynomial in ω_n. Hence,

each iteration consists in finding and testing the three roots of

$$\partial\psi/\partial\omega_n = 0 \tag{13}$$

and selecting the real root ω_n that minimizes ψ. The line searching part of the algorithm is exact and inexpensive to compute. The iterative procedure is halted when the optimal grid, given by V^*, is found, that satisfies

$$\|\nabla F(V^*)\| \leq \varepsilon\,. \tag{14}$$

Here ε is the user-specified convergence criterion.

The gradients of the three functionals in the global objective function are normalized, in order to properly account for smoothness, local orthogonality, and volume control in the iterative procedure [16]. In addition, these gradients are volume weighted, that is,

$$\nabla G_{ij}/J_{ij} \tag{15}$$

is used, where $G \equiv$ SM, ORT, VOC, respectively, and where J_{ij} is the Jacobian of the master cell centered at P_{ij}. It follows that the smaller grid cells are emphasized in the optimal adaptation procedure. This is consistent with the fact that grid distortion due to volume control is more likely to appear in regions of large solution variations, that is in regions where the cells are small. Tighter grid quality control is required there. Moreover, this prevents regions where the solution variations are negligible to become devoid of points.

has been shown [17] that, when grid points can be added to the original grid, the most efficient way to improve numerical accuracy is to add the points only in regions of strong gradients. Otherwise, more numerical diffusion must be added to prevent nonlinear instability when solution gradients are insufficiently resolved [18]. Finally, it has been proved in the one-dimensional case [7] that increased resolution of gradients does reduce the numerical error. Therefore, choosing for W a measure of the gradient of some dependent variable Q is appropriate, in the form:

$$W_{ij} = \|\nabla Q/Q\|_{ij}^2\,. \tag{16}$$

For compressible flow problems with shocks, Q may be the density ρ, the pressure p, or the Mach number M. Alternatively, W_{ij} may reflect the variations over the grid of the current residual in iterative time-asymptotic solutions. Improved resolution in regions of large local residuals would improve the iterative convergence rate of the solution or, alternatively, yield the same rate of convergence with fewer grid points. For finite element applications, it has been recently shown [9] that for proper choice of W in terms of error norms, the variational adaptation procedure [7] will minimize an approximation to the local interpolation error.

Some control over the range of variations of the weight function is desirable. The minimum cell size determines the allowable time step for explicit methods, and the cells cannot be allowed to become too large. The scaling procedure proposed by Brackbill and Saltzman is suitable. It is formulated here in a slightly different manner. Define

$\sigma = \min($

where max W and
a user-supplied pa

$w_{i,j} = [($

The range of varia

$0 \leq \min$

$\min W =$

In both cases, ma

For a given valu
expand in regions
W is large (see (1

Additionally, s
abrupt, localized
evolving numerica
calculated using W

4. Results

The first applic
very fast. All the
In an initial phase
by closely followi
iterations significa

In all the test c
by six orders of m
obtain a nearly op
been enforced by

A sequence of
square domain is
examination of th
algorithm, withou
examples but the
with the grids ob

EXAMPLE 4.1.

$W = A$

where r is the ra

$$\sigma = \min(\sigma_0, \max W/\min W)\,, \tag{17}$$

where max W and min W are the maximum and minimum values of W_{ij} over the grid and σ_0 is a user-supplied parameter ($\sigma_0 \geq 1$). The weight function is scaled by setting

$$w_{i,j} = [(\sigma^2 - 1)W_{i,j}/\sigma \max W] + 1/\sigma\,. \tag{18}$$

The range of variation of W follows from (18):

$$0 \leq \min W \leq 1\,, \quad \text{if } \sigma_0 \leq \max W/\min W\,, \tag{19a}$$

$$\min W = 1\,, \quad \text{if } \sigma_0 \geq \max W/\min W\,. \tag{19b}$$

In both cases, max $W = \sigma$.

For a given value of σ_0, it follows that, when the range of variation of W is wide, cells will expand in regions where W is small, thus allowing for stronger clustering in the regions where W is large (see (19a)).

Additionally, smoothing the scaled weight function w may be necessary in the case of abrupt, localized variations of W or in the case of nonsmooth numerical data supplied by the evolving numerical solution being done on the grid [13]. The volume control functional is then calculated using W.

