VARIATIONAL BARRIER METHOD OF ADAPTIVE GRID GENERATION IN HYPERBOLIC PROBLEMS OF GAS DYNAMICS *

BORIS N. AZARENOK †

Abstract. Application of the harmonic mapping using a variational approach to generate moving adaptive grids in the hyperbolic problems of gas dynamics is considered. On the example of three-point model of adaptation it is demonstrated possibility to generate an unfolded mesh with strong grid lines condensing in the vicinity of discontinuities of the control/(monitor) function. The algorithm of redistributing the boundary nodes is suggested consisting of using constrained minimization of the discrete harmonic functional when constraints define the boundary of the domain. In real computations due to mesh adaptation it is possible to reduce the errors, caused by shock waves smearing over the cells, by many factors of ten. Modeling of the 2-D supersonic gas flow in the channel has shown to achieve the same accuracy on the adaptive grid with the same structure as the quasiuniform mesh it is required less CPU memory by factor of 25 and running time by factor of 50 to 60. Computational tests of the steady transonic and supersonic flow over an airfoil demonstrate the ability of the method in controlling mesh sizes across shocks.

Key words. Moving adaptive grids, harmonic mapping, shock waves, constrained optimization

AMS subject classifications. 65M50, 76-08

1. Introduction. Moving adaptive grid technology has important applications in the problems of fluid dynamics. The essence of such an approach, referred as r-refinement, is in adjusting redistribution of the grid nodes in such a manner so that to catch particularities in the solution of interest with fixed computer cost. Constructive way is that we try to position more grid nodes in the domains of sharp change in the solution being the regions of high gradients while the mesh structure remains invariable that makes more simple the course of computation. Examples of some r-refinement based methods can be found in Carey [10], Charakhch'yan and Ivanenko [12], Hawken et al. [21], Ivanenko [24], Jacquotte [26], Li et al. [28], H.Tang and T.Tang [38], Thompson [39], Zegeling [41]. In Liao et al. [30] the deformation method is suggested, where the grid velocities are determined by solving a scalar Poisson equation. An approach based on Moving Mesh Partial Differential Equations method has been considered in Cao et al [8, 9] and Huang [22]. Here the Euler-Lagrange equations, both stationary and with the time dependent left part, to the variational functional are solved and mesh adaptation is performed using a class of monitor functions being the symmetric positive definite matrices.

When constructing an adaptive moving grid the main difficulty is in maintaining its nondegeneracy. Probably, an approach should consist of extending the equdistribution principle, providing a one-to-one mapping of logic space onto the physical domain in discrete approach, from widely used 1-D case to 2-D. Some ways to define equdistribution-like methods in 2-D dimensions can be found in Baines [6], Cao et al [8, 9], Dwyer et al. [14], Huang [22], Zegeling [41]. In Ivanenko [25] it has been suggested to use a variational functional depending on derivatives of the functions sought and on variable coefficients, being the elements of some symmetric positive defined matrix, and in discrete approach it is proven the following variational principle: a mapping, minimizing that functional, is one-to-one.

*The paper has been appeared in a preliminary form in [1]. This work was supported by the Russian Fund of Fundamental Research (project code 02-01-00236).

†Computing Center of Russian Academy of Sciences, Vavilov str. 40, Moscow, 119991, Russia (E-mail: azarenok@ccas.ru).

One of the mesh generation techniques is in using a harmonic mapping, first applied in Winslow [40], and in particular a variational approach when minimizing the Dirichlet (or harmonic) functional of smoothness we ensure the grid lines to be as smooth as possible, see Brackbill and Saltzman [7]. To include adaptivity in this process it has been suggested to write the Dirichlet functional on the surface of the graph of the control/(monitor) function, being the solution of the basic problem or somehow connected with it, and to perform adaptation by solving the Euler-Lagrange equations to the harmonic functional, see Liseikin [31, 32]. Application of such monitor surfaces has been also considered in Dwyer et al. [14], Eisman [15], Spekreijse [37]. Note that Dvinsky [13] was the first to use the harmonic mapping for mesh adaptation where the gradient of the monitor function is utilized to perform grid lines clustering.

Numerical solution of the discrete Euler-Lagrange equations can not always provide generation of the unfolded mesh even in domains without adaptation, see examples of grids to the backstep in Knupp and Steinberg [27], Ivanenko [24]. It means, though in continuous approach there exists an unique harmonic mapping, being one-to-one, when discretizing the property of the mapping to be one-to-one can be lost. The discrete mapping can be also non-unique, see Garanzha and Kaporin [18].

To provide a one-to-one harmonic mapping at discrete level Charakhch'yan and Ivanenko [11] have suggested a variational barrier method of grid generation in a physical domain without adaptation when the mapping is constructed by minimizing the harmonic functional. The functional is approximated in such a manner that there is an infinite barrier ensuring all grid cells to be convex quadrilaterals. This approach has been extended to adaptive grid generation when the harmonic functional is written on the surface of the control function, see Ivanenko [23, 24], Charakhch'yan and Ivanenko [12]. In Azarenok [2], Azarenok and Ivanenko [4, 5] this approach has been applied in 2-D unsteady problems of gas dynamics when as a control function it is used one of the flow parameters or superposition of several parameters.

The purpose of the present work is to show some theoretical aspects and practical possibilities of using the variational barrier method of constructing structured adaptive grids in hyperbolic problems of gas dynamics with discontinuous solution. In 1-D case based on solving the nonlinear advection equation it is shown that the regularized discrete functional is convex and its minimum is attained on the mesh which can be strongly condensed in the vicinity of discontinuities of the control function, meanwhile this functional keeps the infinite barrier preventing the grid cells from collapsing. In 2-D case it is shown at strong grid lines condensing the infinite barrier disappears, nevertheless, when using the iterative procedure of mesh generation, we can guarantee the grid to be unfolded. The algorithm of redistributing the boundary nodes, consisting of using constrained minimization of the functional when constraints define the boundary of the domain, is suggested. Such an approach allows to perform consistent redistribution of grid nodes inside the domain and on its boundary that increases reliability of grid generation and modeling the flow problem. Modeling of the 2-D supersonic gas flow in the channel has shown to achieve the same accuracy on the adaptive grid with the same structure as on the quasiuniform mesh it is required less CPU memory by factor of 25 and running time by factor of 50 to 60. Computational tests of the steady transonic and supersonic flow over an airfoil demonstrate possibility to control mesh sizes across shocks.

2. Problem formulation. The theory of harmonic mappings is useful for formulating a well-posed variational grid generation problem. The energy of a mapping $\phi : (M, g)\to(N, h)$ between two n-dimensional Riemannian manifolds M and N with metric tensors g_{ij} and h_{ij} is the function $e(\phi) : M \rightarrow \mathcal{R}(\geq 0)$, defined in some local coordinates ξ^i, μ^i as

$$e(\phi) = \frac{1}{2} g^{ij}(\xi) \frac{\partial \mu^k}{\partial \xi^i} \frac{\partial \mu^l}{\partial \xi^j} h_{kl}(\mu),$$

where g^{ij} is the inverse metric. The energy functional (or total energy) of the mapping ϕ is defined as, see Eells and Lemaire [16], Dvinsky [13],

$$E(\phi) = \int_M e(\phi) d\xi, \quad \text{where } d\xi = \sqrt{det(g)} d\xi^1 \cdots \xi^n. \tag{2.1}$$

A smooth mapping $\phi : (M, g) \rightarrow (N, h)$ is harmonic if it is a critical point of the energy functional E.

Sampson [35] and Schoen and Yau [36] have shown that the harmonic mapping $M \rightarrow N$ when $dimN{=}2$ is always a homeomorphism (one-to-one) provided that the curvature of N is nonpositive and the boundary ∂N is convex (with respect to the metric h_{ij}). But it is sometimes not true when $dimN{>}2$ [17].

Now we turn to the problem formulation of the surface grid generation utilized to constructing the adapted mesh [12, 24, 32]. Let a simply connected domain $\Omega \subset \Re^2$ with a smooth boundary in plane $x{-}y$ be given. Consider the surface S^{r2} of the graph of the control/(monitor) function $z{=}f(x, y)$, see Fig. 2.1. The surface S^{r2} can be presented in Euclidean space $\mathcal{R}^3$ by a parametrization $\boldsymbol{r}(\xi, \eta){=}(x(\xi, \eta), y(\xi, \eta), f[x(\xi, \eta), y(\xi, \eta)])$, where ξ, η are the local coordinates on S^{r2}. We seek for a mapping of the unit square in parametric plane $\xi{-}\eta$ onto the domain Ω, when a mapping of the boundary of square onto the boundary of Ω is given, so that the mapping of the surface S^{r2} onto the parametric square is harmonic. The conditions of the above Sampson, Shoen and Yau's statement are obviously satisfied to the mapping of the physical domain Ω onto the parametric domain Ω_p with Euclidean metric $h_{ij}{=}\delta_{ij}$ (here δ_{ij} is the Kronecker symbol), being the square in parametric plane $\xi{-}\eta$. Hence, the nondegenerate harmonic coordinates ξ, η may be constructed on the surface S^{r2}. The coordinate lines ξ, η are then projected on the physical domain Ω and the result is an adaptive-harmonic grid.

To construct the harmonic maping we use the variational formulation of the Dirichlet functional, being the measure of closeness of the mapping to conformal. The task in hand is to minimize the Dirichlet functional written on the surface S^{r2}. This functional of smoothness can be expressed through the invariants $\mathcal{I}_1$, $\mathcal{I}_2$ of the orthogonal transforms of the covariant metric tensor g_{ij} [31, 32], namely,

$$I = \int_{S^{r2}} \frac{\mathcal{I}_1}{\mathcal{I}_2} dS^{r2}.$$

The invariant $\mathcal{I}_2 = det(g_{ij}) = g_{11}g_{22} - (g_{12})^2$ and it equals the area squared of the cell on the surface S^{r2}, the invariant $\mathcal{I}_1 = g_{11} + g_{22}$ means the sum of the length squared of the cell sides. Noting that $dS^{r2} = \sqrt{g_{11}g_{22} - (g_{12})^2} d\xi d\eta$ we get

$$I = \int_0^1 \int_0^1 \frac{g_{11} + g_{22}}{\sqrt{g_{11}g_{22} - (g_{12})^2}} d\xi d\eta, \tag{2.2}$$

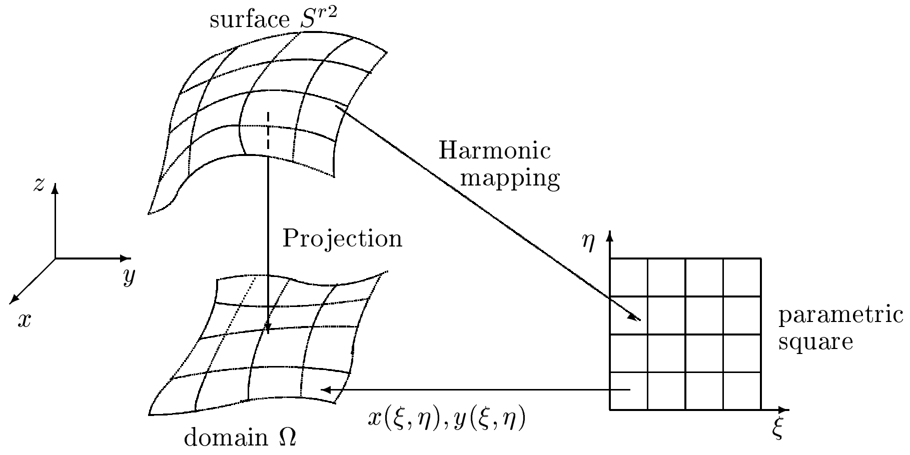


FIG. 2.1. *Adaptive grid generation*

The components of the tensor g_{ij} are defined on the surface S^{r2} with the local coordinates ξ, η in Euclidean space $\mathcal{R}^3$ as follows

$$g_{11} = \boldsymbol{r}_\xi^2 = x_\xi^2 + y_\xi^2 + f_\xi^2 = x_\xi^2 + y_\xi^2 + (f_x x_\xi + f_y y_\xi)^2,$$
$$g_{12} = (\boldsymbol{r}_\xi \cdot \boldsymbol{r}_\eta) = x_\xi x_\eta + y_\xi y_\eta + f_\xi f_\eta = x_\xi x_\eta + y_\xi y_\eta + (f_x x_\xi + f_y y_\xi)(f_x x_\eta + f_y y_\eta),$$
$$g_{22} = \boldsymbol{r}_\eta^2 = x_\eta^2 + y_\eta^2 + f_\eta^2 = x_\eta^2 + y_\eta^2 + (f_x x_\eta + f_y y_\eta)^2.$$

Substituting them in (2.2) we get the following form of the functional

$$I = \int_0^1\int_0^1 \frac{(x_\xi^2 + x_\eta^2)(1 + f_x^2) + (y_\xi^2 + y_\eta^2)(1 + f_y^2) + 2 f_x f_y (x_\xi y_\xi + x_\eta y_\eta)}{(x_\xi y_\eta - x_\eta y_\xi)\sqrt{1 + f_x^2 + f_y^2}} d\xi d\eta. \tag{2.3}$$

This form of the harmonic functional has been derived in [12, 24] from the energy functional (2.1) (if multiplied by 2) written for the surface S^{r2}

$$E = \int_0^1\int_0^1 Tr(g_{ij}^{-1})\sqrt{det(g_{ij})} d\xi d\eta \ ,$$

where $Tr(g_{ij}^{-1}) = g^{ii}$.

With purpose to control the degree of coordinate lines condensing in the domains of high gradients it is convenient to use $c_a f$ instead of the control function f, where c_a is a coefficient of adaptation [24] which can depend on variables x, y. Thus, we work with the control function multiplied by some coefficient c_a in order to increase or decrease adaptation.

In 1-D case to generate the inverse harmonic mapping of the graph of f onto the unit segment in parametric space ξ it is required to minimize the following functional, see Liseikin [31, 32],

$$I = \int_0^1 \frac{1}{x_\xi \sqrt{1 + c_a^2 f_x^2}} d\xi. \tag{2.4}$$

Here we at once use $c_a f$ instead of f and it means we seek for the grid points arc-length equidistribution in metric of the curve $c_a f$.

