REPORT ON APPLIED MATHEMATICS

Boris N. Azarenok [1]
Sergey A. Ivanenko [2]
Tao Tang [3]

GODUNOV'S SCHEME and MOVING ADAPTIVE GRIDS

2002

[1]Computing Center of Russian Academy of Sciences Vavilov str. 40, GSP-1, Moscow, Russia. Email: azarenok@ccas.ru . Part of this work has been performed at the Department of Mathematics, Hong Kong Baptist University.

[2]Computing Center of Russian Academy of Sciences Vavilov str. 40, GSP-1, Moscow, Russia. Email: sivan@ccas.ru

[3]Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong. Email: ttang@math.hkbu.edu.hk

1 Introduction

Coupled algorithm of using the Godunov's type solver and adaptive moving mesh has been offered in [1, 17]. In this approach after every mesh iteration the finite-volume flow solver updates the flow parameters at the new time level directly on the curvilinear moving grid without interpolation from one mesh to another [2]. Thus, we eliminate the errors caused by interpolation procedure which smears the discontinuities. This scheme on one hand utilizes the idea of the Godunov's scheme on the deforming meshes [13] and on the other hand is of the second-order accuracy in time and space.

Method of adaptive grid generation has been suggested in [15, 8, 16] and it is based on the theory of harmonic maps [29, 30]. Method is variational, i.e. we consider the problem of minimizing a finite-difference function approximating the Dirichlet's functional written for surfaces. The discrete functional has an infinite barrier at the boundary of the set of grids with all convex quadrilateral cells and this guarantees unfolded grid generation during computations both in any simply connected, including nonconvex, and multiply connected 2D domains. This folding-resistant property is very important since if any of the cells becomes folded we have to stop calculation of the flow problem and use special procedures to continue modeling. The barrier property is also of particularly importance in the vicinity of shock waves where the cells are very narrow and maximal aspect ratio achieves 50 and larger [3].

When modeling 2D hyperbolic problems with discontinuous solution on the moving adaptive mesh it is possible to reduce the errors, caused by shocks smearing over the cells, by many factors of ten and, therefore, to decrease significantly, by several times [3], the overall error.

Moving grid techniques have been developed in a number of works [33, 26, 36, 5, 6]. All these methods, in contrast to presented in this report, do not ensure the generated meshes to be unfolded at every time step [4].

2 Flow solver

In this section we describe the Godunov Linear Flux Correction (GLFC) scheme to compute the 2D ideal gas flow on a moving mesh [2]. This scheme on one hand utilizes the idea of the Godunov's scheme on the deforming meshes [13] and on the other hand is of the second-order accuracy in time and space.

2.1 System of equations

The governing system of the differential equations relating to 2D gas flow in Euler approach is

$$\frac{\partial \boldsymbol{\sigma}}{\partial t} + \frac{\partial \boldsymbol{a}}{\partial x} + \frac{\partial \boldsymbol{b}}{\partial y} = \boldsymbol{c} \ ,$$

[4] Note the recent paper [18] where it is used the same idea to minimize the discrete Dirichlet's functional possesing the infinite barrier property. This idea of using the Dirichlet's functional has been suggested in [7].

where the vector-valued functions are defined by

$$\boldsymbol{\sigma}{=}(\rho,\rho u,\rho v,E)^\top,\quad \boldsymbol{a}{=}(\rho u,\rho u^2{+}p,\rho uv,u(E{+}p))^\top,$$
$$\boldsymbol{b}{=}(\rho v,\rho uv,\rho v^2{+}p,v(E{+}p))^\top,\quad \boldsymbol{c}{=}-\frac{\nu v}{y}(\rho,\rho u,\rho v,E{+}p)^\top,$$

u and v are the velocity components, p and ρ are the pressure and density, respectively, $\nu{=}0,1$ represent the planar and axi-symmetric flow, respectively. 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 function of flow parameters as $\boldsymbol{f}{=}(u,v,p,\rho)^T$.

We perform calculations utilizing integral conservation laws which can be derived by integrating the above system of the differential equations and transforming the volume integrals in space $x{-}y{-}t$ to the surface integrals by virtue of Gauss's theorem

$$\iiint_V\left[\frac{\partial\boldsymbol{\sigma}}{\partial t}+\frac{\partial\boldsymbol{a}}{\partial x}+\frac{\partial\boldsymbol{b}}{\partial y}-\boldsymbol{c}\right]dV=\oiint_{\partial V}\boldsymbol{\sigma}dxdy+\boldsymbol{a}dydt+\boldsymbol{b}dtdx-\iiint_V\boldsymbol{c}dxdydt=0\ ,$$

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, as follows [13]

$$\oiint_{\partial V}\boldsymbol{\sigma}dxdy+\boldsymbol{a}dydt+\boldsymbol{b}dtdx=\iiint_V\boldsymbol{c}dxdydt, \tag{1}$$

The conservation laws (1) hold for any parameters $\boldsymbol{f}$, both smooth and discontinuous, and, thus, they gover a real gas flow.

2.2 Discretization of equations

We introduce the curvilinear moving grid in $\mathcal{R}^3$ space $x{-}y{-}t$, a hexahedral cell of which is depicted in Fig.1. The bottom face of the cell (or control volume) is taken at time level n and the top face at level $n{+}1$, while four lateral faces generally form ruled surfaces rather than simple planes.

Integrating (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

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

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 [13] where the fluxes in (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.

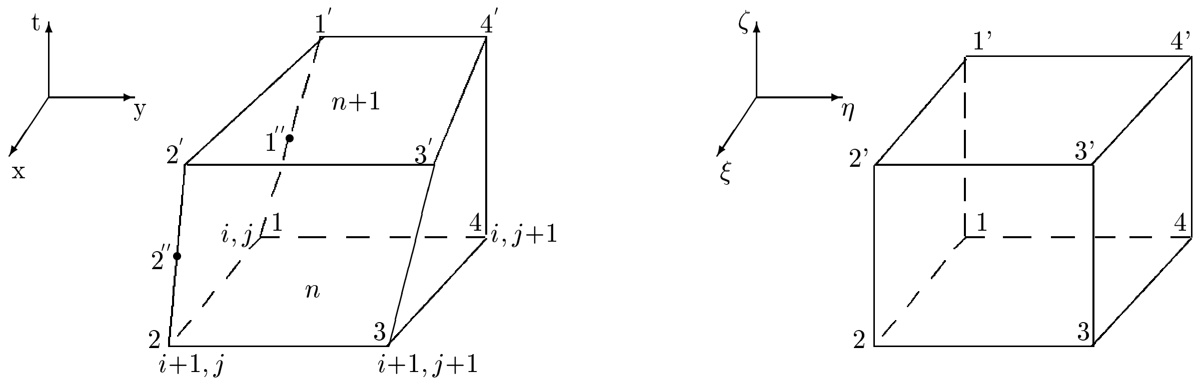

Figure 1: Hexahedral cell in $\mathcal{R}^3$ space $x{-}y{-}t$ being the image of the unit cube $I^3 = \{(\xi,\eta,\zeta) : 0 \le \xi,\eta,\zeta \le 1\}$ in parametric space $\mathcal{R}^3$

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

$$\boldsymbol{Q}_{122'1'} = \boldsymbol{\sigma}^{n+1/2} A^{xy} + \boldsymbol{a}^{n+1/2} A^{yt} + \boldsymbol{b}^{n+1/2} A^{tx}, \tag{3}$$

where $\boldsymbol{\sigma}^{n+1/2}, \boldsymbol{a}^{n+1/2}, \boldsymbol{b}^{n+1/2}$ are calculated using parameters $\boldsymbol{f}^{n+1/2}$ in the center of the face, i.e. at mid point of edge 12 at time $t^{n+1/2}$. 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

$$\begin{aligned} A^{xy} = & \iint_{122'1'} dxdy = & 0.5[(x_{2'} - x_1)(y_{1'} - y_2) - (x_{1'} - x_2)(y_{2'} - y_1)], \\ A^{yt} = & \iint_{122'1'} dydt = & 0.5\triangle t(y_{2'} + y_2 - y_1 - y_{1'}), \\ A^{tx} = & \iint_{122'1'} dtdx = & -0.5\triangle t(x_{2'} + x_2 - x_1 - x_{1'}), \end{aligned} \tag{4}$$

which are obtained from the formula for the quadrangle 1234

$$A_{1234} = A(x_1, y_1; x_2, y_2; x_3, y_3; x_4, y_4) = 0.5[(x_3 - x_1)(y_4 - y_2) - (x_4 - x_2)(y_3 - y_1)]$$

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

The vector-valued function $\boldsymbol{c}^{n+1/2}$ is defined in the center of quadrilateral cell at time $t^{n+1/2}$ (i.e. in the center of control volume), V is the volume of the hexahedral cell. Formulae for the control volume V of a ruled/(hexahedral) cell were derived in [11, 34]. For instance, in [11] V is computed by integrating over the control volume surface ∂V. Contribution to the total volume V from the surface 1234, refered as V_{1234}, when integrating over the bottom face 1234 reads

$$\begin{aligned} V_{1234} = [\boldsymbol{r}_1 \cdot (\boldsymbol{r}_4 \times \boldsymbol{r}_3) + \boldsymbol{r}_2 \cdot (\boldsymbol{r}_1 \times \boldsymbol{r}_4) + \boldsymbol{r}_3 \cdot (\boldsymbol{r}_2 \times \boldsymbol{r}_1) + \boldsymbol{r}_4 \cdot (\boldsymbol{r}_3 \times \boldsymbol{r}_2)]/12 = \\ (\boldsymbol{r}_1 + \boldsymbol{r}_2) \cdot [(\boldsymbol{r}_1 + \boldsymbol{r}_4) \times (\boldsymbol{r}_2 + \boldsymbol{r}_3)]/12\ , \end{aligned} \tag{5}$$

where $\boldsymbol{r}_i{=}(x, y, t)_i$. Formulae to the contribution of the other five cell faces can be obtained from (5) by cyclic permutation of indexes. In [34] it is prooved that V is equal mean of volumes of two dodecahedrons with planar faces which have the same vertexes as the hexahedral cell.