4. Results

The first applications of this method have shown the optimal adaptation procedure to be very fast. All the adapted grids presented next have been obtained in ten or twenty iterations. In an initial phase, grid points are redistributed to minimize the energy of the spring system, by closely following the variations of W. This takes three iterations at most. Subsequent iterations significantly improve the smoothness and local orthogonality of the adapted grid.

In all the test cases but one, the maximum corrections to the grid point coordinates dropped by six orders of magnitude in less than twenty iterations. In practice, ten iterations suffice to obtain a nearly optimal adapted grid. In all the test cases, orthogonality at the boundaries has been enforced by allowing the boundary points to 'float' on their respective boundaries.

A sequence of applications is performed in which a given uniform Cartesian grid on a square domain is made to adapt to a weight function that is given analytically. This allows examination of the behavior of the grid according to the sole influence of the point motion algorithm, without interference from a particular field problem or numerical scheme. All examples but the first one are taken from the literature on adaptive grids, inviting comparison with the grids obtained by some other adaptation methods.

EXAMPLE 4.1. On the unit square ($0 \leq x \leq 1, 0 \leq y \leq 1$), the weight function is defined as:

$$W = A \exp(-Br^2)\,,$$

where r is the radial distance of a point measured from the center of the square.

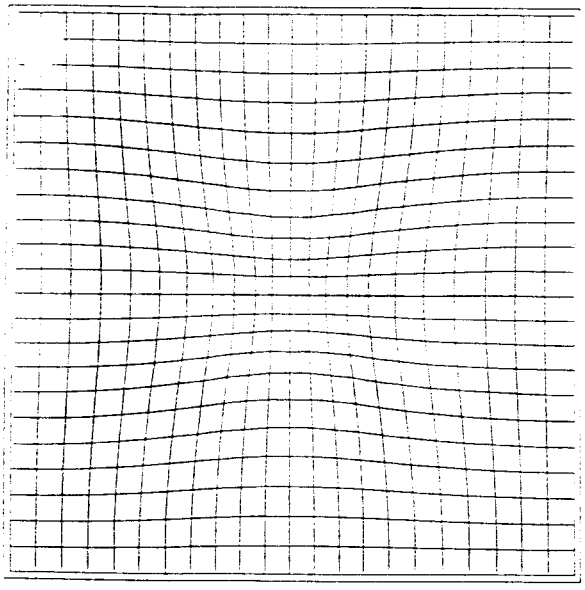


Fig. 3. Adapted grid, $W = 1000 \exp(-20(x^2 + y^2))$, $\alpha = 0.5$, $\beta = 0.5$, $\sigma_0 = 1000$.

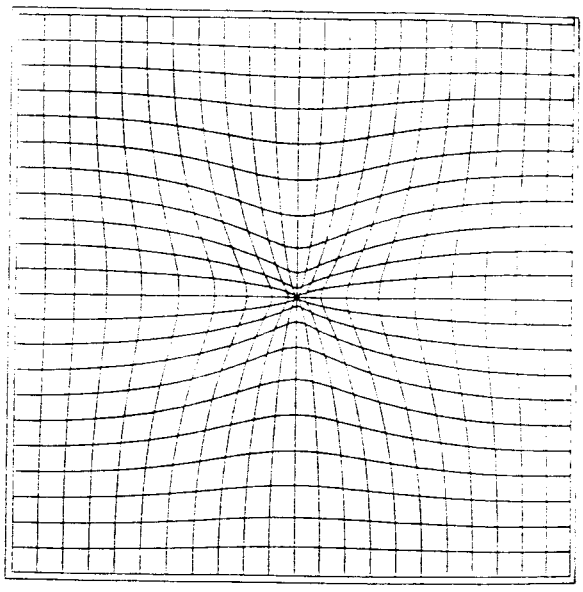


Fig. 4. Same example as Fig. 3 without scaling of W.

For $r = 0$ at $(x, y) = (\frac{1}{2}, \frac{1}{2})$, $A = 1000$, and $B = 20$, the grid is optimally adapted for $\alpha = 0.5$, $\beta = 0.5$, and $\sigma_0 = 1000$ (Fig. 3). Setting $\beta = 1$ results, as expected, in smaller cells at the center of the square. The effect of not scaling W (see (18)) is shown in Fig. 4: the smallest grid cells are highly distorted.