For the direct mapping of the unit segment in parametric space ξ to the curve $c_a f$ the Dirichlet functional has the form

$$I = \int_0^1 x_\xi^2 (1 + c_a^2 f_x^2) d\xi. \tag{2.5}$$

The Euler-Lagrange equations to the functionals (2.4) and (2.5) are similar

$$x_\xi \sqrt{1 + c_a^2 f_x^2} = const \tag{2.6}$$

3. Approximation of functional. We consider a piecewise bilinear mapping of the unit square $i \leq \xi \leq i+1$, $j \leq \eta \leq j+1$ in parametric plane $\xi - \eta$ onto a quadrilateral grid cell in plane $x - y$ formed by nodes with coordinates $(x, y)_{i,j}$, $(x, y)_{i+1,j}$, $(x, y)_{i+1,j+1}$, $(x, y)_{i,j+1}$ numbered from 1 to 4 in an anticlockwise manner, where i, j are positive integers. The functional (2.3) is approximated in such a way that its minimum is attained on a grid of convex quadrilaterals, referred as a convex grid [12, 24]

$$I^h = \sum_{i=1}^{i_{max}} \sum_{k=1}^{4} \frac{1}{4} [F_k]_i \,, \tag{3.1}$$

where F_k is the integrand evaluated in the k-th corner of the i-th cell. If the set of convex meshes is not empty, the system of algebraic equations written to every interior node (here i is a global node number)

$$R_x = \frac{\partial I^h}{\partial x_i} = 0 \,, \quad R_y = \frac{\partial I^h}{\partial y_i} = 0 \,, \tag{3.2}$$

has at least one solution being a convex mesh. To find it one should have an initial convex mesh and then to use a method of unconstrained minimization [4, 12, 24]. It has been shown [11, 12, 24] when generating a curvilinear mesh without adaptation in an arbitrary simply connected domain Ω in plane $x - y$ the discrete functional (3.1) (if no adaptation then in (2.3) we define $f_x = f_y = 0$) has an infinite barrier on the boundary of the set of convex grids, see Fig. 3.1. This is caused by the condition of positiveness to the Jacobian of the mapping $J = x_\xi y_\eta - x_\eta y_\xi$ in (2.3). Should the vertexes of some cell displace so that the cell becomes nearly nonconvex, see Fig. 3.1b, then one of four triangles, into which the cell is divided by its two diagonals, degenerates, its area, equal $0.5J$, tends to zero and, therefore, $I^h \to \infty$. The infinite barrier holds in case of adaptive grid generation as well when the control function $f \in C^1(\Omega)$ since the values f_x, f_y under the square root in (2.3) are bounded everywhere in Ω. When starting from an initial convex mesh, due to the infinite barrier every step of the minimization procedure can be chosen so that the mesh always remains convex. However, if f is of the class of discontinuous functions, what we generally have in hyperbolic problems of gas dynamics, in the vicinity of a discontinuity the derivatives f_x, f_y become unbounded, the infinite barrier disappears and this causes some grid cells to fold and modeling to break. In order to prevent grid lines overlap it is used the procedure of regularization to the discrete functional described in Sec. 5.

We make use the rectangular rule to compute the 1-D functional (2.4)

$$I^h = \sum_{i=1}^{i_{max}} \frac{\triangle \xi}{(x_\xi)_{i+1/2} \sqrt{1 + c_a^2 (f_x)_{i+1/2}^2}}, \tag{3.3}$$

where i_{max} is the number of spacings and the derivatives are computed via

$$(x_\xi)_{i+1/2} = (x_{i+1} - x_i)/\triangle \xi \quad , \qquad (f_x)_{i+1/2} = (f_{i+1} - f_i)/(x_{i+1} - x_i) \,. \tag{3.4}$$

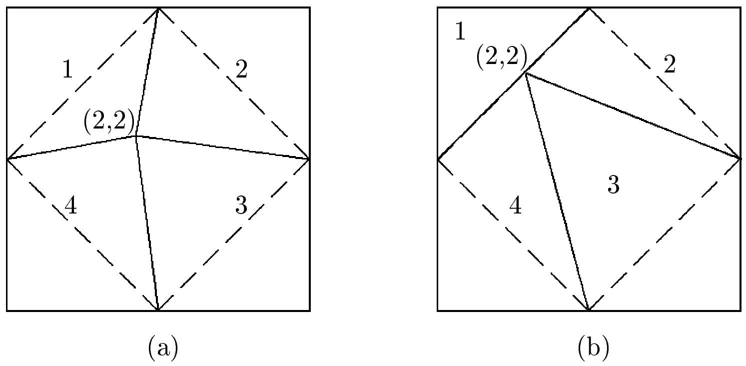


FIG. 3.1. *When minimizing the discrete functional I^h there is an infinite barrier preventing the cells from folding. The shown mesh consists of 4 cells. At minimization only the node $(2,2)$ moves. Dash lines indicate the domain inside of which the node $(2,2)$ shall be placed so as all four cells to be convex (a). If this node approaches the dash line (b), one (right) of two triangles, into which cell 1 is divided by the diagonal, degenerates, its area tends to zero, and, therefore, $I^h \to \infty$.*

4. Three-Point model of adaptation. In this section we demonstrate that when solving the Cauchy problem to the nonlinear advection equation under some conditions the values of the discrete function in the cells of the moving mesh remain invariable. Then the problem of constructing an adaptive mesh can be considered separately from the basic problem as if to use an analytical control function. This allows to analyze some properties of the discrete functional which also hold in general case of computing a real flow with discontinuities.

Consider 1-D adaptive mesh generation when solving the initial value problem (IVP) to the nonlinear advection equation with discontinuous initial data when the shock moves from the left to the right

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0, \quad u(x,0) = \begin{cases} u_l & \text{if} \quad x < 0\ , \\ u_r & \text{if} \quad x \geq 0\ , \end{cases} \qquad u_l > u_r > 0\ . \tag{4.1}$$

We use the integral equation

$$\oint_C u\,dx - \frac{1}{2}u^2 dt = 0, \tag{4.2}$$

which in case of a smooth solution is equivalent to the above differential equation and a discontinuous solution is governed by it as well. Here the contour C is the boundary of an arbitrary domain in plane $x{-}t$.

In plane $x{-}t$ we introduce the moving grid, see Fig. 4.1a, with spacings at the n-th time level $h_{i+1/2}{=}x_{i+1}^n{-}x_i^n$ and at the $n{+}1$-th level $h^{i+1/2}{=}x_{i+1}^{n+1}{-}x_i^{n+1}$ and time step $\triangle t{=}t^{n+1}{-}t^n$, where n, i are integers. Let at t^n the cell-average values of the discrete function $u_{i+1/2}$ are defined in the center of intervals. To update the values $u^{i+1/2}$ at time t^{n+1} we use the Godunov's scheme on the moving grid [20, 19]. This scheme can be obtained if we integrate (4.2) along the contour C being the boundary of the computing cell

$$u^{i+1/2}h^{i+1/2} - u_{i+1/2}h_{i+1/2} - u_{i+1}h_{i+1} + h_i u_i + \frac{\triangle t}{2}\left[(u_{i+1})^2 - (u_i)^2\right] = 0, \tag{4.3}$$

where $h_i{=}x_i^{n+1}{-}x_i^n$. In order to determine a flux through the intercell boundary $[x_i^n, x_i^{n+1}]$ we need the value u_i on it which is defined via the condition across the

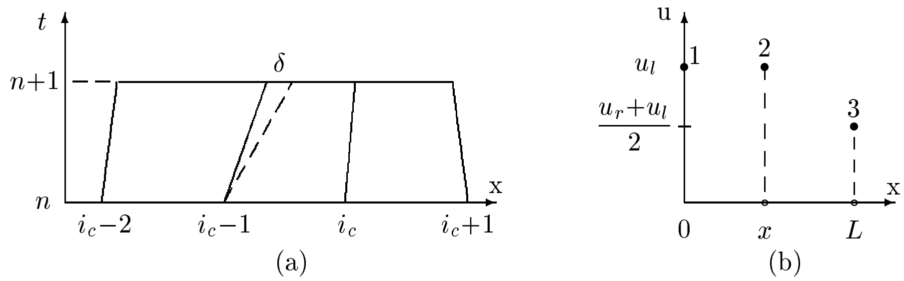


FIG. 4.1. *Moving grid (a); i_c is a center of the shock smeared. If the mesh moves with speed of shock $w = 0.5(u_r+u_l)$ and number of intervals i_{max} is even, then the updated value $u^{i+1/2}$ is not change. Three-point model of adaptation (b); in the coordinate system moving with speed of shock w, $x_1=0$ and $x_3=L$ are the coordinates of the fixed boundary nodes, $x_2=x$ is a moving node.*

discontinuity

$$u_i = \begin{cases} u_{i-1/2} & \text{if } h_i/\triangle t < 0.5(u_{i-1/2} + u_{i+1/2}) \ , \\ u_{i+1/2} & \text{otherwise} \ . \end{cases}$$

The above condition takes into acount the inclinations of the straight-line characteristic dx/dt=$0.5(u_{i-1/2}$+$u_{i+1/2})$ at the point x_i^n and slanted boundary of the cell $[x_i^n, x_i^{n+1}]$ defined as $h_i/\triangle t$. If the i-th node moves faster than perturbations from this node then we set u_i=$u_{i+1/2}$, and if vice versa then u_i=$u_{i-1/2}$.

From (4.3) we get the values at t^{n+1}

$$\text{(4.4)} \quad u^{i+1/2} = \frac{1}{h^{i+1/2}} \left\{ u_{i+1/2}h_{i+1/2} - \frac{\triangle t}{2} \left[(u_{i+1})^2 - (u_i)^2 \right] - u_i h_i + u_{i+1}h_{i+1} \right\} .$$

THEOREM 4.1. *Let without adaptation the grid nodes move with speed of shock w=$0.5(u_r$+$u_l)$. Suppose at time t^n the values $u_{i+1/2}$=u_l if i<i_c and $u_{i+1/2}$=u_r if i≥i_c, where i_c is a node being the center of the shock smeared and the mid mesh node, i.e., i_c=$0.5i_{max}$+1, where i_{max} is an even number of intervals. Then the updated values $u^{i+1/2}$ do not change.*

Proof. First, note that the problem is symmetrical about the i_c-th node. This node moves with the velocity w. Consider the interval (i_c-1, i_c), see Fig 4.1a. Let at time t^{n+1} the i_c−1-th node shift by δ from the position

$$x_{i_c-1}^{n+1} = x_{i_c-1}^n + w\triangle t,$$

where it would be if no adaptation. Denote $h_{i_c-1/2}$=h, then $h^{i_c-1/2}$=h−δ. Observing that projections of the cell lateral edges onto x are h_{i_c}=$w\triangle t$, h_{i_c-1}=$w\triangle t$+δ, and defining the intercell values from the condition across the discontinuity: u_{i_c-1}=u_l and u_{i_c}=u_r, we get from (4.4)

$$u^{i_c-1/2} = \frac{1}{h-\delta} \left[u_l h - \frac{\triangle t}{2} \left(u_r^2 - u_l^2 \right) - (w\triangle t + \delta)u_l + u_r w \triangle t \right]$$

$$= \frac{1}{h-\delta} \left[u_l h - \frac{\triangle t}{2} (u_r + u_l)(u_r - u_l) + w\triangle t \, (u_r - u_l) - \delta u_l \right] = u_l .$$

A similar result holds if u_{i_c}=u_l. □

Consequently, when constructing the adaptive mesh we need not solve the IVP (4.1) and can merely set the values equal u_l in the cells to the left of the $i_c{-}th$ point and u_r to the right.

We shall construct the adaptive mesh minimizing the functional (3.3) and using u as a control function. We can simplify the model considering the left half of the mesh consisting of three points, i.e., when i_c=3. Such an assumption does not change the mesh structure. The general case can be easy obtained from this three-point model. We pass into the new coordinate system moving with the velocity w so that x_1=0, see Fig. 4.1b. Then, when adapting, 1 and 3 are the fixed boundary nodes and x_3=L, where L=2h, h is a spacing of the initial uniform mesh, coordinate x_2 is variable, referred further as x. We also have u_1=u_2=u_l , u_3=$(u_l$+$u_r)/2$.

5. Properties of discrete functional. In this section we consider properties of the discrete functional in 1-D and 2-D cases within the framework of three-point model of adaptation. It will be shown if the coefficient c_a is larger than some critical value, minimization of the functional leads the right cell of the two-cell grid to collapse. In order to get a convex functional in 1-D case, i.e., to provide the unique solution of the minimization problem, it is used a regularized functional. The improved functional holds an infinite barrier preventing the right cell from collapsing to any extent of grid nodes condensation. In 2-D case meaning of the minimization procedure to the functional can be lost due to absence of a solution in the minimization problem. Nevertheless, we shall demonstrate the iterative procedure allows to condense significantly the grid lines towards the discontinuity and guarantee the grid to be unfolded.

First, consider 1-D case. In assumptions of Sec. 4 the approximation (3.3) to the functional on the two-cell grid reads (we set $\triangle\xi = 1$)

$$I^h = \frac{1}{x} + \frac{1}{(L-x)\sqrt{1+c_a^2\triangle u^2/(L-x)^2}}, \tag{5.1}$$

where $\triangle u$=$|u_3-u_2|/2$=$|u_r-u_l|/2$. To minimize the one-parametric functional I^h we apply the Newton method

$$x^{p+1} = x^p - \tau\frac{\partial I^h}{\partial x}\left[\frac{\partial^2 I^h}{\partial x^2}\right]^{-1}, \tag{5.2}$$

where the iterative parameter $\tau\leq 1$, the first derivatives is

$$\frac{\partial I^h}{\partial x} = -\frac{1}{x^2} + \frac{1}{(L-x)^2\left[1+c_a^2\triangle u^2/(L-x)^2\right]^{3/2}},$$

and the second derivative is derived of the first one.