To determine $\boldsymbol{c}^{n+1/2}$ we need also the coordinate y_c of the cell center which is obtained from trilinear mapping forming the hexahedral cell, see Fig. 1. At ξ, η, ζ=0.5 it is given by

$$y_c = 0.125 \sum_{i=1}^{8} y_i \ .$$

The values $\boldsymbol{f}^{n+1}$ are updated by two stages using time splitting [27] that provides the scheme with the second order in time. Another way to provide the scheme with second order accuracy in time has been suggested in [4] where the Riemann problem is solved based on piecewise linear distribution of the flow parameters (generalized Riemann problem).

At the first stage, predictor, via (2) we compute the intermediate values at the n+1-th level $\bar{\boldsymbol{f}}^{n+1}$. 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 (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.

2.3 Determination of fluxes

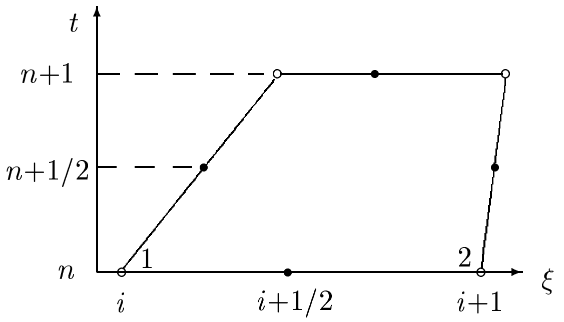

Figure 2: Computing cell of 1D grid along the curvilinear coordinate ξ

For clarity let us consider determination of the fluxes in 1-D case. In 2-D case such an algorithm is executed along each of the curvilinear coordinates. Consider a cell of the 1D grid along the curvilinear coordinate ξ shown in Fig.2. Assume the function $\boldsymbol{f}$ to be linear within the zone $(i, i+1)$. For simplicity we omit the superscript n keeping in mind, if it is not specified, all the values are taken at t^n. Then $\boldsymbol{f}_{1,i+1/2}$ and $\boldsymbol{f}_{2,i+1/2}$, the values at the left and right zone ends, see Fig.2, are defined by

$$\boldsymbol{f}_{1,i+1/2} = \boldsymbol{f}_{i+1/2} - 0.5\delta\boldsymbol{f}_{i+1/2}h_{i+1/2} \quad , \quad \boldsymbol{f}_{2,i+1/2} = \boldsymbol{f}_{i+1/2} + 0.5\delta\boldsymbol{f}_{i+1/2}h_{i+1/2} \ . \qquad (6)$$

Here the spacing along the ξ coordinate line at fixed index j is given

$$h_{i+1/2} = 0.5\sqrt{(x_{i+1,j} + x_{i+1,j+1} - x_{i,j} - x_{i,j+1})^2 + (y_{i+1,j} + y_{i+1,j+1} - y_{i,j} - y_{i,j+1})^2}.$$

The "effective" derivative $\delta \boldsymbol{f}_{i+1/2}$ is defined via the auxiliary values $\tilde{\boldsymbol{f}}_{1,i+1/2}$ and $\tilde{\boldsymbol{f}}_{2,i+1/2}$ at the right and left zone ends

$$\delta \boldsymbol{f}_{i+1/2} = (\tilde{\boldsymbol{f}}_{2,i+1/2} - \tilde{\boldsymbol{f}}_{1,i+1/2})/h_{i+1/2} \ . \tag{7}$$

The auxilairy values at the left zone end $\tilde{\boldsymbol{f}}_{1,i+1/2}$ and right zone end $\tilde{\boldsymbol{f}}_{2,i+1/2}$ are obtained using the Taylor expansion about the points ξ_i and ξ_{i+1}, respectively,

$$\boldsymbol{f}_{i+1/2} = \boldsymbol{f}_i + \frac{1}{2}h_{i+1/2}\boldsymbol{f}_i^{(1)} + \frac{1}{8}h_{i+1/2}^2\boldsymbol{f}_i^{(2)} + O(h_{i+1/2}^3) \ ,$$

$$\boldsymbol{f}_{i+1/2} = \boldsymbol{f}_{i+1} - \frac{1}{2}h_{i+1/2}\boldsymbol{f}_{i+1}^{(1)} + \frac{1}{8}h_{i+1/2}^2\boldsymbol{f}_{i+1}^{(2)} - O(h_{i+1/2}^3) \ .$$

From the above two expressions, neglecting the terms of $O(h^3)$, we get

$$\begin{aligned} \tilde{\boldsymbol{f}}_{1,i+1/2} &= \boldsymbol{f}_i = \boldsymbol{f}_{i+1/2} - \frac{1}{2}h_{i+1/2}\boldsymbol{f}_i^{(1)} - \frac{1}{8}h_{i+1/2}^2\boldsymbol{f}_i^{(2)} \ , \\ \tilde{\boldsymbol{f}}_{2,i+1/2} &= \boldsymbol{f}_{i+1} = \boldsymbol{f}_{i+1/2} + \frac{1}{2}h_{i+1/2}\boldsymbol{f}_{i+1}^{(1)} - \frac{1}{8}h_{i+1/2}^2\boldsymbol{f}_{i+1}^{(2)} \ . \end{aligned} \tag{8}$$

The first and second derivatives are given by relations

$$\begin{aligned} \boldsymbol{f}_i^{(1)} &= \frac{2}{h_{i+1/2}+h_{i-1/2}}\left(\boldsymbol{f}_{i+1/2} - \boldsymbol{f}_{i-1/2}\right) \ , \\ \boldsymbol{f}_i^{(2)} &= \frac{2}{h_{i+1/2}+h_{i-1/2}}\left(\boldsymbol{f}_{i+1/2}^{(1)} - \boldsymbol{f}_{i-1/2}^{(2)}\right) \ . \end{aligned} \tag{9}$$

Substituting them into (8) gives

$$\begin{aligned} \tilde{\boldsymbol{f}}_{1,i+1/2} &= \boldsymbol{f}_{i+1/2} - \frac{h_{i+1/2}}{h_{i+1/2}+h_{i-1/2}}\left(\boldsymbol{f}_{i+1/2} - \boldsymbol{f}_{i-1/2}\right) - \frac{h_{i+1/2}^2}{4(h_{i+1/2}+h_{i-1/2})}\left(\boldsymbol{f}_{i+1/2}^{(1)} - \boldsymbol{f}_{i-1/2}^{(1)}\right) \ , \\ \tilde{\boldsymbol{f}}_{2,i+1/2} &= \boldsymbol{f}_{i+1/2} + \frac{h_{i+1/2}}{h_{i+3/2}+h_{i+1/2}}\left(\boldsymbol{f}_{i+3/2} - \boldsymbol{f}_{i+1/2}\right) - \frac{h_{i+1/2}^2}{4(h_{i+3/2}+h_{i+1/2})}\left(\boldsymbol{f}_{i+3/2}^{(1)} - \boldsymbol{f}_{i+1/2}^{(1)}\right) \ . \end{aligned} \tag{10}$$

2.4 Monotonicity algorithm

In (10) every components of the derivative $\boldsymbol{f}^{(1)}$ is defined by the Kolgan's principle of minimal derivatives [19]. For example, to the velocity u it reads

$$u_{i+1/2}^{(1)} = \begin{cases} u_i^{(1)} \ , & \text{if } |u_i^{(1)}| \le |u_{i+1}^{(1)}| \\ u_{i+1}^{(1)} \ , & \text{otherwise} \ , \end{cases}$$

where the derivative is given by the first relation of (9).

Afterwards we apply the Collela-Woodward's monotonicity algorithm [9] to reset the auxiliary values $\tilde{\boldsymbol{f}}_{1,i+1/2}$ and $\tilde{\boldsymbol{f}}_{2,i+1/2}$. We construct an interpolating parabola within the zone for every component of the vector $\boldsymbol{f}$. For example, to u we have

$$U(\zeta) = a_0 + a_1\zeta + a_2\zeta^2 \ , \quad \zeta = (\xi - \xi_{i+1/2})/h_{i+1/2} \ , \quad \xi_i \le \xi \le \xi_{i+1} \ , \quad -0.5 \le \zeta \le 0.5 \ .$$

The parabola takes on the following values at the zone ends $U(-0.5){=}u_{1,i+1/2}$, $U(0.5){=}u_{2,i+1/2}$ and for it the value $u_{i+1/2}$ is a zone average $u_{i+1/2} = \int_{-0.5}^{0.5} U(\xi)d\xi$. Coefficients of the parabola are equal $a_1{=}\tilde{u}_{2,i+1/2}{-}\tilde{u}_{1,i+1/2}$, $a_2{=}3(\tilde{u}_{1,i+1/2}{+}\tilde{u}_{2,i+1/2}){-}6u_{i+1/2}$ and we need not a_0.