EXAMPLE 4.2. A grid on the unit square is made to adapt to the variations of sinusoidal functions. This example demonstrates that the volume control capability does not depend on the direction of ∇W relative to the x- and y-coordinate directions. The input parameters are: $\alpha = 0.5$, $\beta = 0.5$, and $\sigma_0 = 100$, unless otherwise noted.

$$W = \sin(2\pi x) + 1 + 1/\sigma_0 . \tag{20a}$$

The cells vary only in the x-coordinate direction (Fig. 5).

$$W = \sin(2\pi x) \times \sin(2\pi y) + 1 + 1/\sigma_0 . \tag{20b}$$

Satisfactory volume control is achieved: where W is large ($(x, y) = (\frac{1}{4}, \frac{1}{4})$) the cells are small; where W is small ($(x, y) = (\frac{3}{4}, \frac{1}{4})$) the cells are large (Fig. 6). For $\beta = 16$, the procedure still results in a fully converged, stable adapted grid (Fig. 7). The variational method [7] failed to converge on this same example for such high values of β (or λ_v as defined in (2)).

$$W = \sin[4\pi(x + y)] + 1 + 1/\sigma_0 . \tag{20c}$$

W varies along the diagonal of the grid, and so do the cells (Fig. 8). The grid obtained for $\beta = 1$ shows that the increased grid clustering does not cause excessive skewness of the largest

Fig

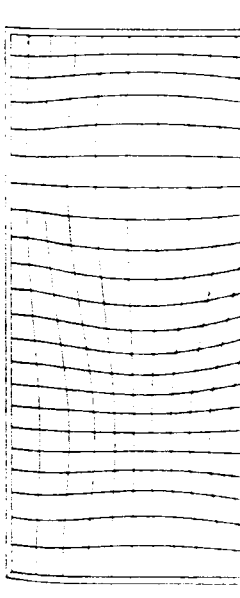
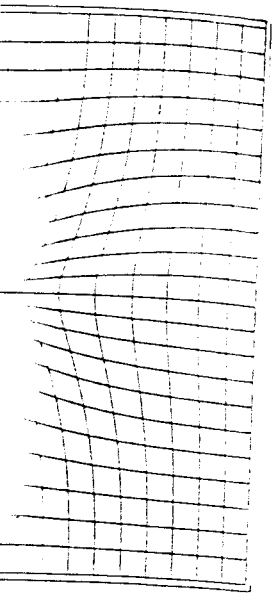


Fig. 6. Adapted grid.
$1/\sigma_0$, $\alpha = 0.5$, $\beta = 0.5$

cells (Fig. 9). A
maxima and mini
accurate and the
highly skewed grid
and orthogonality
procedure for $\beta =$



without scaling of W.

apted for $\alpha = 0.5$,
cel' t the center
smallest grid cells

ions of sinusoidal
es not depend on
parameters are:

(20a)

(20b)

e cells are small;
e procedure still
nod [7] failed to
n (2)).

(20c)

rid obtained for
ess of the largest

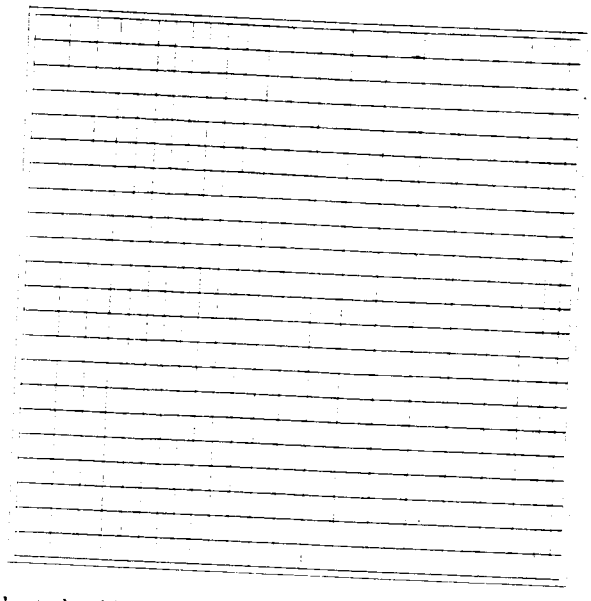


Fig. 5. Adapted grid, $W = \sin(2\pi x) + 1 + 1/\sigma_0$, $\alpha = 0.5$, $\beta = 0.5$, $\sigma_0 = 100$.