In Fig. 5.1 the distributions of I^h and $\partial I^h/\partial x$ are presented for several values of c_a at u_l=2, u_r=1, and initial uniform spacing h=0.1. One can see at c_a=0 the functional I^h has the minimum at x=0.1 which corresponds to the point's equidistribition or a uniform mesh. When c_a>0 the functional I^h loses convexity in the interval $(0, L)$ and besides the minimum there appears a maximum to the left of point 3. Therefore, solution of the problem on finding its extremum becomes non-unique. When increasing c_a on one hand the value x_{min}, where I^h reaches the minimum in $(0, L)$, shifts to the right corresponding to point 2 moving towards point 3 or grid

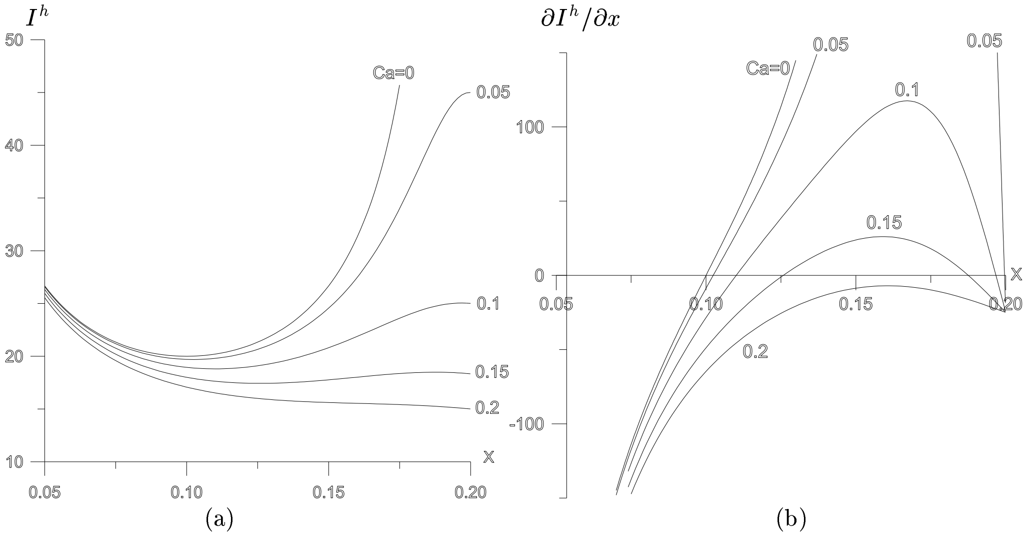


FIG. 5.1. *Dependence of distribution for the functional I^h (a) and $\partial I^h/\partial x$ (b) on c_a within the interval $(0, L)$, where $L{=}2h{=}0.2$. Here the part of the interval $(0, 0.05)$ is cut off since it is out of interest. At $c_a{>}0.185$ there is no minimum of the functional, see the curve $c_a{=}0.2$, and minimization of I^h causes the right cell to collapse.*

clustering, see Fig. 5.1b, on the other hand x_{max}, where I^h reaches the maximum in $(0, L)$, shifts to the left from point 3 causing grid rarefaction. In 2-D problems it can cause harsh displacements of the mesh nodes due to jumps of the solution from the minimum to maximum and vice versa during iterations, i.e., grid foldness and instability in the solution of the flow problem. Furthermore, for some critical value c_a^{crt}, both extrema merge at the point x^{crt}. To find c_a^{crt} it is required to solve the system of two equations $\partial I^h/\partial x{=}0$ and $\partial^2 I^h/\partial x^2{=}0$ about c_a^{crt} and x^{crt}. For the above parameters $c_a^{crt}{\simeq}0.185$, $x^{crt}{\simeq}0.159$. When $c_a{>}c_a^{crt}$ there is no extremum and minimization of I^h causes the right cell to collapse, see the curve $c_a{=}0.2$ in Fig. 5.1b. Consequently, significant mesh clustering is impossible.

Let us write the discrete functional to the left half of the mesh including $0.5i_{max}$ intervals

$$I^h = \frac{(0.5i_{max} - 1)^2}{x} + \frac{1}{(L-x)\sqrt{1 + c_a^2 \triangle u^2/(L-x)^2}}, \tag{5.3}$$

here $L{=}0.5i_{max}h$. The functional (5.3) is also one-parametric since all intervals except on the right have the same length. Note the larger i_{max} the smaller c_a^{crt} is. For example, at $i_{max}{=}100$ the value $c_a^{crt}{=}0.126$ that reduces possibility to condense the mesh towards discontinuity.

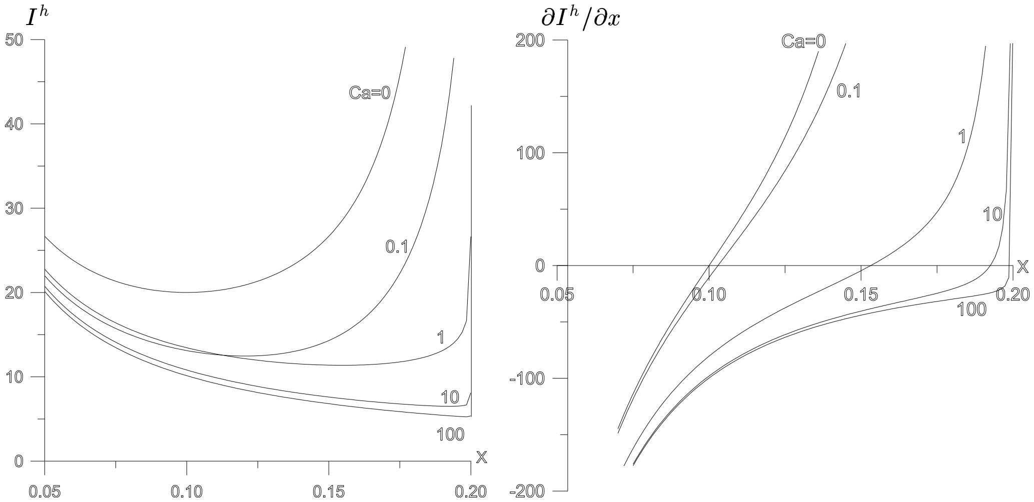
In order to preserve convexity of the discrete functional we assume, when varying x, the derivative of the control function f_x in (3.3) (i.e. u_x in our case) remains fixed (invariable metric) as it was used in [24]. Assuming in (5.1) the derivative $(u_x)_{i+1/2}$ does not depend on x, we obtain the derivative of a new functional I_1^h

$$\frac{\partial I_1^h}{\partial x} = -\frac{1}{x^2} + \frac{1}{(L-x)^2\sqrt{1 + c_a^2 \triangle u^2/(L-x)^2}}. \tag{5.4}$$

To obtain an explicit expression for I_1^h, referred as the regularized functional, we integrate (5.4) and get

$$I_1^h = \frac{1}{x} + \frac{1}{c_a \triangle u} ln \left[\frac{c_a \triangle u}{L - x} + \sqrt{1 + \frac{c_a^2 \triangle u^2}{(L-x)^2}} \right]. \tag{5.5}$$

Distributions of I_1^h and $\partial I_1^h/\partial x$, presented in Fig. 5.2, illustrate three important properties of this new class of functionals,



(a) (b)

FIG. 5.2. *Dependence of distribution for I_1^h (a) and $\partial I_1^h/\partial x$ (b) on c_a. Functional I_1^h is convex within the interval (0,L) for any c_a that guarantees existence of an unique solution of the minimization problem. The infinite barrier prevents the right cell from collapsing.*

1. *The functional I_1^h is convex within the interval (0,L) for any c_a.*
2. *When $c_a \to \infty$ the position of x_{min} for I_1^h tends to L from the left.*
3. *There is an infinite barrier preventing the right cell from collapsing.*

The first property guarantees existence and uniqueness of the solution of the minimization problem within the interval $(0, L)$. According to the second one point 2 can approach point 3 to within any small distance. The third one states that the infinite barrier keeps the mesh nondegenerate. The infinite barrier allows the clustering of the mesh in the vicinity of the discontinuity up to any small size. When modeling a flow with shock waves it allows the accuracy to increase significantly, since the error caused by shock wave smearing, in the integral norms L_1 or L_2 is proportional to the shock thickness. A corollary is that two adjacent cells, one located in the shock zone and the other in the domain of smooth flow, with sizes differing by orders of magnitude, do not degrade the accuracy of the solution. Thus, the mesh can be sharply clustered within one cell towards discontinuity. Those properties hold in case of a mesh with any number of intervals i_{max} as well.

Note, when adapting with using the regularized functional (5.5), the velocity of grid nodes condensing, defined by dependence of the node 2 position on c_a at fixed L and $\triangle u$, is reduced in comparison with the case of the functional (5.1). It can be shown that at small c_a the ratio of the velocities to the functionals (5.1) and (5.5) equal 3. It leads one to have to perform greater number of mesh iterations to obtain necessary degree of nodes condensing. We can increase the velocity of condensing raising the expression under the square root in (5.4) to the power of $1{-}1/2^m$ (here substitution $m{=}1$ gives (5.4)) and such a functional will be referred as I^h_m. In practice it can be used $m{=}2$ as well.

Now we turn to a special case of constructing the 2-D adaptive mesh when the discrete functional can be reduced to one-parametric. Suppose the control function f in (2.3) depends only on the variable x. Therefore, when seeking the mapping of the parametric square onto the domain Ω we have $x{=}x(\xi)$, $y{=}a\eta$ (a=const) and $f_y{=}x_\eta{=}y_\xi{=}0$. Then the harmonic functional (2.3) reads (we set $a{=}1$)

$$I = \int_0^1\int_0^1 \frac{x_\xi^2(1+c_a^2f_x^2)+1}{x_\xi\sqrt{1+c_a^2f_x^2}}d\xi d\eta = \int_0^1 \frac{x_\xi^2(1+c_a^2f_x^2)+1}{x_\xi\sqrt{1+c_a^2f_x^2}}d\xi. \tag{5.6}$$

The functional (5.6) differs from (2.4) by the additional term $\int x_\xi\sqrt{1+c_a^2f_x^2}d\xi$ expressing the curve c_af length in a cross-section $y{=}const$. This term defines the difference in properties of the 1-D and 2-D regularized discrete functionals.

We follow assumptions of Sec. 4 when approximating (5.6) on the two-cell grid

$$I^h{=}\, x+\frac{1}{x}+(L-x)\sqrt{1+c_a^2\triangle u^2/(L-x)^2}+\frac{1}{(L-x)\sqrt{1+c_a^2\triangle u^2/(L-x)^2}}.$$

This discrete functional possesses the similar properties as (5.1) does. Within the interval $(0,L)$ at $c_a{<}c_a^{crt}$ there are maximum and minimum of I^h which disappear at $c_a{>}c_a^{crt}$. Here the value of c_a^{crt} a bit differs from the one in 1-D case.

To regularize I^h we again fix the metric when deriving the first derivative. Derivative of a new functional reads

$$\frac{\partial I_1^h}{\partial x} = 1-\frac{1}{x^2}-\sqrt{1+c_a^2\triangle u^2/(L-x)^2}+\frac{1}{(L-x)^2\sqrt{1+c_a^2\triangle u^2/(L-x)^2}}.$$

Integrating it we get the regularized functional

$$I_1^h = \frac{1}{x}+x+\sqrt{(L-x)^2+c_a^2\triangle u^2}+\frac{1}{c_a\triangle u}(1-c_a^2\triangle u^2)ln\left[\frac{c_a\triangle u}{L-x}+\sqrt{1+\frac{c_a^2\triangle u^2}{(L-x)^2}}\right].$$

Distributions of I_1^h and $\partial I_1^h/\partial x$ are presented in Fig. 5.3. We see as soon as the term $1{-}c_a^2\triangle u^2$ becomes negative, in the above case at $c_a{>}1/\triangle u{=}2$, the functional loses convexity. First, it seems to lead the right cell to collapse when finding the minimum of I_1^h via the iterative procedure (5.2). In practice it does not happen by the following reason. Derive the second derivative of I_1^h

$$\frac{\partial^2 I_1^h}{\partial x^2} = \frac{1}{x^3}+\frac{2(L-x)^2+c_a^2\triangle u^2}{(L-x)^2\sqrt{(L-x)^2+c_a^2\triangle u^2}}\left[\frac{1}{(L-x)^2+c_a^2\triangle u^2}+1\right].$$

When point 2 tending to point 3, i.e. x to L, and consequently we have $L{-}x{\ll}c_a\triangle u$, ratio of the derivatives in (5.2) gives

$$\frac{\partial I_1^h}{\partial x} : \frac{\partial^2 I_1^h}{\partial x^2} \approx (L{-}x^p)\frac{1-c_a^2\triangle u^2}{1+c_a^2\triangle u^2} \quad .$$

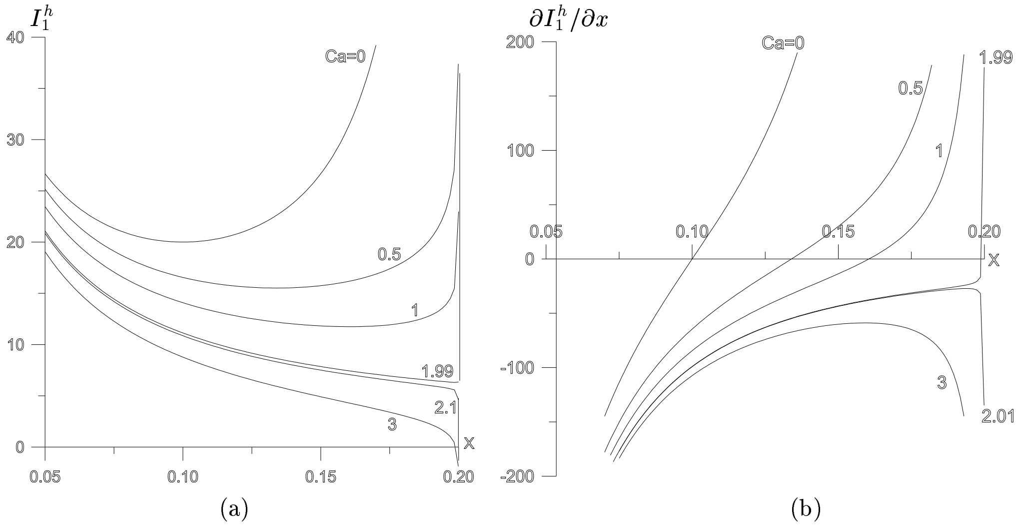


FIG. 5.3. *1-D approach to 2-D functional (2.3). Dependence of distribution for I_1^h (a) and $\partial I_1^h/\partial x$ (b) on c_a. At $c_a>2$ the functional loses convexity.*

Since $1-c_a^2\triangle u^2<0$ at every iteration x^p gets an increment being smaller than the distance to the right node 3. Length of the right cell remains greater than zero within truncation error or prescribed accuracy of calculation at any c_a. Thus, although it can turn out beginning from some value of c_a the regularized functional will not have a minimum, nevertheless, the iterative procedure (5.2) allows to condense significantly the grid lines towards the discontinuity and to guarantee the grid to be unfolded.

Approach of three-point model can be also applied in general case of computing real 2-D flows with shocks. In the neighborhood of any point of discontinuity we introduce a local Cartesian system of coordinates with axis y directed along the tangent line towards the discontinuity at the point considered. Then, to a first approximation, we assume $x=x(\xi)$, $y=a\eta$, neglect the small terms in (2.3), i.e., f_x, x_η, y_ξ and get (5.6). On the other hand presence of these small terms in the functional causes the nodes "to wander" permanently along some trajectory about an average position keeping strong grid lines condensing in the vicinity of shocks. Further, for the sake of simplicity, we shall refer the iterative procedure (quasi-Newton method in a real 2-D case, see Sec. 7.2) as a minimization of the discrete functional independently on whether there is a solution of the minimization problem or not.

Performed analysis of the properties to the 1-D and 2-D functionals shows these functionals are inconsistent, i.e., nodes clustering towards the discontinuity is performed in a different manner inside the domain Ω via minimization of the 2-D functional and on the boundary $\partial\Omega$ via 1-D functional. It is arisen the necessity in consistent redistribution of the grid nodes in Ω and on $\partial\Omega$. This matter will be considered in Sec. 8.

6. Another method of adaptation. Let us see what using of the functional (2.5) to the direct mapping or Euler-Lagrange equation (2.6) gives to constructing the adaptive mesh.

Consider the problem of minimizing the Dirichlet functional to the direct mapping of the domain $\Omega(x)$ onto ξ. Approximation of (2.5) on the two-cell mesh gives the discrete functional

$$I^h = x^2 + (L-x)^2 + c_a^2 \triangle u^2 \ .$$

Its derivative

$$\frac{\partial I^h}{\partial x} = 4x - 2L$$

vanishes at $x{=}L/2$. Thus, independently on c_a minimum of I^h reaches on the set of uniform meshes and grid clustering is not performed towards discontinuity.