A condition to be imposed is that the interpolating parabola shall not achieve its maximum or minimum within the zone. Otherwise the values $\tilde{u}_{1,i+1/2}$,$\tilde{u}_{2,i+1/2}$ are corrected to ensure the parabola to be monotone. There are two possible cases when it takes place:

a) $u_{i+1/2}$ does not lie inside the interval from $\tilde{u}_{1,i+1/2}$ to $\tilde{u}_{2,i+1/2}$. In such a case the interpolating function is set to be a constant, i.e.

$$\tilde{u}_{1,i+1/2} = \tilde{u}_{2,i+1/2} = u_{i+1/2} \ , \text{ if } \quad (\tilde{u}_{2,i+1/2} - u_{i+1/2})(u_{i+1/2} - \tilde{u}_{1,i+1/2}) \le 0 \ ;$$

b) $u_{i+1/2}$ lies inside the above interval, but sufficiently close to one of these values. Then we reset $\tilde{u}_{1,i+1/2}$ or $\tilde{u}_{2,i+1/2}$ (which is farther from $u_{i+1/2}$) so that the interpolating parabola becomes monotone as follows

$$\begin{cases} \tilde{u}_{1,i+1/2} = u_{i+1/2} + 2(u_{i+1/2} - \tilde{u}_{2,i+1/2}) \ , & \text{if } a_1^2 < -a_1 a_2 \ , \\ \tilde{u}_{2,i+1/2} = u_{i+1/2} + 2(u_{i+1/2} - \tilde{u}_{1,i+1/2}) \ , & \text{if } a_1^2 < a_1 a_2 \ . \end{cases}$$

Next from (7) we get $\delta\boldsymbol{f}_{i+1/2}$ and from (6) the zone end values $\boldsymbol{f}_{1,i+1/2}$ and $\boldsymbol{f}_{2,i+1/2}$ used to calculate the fluxes $\boldsymbol{Q}_{411'4'}$ and $\boldsymbol{Q}_{233'2'}$ via formulas being similar to (3). In the same manner we evaluate $\boldsymbol{f}_{1,j+1/2}$ and $\boldsymbol{f}_{2,j+1/2}$ used to calculate the fluxes $\boldsymbol{Q}_{122'1'}$ and $\boldsymbol{Q}_{433'4'}$. Further, from (2) we obtain the intermediate value $\bar{\boldsymbol{\sigma}}^{n+1}$ at time t^{n+1}.

At the second stage, corrector, we first get the pre-wave values $\boldsymbol{f}$ at the center of the intercell faces at time $t^{n+1/2}$. For it we set the effective derivative at t^{n+1} equal the one at t^n, i.e. $\delta\bar{\boldsymbol{f}}^{n+1}_{i+1/2}{=}\delta\boldsymbol{f}^n_{i+1/2}$. Then the zone end values to the intermediate function $\bar{\boldsymbol{f}}^{n+1}_{i+1/2}$ are obtained from (6). Taking the mean of edge values at t^n and t^{n+1} we get the pre-wave states at the faces 411'4' and 233'2' at time $t^{n+1/2}$ as follows:

$$\begin{aligned} \boldsymbol{f}^{n+1/2}_{1,i+1/2} &= 0.5\left[\boldsymbol{f}^n_{i+1/2} + \bar{\boldsymbol{f}}^{n+1}_{i+1/2} - 0.5\delta\boldsymbol{f}^n_{i+1/2}(h^n_{i+1/2} + h^{n+1}_{i+1/2})\right] \ , \\ \boldsymbol{f}^{n+1/2}_{2,i+1/2} &= 0.5\left[\boldsymbol{f}^n_{i+1/2} + \bar{\boldsymbol{f}}^{n+1}_{i+1/2} + 0.5\delta\boldsymbol{f}^n_{i+1/2}(h^n_{i+1/2} + h^{n+1}_{i+1/2})\right] \ . \end{aligned} \tag{11}$$

Similarly we calculate $\boldsymbol{f}^{n+1/2}_{1,j+1/2}, \boldsymbol{f}^{n+1/2}_{2,j+1/2}$, the pre-wave states at the faces 122'1'and 433'4' at time $t^{n+1/2}$.

2.5 Riemann problem on the moving mesh

In this subsection we solve the Riemann problem to each of four intercell faces taking into account the velocity of the intercell face and using the Riemann solver suggested in [13] (for description

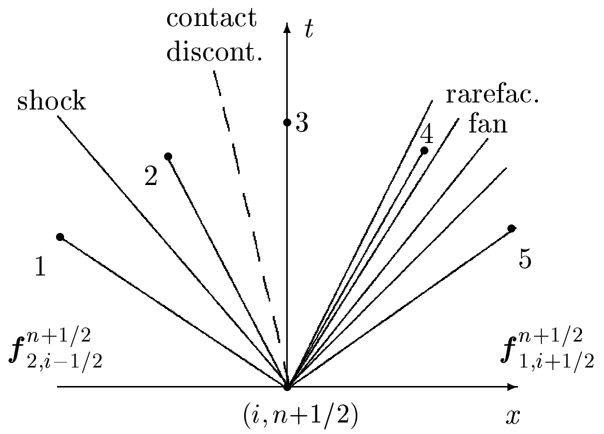

Figure 3: Five cases of location of the intercell boundary $[(i, n{+}1/2)(i, n{+}1)]$ in the wave pattern. Points $1, \ldots, 5$ denote the node $(i, n{+}1)$.

of this Riemann solver see also [20]). Numerical calculations have shown that for 2D problems with intensive shocks using this Riemann solver based on the exact solution of the nonlinear PDE's system gives loss in computing time by only ≈5% in comparison with the time when the Riemann solver based on the linearized system (cf. [12]) is employed. The idea of taking into account the velocity of the intercell face can be demonstrated to the 1D case. Consider the intercell boundary of the cell within the time interval $(t^{n+1/2}, t^{n+1})$. Let after solving the Riemann problem at point $(i, n{+}1/2)$ we have the wave pattern depicted in Fig.3. There are 5 cases of location of the intercell boundary $[(i, n{+}1/2)(i, n{+}1)]$ in the wave pattern depending on velocity of the i-th node w_i. As the post-wave values $\boldsymbol{f}_i^{n+1/2}$ we use:

1. $\boldsymbol{f}_i^{n+1/2} = \boldsymbol{f}^{n+1/2}_{2,i-1/2}$ if $w_i < u_{shock}$, where u_{shock} is the speed of the left shock.

2. $\boldsymbol{f}_i^{n+1/2} = \boldsymbol{f}_2$ if $u_{shock} < w_i < u_{cont}$, where the vector $\boldsymbol{f}_2$ defines the flow parameters after the shock, u_{cont} is the speed of the contact discontinuity.

3. $\boldsymbol{f}_i^{n+1/2} = \boldsymbol{f}_3$ if $u_{cont} < w_i < u^{lft}_{raref}$, where the vector $\boldsymbol{f}_3$ defines parameters in the domain between the contact discontinuity and left characteristic of the rarefaction wave expanding with the speed u^{lft}_{raref}. We should chose these flow parameters if calculate on the fixed mesh.

4. $\boldsymbol{f}_i^{n+1/2} = \phi(x/t)$ if $u^{lft}_{raref} < w_i < u^{rght}_{raref}$, i.e. we calculate the flow parameters in the rarefaction wave using the variable x/t. Here u^{rght}_{raref} is the speed of the right characterisic in the rarefaction fan.

5. $\boldsymbol{f}_i^{n+1/2} = \boldsymbol{f}^{n+1/2}_{1,i+1/2}$ if $w_i > u^{rght}_{raref}$

In 2D case we by analogy consider the Riemann problem on the moving mesh. To get the post-wave states $\boldsymbol{f}^{n+1/2}$ in the center of face $122'1'$, i.e. at the mid point of the segment $1''2''$, being the edge 12 at $t^{n+1/2}$, see Fig.1, we solve the Riemann problem with the pre-wave states

$(r,p,\rho)^{n+1/2}$ at this point on the both sides of the face (one state relates to the cell considered and the other to the cell adjacent to the face $122'1'$), here $r^{n+1/2}$ is a normal components of the velocity, and use the tangential components of the velocity $q^{n+1/2}$ on those sides. The normal and tangential components of the velocity are given

$$r^{n+1/2} = n_x u^{n+1/2} + n_y v^{n+1/2} \quad , \qquad q^{n+1/2} = n_y u^{n+1/2} - n_x v^{n+1/2} \ ,$$

here n_x and n_y are the components of the outward normal vector to the segment $1''2''$.

After solving the Riemann problem we find the tangential component $q^{n+1/2}$ via

$$q^{n+1/2} = \begin{cases} q^{n+1/2} & \text{if } \ w_{12} \leq u_{cont} \ , \\ \tilde{q}^{n+1/2} & \text{otherwise} \ , \end{cases} \tag{12}$$

here u_{cont} is the speed of the contact discontinuity in the Riemann problem, w_{12} is the velocity of the edge 12 in the normal direction to this edge, $\tilde{q}^{n+1/2}$ is the tangential component of the velocity in the cell adjacent to the face $122'1'$. The condition (12) expresses the fact that tangential component of the velocity is discontinuous across the tangential/(contact) discontinuity [10]. Velocity w_{12} can be defined from the equality

$$\triangle t l_{1''2''} w_{12} = A^{xy} \ , \tag{13}$$

here $l_{1''2''}$ is the distance between the points $1''$ and $2''$.

Afterwards we restore the Cartesian components of the velocity in the center of the face

$$u^{n+1/2} = n_x r^{n+1/2} + n_y q^{n+1/2} \quad , \qquad v^{n+1/2} = n_y r^{n+1/2} - n_x q^{n+1/2} \ .$$

When calculating $\boldsymbol{c}^{n+1/2}$ the average value component of the velocity in the $(i+1/2, j+1/2)$-th point $u^{n+1/2}$ is

$$u^{n+1/2} = 0.5(u^n + \bar{u}^{n+1})$$

where $\bar{u}^{n+1}$ is the intermediate value obtained at the predictor stage. Similarly we get $(v,p,\rho)^{n+1/2}$.

After solving the Riemann problem we obtain the post-wave values $\boldsymbol{f}$ at the center of 4 intercell faces. Calculating the fluxes through the intercell face 122'1' via (3) and similarly through the other 3 intercell faces, and substituting the values of the fluxes into (2) we get the final values of mass, momentum and energy $\boldsymbol{\sigma}^{n+1}$ at time t^{n+1}.

Note that, when calculating by (10) the auxilairy values, at the boundary nodes we should use the boundary conditions. For instance, if at the left node i=0 the boundary condition is defined as $\boldsymbol{f}_{left}$ then we set $\tilde{\boldsymbol{f}}_{1,1/2}$=$\boldsymbol{f}_{left}$. If the boundary condition is defined to the fluxes, e.g. reflecting boundary condition, we set $\tilde{\boldsymbol{f}}_{1,1/2}$=$\boldsymbol{f}_0^{n-1/2}$, where $\boldsymbol{f}_0^{n-1/2}$ are the post-wave values at the preceding time step.