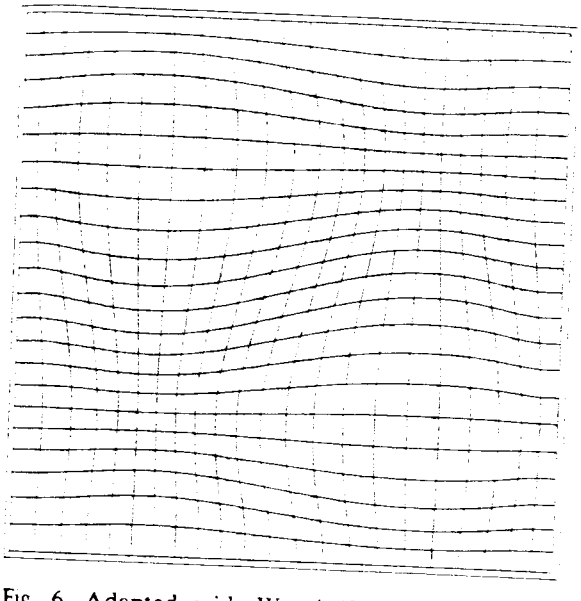


Fig. 6. Adapted grid, $W = \sin(2\pi x) \times \sin(2\pi y) + 1 + 1/\sigma_0$, $\alpha = 0.5$, $\beta = 0.5$, $\sigma_0 = 100$.

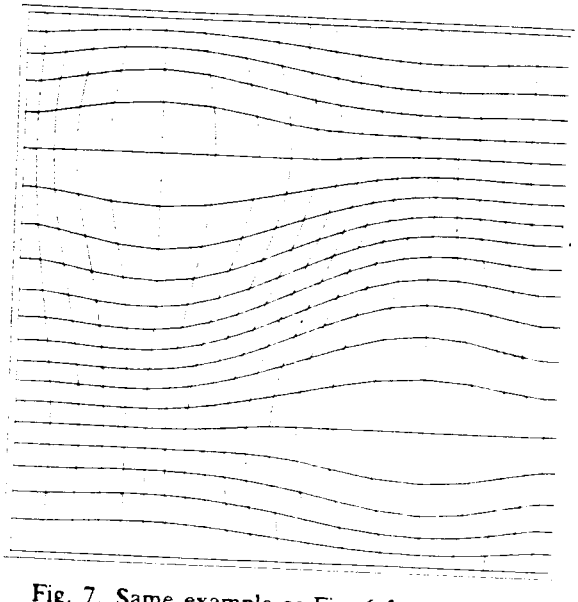
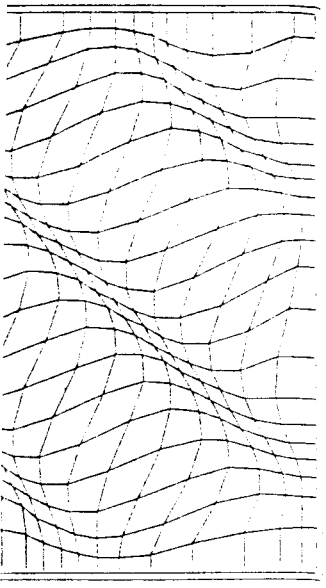


Fig. 7. Same example as Fig. 6 for $\beta = 16$.

cells (Fig. 9). A contour plot of W demonstrates that the adapted grid closely follows the maxima and minima of W (Fig. 10). In the original example [7], volume control is not as accurate and the largest cells are highly skewed. Minimizing $\sum_i \sum_j \mathrm{VOC}_{ij}$ alone results in a highly skewed grid (Fig. 11): it is seen that optimizing the adapted grid in terms of smoothness and orthogonality is an essential part of this method. The convergence history of the iterative procedure for $\beta = 1$ is illustrated in Fig. 12, for optimal adaptation (Fig. 9) as well as for



ple as Fig. 8 for $\beta = 1$.