Approximation of the Euler-Lagrange equation (2.6) is given

$$x = (L-x)\sqrt{1 + c_a^2 \triangle u^2/(L-x)^2}$$

and from it we obtain

$$x_{min} = \frac{L}{2} + \frac{c_a^2 \triangle u^2}{2L} \ .$$

Consequently, at $c_a \triangle u{=}L$ the right cell collapses. In this case there is no barrier and in multidimensional problems, when grid lines are strongly bent near the shock wave, the derivative of a control function f towards the normal of the shock changes from one cell to another. In some cells we will not get enough condensation of grid lines and the other cells will have already been folded.

7. Optimization method and coupled algorithm.

7.1. 1-D case. We use the approximation (3.3) to the functional (2.4) and seek for the minimum of I_m^h ($m{=}1,2$) applying the Newton method (5.2) when $x{=}(x_1, \cdots, x_{i_{max}+1})$. Using the approximations (3.4) to $(x_\xi)_{i+1/2}$ and $(f_x)_{i+1/2}$, fixing $(f_x)_{i+1/2}$ and setting $\triangle\xi{=}1$ we get the first derivative

$$\frac{\partial I_m^h}{\partial x_i} = \frac{-1}{(x_i - x_{i-1})^2 \left[1 + c_a^2 (f_x)^2_{i-1/2}\right]^{1-2^{-m}}} + \frac{1}{(x_{i+1} - x_i)^2 \left[1 + c_a^2 (f_x)^2_{i+1/2}\right]^{1-2^{-m}}}.$$

The Hessian is a diagonal matrix with components $\mathcal{H}_{ii} = \partial^2 I_m^h / \partial x_i^2$. As above we fix $(f_x)_{i+1/2}$ while deriving the second derivative

$$\frac{\partial^2 I_m^h}{\partial x_i^2} = \frac{2}{(x_i - x_{i-1})^3 \left[1 + c_a^2 (f_x)^2_{i-1/2}\right]^{1-2^{-m}}} + \frac{2}{(x_{i+1} - x_i)^3 \left[1 + c_a^2 (f_x)^2_{i+1/2}\right]^{1-2^{-m}}}.$$

Here the case $m{=}2$ is used to increase velocity of grid nodes condensing.

If we need to perform adaptation along the curve, e.g., boundary $\partial\Omega$ of the domain Ω, the 1-D functional (2.4) can be written in the parametric form [31, 32],

$$I = \int_0^1 \frac{1}{t_\xi \sqrt{1 + c_a^2 f_t^2}} d\xi, \tag{7.1}$$

where the control function $f{=}f(t)$, parameter t defines the length of the boundary $\partial\Omega$. Then the Newton method (5.2) of minimization can be applied.

7.2. 2-D case. We use the optimization algorithm suggested in [12, 24]. Approximation of the functional (2.3) is performed on the mesh of quadrilaterals and is given by (3.1). If the set of convex grids is not empty, the system of algebraic equations (3.2) has at least one solution being the convex mesh. Assuming the grid to be convex at the p-th step of the iterative procedure we find the coordinates of the i-th node at the p+1-th step using the quasi-Newton method in the sense that the Hessian is a diagonal matrix, see [4, 12, 24] for detail,

$$\begin{aligned} x_i^{p+1} &= x_i^p - \tau \left(R_x \frac{\partial R_y}{\partial y_i} - R_y \frac{\partial R_x}{\partial y_i} \right) \left(\frac{\partial R_x}{\partial x_i} \frac{\partial R_y}{\partial y_i} - \frac{\partial R_y}{\partial x_i} \frac{\partial R_x}{\partial y_i} \right)^{-1}, \\ y_i^{p+1} &= y_i^p - \tau \left(R_y \frac{\partial R_x}{\partial x_i} - R_x \frac{\partial R_y}{\partial x_i} \right) \left(\frac{\partial R_x}{\partial x_i} \frac{\partial R_y}{\partial y_i} - \frac{\partial R_y}{\partial x_i} \frac{\partial R_x}{\partial y_i} \right)^{-1}. \end{aligned} \tag{7.2}$$

here τ is the iterative parameter.

Note when finding the first and second derivatives of the functional (3.1) we fix the metric (derivatives f_x and f_y in (2.3)) and it is referred as I_1^h.

To increase velocity of grid nodes condensing in the discrete functional instead of $\sqrt{1 + f_x^2 + f_y^2}$ it can be used the term

$$\left[1 + (f_x)^2 + (f_y)^2\right]^{1-2^{-m}},$$

where m=1, 2, and as in 1-D case such a discrete functional will be referred as I_m^h

If a flow solver gives the values of the control function in the cells center, it is required to update them to the nodes. It can be done by interpolation and it is sufficient to use a first-order interpolation formula, e.g., to the i-th node (except boundary nodes) which is surrounded by 4 cells, we have

$$f_i = c_a \sum_{l=1}^{4} f_l^c A_l \bigg/ \sum_{l=1}^{4} A_l,$$

where f_l^c is the value in the l-th cell center, A_l is the area of the triangle, one vertex of which is the i-th node and two others are adjacent vertexes of the l-th cell.

The coefficient of adaptation can depend on the node position, i.e., c_a=$c_a(x, y)$.

7.3. Coupled algorithm. One time step to solve the 1-D or 2-D equations of gas dynamics with grid adaptation contains the following steps

1. Generate the mesh at the next time level n+1.
2. Compute the gas dynamics values at time t^{n+1}.
3. Make one iteration step by mesh and compute the new values of $(x, y)_i$ at t^{n+1} by formulas (5.2) or (7.2).
4. Repeat starting with step 2 to convergence or within given number of iterations p_{iter}.
5. Compute the final gas dynamics values at t^{n+1}.

The matter of preparing an initial quasiuniform mesh, including the procedure of untangling the initial prepared folded mesh, is considered in [12, 24].

We now consider how to select c_a. As shown in Sec. 5 the present method generates an unfolded mesh at any c_a and L_1- or L_2-errors depend on the shock thickness. Besides, used here the Godunov's-type solver smears the shocks within 2 to 3 cells. Theoretically the thinner the discontinuity (i.e. the larger c_a) is the higher accuracy we get. Therefore, there are no grounds for automatical choice of c_a in contrast to, for instance, [33] where it is used a smothing of the solution and variation of the number of grid points in the shock zone. In practical computations, however, definition of c_a should satisfy some reasonable requirements. On one hand c_a must not be too large, otherwise the mesh begins "to feel" weak compression and rarefaction waves that leads undesirable and useless distortion of the cells. Besides, the larger the value of c_a is the less iterative parameter τ is needed in order not to overcome the admissible set of convex grids, see Tables 10.1,10.2, and the more number of iterations p_{iter} we have to perform. On the other hand if c_a is too small then the mesh can not be condensed in the vicinity of shocks of interest. Thus, we should choose c_a so that to avoid these lacks in mesh adaptation. Further, as it will be shown in Sec.10, on one hand to get a substantial winning in accuracy we need to perform very strong grid lines condensing near the shocks. On the other hand in gas dynamics calculations at strong grid lines clustering the admissible time step $\triangle t$ becomes rather small. It is not burdensome to steady flows. But when computing unsteady flows too small $\triangle t$ will cause overall time of modeling to increase significantly. To avoid too large running time in [4, 5] the basic calculation of the 2D unsteady flow was performed with not very large c_a when $\triangle t \approx 0.1 \triangle t_u$, where $\triangle t_u$ is the step on the quasiuniform mesh. At some time before the control time c_a is increased so that $\triangle t \approx 0.01 \triangle t_u$. And such a technique gave very high resolution of the shocks and contact discontinuity. One more reason not to set c_a too large is that controlling cell sizes we can eliminate only the errors caused by shock smearing, see Secs. 10.3,10.4. There presents the errors gained thoughout the domains of smooth flow. And it is useless to try to reduce the first type errors to zero.

Numerical experiments have shown that to get a suitable mesh it is sufficient to define c_a to be constant, linear or piecewise linear function along some distinctive direction in Ω, and its values are within the interval $0.05 \leq c_a \leq 0.5$.

Above discussion shows that both theoretically and in practice the parameter c_a is free and the user should find an optimal solution taking into account particularity of the concrete problem.

As it was shown in Sec. 5 in the 2-D case it may happen that the iteration procedure (7.2) will not converge and we define the number of iterations p_{iter} at every time step. In real 2-D flow computations one should set p_{iter}=8 to 10 to the rapidly developing unsteady flows and p_{iter}=1 to 2 to the nearly steady flows [5].

8. Redistribution of nodes along the boundary curve. There are several ways to redistribute the grid nodes along the boundary $\partial\Omega$ during adaptation. The simplest one is a fixed position of every point on $\partial\Omega$, referred as "fixed position". When moving the interior nodes towards a discontinuity some instability in mesh generation can arise and, consequently, in the flow near the points where the discontinuity joins $\partial\Omega$. In next method the boundary nodes are treated as interior and the vectors of shift are projected onto $\partial\Omega$ [26], we call it "unconstrained minimization". This way can be used only if the discontinuity is nearly orthogonal to $\partial\Omega$. If no then, when condensing, the boundary nodes overlap, adjacent cells degenerate and modeling breaks. The next method consists of using the 1-D functional (7.1) [12] referred as "1-D minimization". It is more robust than two methods discussed above and can be usually used at adaptation. However, as it has been shown in Sec. 5, the 1-D and 2-D functionals are inconsistent. By this reason the parameters of adaptation c_a and τ should be selected separately. It requires additional work and is particularly cumbersome when modeling unsteady flows. Sometimes we get undesirable displacement of the boundary nodes up to their overlap.

It is required to perform redistribution of the interior and boundary nodes consistently. In the suggested method we perform constrained minimization of the discrete functional (3.1) under constraints defining $\partial\Omega$, referred as "constrained minimization". Constrained minimization on $\partial\Omega$ has been also applied in [29]. We minimize the following functional [3]

$$\tilde{I}_1^h = \sum_{i=1}^{i_{max}} \sum_{k=1}^{4} \frac{1}{4} [F_k]_i + \sum_{l\in\mathcal{L}} \lambda_l G_l = I_1^h + \sum_{l\in\mathcal{L}} \lambda_l G_l, \tag{8.1}$$

where the constraints $G_l{=}G(x_l, y_l){=}0$ define $\partial\Omega$, λ_l are the Lagrange multipliers, $\mathcal{L}$ is the set of the boundary nodes. Since the function $G(x, y)$ is assumed piecewise differentiable, the functional $\tilde{I}_1^h$ holds the infinite barrier on the boundary of the set of convex grids as I_1^h does if $f \in C^1$. If f is of the class of discontinuous functions then the analysis from Sec. 5 can be applied here.

If the set of convex grids is not empty, the system of algebraic equations has at least one solution being the convex mesh

$$R_x = \frac{\partial I_1^h}{\partial x_i} + \lambda_i \frac{\partial G_i}{\partial x_i} = 0, \qquad R_y = \frac{\partial I_1^h}{\partial y_i} + \lambda_i \frac{\partial G_i}{\partial y_i} = 0, \qquad G_i = 0, \tag{8.2}$$

here $\lambda_i{=}0$ if $i \notin \mathcal{L}$ and constraints are defined to the boundary nodes $i \in \mathcal{L}$.

Consider the method of minimizing the functional (8.1) assuming the grid to be convex at the p-th step of the iterative procedure. We use the quasi-Newton procedure to find the coordinates of the i-th node x_i^{p+1} and y_i^{p+1} from the system (8.2)

$$\begin{aligned}
\tau R_x + \frac{\partial R_x}{\partial x_i}(x_i^{p+1} - x_i^p) + \frac{\partial R_x}{\partial y_i}(y_i^{p+1} - y_i^p) + \frac{\partial R_x}{\partial \lambda_i}(\lambda_i^{p+1} - \lambda_i^p) &= 0,\\
\tau R_y + \frac{\partial R_y}{\partial x_i}(x_i^{p+1} - x_i^p) + \frac{\partial R_y}{\partial y_i}(y_i^{p+1} - y_i^p) + \frac{\partial R_y}{\partial \lambda_i}(\lambda_i^{p+1} - \lambda_i^p) &= 0,\\
\tau G_i + \frac{\partial G_i}{\partial x_i}(x_i^{p+1} - x_i^p) + \frac{\partial G_i}{\partial y_i}(y_i^{p+1} - y_i^p) &= 0,
\end{aligned} \tag{8.3}$$

where

$$\begin{aligned}
\frac{\partial R_x}{\partial x_i} &= \frac{\partial^2 I_1^h}{\partial x_i^2} + \lambda_i \frac{\partial^2 G_i}{\partial x_i^2}, \quad & \frac{\partial R_x}{\partial y_i} &= \frac{\partial^2 I_1^h}{\partial x_i \partial y_i} + \lambda_i \frac{\partial^2 G_i}{\partial x_i \partial y_i}, \quad & \frac{\partial R_x}{\partial \lambda_i} &= \frac{\partial G_i}{\partial x_i},\\
\frac{\partial R_y}{\partial x_i} &= \frac{\partial^2 I_1^h}{\partial x_i \partial y_i} + \lambda_i \frac{\partial^2 G_i}{\partial x_i \partial y_i}, \quad & \frac{\partial R_y}{\partial y_i} &= \frac{\partial^2 I_1^h}{\partial y_i^2} + \lambda_i \frac{\partial^2 G_i}{\partial y_i^2}, \quad & \frac{\partial R_y}{\partial \lambda_i} &= \frac{\partial G_i}{\partial y_i}.
\end{aligned}$$

Resolving the last equation of (8.3) about $y_i^{p+1} - y_i^p$ and substituting it in two remaining equations, we get the system

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_i^{p+1} - x_i^p \\ \lambda_i^{p+1} - \lambda_i^p \end{pmatrix} = \begin{pmatrix} a_{13} \\ a_{23} \end{pmatrix},$$

where

$$\begin{aligned}
a_{11} &= \frac{\partial R_x}{\partial x_i} - \frac{\partial R_x}{\partial y_i} \frac{\partial G_i}{\partial x_i} \Big/ \frac{\partial G_i}{\partial y_i}, \quad a_{12} = \frac{\partial G_i}{\partial x_i}, \quad a_{13} = \tau \left[\frac{\partial R_x}{\partial y_i} G_i \Big/ \frac{\partial G_i}{\partial y_i} - R_x \right],\\
a_{21} &= \frac{\partial R_y}{\partial x_i} - \frac{\partial R_y}{\partial y_i} \frac{\partial G_i}{\partial x_i} \Big/ \frac{\partial G_i}{\partial y_i}, \quad a_{22} = \frac{\partial G_i}{\partial y_i}, \quad a_{23} = \tau \left[\frac{\partial R_y}{\partial y_i} G_i \Big/ \frac{\partial G_i}{\partial y_i} - R_y \right].
\end{aligned}$$

Denoting

$$\triangle = a_{11}a_{22} - a_{12}a_{21}, \quad \triangle_1 = a_{13}a_{22} - a_{23}a_{12}, \quad \triangle_2 = a_{11}a_{23} - a_{21}a_{13},$$

we obtain

$$x_i^{p+1} = x_i^p + \triangle_1/\triangle, \quad \lambda_i^{p+1} = \lambda_i^p + \triangle_2/\triangle, \tag{8.4}$$

and y_i^{p+1} is determined from the third equation of (8.3). If the constraints are resolved about y in the form $G(x,y) = y - g(x) = 0$, then

$$\frac{\partial G_i}{\partial x_i} = -\frac{\partial g_i}{\partial x_i}, \quad \frac{\partial G_i}{\partial y_i} = 1,$$

and above formulas are simplified. Constraints can be resolved about x in the form $G(x,y) = x - \tilde{g}(y) = 0$ and then (here it is better to resolve the third equation of (8.3) about $x_i^{p+1} - x_i^p$)

$$\frac{\partial G_i}{\partial x_i} = 1, \quad \frac{\partial G_i}{\partial y_i} = -\frac{\partial \tilde{g}_i}{\partial y_i}.$$

These two forms of $G(x,y)$ can substitute each other. For example, on the part of $\partial\Omega$ being nearly parallel the axis x the boundary should be defined in the form $y{=}g(x)$, and where $\partial\Omega$ is nearly parallel the axis y it should be defined as $x = \tilde{g}(y)$.