2.6 Stability condition

In 1D case the choice of admissible time step $\triangle t$ has a clear physical sense. On the moving grid $\triangle t$ is given by

$$\triangle t = c_{cfl} \ \min_i \ \triangle t_{i+1/2} \tag{14}$$

where in every cell shown in Fig.1b the local time step is determined via [13]

$$\triangle t_{i+1/2} = \frac{h_{i+1/2}}{\max(d_i^{II} - w_{i+1} \ , \ -d_{i+1}^{I} - w_i)} \ , \tag{15}$$

here d_i^{II} and d_{i+1}^I are the extreme right and left wave speeds at the points x_i and x_{i+1}, respectively, obtained from solving the Riemann problem, w_i is the velocity of the node x_i, i.e., the slope of the intercell boundary. The condition (15) means that we estimate the time within which the left characteristic (in linearized analysis this is a straight line), emanating from the i+1-th node, achives the i-th node, then estimate the time within which the right characteristic, emanating from the i-th node, achives the i+1-th node. From these two time steps we take the minimal one.

The Courant coefficient c_{cfl} (or coefficient of reserve [13]) is a correction to non-linearity of the PDE system (1). To calculate the velocity w_i on one hand it is necessary to know the time step $\triangle t$. On the other hand w_i participates in calculation of $\triangle t$. By this reason at time level n+1 we use $\triangle t$ calculated at the preceding level n. The coefficient c_{cfl}<1 (usualy 0.5 to 0.9) and it may be corrected during computing. In the problems where the grid moves with high velocity it may be necessary to decrease c_{cfl} greatly to provide the stability of calculations.

In linearized analysis on the fixed grid we use a sound approach to get $\triangle t_{i+1/2}$

$$\triangle t_{i+1/2} = \frac{h_{i+1/2}}{\max(u_{i+1/2} + a_{i+1/2} \ , \ a_{i+1/2} - u_{i+1/2})} \tag{16}$$

where $a_{i+1/2}$ is the sound speed. The formula (16) can be also used in case of the meshes moving with not very high velocity.

In 2D case the choice of admissible $\triangle t$ is estimated to the t-hyperbolic by Friedrichs system [13]. The system of equations (1) in the differential form is reduced to the following

$$\tilde{A}(\boldsymbol{f})\frac{\partial \boldsymbol{f}}{\partial t} \ + \tilde{B}(\boldsymbol{f})\frac{\partial \boldsymbol{f}}{\partial x} \ + \tilde{C}(\boldsymbol{f})\frac{\partial \boldsymbol{f}}{\partial y} \ = 0 \ ,$$

where $\tilde{A}$,$\tilde{B}$,$\tilde{C}$ are symmetric matrices $n \times n$ and $\tilde{A}$ is positive defined. Next the above system is linearized as follows

$$A\frac{\partial \boldsymbol{f}}{\partial t} \ + B\frac{\partial \boldsymbol{f}}{\partial x} \ + C\frac{\partial \boldsymbol{f}}{\partial y} \ = 0 \ ,$$

where A, B, C are the matrices with constant elements. The integral of energy to this system is

$$\frac{\partial}{\partial t}\,(A\boldsymbol{f},\boldsymbol{f}) + \frac{\partial}{\partial t}\,(B\boldsymbol{f},\boldsymbol{f}) + \frac{\partial}{\partial y}\,(C\boldsymbol{f},\boldsymbol{f}) = 0$$

Discretizing this sytem on the rectangular mesh with steps h_x, h_y it can be shown that, for instance to the Cauchy problem, the energy norm of the discrete function $\boldsymbol{f}^{n+1}_{i+1/2,j+1/2}$ does not increase, i.e. the scheme is stable

$$\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}(A\boldsymbol{f}^{n+1}_{i+1/2,j+1/2},\boldsymbol{f}^{n+1}_{i+1/2,j+1/2}) \le \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}(A\boldsymbol{f}^{n}_{i+1/2,j+1/2},\boldsymbol{f}^{n}_{i+1/2,j+1/2})$$

subject to

$$\triangle t\left(\frac{1}{\triangle t_x}+\frac{1}{\triangle t_y}\right)\le 1 \tag{17}$$

where $\triangle t_x, \triangle t_y$ are admissible 1D time steps in the x and y directions. The formula (17) has been proved on the parallelogram mesh as well [13].

In the case of the curvilinear mesh to calculate the admissible time step in the $(i+1/2, j+1/2)$-th cell the formula (17) is generalized as follows [13]

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

$$\text{where}\quad \triangle t^{'} = \frac{h^{'}}{\max(d^{II}_{41}-w_{41};-d^{I}_{23}-w_{23})},\qquad \triangle t^{''} = \frac{h^{''}}{\max(d^{II}_{12}-w_{12};-d^{I}_{34}-w_{34})},$$

$$h^{'} = \frac{A_{1234}}{0.5\sqrt{(x_4+x_3-x_1-x_2)^2+(y_4+y_3-y_1-y_2)^2}},$$

$$h^{''} = \frac{A_{1234}}{0.5\sqrt{(x_3+x_2-x_4-x_1)^2+(y_3+y_2-y_4-y_1)^2}}.$$

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 in the normal direction to this edge, e.g., w_{12} is the velocity of the edge 12 in the normal direction to the edge 12 determined from (13). Next, d^{II}_{12} and d^{II}_{41} are the "extreme right wave" speeds defined from solving the Riemann problem to the faces $122'1'$ and $11'4'4$, respectively; d^{I}_{23} and d^{I}_{34} are the "extreme left wave" speeds to the faces $233'2'$ and $433'4'$, respectively.

The resulting time step over all mesh is given by

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

The $GLFC$ scheme is of the second-order accuracy in time and space in the domains of smooth flow subject to spacings in the ξ and η directions are satisfied

$$h_{i+1/2}-h_{i-1/2} = O((h_{i+1/2})^2)\ ,\quad h_{j+1/2}-h_{j-1/2} = O((h_{j+1/2})^2)$$

and the nodes move with small velocity

$$x^{n+1}_{i,j}-x^{n}_{i,j} = O((h_{i+1/2})^2)\ ,\quad x^{n+1}_{i,j}-x^{n}_{i,j} = O((h_{j+1/2})^2)\ .$$

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.

2.7 1D case

Now we shall describe briefly the $GLFC$ scheme in 1D case. The gas flow in Euler approach is governed by the following system of hyperbolic quasi-linear differential equations

$$\frac{\partial \boldsymbol{\sigma}}{\partial t} + \frac{\partial \boldsymbol{a}}{\partial x} = 0 , \tag{18}$$

where the vector-valued functions are $\boldsymbol{\sigma}=(\rho, \rho u, E)^\top$ and $\boldsymbol{a}=(\rho u, p+\rho u^2, u(E+p))^\top$. For the smooth flow the system (18) is equivalent to the system of integral equations

$$\oint_C \boldsymbol{\sigma} dx - \boldsymbol{a} dt = 0 . \tag{19}$$

We consider the cell of the moving mesh depicted in Fig.2. Note that instead the variable ξ we use x. After integrating (19) along the contour C of the cell we get the system of the finite-difference equations

$$\boldsymbol{\sigma}^{n+1}_{i+1/2} h^{n+1}_{i+1/2} - \boldsymbol{\sigma}^{n}_{i+1/2} h^{n}_{i+1/2} - \boldsymbol{\sigma}^{n+1/2}_{i+1} h_{i+1} + \boldsymbol{\sigma}^{n+1/2}_{i} h_{i} + \triangle t(\boldsymbol{a}^{n+1/2}_{i+1} - \boldsymbol{a}^{n+1/2}_{i}) = 0 , \tag{20}$$

where $h^n_{i+1/2}=x^n_{i+1}-x^n_i$, $h_i=x^{n+1}_i-x^n_i$. We update the cell average values $\boldsymbol{f}^n_{i+1/2}$ to the values $\boldsymbol{f}^{n+1}_{i+1/2}$ following the two-step procedure described to the 2D scheme. At the fist stage, predictor, assuming piece-wise linear distribution of the flow parameters along x, we compute the intermediate values $\bar{\boldsymbol{f}}^{n+1}_{i+1/2}$ via (20), when instead of $\boldsymbol{f}^{n+1/2}_i, \boldsymbol{f}^{n+1/2}_{i+1}$ we utilize $\boldsymbol{f}^n_{1,i+1/2}$ and $\boldsymbol{f}^n_{2,i+1/2}$ computed via (6) with applying the monotonicity algorithm. At the second stage, corrector, after computing pre-wave values $\boldsymbol{f}^{n+1/2}_{2,i-1/2}$, $\boldsymbol{f}^{n+1/2}_{1,i+1/2}$ via (11), we solve the Riemann problem taking into account the velocity of nodes and determine the post-wave values $\boldsymbol{f}^{n+1/2}_i$. Substituting them into (20) we get $\boldsymbol{f}^{n+1}_{i+1/2}$.

3 Problem formulation for grid generation

To generate a regular adaptive-harmonic mesh we formulate the problem to minimize the harmonic (Dirichlet's) functional, written for a surface [22, 23, 8, 16]. Notations are shown in Fig.4.