$\times W$, $m \sim \min W$.

ontrol still allows for a

(21)

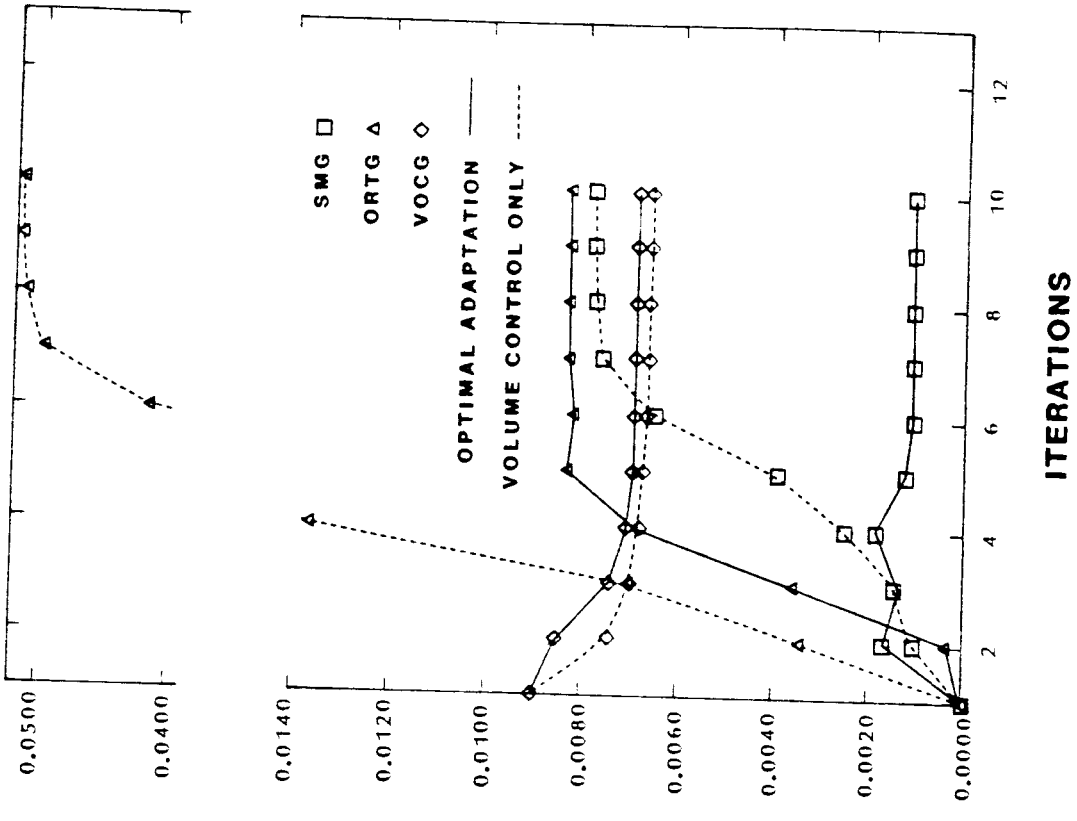


Fig. 12. Convergence history for the example of Figs. 9 and 11.

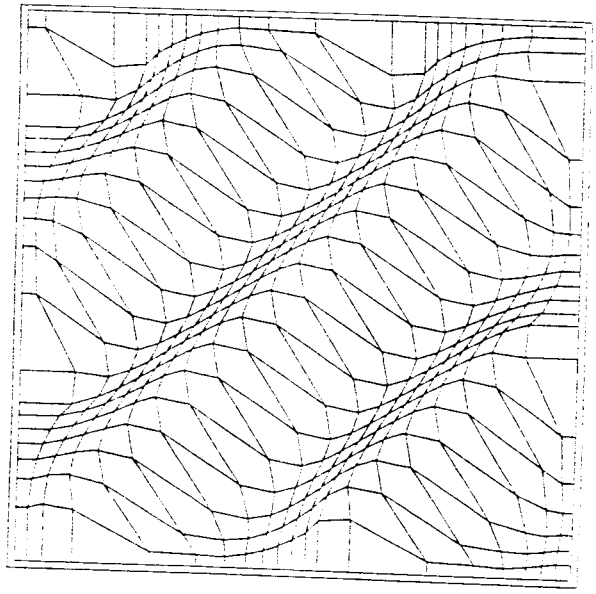


Fig. 11. Same example as Fig. 9, straightforward adaptation.

while greatly reducing the jaggedness and skewness of the adapted grid. The latter are quantified by

$$\mathrm{SMG} = \sum_i \sum_j \mathrm{SM}_{ij} , \tag{22}$$

$$\mathrm{ORTG} = \sum_i \sum_j \mathrm{ORT}_{ij} . \tag{23}$$

EXAMPLE 4.3 [10]. A grid on the square domain $[-2, 2] \times [-2, 2]$ is made to adapt to the gradient of the hyperbolic tangent function T defined by

$$T = \tau(x^2 + y^2 - 1) + \tau(y - x) , \tag{24}$$

where

$$\tau(r) = A \tanh(D \times r) .$$

Disturbances on the unit circle and on the diagonal of the grid are thus created, in the form of a jump at $r = 0$ from $-A$ to $+A$ with a jumping rate increasing with D. A and D were set to 1 and 3, respectively, as in the original example [10]. The input parameters are $\alpha = 0.5$, $\beta = 0.5$, and $\sigma_0 = 100$. The optimally adapted grid is perfectly symmetrical with respect to the diagonal $y = x$ and exhibits a high quality [10] (Fig. 13).