If $\partial\Omega$ is given by parametric functions $x{=}x(t)$, $y{=}y(t)$ or tabular values $(x,y)_i$, the following algorithm can be used. When calculating the coordinates of the i-th node, in the interval (x_{i-1}, x_{i+1}) we construct an interpolating parabola $t{=}t(x)$ using the values in three nodes $i{-}1, i, i{+}1$. From (8.4) we compute an intermediate value $\tilde{x}_i^{p+1}$, further from the interpolation formula we determine $t_i = t(\tilde{x}_i^{p+1})$ and final values x_i^{p+1}, y_i^{p+1} from the parametric formulas.

Another way of redistributing the nodes along $\partial\Omega$, given as parametric functions or by tabular values, employs an unconstrained minimization of the functional in parametric form and is based on solving the following system of algebraic equations [3], referred as "parametric minimization",

$$R_t = R_x \frac{\partial x_i}{\partial t_i} + R_y \frac{\partial y_i}{\partial t_i} = 0,$$

via the quasi-Newton procedure

$$\tau R_t + \frac{\partial R_t}{\partial t_i} (t_i^{p+1} - t_i^p) = 0. \tag{8.5}$$

Here

$$\frac{\partial R_t}{\partial t_i} = \frac{\partial R_x}{\partial x_i} \left(\frac{\partial x_i}{\partial t_i}\right)^2 + \frac{\partial R_y}{\partial y_i} \left(\frac{\partial y_i}{\partial t_i}\right)^2 + \left(\frac{\partial R_x}{\partial y_i} + \frac{\partial R_y}{\partial x_i}\right) \frac{\partial x_i}{\partial t_i} \frac{\partial y_i}{\partial t_i}$$
$$+ R_x \frac{\partial^2 x_i}{\partial t_i^2} + R_y \frac{\partial^2 y_i}{\partial t_i^2}, \qquad R_x = \frac{\partial I_1^h}{\partial x_i}, \quad R_y = \frac{\partial I_1^h}{\partial y_i}.$$

To the analytical control functions constrained and parametric minimization give similar results. Real-world 2-D flow computations have shown it is better to perform adaptation along the boundary using constrained minimization (8.3), (8.4) since the procedure (8.5) does not always ensure consistent redistribution of the nodes in Ω and on $\partial\Omega$.

9. Flow solver. In this section we briefly describe the Godunov Linear Flux Correction ($GLFC$) scheme [2] to compute the 2-D gas flow in Euler approach.

Equations of gas dynamics are written in the integral form which can be derived by transformation of the volume integrals in space $x{-}y{-}t$ to the surface integrals by virtue of Gauss's theorem as shown below for the law of conservation of mass

$$\iiint_V \left[\frac{\partial\rho}{\partial t} + div(\rho \boldsymbol{V})\right] dV = \oiint_{\partial V} \rho dxdy + \rho udydt + \rho vdtdx = 0\ ,$$

here V is an arbitrary control volume, homeomorphic sphere in space $x{-}y{-}t$, ∂V is the boundary of V.

Hence, the laws of conservation of mass, momentum and total energy, can be written in the integral form, or generalized formulation [20], as follows

$$\oiint_{\partial V} \boldsymbol{\sigma} dxdy + \boldsymbol{a} dydt + \boldsymbol{b} dtdx = \boldsymbol{0}, \tag{9.1}$$

where

$$\boldsymbol{\sigma} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}, \quad \boldsymbol{a} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E+p) \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E+p) \end{bmatrix}.$$

Here u and v are the velocity components, p and ρ are the pressure and density. The total energy $E{=}\rho[e{+}0.5(u^2{+}v^2)]$, e is the specific internal energy. The equation of state is $p{=}(\gamma{-}1)\rho e$, where γ is the ratio of specific heats. Denote the vector-valued unknown functions as $\boldsymbol{f} = (u, v, p, \rho)^T$. The conservation laws (9.1) hold for any parameters $\boldsymbol{f}$, both smooth and discontinuous, describing a real gas flow.

We introduce the curvilinear moving grid in space $x{-}y{-}t$ and consider the hexahedral computing cell, see Fig. 9.1. The bottom face of the cell (or control volume) is taken at time level n and the top face at level $n{+}1$, four lateral faces generally form ruled surfaces rather than simple planes.

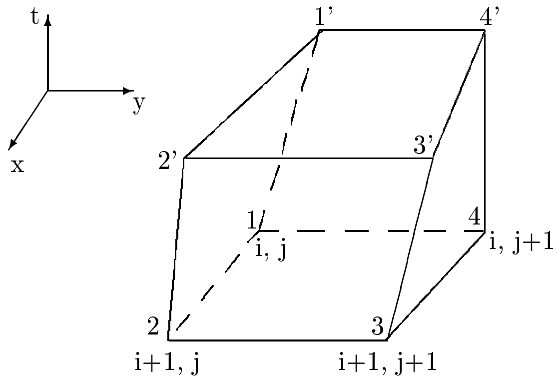


FIG. 9.1. *Computing cell*

Integrating (9.1) over the oriented surface, being the boundary ∂V of the computing cell, we obtain a cell-centered finite-volume discretization of the governing equations

$$\boldsymbol{\sigma}^{n+1} A^{n+1} - \boldsymbol{\sigma}^n A^n + \boldsymbol{Q}_{411'4'} + \boldsymbol{Q}_{233'2'} + \boldsymbol{Q}_{122'1'} + \boldsymbol{Q}_{344'3'} = \boldsymbol{0}, \tag{9.2}$$

where $\boldsymbol{\sigma}^{n+1}$ and $\boldsymbol{\sigma}^{n}$ are the average values at time t^{n+1} and t^n in the center of the top and bottom faces, A^{n+1} and A^n are the areas of these faces. Each of four vector values $\boldsymbol{Q}_{411'4'}$, $\boldsymbol{Q}_{233'2'}$, $\boldsymbol{Q}_{122'1'}$ and $\boldsymbol{Q}_{344'3'}$ is an average flux of mass, momentum and energy through the corresponding intercell surface in the direction of the outward normal vector. Unlike the original Godunov's scheme [20] where the fluxes in (9.2) are taken at time t^n, in the present scheme those values are computed at $t^{n+1/2}$ that provides it with the second-order accuracy in time.

For example, at the face $122'1'$ the value $\boldsymbol{Q}_{122'1'}$ has the following structure

$$\boldsymbol{Q}_{122'1'} = \boldsymbol{\sigma}^{n+1/2}A^{xy} + \boldsymbol{a}^{n+1/2}A^{yt} + \boldsymbol{b}^{n+1/2}A^{tx}, \tag{9.3}$$

where A^{xy}, A^{yt}, A^{tx} are the areas of projections of the face $122'1'$ onto the coordinate planes $x{-}y$, $y{-}t$ and $t{-}x$, respectively, given by

$$A^{xy} = \iint_{122'1'} dxdy = \frac{1}{2}[(x_{2'} - x_1)(y_{1'} - y_2) - (x_{1'} - x_2)(y_{2'} - y_1)],$$
$$A^{yt} = \iint_{122'1'} dydt = \frac{1}{2}\triangle t(y_{2'} + y_2 - y_1 - y_{1'}),$$
$$A^{tx} = \iint_{122'1'} dtdx = -\frac{1}{2}\triangle t(x_{2'} + x_2 - x_1 - x_{1'}),$$

which are obtained from the formula for the quadrangle 1234

$$A_{1234} = A(x_1, y_1; x_2, y_2; x_3, y_3; x_4, y_4) = \frac{1}{2}[(x_3 - x_1)(y_4 - y_2) - (x_4 - x_2)(y_3 - y_1)],$$

when running along its contour in an anticlock-wise manner, time step $\triangle t{=}t^{n+1}{-}t^n$.

The values $\boldsymbol{f}^{n+1}$ are updated by two stages using time splitting. At the first stage, predictor, via (9.2) we compute the intermediate values $\tilde{\boldsymbol{f}}^{n+1}$. Here we apply the piecewise linear interpolation along each curvilinear coordinate line ξ, passing through the center of cells $j{+}1/2{=}const$, and η, passing through the center of cells $i{+}1/2{=}const$, i.e., determine the derivatives $\boldsymbol{f}_\xi$ and $\boldsymbol{f}_\eta$ in every cell, to get the fluxes via (9.3) on the lateral faces with the second-order accuracy in space but still at t^n. To suppress spurious oscillations in the vicinity of discontinuities a monotonicity algorithm is applied [2]. At the second stage, corrector, using $\boldsymbol{f}^n, \tilde{\boldsymbol{f}}^{n+1}$ and derivatives $\boldsymbol{f}_\xi$, $\boldsymbol{f}_\eta$ at t^n, we get pre-wave values on the both sides of each of four lateral faces of the cell at $t^{n+1/2}$. After solving the Riemann problem we obtain $\boldsymbol{f}^{n+1/2}$, the values at each of four lateral faces, and again calculate the fluxes from (9.3). Then, via (9.2), we compute the final values $\boldsymbol{f}^{n+1}$.

In the $(i{+}1/2, j{+}1/2)$-th cell the admissible time step $\triangle t_{i+1/2,j+1/2}$ is defined via [20]

$$\triangle t_{i+1/2,j+1/2} = \frac{\triangle t' \, \triangle t''}{\triangle t' + \triangle t''},$$

where

$$\triangle t' = \frac{h'}{\max(d^{II}_{41} - w_{41}; -d^{I}_{23} - w_{23})}, \qquad \triangle t'' = \frac{h''}{\max(d^{II}_{12} - w_{12}; -d^{I}_{34} - w_{34})},$$
$$h' = \frac{A_{1234}}{0.5\sqrt{(x_4 + x_3 - x_1 - x_2)^2 + (y_4 + y_3 - y_1 - y_2)^2}},$$
$$h'' = \frac{A_{1234}}{0.5\sqrt{(x_3 + x_2 - x_4 - x_1)^2 + (y_3 + y_2 - y_4 - y_1)^2}}.$$

Here $\triangle t^{'}$ and $\triangle t^{''}$ are the admissible time steps to 1-D schemes in ξ and η direction, respectively, $h^{'}, h^{''}$ are the "average heights" of the bottom face A_{1234} , w is the velocity of the corresponding cell edge, e.g., w_{12} is the velocity of the edge 12 which defines inclination of the face $122'1'$, see Fig. 9.1. Next, d_{12}^{II} and d_{41}^{II} are the "extreme right wave" speeds defined from solving the Riemann problem to the faces $122'1'$ and $11'4'4$, respectively; d_{23}^{I} and d_{34}^{I} are the "extreme left wave" speeds to the faces $233'2'$ and $344'3$, respectively.

As an admissible time step we take the minimal of all cells

$$\triangle t = \nu \min_{i,j} \triangle t_{i+1/2,j+1/2} \ .$$

The coefficient ν is less than 1 (usually close to 1) and it is introduced as a correction to non-linearity of the problem. Note that the time step depends on both post-wave values and velocity of every intercell face. In computations the value of $\triangle t$, obtained at the preceding time step, is used to the next time step. By this reason in the case of essentially nonstationary processes the coefficient ν should be greatly decreased.

The *GLFC* scheme is of the second-order accuracy in time and space in the domains of smooth flow. We get the values $\boldsymbol{f}^{n+1}$ directly on the moving mesh and need not perform interpolation at t^{n+1} from one mesh to the other.

10. Examples of modeling.

10.1. Analytical control function. First we demonstrate a simple test illustrating inconsistency of redistributing the boundary and interior nodes when using various methods from Sec. 8 and vice versa their consistency when using another.

The adaptive mesh 50×50 is generated in the unit square $0<x, y<1$ when the control function is defined to be

$$f(x,y) = \begin{cases} 1 & \text{if} \quad y < 0.5 \ , \\ 0 & \text{if} \quad y \geq 0.5 \ . \end{cases}$$

Fragments of the adapted meshes in the vicinity of the discontinuity are presented in Fig. 10.1. In the first case the coefficient c_a=0.1, see Fig. 10.1a,b,c. When we apply fixed position and 1-D minimization methods of redistributing the boundary nodes, see Fig. 10.1a,b, the horizontal grid lines are not parallel and in case of the other 3 methods they are parallel. In the next case the coefficient c_a=0.15, see Fig. 10.1d,e,f. To the fixed position method the coordinate lines become more bent, see Fig. 10.1d. Using 1-D minimization leads the boundary nodes to overlap, see Fig. 10.1e, i.e., the mesh to fold. This happens due to inconsistency of the nodes redistribution in Ω and on $\partial\Omega$ despite of the fact that 1-D and 2-D algorithms separately provide unfolded grid generation. In this test unconstrained minimization gives the same result as constrained and parametric minimization due to the discontinuity is orthogonal to $\partial\Omega$ and here the horizontal lines almost merge near the discontinuity and remain parallel keeping the mesh unfolded, see Fig. 10.1f.