Let us consider the functional defining the adaptive-harmonic grid clustered in regions of high gradients of the function $f(x, y)$

$$I = \iint \frac{((1+f_x^2)(x_\xi^2+x_\eta^2) + 2 f_x f_y (x_\xi y_\xi + x_\eta y_\eta) + (1+f_y^2)(y_\xi^2+y_\eta^2)}{(x_\xi y_\eta - x_\eta y_\xi)\sqrt{1+f_x^2+f_y^2}} d\xi d\eta. \tag{21}$$

The problem of grid adaptation (r-refinement) is formulated as follows. Let the coordinates of the grid nodes be given. The grid is formed by quadrilateral elements. The problem is to find new coordinates of the grid nodes minimizing the functional (21) values computed for a mapping of the unit square onto every grid cell. In unsteady problems such a formulation is considered

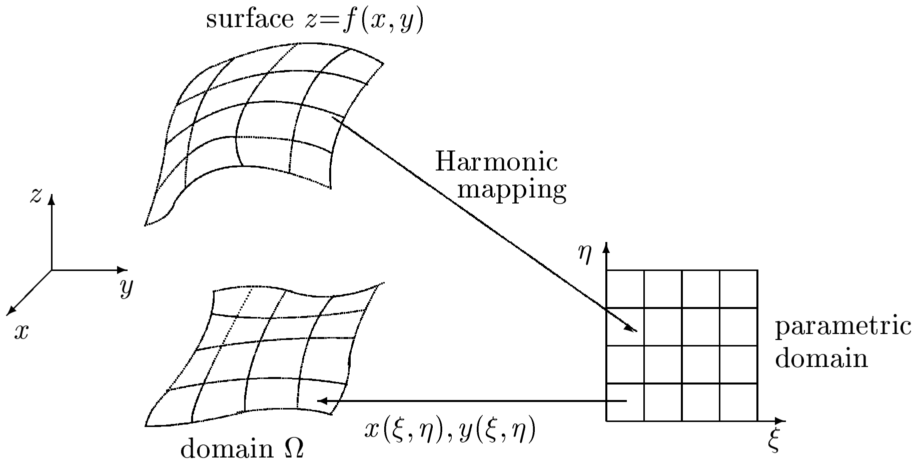

Figure 4: Harmonic coordinates on the surface of the graph of a control function $z=f(x,y)$.

at every time step and grid adaptation is performed to the control function considered as a function $f(x,y)$ at this time level.

In order to control the number of grid nodes in the layer of high gradients it has been suggested to use the function $c_a f(x,y)$ instead of $f(x,y)$ [16, 5]. In general, the larger the coefficient of adaptation c_a is, the stronger grid lines condensation in the layer of large gradients we get.

If the control function f is discontinuous then, when adapting, the cell size might approach zero in the vicinity of the discontinuity. In [3] it has been shown that the discrete functional in 1D case has infinite barrier preventing the grid cells from degenerating. In 2D case despite the infinite barrier for some c_a disappears, the iterative procedure of minimizating the functional prevents grid lines overlap. This allows for obtaining highly condensed grid lines in the vicinity of discontinuities when computing 1D and 2D flows and increasing significantly the accuracy of computations. However, in practical computation of 2D unsteady flows redundant grid lines compression leads, firstly, time step to become too small and, secondly, we have to reduce c_a in order not to leave the admissible set of convex grids (when all quadrilateral cells are convex). This, in turn, leads the mesh stops "to feel" the discontinuities of mid and low intensity and it reacts only to discontinuities of large intensity. Thus, sometimes (its mainly relates to 2D unsteady flows) it is necessary to modify the discontinious function f in such a way to get a smooth function in regions of high gradients.

There are several ways to get it. In [25, 24, 32] the monitor function is smoothed in the vicinity of discontinuity. We use another approach based on introducing an additional parameter restricting the gradient rise of the control function during adaptation. In our approach the number of grid points in the vicinity of the shock is defined by the flow solver property to smear the shocks. The second order Godunov's type scheme smears shocks within 2 to 3 cells independently on cells thickness [35]. In fact, it is not necessary to place in the shock zone more than 2 points, since it is only a discontinuity with infinitely small thickness. In contrast, for instance, to the boundary or internal layers in viscous flows where we have real physical gradients of the functions and should describe them precisely. Thus, in the hyperbolic problems the coefficient c_a plays the following role: when increasing c_a the mesh begings "to feel" the discontinuities of less intensity. Decrease of c_a results in decreasing sensitivity of the mesh to the discontinuities of greater intensity and in the limit of c_a=0 we obtain a quasiuniform harmonic mesh not depending on $f(x,y)$.

Thus, the coefficient c_a is only responsible for selecting those discontinuities of the function $f(x,y)$ to which the mesh should be adapted. Therefore, it is necessary to have one more controlling parameter determining the range of the gradients that approximates the discontinuity line in the solution and simultaneously restricts maximum gradient of the grid control function. It is accomplished as follows. First, at the mesh nodes we define the function $\tilde{f} = c_a f^h$. Then we specify D_{max}, maximal value of modulus of gradient to the control function f, for example, as D_{max}=$\lambda \max(|\nabla\tilde{f}|)$, where the coefficient λ<1 and $|\nabla\tilde{f}|$=$\sqrt{\tilde{f}_x^2+\tilde{f}_y^2}$. Next the gradient of the function is updated via

$$\nabla\tilde{f}^* = \begin{cases} D_{max}\nabla\tilde{f}/|\nabla\tilde{f}| & \text{if } |\nabla\tilde{f}| > D_{max}\ , \\ \nabla\tilde{f} & \text{otherwise}\ . \end{cases} \tag{22}$$

Then obtained values $\tilde{f}_x^*$ and $\tilde{f}_y^*$ are substituted into (21) instead of f_x and f_y.

It is not necessary to use this procedure in simulation of 1D and steady 2D flows, however, in the case of rapidly developed unsteady supersonic flows we should apply it to prevent the cells to be folded in the vicinity of the shock waves.

The question of mesh adaptation to the vector-function with components $f_i(x,y)$, i.e. using several control functions, has been considered in [1, 17].

In 1D 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 [22, 23]

$$I = \int_0^1 \frac{1}{x_\xi\sqrt{1+c_a^2 f_x^2}}d\xi. \tag{23}$$

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 $c_a f$.

4 Minimization of the functional

4.1 1D case

Using the rectangular rule the discretized functional (23) reads

$$I^h = \sum_{i=1}^{i_{max}} \frac{\triangle\xi}{(x_\xi)_{i+1/2}\sqrt{1+c_a^2(f_x)^2_{i+1/2}}}, \tag{24}$$

where i_{max} is the number of spacings, step $\triangle\xi$ is set equal 1 and the derivatives are computed via

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

To minimize (24) it is applied the Newton method

$$x_i^{p+1} = x_i^p - \tau \frac{\partial I^h}{\partial x_i} \left[\frac{\partial^2 I^h}{\partial x_i^2} \right]^{-1}, \tag{25}$$

where the iterative parameter $\tau \leq 1$, first and second derivatives are

$$\frac{\partial I^h}{\partial x_i} = \frac{-1}{(x_i - x_{i-1})^2 \sqrt{1 + c_a^2 (f_x)_{i-1/2}^2}} + \frac{1}{(x_{i+1} - x_i)^2 \sqrt{1 + c_a^2 (f_x)_{i+1/2}^2}},$$

$$\frac{\partial^2 I^h}{\partial x_i^2} = \frac{2}{(x_i - x_{i-1})^3 \sqrt{1 + c_a^2 (f_x)_{i-1/2}^2}} + \frac{2}{(x_{i+1} - x_i)^3 \sqrt{1 + c_a^2 (f_x)_{i+1/2}^2}}.$$

When deriving the above derivatives we fix metric, i.e. $(f_x)_{i+1/2}$, that allows for holding the infinite barrier for any $c_a > 0$ and prevents grid cells overlap [3].

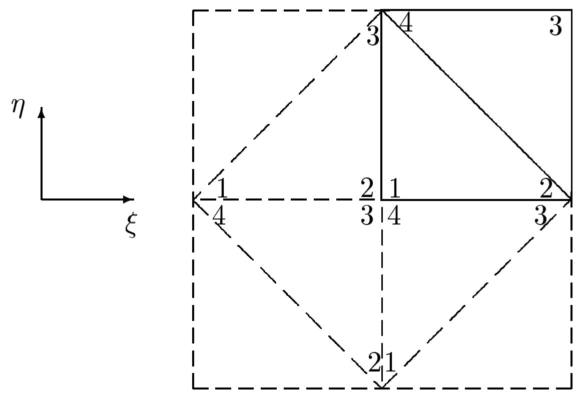

Figure 5: Computing cell is a unit square in parametric plane. Solid lines indicate the cell considered, dash lines show the adjacent cells.

4.2 2D case

Approximation of the functional (21) is performed on the mesh of quadrilaterals, every of them is a unit square in logic plane, see Fig.5

$$I^h = \sum_{i=1}^{i_{max}} \sum_{k=1}^{4} \frac{1}{4} \left[F_k \right]_i , \tag{26}$$

where i_{max} is a number of grid cells, k indicates the vertex number in a cell

$$F_k = \frac{D_1[1 + (f_x)_k^2] + D_2[1 + (f_y)_k^2] + 2D_3 (f_x)_k (f_y)_k}{J_k \left[1 + (f_x)_k^2 + (f_y)_k^2\right]^{1/2}},$$

$$D_1 = (x_{k+1} - x_k)^2 + (x_{k-1} - x_k)^2, \quad D_2 = (y_{k+1} - y_k)^2 + (y_{k-1} - y_k)^2,$$
$$D_3 = (x_{k+1} - x_k)(y_{k+1} - y_k) + (x_{k-1} - x_k)(y_{k-1} - y_k),$$
$$J_k = (x_{k+1} - x_k)(y_{k-1} - y_k) - (x_{k-1} - x_k)(y_{k+1} - y_k). \tag{27}$$

If the set of convex grids is not empty, the system of algebraic equations has at least one solution being the convex mesh (here i is a global node number)

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

Consider the method of minimizing the functional (26) assuming the grid to be convex at the p-th step of the iterative procedure. We use the quasi-Newton method in the sense that the Hessian is a diagonal matrix

$$x_i^{p+1} = x_i^p - \tau \left(R_x \frac{\partial R_y}{\partial y_i} - R_y \frac{\partial R_x}{\partial y_n} \right) \left(\frac{\partial R_x}{\partial x_i} \frac{\partial R_y}{\partial y_i} - \frac{\partial R_y}{\partial x_i} \frac{\partial R_x}{\partial y_i} \right)^{-1},$$
$$y_i^{p+1} = y_i^p - \tau \left(R_y \frac{\partial R_x}{\partial x_i} - R_x \frac{\partial R_y}{\partial x_i} \right) \left(\frac{\partial R_x}{\partial x_i} \frac{\partial R_y}{\partial y_i} - \frac{\partial R_y}{\partial x_i} \frac{\partial R_x}{\partial y_i} \right)^{-1}. \tag{28}$$