The reduction in numerical error allowed by an optimally adapted grid is investigated on a ple test case, which has been studied by Brackbill and Saltzman [7]. It deals with the stationary solution of the convection-diffusion equation

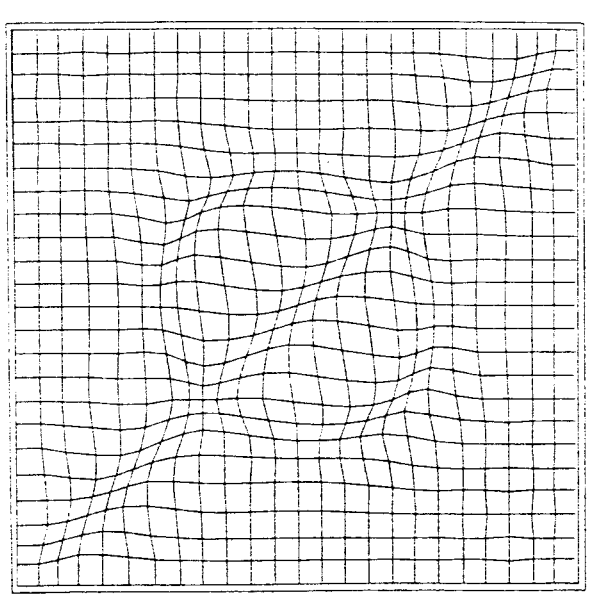


Fig. 13. Adapted grid, $W = \nabla[T(x^2 + y^2 - 1) + T(y - x)]$, T = hyperbolic tangent function, $\alpha = 0.5$, $\beta = 0.5$, $\sigma_0 = 100$.

$$\phi_t + \nabla \cdot (\phi U) - \kappa \nabla \cdot \nabla \phi = 0 , \tag{25}$$

where $U = U(r)r$ is given, and κ is a positive scalar constant. Here, $r = (x^2 + y^2)^{0.5}$ and $r = r\hat{r}$. The first integral of (25) on an infinite domain is simply

$$\phi U - \kappa \nabla \phi = 0 . \tag{26}$$

When κ is small, $\nabla\phi/\phi$ will be very large where U is finite. Choosing for $U(r)$ a function which is nonzero only in a narrow annulus at $r = r_0$ will result in the formation of a singularity on the circle of radius r_0. Considering the unit square with $r = 0$ at its center and $r_0 = \frac{1}{4}$, U is taken [7] as

$$U(r) = -ru_0^2 h_+ h_- / \kappa ,$$
$$h_\pm = 1/[1 + \exp(\pm u_0(r - r_0)/\kappa)] . \tag{27}$$

where

Using $\kappa = 0.08$ and $u_0 = 6$ results in the contour plot of ϕ shown in Fig. 14. An analytic expression for ϕ is then obtained from (25), (26) as

$$\phi(r) = \phi(0) \exp[u_0 r h_+ / \kappa - \kappa \ln(h_- / h_+(r = 0))/u_0] . \tag{28}$$

The departure from cylindrical symmetry noticeable in Fig. 14 is simply due to the interpolation of ϕ-values performed by the contour routine on the 48×48 Cartesian grid. The numerical error ε is defined as