In the next example the discontinuity is not orthogonal to $\partial\Omega$. The control function is defined to be

$$f(x,y) = \begin{cases} 1 & \text{if} \quad y > 5x - 2 \ , \\ 0 & \text{if} \quad y \leq 5x - 2 \ . \end{cases}$$

Fragments of the adapted meshes near the top boundary are presented in Fig. 10.2. Here using unconstrained and 1-D minimization leads the boundary nodes to overlap

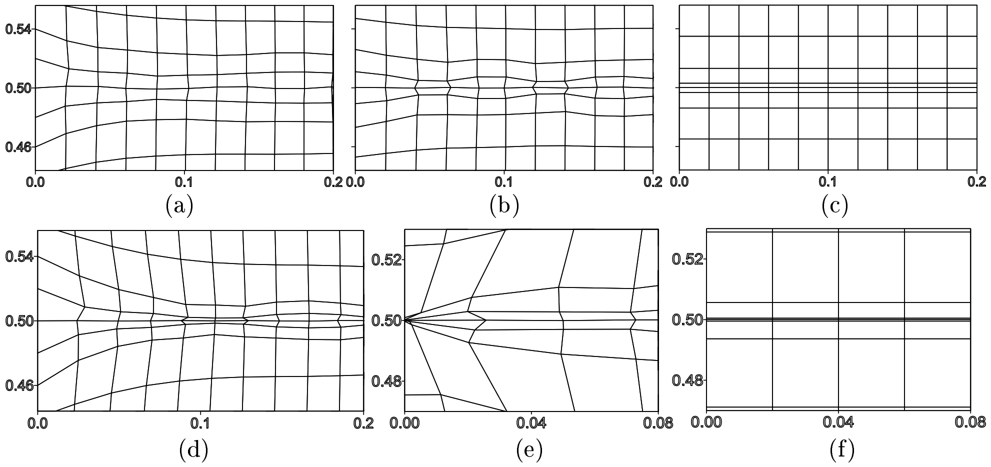


FIG. 10.1. *Fragment of the adapted mesh. The boundary nodes are redistributed using fixed position (a,d), 1-D minimization (b,e) and unconstrained or constrained or parametric minimization (c,f). Coefficient* c_a=0.1 *in the cases (a-c) and* c_a=0.15 *in the cases (d-f), iterative parameter* τ=0.15.

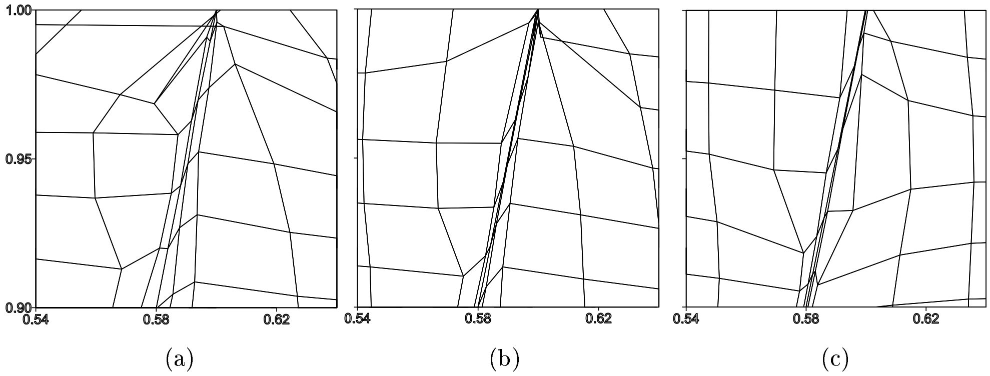


FIG. 10.2. *Fragment of the adapted mesh. The boundary nodes are redistributed using unconstrained (a), 1-D (b) and constrained or parametric minimization (c) methods. Coefficient* c_a=0.3, *iterative parameter* τ=0.1.

in several tenths mesh iterations, see Fig. 10.2a,b, respectively. Constrained and parametric minimization maintain unfolded mesh, see Fig. 10.2c. To the analytical control functions constrained and parametric minimization give similar results.

10.2. Contact discontinuity. To observe how the adaptive procedure handles the contact discontinuities the Cauchy problem for the 1-D linear advection equation has been computed with the Godunov's scheme, see Godunov and Ryaben'kii [19]. Adaptation is executed via (5.2). It appears once the grid points have been condensed towards the contact discontinuity the thickness of discontinuity smeared increases with time proportionally to $\sqrt{t}$ as the theoretical estimation gives on the uniform fixed mesh.

10.3. Inviscid Burger's equation. Consider the IVP (4.1) with initial values u_l=2 (in 3 left cells), u_l=1. Initial uniform mesh has spacing h=0.1 and zones number i_{max}=50. The boundary nodes are fixed. Calculations are performed up to t=1 using

TABLE 10.1

IVP to the inviscid Burger's equation (4.1). Adaptation is performed by minimization of I_1^h. Here h_{min} is the smallest interval, n is the number of time steps, τ is the iterative parameter, $||Er||_{L_1}$ is the error.

	Godunov's scheme				GLFC's scheme			
c_a	h_{min}	n	τ	$\|\|Er\|\|_{L_1}$	h_{min}	n	τ	$\|\|Er\|\|_{L_1}$
0	0.1	21	-	0.078046	0.1	21	-	0.023551
0.1	0.056821	30	0.95	0.039784	0.049486	35	0.95	0.016755
0.25	0.022115	77	0.95	0.011285	0.020587	86	0.95	0.005337
0.5	0.010712	179	0.8	0.004249	0.010979	191	0.8	0.003395
1	0.004683	406	0.5	0.000540	0.004637	419	0.4	0.000204
2	0.002382	897	0.3	0.000297	0.001924	903	0.2	0.000043
4	0.001116	2065	0.15	0.000187	0.000918	1835	0.1	0.000049
8	0.000526	4438	0.07	0.000057	0.000624	3733	0.07	0.000068
16	0.000209	9355	0.05	0.000018	0.000246	7612	0.05	0.000065

the first-order Godunov's scheme (4.4) and second-order $GLFC$ scheme [2]. In this case the mesh structure a bit differs from the one in Sec. 4. The shock smears over 3 cells with approximately similar length h_{min}. To the left and right from those 3 cells there is a cell with intermediate length and remaining 45 cells are similar (if sufficient number of mesh iterations are provided). In computations, presented in Table 10.1, every time step includes p_{iter}=100 mesh iterations.

Data of Table 10.1 show that increasing the coefficient of adaptation c_a from 0 to 16 decreases thickness of smearing, equal approximately $3h_{min}$, by 478 times to the Godunov's scheme and by 369 times to the $GLFC$ scheme that leads the accuracy to increase by factors of 4335 and 362 to those schemes, respectively. We see if on the uniform mesh the second-order scheme delivers a higher accuracy than the first-order one, on the strongly compressed mesh the situation changes. Moreover, to the $GLFC$ scheme the accuracy is not increased at c_a>2. This can be explained by presence of spurious oscillations near the discontinuity to the second-order scheme.

10.4. Shock tube problem. Consider 1-D flow of ideal gas with initial parameters $(u,p,\rho)_l$=$(2.253928, 2.296074, 1.7811378)$ in the left 3 cells and $(u,p,\rho)_r$=$(1.5,1,1)$ selected in such a way that at t>0 these two domains are divided by the shock wave, moving from the left to the right. Ratio of specific heats γ=1.4. Initial uniform mesh is as in Sec. 10.3 and calculations are performed up to t=1 with the same schemes. Every time step includes p_{iter}=100 mesh iterations.

Data of Table 10.2 show that increasing the coefficient of adaptation c_a from 0 to 4 decreases thickness of the shock smeared, equal approximately $3h_{min}$, by 80 times and accuracy increases by the factor of 194 and 138 to the Godunov's and $GLFC$ scheme, respectively. The second-order scheme gives the lesser error $||Er||_{L_1}$ estimated by the density. Further increase of c_a does not deliver a higher accuracy since the error, caused by shock smearing, becomes much less than the error gained throughout the other cells. If on the uniform mesh c_a=0 the second-order scheme provides a higher accuracy than the first-order one by factor of 3.1, at c_a=4 the ratio of errors falls to 2.2. The last column shows $\triangle\rho$, amplitude of spurious oscillations to the density in the vicinity of the shock computed with the GLFC scheme. We see adaptation allows to reduce significantly $\triangle\rho$ from 1.1% down to 0.01%. Using the functional I_2^h causes strong grid nodes clustering at rather small c_a and, consequently, $||Er||_{L_1}$ to reduce.

Note in two above examples p_{iter} at every time step was equal 100 to get similar cell spacings in the shock zone and domains of constant flow parameters to see what adaptation provides in the limit at the extreme large values of c_a. In Table 10.3 we present two computations of this problem with $GLFC$ scheme when p_{iter}=10 and 100

TABLE 10.2

Shock tube problem. Adaptation is performed by minimization of I_1^h except the last row where I_2^h is used. Here h_{min} is the smallest interval, n is the number of time steps, τ is the iterative parameter, $||Er||_{L_1}$ is the error estimated by ρ, $\triangle\rho$ is the amplitude of spurious oscillations.

	Godunov's scheme				GLFC's scheme				
c_a	h_{min}	n	τ	$\|\|Er\|\|_{L_1}$	h_{min}	n	τ	$\|\|Er\|\|_{L_1}$	$\triangle\rho,\%$
0	0.1	37	-	0.075736	0.1	38	-	0.024554	1.10
0.1	0.07469	46	0.9	0.088294	0.05827	54	0.8	0.028529	0.51
0.25	0.04080	87	0.9	0.088695	0.02784	113	0.5	0.017749	0.23
0.5	0.01583	211	0.7	0.025053	0.01307	248	0.5	0.011859	0.08
1	0.00702	486	0.7	0.005929	0.00604	584	0.5	0.001864	0.03
2	0.00259	1307	0.5	0.000581	0.00242	1396	0.4	0.000317	0.04
4	0.00126	3086	0.2	0.000391	0.00124	3245	0.2	0.000178	0.01
functional I_2^h									
0.25	0.00103	4891	0.13	0.003058	0.00103	5196	0.13	0.000897	0.06

TABLE 10.3

Shock tube problem. Parameters τ=0.5, c_a=1

p_{iter}=10, $||Er||_{L_1}$=0.006681

i	$h_{i+1/2}$	$\rho_{i+1/2}$	$\rho^{exact}_{i+1/2}$
37	0.10264	1.78201	1.78114
38	0.10547	1.78128	1.78114
39	0.10980	1.78037	1.78114
40	0.11837	1.77947	1.78114
41	0.09642	1.77808	1.78114
42	0.01553	1.70929	1.78114
43	0.01635	1.17676	1.00000
44	0.02131	1.00001	1.00000
45	0.04872	1.00000	1.00000
46	0.25688	1.00000	1.00000
47	0.27041	1.00000	1.00000
48	0.28275	1.00000	1.00000
49	0.29084	1.00000	1.00000

p_{iter}=100, $||Er||_{L_1}$=0.002563

i	$h_{i+1/2}$	$\rho_{i+1/2}$	$\rho^{exact}_{i+1/2}$
30	0.10704	1.78127	1.78114
31	0.10727	1.78120	1.78114
32	0.10751	1.78093	1.78114
33	0.10692	1.77965	1.78114
34	0.01884	1.77406	1.78114
35	0.00666	1.61892	1.78114
36	0.00604	1.00487	1.00000
37	0.00822	1.00000	1.00000
38	0.07492	1.00000	1.00000
39	0.11228	1.00000	1.00000
40	0.11305	1.00000	1.00000
41	0.11383	1.00000	1.00000
42	0.11459	1.00000	1.00000

with parameters τ=0.5, c_a=1. The spacings $h_{i+1/2}$, values of $\rho_{i+1/2}$ and exact values of $\rho^{exact}_{i+1/2}$ in the segments center are shown in the vicinity of the shock. It can be seen that in the first case the value of p_{iter} is insufficient and nodes did not manage to come to their "final" positions if to judge by the values of $h_{i+1/2}$. Spacings before and behind the shock differ by factor of 3. By this reason h_{min} is larger by factor of ≈ 2.7 than the one in the second case. This leads the L_1-error is too large. Thus, in the second case winning in accuracy we lose in number of iterations p_{iter}.

10.5. Flow in a channel. The following test demonstrates the capability of the grid adaptation technique to reduce computer costs while obtaining the same accuracy as on the fixed quasiuniform mesh. Into the channel, shown in Fig. 10.3, from the left it is introduced an ideal gas with M_∞=2.5. There are two edges in the top and bottom with inclination of 5^o. The steady flow has the following structure: two shock waves attached to the top of wedges intersect each other, reflect from the walls, again intersect, etc., dividing the domain into a set of subdomains which contain constant flow parameters.

The problem is calculated with the GLFC scheme on the successively refined grids with 64×16, 128×32, 256×64, and 512×128 zones number. At the first stage the calculation on the quasiuniform mesh is performed until the solution achieves its steady state. Then, at the second stage, we switch on the adaptive procedure. As

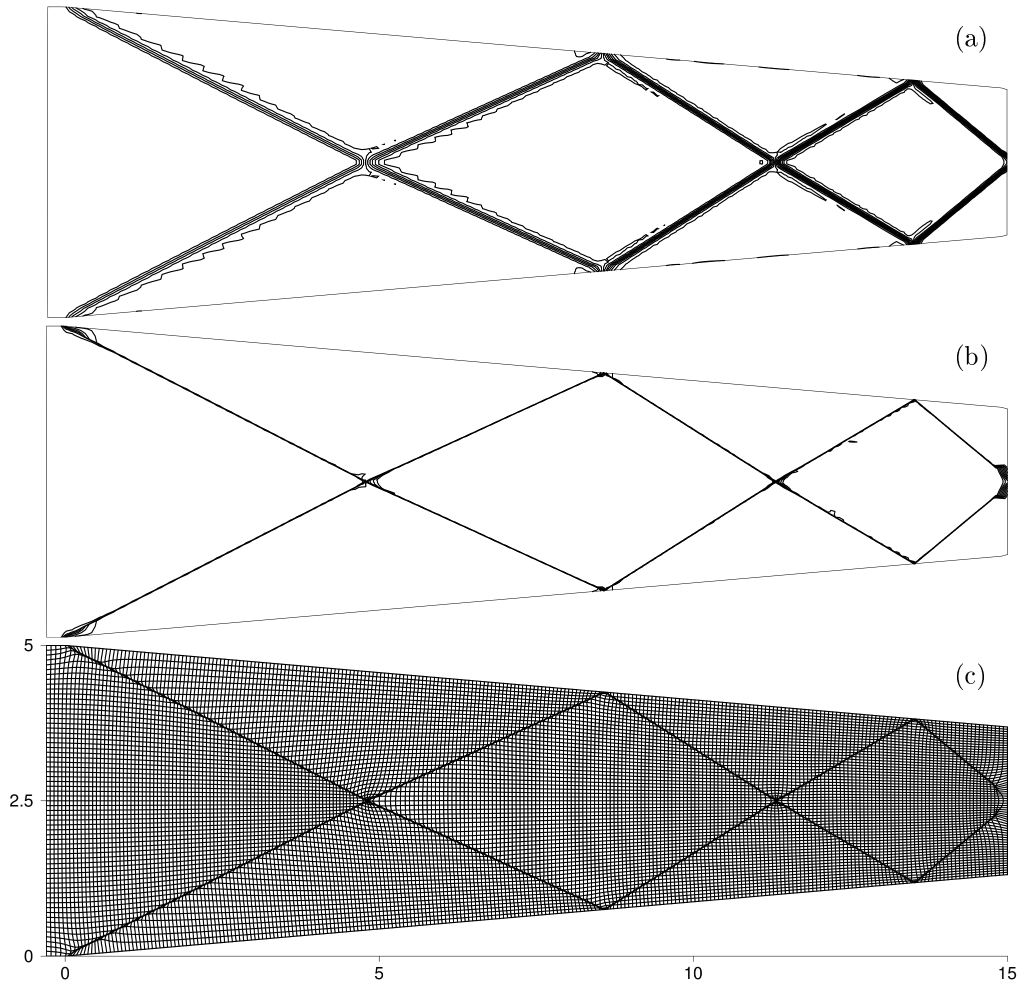


FIG. 10.3. *Supersonic flow in the channel. As a control function it is used ρ. Density contours (from 1 to 2.7 with $\triangle\rho$=0.05) computed on the quasiuniform (a) and adapted (b) meshes* 264×64*; adapted mesh* 264×64 *(c).*

a control function it is used ρ. At every time step we perform one mesh iteration, i.e., p_{iter}=1. To provide the best accuracy of computations c_a is selected so to get similar grid clustering towards the shocks within the whole flow and this is achieved by defining a linear/(piecewise linear) dependence of c_a on x-coordinate.