To find R_x, R_y and their derivatives consider one of four triangles, into which the cell is divided by two diagonals, e.g., with vertex 1,2 and 4, see Fig.5. Integrand (27) does not depend on rotation of the coordinates ξ, η, therefore, the formulas below hold for remaining 3 triangles. For the above triangle in (27) we set k=1, k+1=2, k−1=4 and then

$$D_1 = (x_2 - x_1)^2 + (x_4 - x_1)^2, \quad D_2 = (y_2 - y_1)^2 + (y_4 - y_1)^2,$$
$$D_3 = (x_2 - x_1)(y_2 - y_1) + (x_4 - x_1)(y_4 - y_1),$$
$$J_1 = (x_2 - x_1)(y_4 - y_1) - (x_4 - x_1)(y_2 - y_1). \tag{29}$$

We introduce notation $F_1 = U/V$, where

$$U = D_1\alpha + D_2\beta + D_3\gamma, \quad V = J_1, \quad \alpha = \frac{[1 + (f_x)_1^2]}{M_1},$$
$$\beta = \frac{[1 + (f_y)_1^2]}{M_1}, \quad \gamma = \frac{(f_x)_1 (f_y)_1}{M_1}, \quad M_1 = \sqrt{1 + (f_x)_1^2 + (f_y)_1^2}, \tag{30}$$

Further for F_1 we omit the low index and for convenience write F. Using the chain rule to find the derivatives of ratio F=U/V we get

$$F_x = \frac{U_x - FV_x}{V}, \quad F_y = \frac{U_y - FV_y}{V}, \quad F_{xx} = \frac{U_{xx} - 2F_xV_x - FV_{xx}}{V},$$
$$F_{yy} = \frac{U_{yy} - 2F_yV_y - FV_{yy}}{V}, \quad F_{xy} = F_{yx} = \frac{U_{xy} - F_xV_y - F_yV_x - FV_{xy}}{V}.$$

When differentiating we assume $f_x,\ f_y$ are not dependent on x, y. Thus, we have for vertex 1 of the triangle

$$\begin{aligned} U_x &= 2(2x_1 - x_2 - x_4)\alpha + 2(2y_1 - y_2 - y_4)\gamma, \\ U_y &= 2(2y_1 - y_2 - y_4)\beta + 2(2x_1 - x_2 - x_4)\gamma, \\ U_{xx} &= 4\alpha, \quad U_{yy} = 4\beta, \quad U_{xy} = 4\gamma, \\ V_x = y_2 - y_4, \quad V_y &= x_4 - x_2, \quad V_{xx} = V_{xy} = V_{yy} = 0, \end{aligned}$$

for vertex 2

$$\begin{gathered} U_x{=}2(x_2 - x_1)\alpha + 2(y_2 - y_1)\gamma, \quad U_y{=}2(y_2 - y_1)\beta + 2(x_2 - x_1)\gamma, \quad U_{xx}{=}2\alpha, \\ U_{yy}{=}2\beta, \quad U_{xy}{=}2\gamma, \quad V_x = y_4 - y_1, \quad V_y = x_1 - x_4, \quad V_{xx} = V_{xy} = V_{yy} = 0, \end{gathered}$$

for vertex 4

$$\begin{gathered} U_x{=}2(x_4 - x_1)\alpha + 2(y_4 - y_1)\gamma, \quad U_y{=}2(y_4 - y_1)\beta + 2(x_4 - x_1)\gamma, \quad U_{xx}{=}2\alpha, \\ U_{yy}{=}2\beta, \quad U_{xy}{=}2\gamma, \quad V_x = y_1 - y_2, \quad V_y = x_2 - x_1, \quad V_{xx} = V_{xy} = V_{yy} = 0. \end{gathered}$$

If vertex 1 of a triangle corresponds to the i-th node of global numeration, contributions of four adjacent cells are summarized

$$[R_x]_i = \sum_{l=1}^{4}[F_x]_l, \quad [R_y]_i = \sum_{l=1}^{4}[F_y]_l, \quad [R_{xx}]_i = \sum_{l=1}^{4}[F_{xx}]_l,$$

$$[R_{xy}]_i = \sum_{l=1}^{4}[F_{xy}]_l, \quad [R_{yy}]_i = \sum_{l=1}^{4}[F_{yy}]_l.$$

Here $l{=}1$ corresponds to triangle $1, 2, 4$ of the cell considered, $l{=}2$ corresponds to triangle $2, 3, 1$ of the left cell, $l{=}3$ corresponds to triangle $3, 4, 2$ of the left-down cell, $l{=}4$ corresponds to triangle $4, 1, 3$ of the down cell.

To find derivatives $(f_x)_i$ and $(f_y)_i$ we discretize the following relations

$$f_x = (y_\eta f_\xi - y_\xi f_\eta)/J, \quad f_y = (-x_\eta f_\xi + x_\xi f_\eta)/J,$$

and calculate them as average values over 4 adjacent cells to the $i-th$ node

$$[f_x]_i = \sum_{l=1}^{4}[f_x]_l \Bigg/ \sum_{l=1}^{4} J_l, \quad [f_y]_i = \sum_{l=1}^{4}[f_y]_l \Bigg/ \sum_{l=1}^{4} J_l \quad .$$

Here index l corresponds to the same triangles as above and, e.g., if $l{=}1$ then for the considered cell we have

$$[f_x]_1 = [(y_4{-}y_1)(f_2{-}f_1) - (y_2{-}y_1)(f_4{-}f_1)]\,, \ [f_y]_1 = [(x_2{-}x_1)(f_4{-}f_2) - (x_4{-}x_1)(f_2{-}f_1)]\,,$$

J_1 is computed via (29).

If a flow solver gives the values of the control function in the cell center, it is required to update them to the nodes. Since every i-th node (except boundary nodes) is surrounded by 4 cells, it can be done as

$$f_i = c_a \sum_{l=1}^{4} f_l^c J_l \Big/ \sum_{l=1}^{4} J_l \ , \tag{31}$$

where i is a global node number, f_l^c is the value in the cell center, c_a is the coefficient of adaptation which can depend on the node position, i.e., $c_a{=}c_a(x,y)$. It is convenient to normalize the control function

$$\tilde{f}_i = \frac{f_i - f_{min}}{f_{max} - f_{min}} \sqrt{(x_{max} - x_{min})^2 + (y_{max} - y_{min})^2}.$$

The above formulae can be used for generation the quasiuniform mesh in physical domain Ω by defining $f{=}const.$ and, therefore, $f_x{=}f_y{=}0$. The matter of untangling the initial prepared folded mesh has been considered in [8, 16].

4.3 Redistribution of the boundary nodes

The algorithm of redistributing the boundary nodes, consisting of using constrained minimization of the functional when constraints define the boundary of the domain, has been suggested in [3]. Such an approach allows for performance of consistent redistributing the grid nodes inside the domain Ω and on its boundary $\partial\Omega$ that increases reliability of grid generation and modeling the flow problem. In this approach we minimize the following functional

$$\tilde{I}^h = \sum_{i=1}^{i_{max}} \sum_{k=1}^{4} \frac{1}{4} \left[F_k\right]_i + \sum_{l \in \mathcal{L}} \lambda_l G_l = I^h + \sum_{l \in \mathcal{L}} \lambda_l G_l, \tag{32}$$

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. The function $G(x,y)$ is assumed piecewise differentiable.

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

$$R_x = \frac{\partial I^h}{\partial x_i} + \lambda_i \frac{\partial G_i}{\partial x_i} = 0, \qquad R_y = \frac{\partial I^h}{\partial y_i} + \lambda_i \frac{\partial G_i}{\partial y_i} = 0, \qquad G_i = 0 \ . \tag{33}$$

Here $\lambda_i{=}0$ if $i \notin \mathcal{L}$ and constraints are defined to the boundary nodes $i \in \mathcal{L}$. The method of solving the system (33) is described in [3].

4.4 Coupled algorithm

One time step to solve the 1-D or 2-D equations of gas dynamics with grid adaptation contains the following stages:

1 Generate the mesh at the next time level t^{n+1}.

2 Compute the gas dynamics values at time t^{n+1}.

3 Evaluate the derivatives $(f_x)_i$ and $(f_y)_i$ at every mesh node (in 1-D case $(f_x)_{i+1/2}$) at t^{n+1}.

4 Make one iteration step and compute the new values of $(x,y)_i$ at t^{n+1} by formulas (25) or (28).

5 Repeat starting with step 2 to convergence or within given number of iterations.

6 Compute the final gas dynamics values at t^{n+1}.

5 Numerical results

Two numerical examples are considered: a 1D example consisting of simple waves (without wave interactions) and a 2D example having complicated wave structures. In the 1D case some detailed error analysis is made.

Example 5.1 Consider 1D flow of ideal gas with initial parameters [31]

$$(\rho, u, p) = \begin{cases} (1,0,1)\ , & \text{if } 0\leq x < 0.5 \\ (0.125, 0, 0.1)\ , & \text{if } 0\leq x\leq 1\ \ , \end{cases}$$

ratio of specific heats γ=1.4.

At t>0 the wave pattern consists of the moving to the left rarefaction fan with left and right characteristics $u_{rar}^{lft}\approx -1.18322$ and $u_{rar}^{rght}\approx -0.07027$, contact discontinuity with $u_{cont}\approx 0.92745$ and moving to the right shock wave with $u_{sh}\approx 1.75216$. We compute this problem up to t=0.25 and need not define a boundary conditions at the ends x=0, 1, i.e., we solve the Cauchy problem for the 1D Euler equations.

In the first series of computations the initial grid is uniform with i_{max}=60 intervals. Since by t=0.25 the shock nearly approches the right end x=1 in the calculations with adaptation we add to the right end 10 segments more to exclude influence of the edge to the nodes being attracted into the shock zone. This has not any practical effect to the error estimation since only 3 to 7 of them enter into the control interval $0\leq x\leq 1$ when adapting. The admissible time step $\triangle t$ is calculated from the stability condition (14) and coefficient c_{cfl}=0.5 for computing on the fixed mesh and c_{cfl}=0.6 on the moving mesh. The first time step $\triangle t$ is obtained from the linearized condition (16) on the fixed mesh both for computing on the fixed and moving mesh.