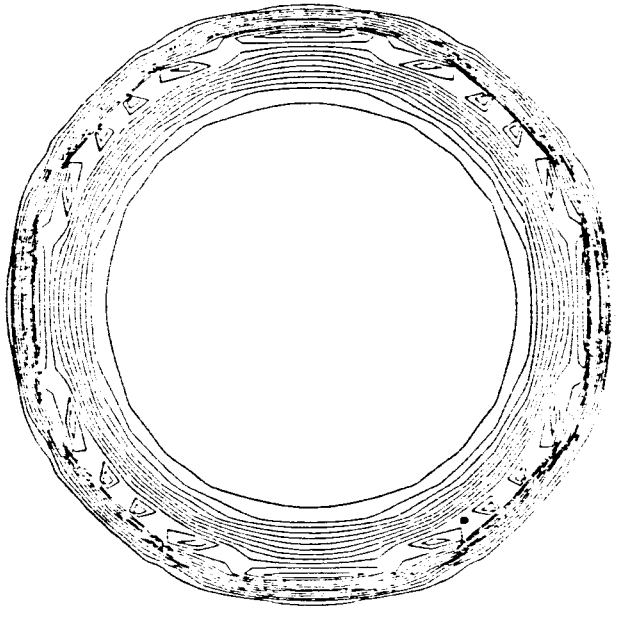


Fig. 14. Contours of ϕ on 48×48 Cartesian grid.

$$\varepsilon = \max_{ij}[\|U - \kappa\nabla\phi/\phi\|] . \tag{29}$$

The weight function is taken as

$$W = \|\nabla\phi/\phi\|^2 . \tag{30}$$

Fig. 15 shows a contour plot of W on the same grid. The outermost contours correspond to $W = 0$, and W exhibits a very steep variation on the annulus centered at $r = r_0$.

The numerical experiments are performed as follows. For $\sigma_0 = 100$, successive optimal adaptation cycles consisting of ten iterations each are carried out for different values of α and β. Function ϕ and the weight function W are recalculated on the grid resulting from each adaptation cycle. Due to the extreme singular character of W, the convergence of the iterative procedure is not as good as in the static adaptation examples, the corrections to the grid point coordinates dropping by two orders of magnitude only after ten cycles. One cycle performed for $\alpha = 0$ and $\beta = 1$ yields a numerical error ε that is 68.8% smaller than on the initial uniform grid. A second cycle brings a total error reduction of 87.3%. At this point, the grid is optimally adapted. Further cycles show that the numerical error ε starts increasing again, although very slowly (after three additional cycles ε is 8% greater than after the first two cycles). The influence of α on the attainable error reduction is investigated by repeating the above experiment for $\alpha = 0.5$. The results are summarized in Table 1. Clearly, the minimization of the volume control functional competes with that of the orthogonality measure. A

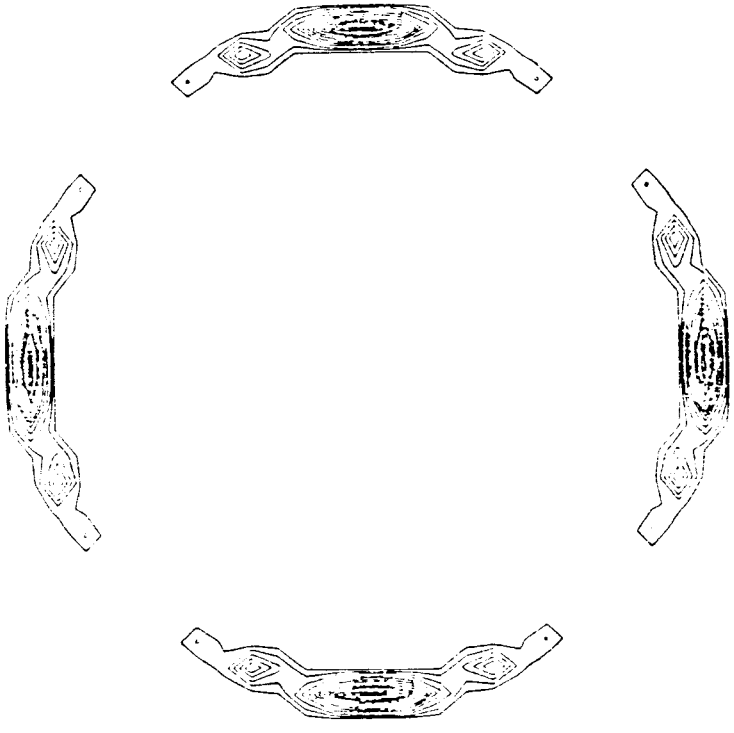


Fig. 15. Contours of W prior to scaling on 48×48 Cartesian grid.

Table 1
Influence of α on erro

Initial grid	
Adapted grid (10 it.)	
Adapted grid (10 + 10	

similar observatio
the adaptation pro
conclusion based
the adaptive grid
The grids resultin
than in the origina
singularity are nea
the adapted grid.