Fig. 10.3 shows the density contours, when modeling on the quasiuniform and adapted meshes 256×64, and adapted mesh. Here the first stage, to get a steady state on the quasiuniform mesh by t=15, takes 1739 time steps. Applying adaptation we get strong grid lines condensing in 200 time steps and a winning in accuracy by factor of 2. Coefficient of adaptation is defined to be c_a=0.2−0.15(15−x)/15, iterative parameter τ=0.3. But condensed grid lines are still far from the shock location in the exact solution, see Fig. 10.4a. We have to perform 2000 time steps more to obtain maximal accuracy. After this rather long shocks "capturing" by the grid lines we get additional winning in accuracy by factor of 2.5, see the final mesh in Fig. 10.4b. To increase the grid velocity we use the functional I_2^h, and this reduces time of shocks capturing by factor of 1.7. Nevertheless, during the last 300 steps we use the functional

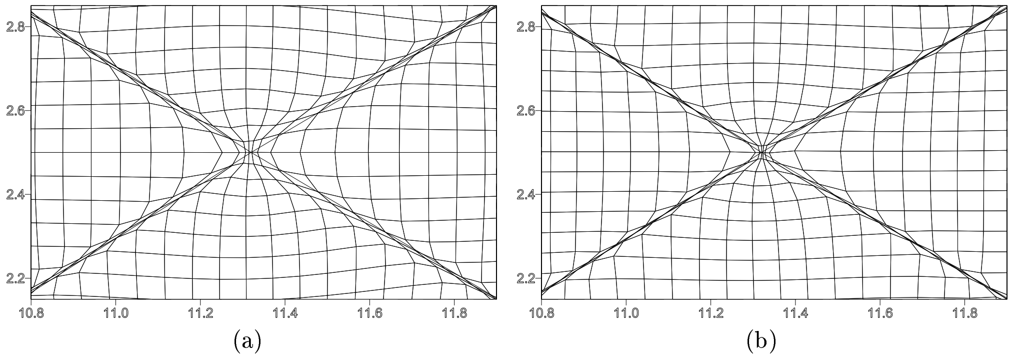


FIG. 10.4. *Fragment of adapted mesh in 200 time steps (a) and final mesh (b). Two diagonal straight-lines indicate position of shocks for the exact solution.*

TABLE 10.4
Dependence of error, estimated by ρ, on zones number computed to quasiuniform and adaptive meshes.

	Error $\|\|Er\|\|_{L_1}$	
Zones number	Quasiuniform mesh	adaptive mesh
64×16	1.5598	0.4070
128×32	0.8274	0.1666
256×64	0.4290	0.0846
512×128	0.2207	0.0444

I_1^h, since I_2^h does not provide very strong grid nodes condensing. Note that using I_2^h on the rough mesh 64×16 does not provide increase of the grid velocity. In real computing the steady problems, when exact solution is not known, user shall watch whether location of a shock changes with time, if yes using the functional I_2^h can increase grid velocity, if no one should at once apply I_1^h. In Fig. 10.4a,b it can be seen the shock waves are smeared over three cells as in 1-D case. Near the shocks the cells are very narrow and maximal aspect ratio achieves 40. As for 1-D case due to adaptation the amplitude of oscillations in the solution is reduced by many factors of ten.

Knowing the exact solution, one can estimate the error, see Table 10.4. We see adaptation allows to increase the accuracy by factor of 4 to 5. Keeping in mind that doubling the points in each space direction on the quasiuniform mesh causes the accuracy to increase twice we can estimate that adaptation in the last three variants is equivalent to the mesh refinement by factor of 5. Thus, adaptation allows to get winning of CPU memory by factor of 25 and running time by factor of 50 to 60 taking into account additional time steps required at adaptation.

10.6. Flow over an airfoil. The GLFC scheme with adaptive procedure is applied to calculating transonic and supersonic Euler flow over NACA0012 airfoil. The first test is a transonic case with M_∞=0.85 and angle of attack α=1°. Fig. 10.5 presents the quasiuniform O-mesh 140×80 and plots the Mach number contours. We see that shock waves, one (stronger) on the upper side of the airfoil and the other (weakear) on the low side, are rather thick. Fig. 10.6 presents the adapted mesh and Mach number contours calculated on this grid. As a control function we use ρ. Coefficient of adaptation c_a is defined to be

$$c_a=\begin{cases} c_{min}+(c_{max}-c_{min})|y|/0.8 & \text{if} \quad |y|\le 0.8, \\ c_{max} & \text{if} \quad |y|>0.8, \end{cases} \quad c_{min}=0.1,\ c_{max}=\begin{cases} 0.2 & \text{if} \quad y<0, \\ 0.15 & \text{if} \quad y>0. \end{cases}$$

(a)

(b)

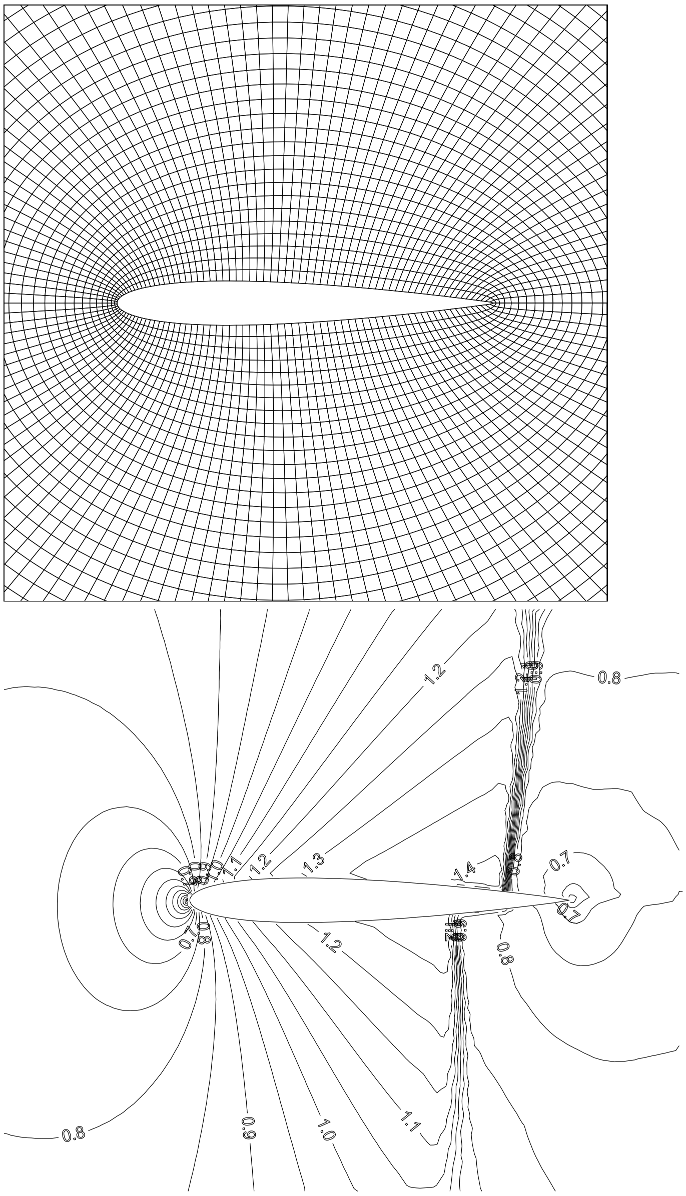


FIG. 10.5. *Transonic flow around NACA0012 airfoil. Quasiuniform mesh (a) and Mach number contours (b).*

A linear dependence of c_a on y is used to strengthen adaptation in the domains where the shocks grow weaker. Around the leading edge where there are rarefaction waves we switch off adaptation setting c_a=0 to exclude cells distortion. Iterative parameter τ=0.3, number of tierations at every time step p_{iter}=1. Constrained minimization is applied to redistribute the boundary nodes along the airfoil contour and along the line y=0 passing from trailing edge which the boundaries of the parametric square ξ=0, 1 correspond to. Using 1-D or parametric minimization with the same parameters c_a, τ on $\partial\Omega$ as in Ω leads the boundary nodes to overlap in the vicinity of the shocks.

(a)

(b)

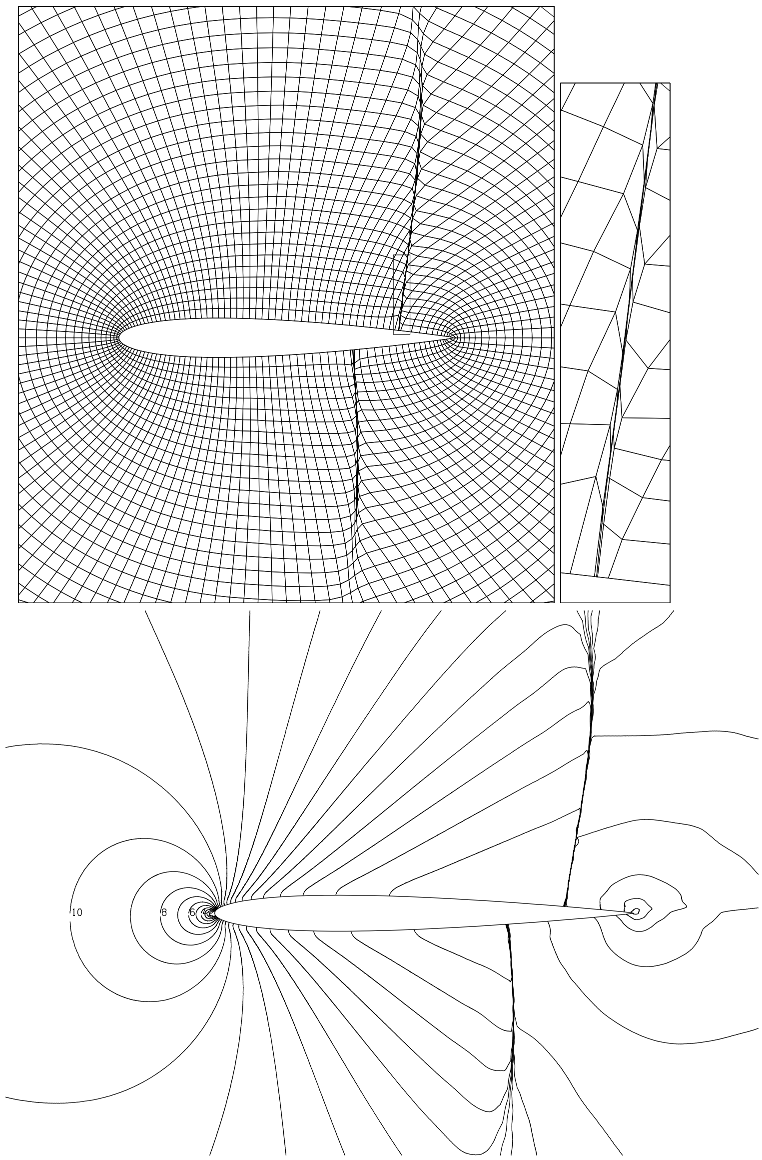


FIG. 10.6. *Transonic flow around NACA0012 airfoil. Adapted mesh (a) and Mach number contours (b).*

Thickness of the shocks is reduced by 50 times in comparison with the nonadapted mesh that provides with capturing the discontinuities very accurately. The first stage takes 7345 time steps to compute by t=20, adaptation does 150 steps that is about 4% of the first stage time.

Another test is a supersonic flow over the same airfoil with M_∞=1.3 and α=0°. Fig. 10.7 plots the Mach number contours computed on the quasiuniform and adapted O-meshs 120×50. As can be seen in Fig. 10.7a, a strong bow shock wave appears in front of the airfoil leading edge and two weak shocks emanate from the trailing edge.

(a)

(b)

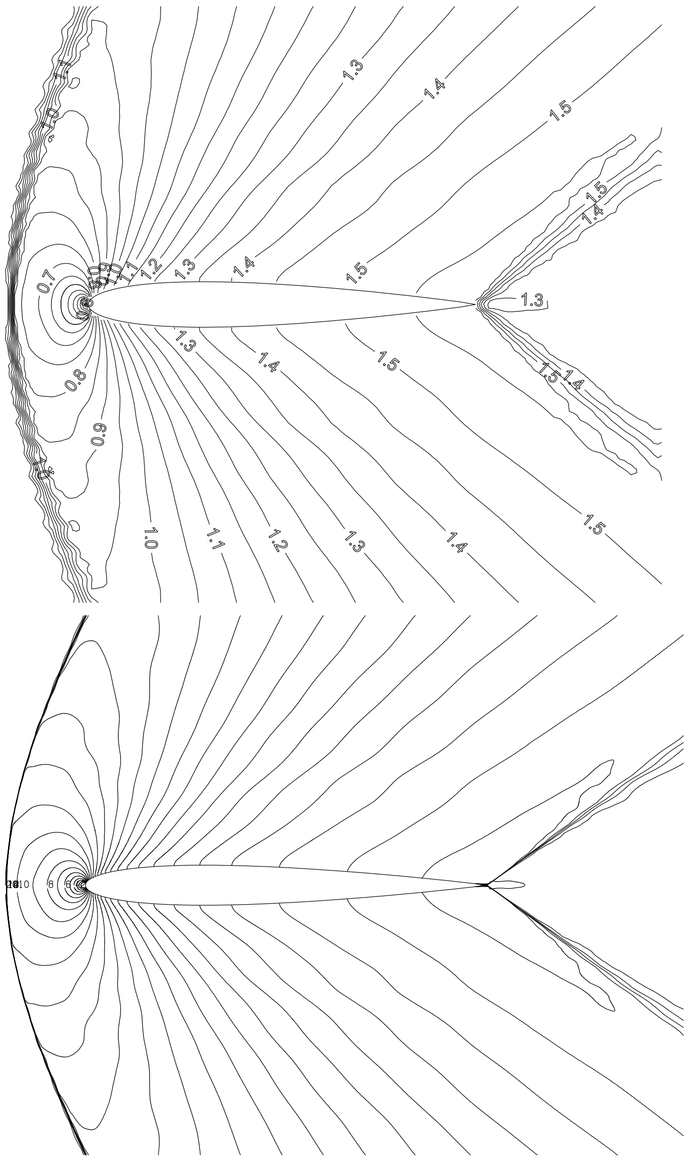


FIG. 10.7. *Supersonic flow around NACA0012 airfoil. Mach number contours computed on quasiuniform (a) and adapted (b) grids.*

Using adaptation provides with very strong reducing the bow shock thickness and rather strong reducing the trailing edge shocks thickness that is demonstrated by the both Mach number contours in Fig. 10.7b and adapted grid in Fig. 10.8. Coefficient of adaptation c_a is defined to be

$$c_a = \begin{cases} c_{min} + (c_{max} - c_{min})|y|/1.5 & \text{if } \ |y| \le 1.5, \\ c_{max} & \text{if } \ |y| > 1.5, \end{cases} \quad c_{min} = \begin{cases} 0.05 & \text{if } \ x \le 0.6, \\ 0.1 & \text{if } \ x > 0.6, \end{cases} \quad c_{max} = 0.15,$$

and as above we set $c_a = 0$ near the airfoil leading edge. Parameters $\tau = 0.3$, $p_{iter} = 1$.