The error of calculation is estimated for density using the numerical solution ρ^h and exact ρ^{ex} in the L_1 norm via the formula

$$\|Er\|_{L_1} = \int_0^1 \left|\rho^h(x) - \rho^{ex}(x)\right| dx \approx$$

$$\sum_{i=1}^{i_{max}} \sum_{j=1}^{j_{max}} \left|\rho_{i+1/2} + q_j h_{i+1/2}\rho^{(1)}_{i+1/2} - \rho^{ex}(x_{i+1/2} + q_j h_{i+1/2})\right| \cdot \triangle x_{i,j}\ ,$$

Table 1: Comparison of fixed (c_a=0) and moving grid solutions for shock tube problem. Here n is the number of time steps, τ is the iterative parameter in procedure (25).

c_a	n	τ	$\|Er\|_{L_1}$	$\|Er\|_{L_1}^{rar}$	$\|Er\|_{L_1}^{cnt}$	$\|Er\|_{L_1}^{sh}$
mesh $i_{max}=60$						
0	67	–	0.005926	0.002229	0.002256	0.001441
1	127	0.6	0.004125	0.001529	0.001723	0.000874
2	179	0.6	0.002950	0.001002	0.001340	0.000608
4	236	0.35	0.002443	0.001220	0.000975	0.000249
8	319	0.35	0.001787	0.000913	0.000725	0.000149
mesh $i_{max}=120$						
0	135	–	0.003159	0.001058	0.001339	0.000762
1	324	0.7	0.001564	0.000671	0.000740	0.000153
2	469	0.6	0.001330	0.000598	0.000576	0.000157
3	577	0.5	0.001199	0.000663	0.000471	0.000065

where every zone $(i, i+1)$ is partitioned uniformly into j_{max}=100 segments and, therefore, parameter $q_j=(j-0.5)/j_{max}-0.5$ and step $\triangle x_{i,j}=h_{i+1/2}/j_{max}$. The derivative of ρ is computed using a linear interpolation

$$\rho^{(1)}_{i+1/2} = \frac{\rho_{i+3/2}-\rho_{i+1/2}}{h_{i+3/2}+h_{i+1/2}} + \frac{\rho_{i+1/2}-\rho_{i-1/2}}{h_{i+1/2}+h_{i-1/2}} .$$

As a control function f in the functional (24) we use the density ρ being sensitive both to the shock and contact discontinuity. When adapting via (25) at every time step we perform p_{iter}=50 mesh iterations. We have found that using the indicator to stop mesh iterations at t^n

$$\max_i \left| x_i^{p+1} - x_i^p \right| < \delta ,$$

where $\delta \approx 10^{-5} - 10^{-4}$, gives a lower accuracy. To comply with the stability condition within the first 5 time steps we set p_{iter}=5 since we begin calculation on the uniform mesh. The computed density are displayed in Fig.6, from which we observe that the moving mesh solution with c_a=8 provides better resolution of all structures in the wave pattern. In Fig.6d we observe that a bit oscillations ($\approx$1.8%) being present on the fixed mesh solution disappear on the moving one. In Table 1 we present the error depending on degree of adaptation, i.e., coefficient c_a. To clarify the influence of adaptation on different waves we calculate the errors gained throughout the rarefacton wave, contact discontinuity and shock

$$\|Er\|_{L_1}^{rar} = \int_0^{x_a} \left|\rho^h - \rho^{ex}\right| dx, \quad \|Er\|_{L_1}^{cnt} = \int_{x_a}^{x_b} \left|\rho^h - \rho^{ex}\right| dx, \quad \|Er\|_{L_1}^{sh} = \int_{x_b}^{1} \left|\rho^h - \rho^{ex}\right| dx,$$

where $x_a = 0.5(u_{rar}^{rght} + u_{cont})t$, $x_b = 0.5(u_{cont} + u_{sh})t$, and they are also presented in Table 1.

We observe that the shock is smeared over two cells on the both meshes, see Fig.6d, but due to the spacings in the shock zone are reduced about by factor of 8 from initial value and due to suppressing the oscillations, the error $\|Er\|_{L_1}^{sh}$ is decreased by factor of 9.67, see the cases

c_a=0 and c_a=8 in Table 1. Decrease of $\|Er\|_{L_1}^{rar}$ by factor of 2.44 occurs because the moving mesh solution more precisely describes the weak discontinuity zones where the left and right characteristics of the rarefaction fan join the flow with constant parameters. Besides in the domain of the rarefaction wave the spacings on the moving mesh are twice smaller than on the fixed one, see Fig.6b. Thickness of the contact discontinuity is also reduced, see Fig.6c, and adaptation gives the winning in accuracy by factor of 3.11. Note that the contact zone, when treated by the second-order scheme, is smeared with time (on the uniform mesh proportionally to $t^{1/3}$ [35]) and the mesh adaptation can not prevent this process. Thus, as expected, the maximal contribution into increase of the accuracy is provided by the shock zone. And to such domains it should be paid especial attention when modeling 2D flows.

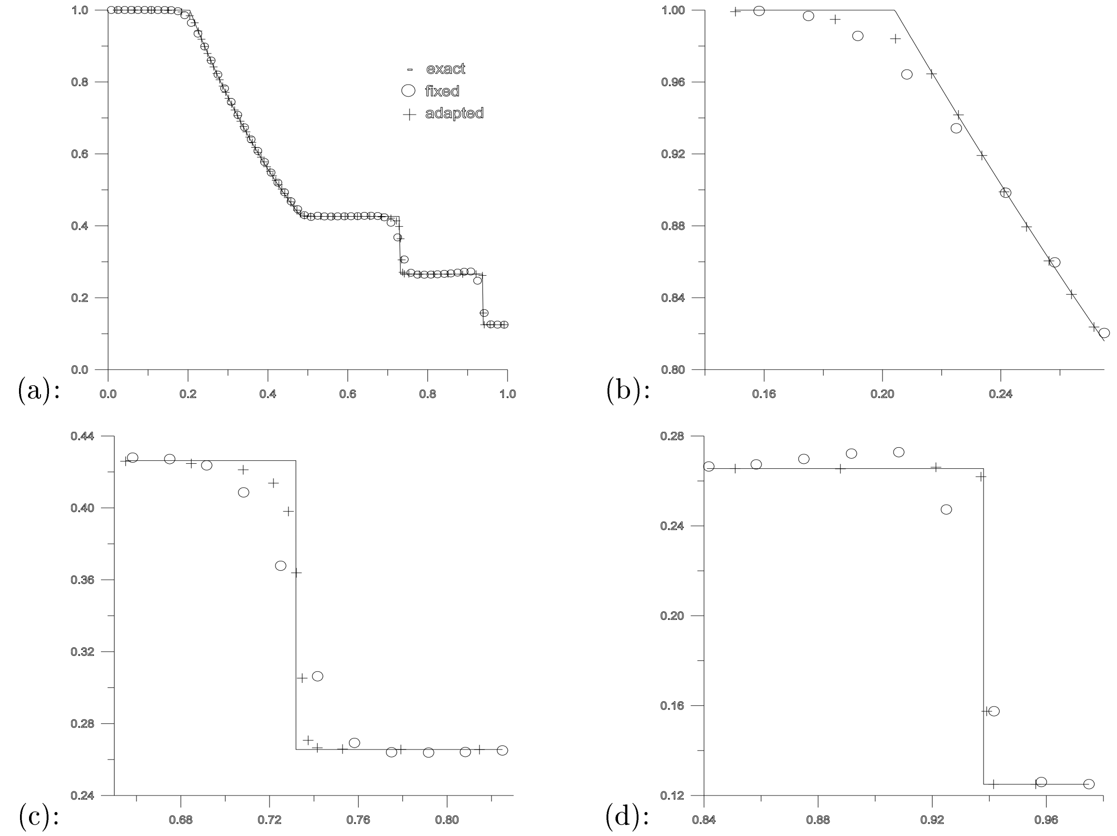

Figure 6: Example 5.1. (a): Plots of density; (b): Close-up of solution in the vicinity of rarefaction fan; (c): contact discontinuity; and (d): shock.

One can compare the above results with other moving mesh results presented in [25] where this problem was treated using the moving mesh PDE technique [5]. Different monitor functions on the moving mesh have allowed to get increase of the accuracy by factor of 1.09 to 1.81.

To compare the computer costs the number of mesh iterations at every time step p_{iter} should be multiplied by ratio of admissible time steps on the moving and fixed mesh, see Table 1. We observe that in the case of c_a=8 one have to solve the flow problem greater by factor of ≈ 250 in comparison with calculation on the fixed mesh. And this, surely, in no case can not be justified by the winning of accuracy by factor of 3.32 we get instead. But we was guided by the following reasoning performing this work with the adaptive procedure. In 1D case to get a maximal winning in accuracy without respect of running costs to see where we can get substantial improvements in the solution when treating 2D flow problems. Further it will be demonstrated that in 2D case at strong grid lines compression we need to perform only 1 or 2 mesh iterations at every time step. Nevertheless the problem of decreasing the running time is to be solved, since stability requires the time step $\triangle t$ to be small enough and in 2D cases the reduce of $\triangle t$ can be by factor of 100 and larger.

With purpose to derease the computer costs we have performed the following numerical experiment. Up to time t_1 (we varied t_1 from 0.15 to 0.22) we first performed at every time step p_{iter}=5 mesh iterations, and then within interval $(t_1, 0.25)$ we defined p_{iter}=50. Increase of p_{iter} from 5 to 50 can be executed gradually within several time steps. Results turned out unsatisfactory. In all cases the error was increased even in comparison with the fixed mesh solution.

In the second series of computations the grid consists of i_{max}=120 intervals. Results are presented in Table 1. We get the winning in accuracy by factor of 2.63. As in the preceding case the maximal rise by factor of 11.72 is obtained to the shock zone and minimal by factor of 1.60 to the rarefaction fan domain.