The influence o
decrease in ε as
increased. Finally
The numerical err
96×96 (Fig. 19).
error on the corr
cycle of ten iterati
800-II computer.

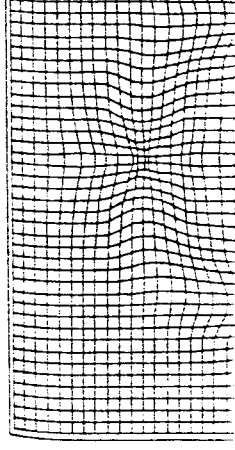


Fig. 16. Adapted gri

Table 1
Influence of α on error reduction ($\beta = 1$)

	$\alpha = 0$		$\alpha = 0.5$	
	ε (%)	% reduction	ε (%)	% reduction
Initial grid	45.31	—	45.31	—
Adapted grid (10 it.)	14.13	68.8	28.27	37.6
Adapted grid (10 + 10 it.)	5.64	87.6	28.16	37.8

similar observation was made when using the variational method [7]. Note that for $\alpha = 0.5$, the adaptation procedure does converge as well as in the static grid adaptation examples. The conclusion based on the numerical experimentation is that to maximize the error reduction, the adaptive grid should be optimized in terms of smoothness rather than local orthogonality. The grids resulting from the first two adaptation cycles (Figs. 16, 17) are noticeably smoother than in the original example using the variational method [7]. Specifically, cells away from the singularity are nearly undisturbed, and reasonable cell aspect ratios are maintained throughout the adapted grid.

The influence of β on the adaptation procedure is illustrated in Fig. 18. It is seen that the decrease in ε as β increases from 0 to 1 is significantly greater than when β is further increased. Finally, optimal grid adaptation is compared to straightforward grid refinement. The numerical error ε decreases in both cases as the number of cells increases from 12×12 to 96×96 (Fig. 19). The error on the adapted grid is consistently one-sixth to one-eighth the error on the corresponding uniform grid. As for computer time requirements, an adaptation cycle of ten iterations on the initial 48×48 grid required one minute of CPU time on a Harris 800-II computer.

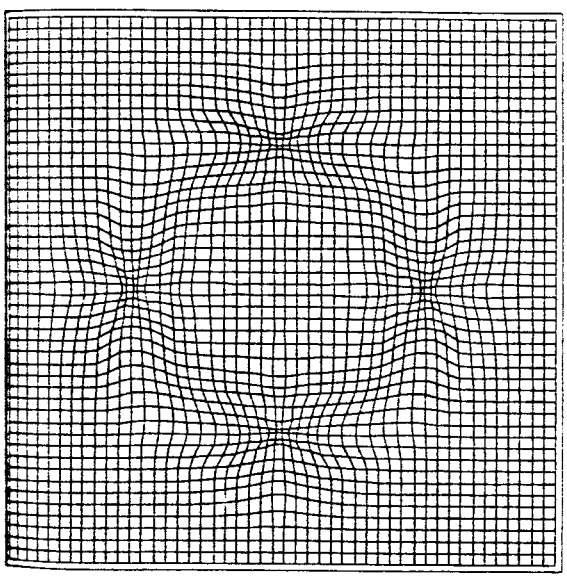


Fig. 16. Adapted grid after one cycle (10 iterations).

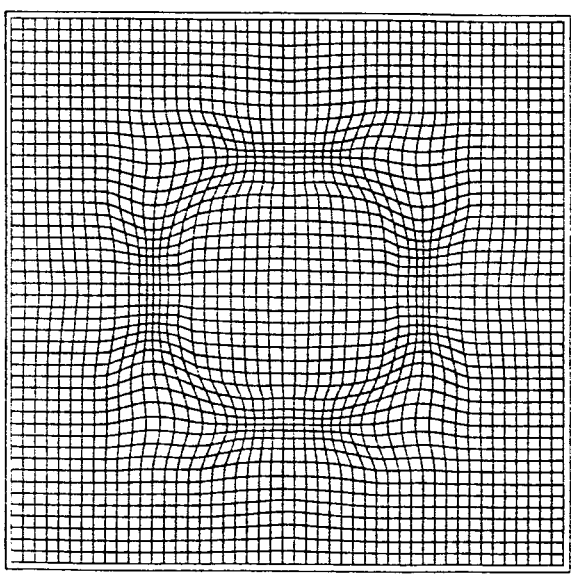


Fig. 17. Adapted grid after two cycles (10 + 10 iterations).