Grid points clustering allows to hope that we nearly eliminated the errors caused by shock waves smearing and increased significantly the accuracy of computations.

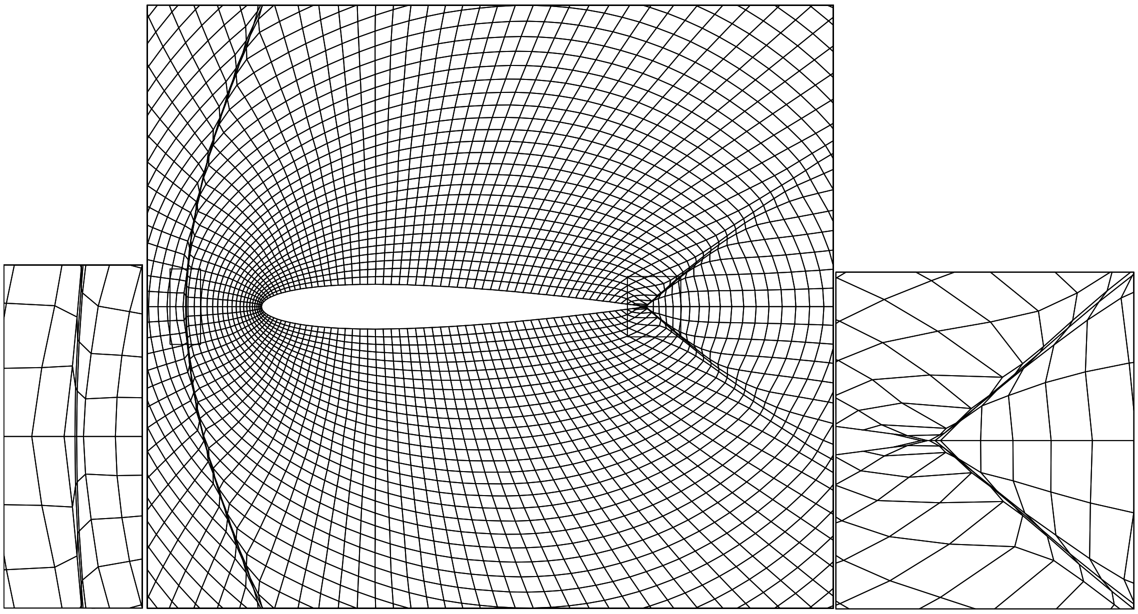


FIG. 10.8. *Mesh around NACA0012 airfoil after adaptation for supersonic flow calculations.*

First stage takes 5882 time steps to compute by t=5, adaptation does 200 steps that is about 7% of the first stage time. In the both tests we use only minimization of the functional I_1^h.

11. Concluding remarks. Results of computations presented show adaptive-harmonic grid generation increases significantly the accuracy of calculations in comparison with modeling on quasiuniform meshes due to reducing the thickness of smearing to the shock waves, while keeping the same simple grid structure and requiring less computer costs.

Theoretical analysis based on the three-point model of adaptation has shown minimization of the regularized discrete functional in 1-D and 2-D cases delivers strong grid lines compression towards the discontinuities, while keeping the mesh unfolded. The errors of computations can be conditionally divided into two kinds. The first kind errors are gained in the vicinity of the shock waves and second throughout the subdomains of smooth flow. Value of the first in the integral norms L_1 or L_2 is proportional to the thickness of the shocks smeared and, consequently, grid clustering enables to eliminate those errors or, at least, provides their magnitude to be insignificant in comparison with the second type errors, value of which depends on the numerical scheme accuracy.

Constrained minimization leads to consistent redistributing the boundary and interior mesh nodes that increases reliability of the adaptive procedure and modeling.

In 2-D tests, presented in Secs. 10.5,10.6, the mesh with strong grid lines compression looks like a set of subdomains into each the flow is smooth and the mesh is quasiuniform. Boundaries of those subdomaines coincide with the shocks and are built automatically by the condensed grid lines. If to draw a closed contour along the shock, one side to the left of the discontinuity another to the right, and to direct the contour width to zero, from the system of conservation laws, written in the integral form (9.1), we get in the limit the Rankine-Hugoniot conditions across the shock [34]. The line passing through the centers of narrow cells, stretched along the shocks, looks like that contour. To draw such limiting contour we need two adjacent narrow cells in the shock zone as three-point model of adaptation provides. In real computations, see Secs. 10.5,10.6, the shocks are smeared over 2 to 3 cells and this is nearly the same that we would like to have in the ideal case, when the shocks are treated using the Rankine-Hugoniot conditions with a special "shock-fitting" procedure, and the subdomains of smooth flow are computed using the numerical scheme. Adaptive procedure, changing the cells width in the shock zone, adjusts automatically the flow solver to those two kinds of computations that increases significantly accuracy of modeling.

Performed in Sec. 5 theoretical analysis shows in contrast to the boundary layers or shocks smeared in viscous flows, in the vicinity of the shocks in the hyperbolic problems the adjacent cells, one located in the shock zone and the other in the domain of smooth solution, with sizes in the direction of normal towards the shock differing by orders of magnitude, do not deteriorate the accuracy of the solution. Therefore, it is not necessary to change cells size gradually when passing across the shock. If to apply a numerical scheme using an artificial viscosity, the shock is smeared over several cells and we get a mesh clustered gradually towards the shock. But when decreasing the artificial viscosity and tending the smooth solution to discontinuous, the mesh will come to the above described structure when cells size changes sharply while approaching the shock. From this point of view when computing on the adaptive grids the flow solvers, using piecewise polynomial approximation of the functions, especially Godunov's-type schemes, have some advantages over the others based on continuous distribution of the flow parameters.

From the above we can state, probably, the adapted meshes to the Euler and Navier-Stokes flows must have different structure and be generated in a different way.

Acknowledgements. The author thanks Dr. Sergey Ivanenko for the attention to the work and fruitful discussions.

REFERENCES

[1] B. N. Azarenok, Adaptive Moving Grids in Supersonic Flow Simulation, Numerical Grid Generation in Computational Field Simulations, B. K. Soni, J. Haeuser, J. F. Thompson, P. Eiseman, (Eds.), Proceedings of the 7th International Conference, Whistler, British Columbia, Canada, Sept. 2000, pp. 629–638.

[2] ———, Realization of a second-order Godunov's scheme, Comp. Meth. in Appl. Mech. and Engin. 189(2000), pp. 1031–1052.

[3] ———, Adaptive moving grids in problems of gas dynamics, Grid Generation: New Trends and Applications in Real-World Simulations, S. A. Ivanenko et al. (Eds.), Proceedings of the minisimposium in Intern. Confer. "Optimization of finite-element approximations, splines and wavelets", St. Petersburg State University, June 25-29, 2001, pp. 30–44. See also http://www.ccas.ru/gridgen/.

[4] B. N. Azarenok and S. A. Ivanenko, Application of adaptive grids in numerical analysis of time-dependent problems in gas dynamics, Comput. Maths. Math. Phys., 40(2000), No. 9, pp. 1330–1349.

[5] ———, Application of moving adaptive grids for numerical solution of nonstationary problems in gas dynamics, Int. J. for Numer. Meth. in Fluids, 39(2002), No. 1, pp. 1–22. See also http://www.math.ntnu.no/conservation/2001/043.html/.

[6] M. J. Baines, Moving Finite Elements. Clarendon Press, Oxford, 1994.

[7] J. U. Brackbill and J. S. Saltzman, Adaptive zoning for singular problems in two dimensions, J. Comp. Phys. 46(1982), pp. 342–368.

[8] W. M. Cao, W. Z. Huang and R. D. Russel, A study of monitor functions for two dimensional adaptive mesh generation, SIAM J. Sci. Comput. 20(1999), pp. 1978–1994.

[9] ———, An r-adaptive finite element method based upon moving mesh PDEs, J. Comp. Phys. 149(1999), pp. 221–244.

[10] G. Carey, Computational Grids: Generation, Adaptation, and Solution Strategies, Taylor & Francis, Florida, 1997.

[11] A. A. Charakhch'yan and S. A. Ivanenko, Curvilinear grids of convex quadrilaterals, USSR Comput. Math. Math. Phys. 28(1988), No. 2, pp. 126–133.

[12] ———, A variational form of the Winslow grid generator, J. Comput. Phys. 136(1997), pp. 385–398.

[13] A. S. Dvinsky, Adaptive grid generation from harmonic maps on Riemannian manifolds, J. Comp. Phys. 95(1991), pp. 450–476.

[14] H. A. Dwyer, B. R. Sanders, and F. Raiszadek, Ignition and flame propagation studies with adaptive numerical grids, Combustion and Flame, 52 (1984), pp. 11–23.

[15] P. S. Eisman, Adaptive grid generation, Comput. Methods in Appl. Mech. and Engineering, 64(1987), pp. 321–376.

[16] J.E. Eells and L. Lemaire, Another report on harmonic mapps, Bull. London Math. Soc. 20, No. 387 (1988), 387.

[17] F. T. Farrell and L. E. Jones, Some non-homeomorphic harmonic homotopy equivalences, Bull. of London Math. Society, 28(1996), No. 131, pp. 177–182.

[18] V. A. Garanzha and I. E. Kaporin, Regularization of the barrier variational grid generation method, Comp. Math. Math. Phys., 39(1999), No. 9, pp. 1489–1503.

[19] S. K. Godunov and V. S. Ryabenkii, Difference Schemes: An Introduction to the Underlying Theory, North-Holland, Amsterdam, 1987.

[20] S. K. Godunov (Ed.), et al., Numerical Solution of Multi-Dimensional Problems in Gas Dynamics, Nauka, Moscow, 1976, (In Russian); See also: S. K. Godunov, A. V. Zabrodin, M. Ya. Ivanov, A. N. Kraiko and G. P. Prokopov, Résolution Numérique des Problémes Multidimensionnels de la Dynamique des Gaz, Mir, Moscou, 1979 (In French).

[21] D. F. Hawken, J. J. Gottlieb and J. S. Hansen, Review of some adaptive node-movement techniques in finite-element and finite-difference solutions of partial differential equations, J. Comput. Phys. 95(1991), pp. 254–302.

[22] W. Z. Huang, Practical aspects of formulation and solution of moving mesh PDEs, J. Comp. Phys. 171(2001), pp. 753–775.

[23] S. A. Ivanenko, Adaptive grids and grids on surfaces, Comput. Math. Math. Phys. 33(1993), No. 9, pp. 1179–1193.

[24] ———, Harmonic mappings, Chap. 8 in Handbook of Grid Generation. J. F. Thompson et al. (Eds.), CRC Press, Boca Raton, Fl., 1998.

[25] ———, Optimality principle for nondegenerate grids, Grid Generation: New Trends and Applications in Real-World Simulations, S. A. Ivanenko et al. (Eds.), Proceedings of the minisimposium in Intern. Confer. "Optimization of finite-element approximations, splines and wavelets", St. Petersburg State University, June 25-29, 2001, pp. 85-99. See also http://www.ccas.ru/gridgen/.

[26] O-P. Jacquotte, Grid optimization methods for quality improvement and adaptation, Chap. 33 in Handbook of Grid Generation. J. F. Thompson et al. (Eds.), CRC Press, Boca Raton, Fl., 1998.

[27] P. Knupp and S. Steinberg, Fundamentals of Grid Generation, CRC Press, Boca Raton, Fl., 1994.

[28] R.Li, T.Tang, and P.Zhang, Moving mesh methods in multiple dimensions based on harmonic maps, J. Comp. Phys. 170(2001), pp. 562–588.

[29] ———, A moving mesh finite element algorithm for singular problems in two and three dimensions, Preprint, 2001. See also http://www.math.ntnu.no/conservation/ .

[30] F. Liu, S. Ji, and G. Liao, An adaptive grid method and its application to steady Euler flow calculations, SIAM J. Sci. Comput. 20(1998), pp. 811–825.

[31] V. D. Liseikin, On generation of regular grids on n-dimensional surfaces, USSR Comput. Math. Math. Phys. 31(1991), No. 11, pp. 47–57.

[32] ———, Grid Generation Methods, Springer-Verlag, New York, 1999.

[33] J. A. Mackenzie, R. D. Russell and J. M. Stockie, A moving mesh method for one dimensional hyperbolic conservation law, SIAM J. Sci. Comput. 22(2000), pp. 1791–1813.

[34] B. L. Rozhdestvenskii and N. N. Yanenko, Systems of Quasilinear Equations and Their Applications to Gas Dynamics, Amer. Math. Soc. Transl. of Math. Monographs, 55, AMS, Providence, RI, 1983.

[35] J. S. Sampson, Some properties and applications of harmonic mappings, Ann. École Norm. Sup. 11 (1978), pp. 211–228.

[36] R. Schoen and S. T. Yau, On univalent harmonic maps between surfaces, Invent. Math. 44 (1978), pp. 265–278.

[37] S. P. Spekreijse, R. Hagmeijer, J. M. Boerstoel, Adaptive grid generation by using Laplace-

Beltrami operator on a monitoring surface, Numerical Grid Generation in Computational Field Simulations, B. K. Soni, J. Haeuser, J. F. Thompson, P. R. Eiseman, (Eds.), Proceedings of the 5th International Conference, Missisipi State, MS, U.S., April 1996, pp. 137–146.

[38] H. Tang and T. Tang, Moving mesh methods for one- and two-dimensional hyperbolic conservation laws, SIAM J. Numer. Anal., submitted for publication.

[39] J. F. Thompson, A survey of dynamically-adaptive grids in the numerical solution of partial differential equations, Appl. Numer. Math. 1(1985), pp. 3–27.

[40] A. Winslow, Numerical solution of the quasi-linear Poisson equation in a nonuniform triangle mesh, J. Comp. Phys. 1(1966), pp. 149–172.

[41] P. A. Zegeling, Moving grid techniques, Chap. 37 in Handbook of Grid Generation. J. F. Thompson et al. (Eds.), CRC Press, Boca Raton, Fl., 1998.