Example 5.2 As 2D example we consider a spherical explosion between two parallel walls at z=0 and z=1 [21]. Initially the gas is at rest with parameters $(p,\rho)_{out}$=$(1,1)$ evewhere except in a sphere centered at $(0,0,0.4)$ with radius 0.2. Inside the sphere $(p,\rho)_{in}$=$(5,1)$, γ=1.4. The initial jump in pressure results in an outward moving shock and a contact discontinuity and an inward moving rarefaction wave. There occurs interactions between these waves and between waves and the walls. The flow by t=0.7 consists of several shocks and strong contact/(tangential) discontinuity surrounding the low density region near the center.

First, until the initial outward shock reaches the wall z=0, the solution is spherically symmetric. Further, it remains cylindrically symmetric and we use the 2D axisymmetric formulation of the problem, see system (1) where we use the variable z instead x and x instead y and parameter ν=1. In half-plane ϕ=$const.$ (where ϕ is an angle in cylindrical coordinates) we compute the domain $(x,z) \in [0,1.4] \times [0,1]$ and on the axis x=0 define the symmetric boundary conditions. On the walls z=0, 1 we use the condition of reflection and at line x=1.4 undisturbed values $(u,v,p,\rho)_{out}$ are defined. The fixed mesh is rectangular, in all computations the coefficient c_{cfl} is 0.6. The pressure contours at t=0.7 computed on the fixed mesh 100×140 are presented in Fig.7.

We perform adaptation by minimizing the discrete functional (26). On the walls and axis of symmetry we apply constrained minimization solving the system (33). The adaptive procedure is switched on beginning from t=0.45. Since strong grid lines compression results in decreasing the admissible time step (value of $\triangle t$ falls proportionally to the increase of the cell aspect ratio) we use the following strategy. Main part of the time interval we compute on not very condensed mesh and at some moment t'=$0.7-\delta t$ we attain a maximum compression of grid lines in the

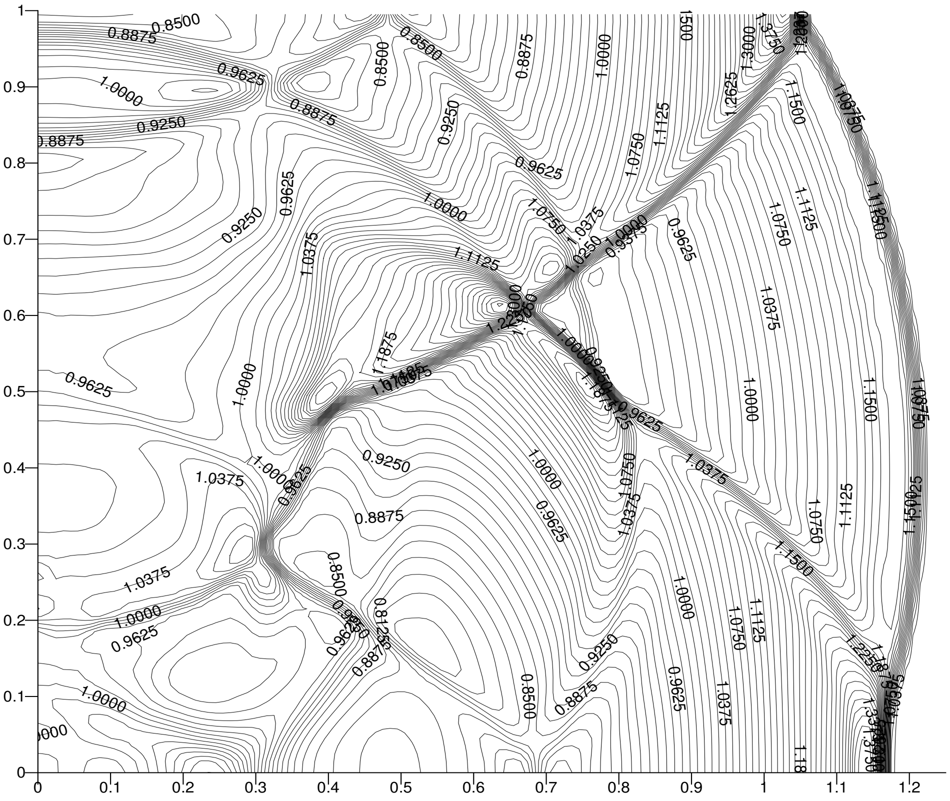

Figure 7: Pressure contours from 0.775 to 1.5 with increment 0.0125 in the $x-z$ plane at t=0.7 computed on the rectangular mesh 100×140.

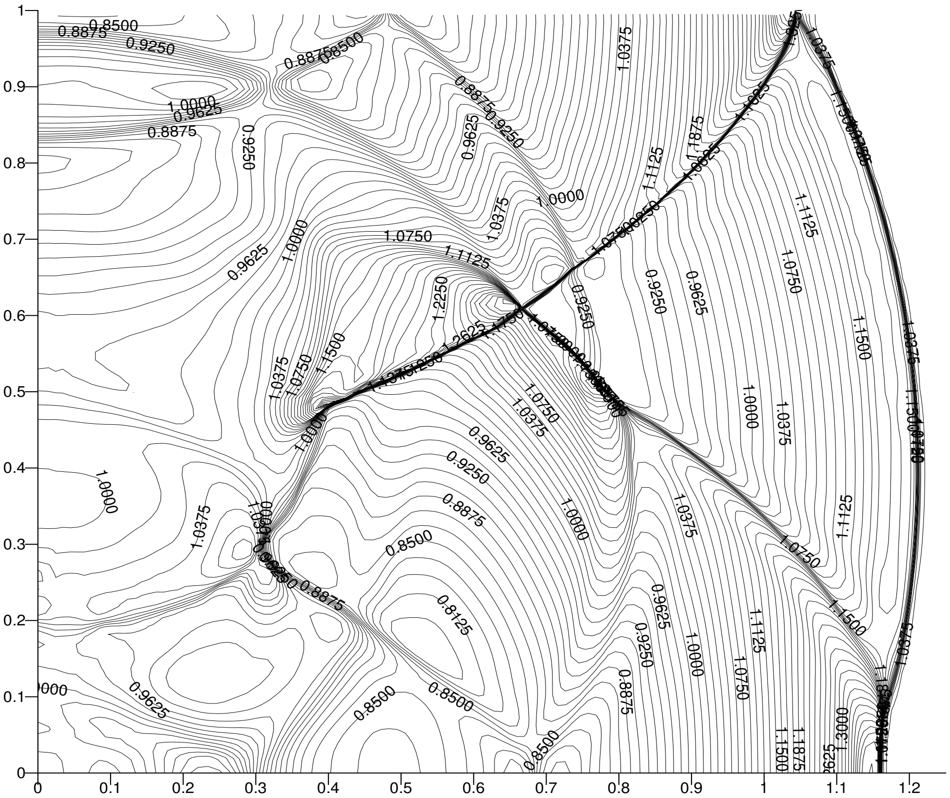

Figure 8: Pressure contours computed on the adapted mesh 100×140.

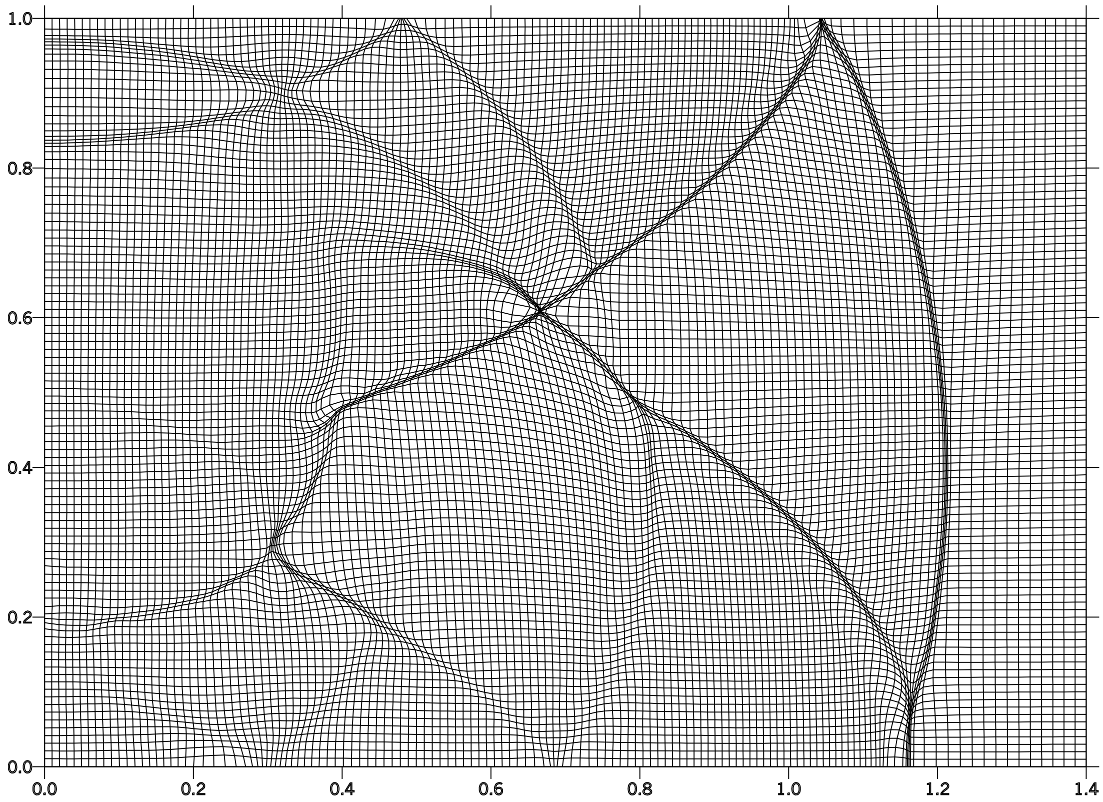

Figure 9: Adapted mesh 100×140, p is a control function, c_a=0.21, D_{max}=5.5 .

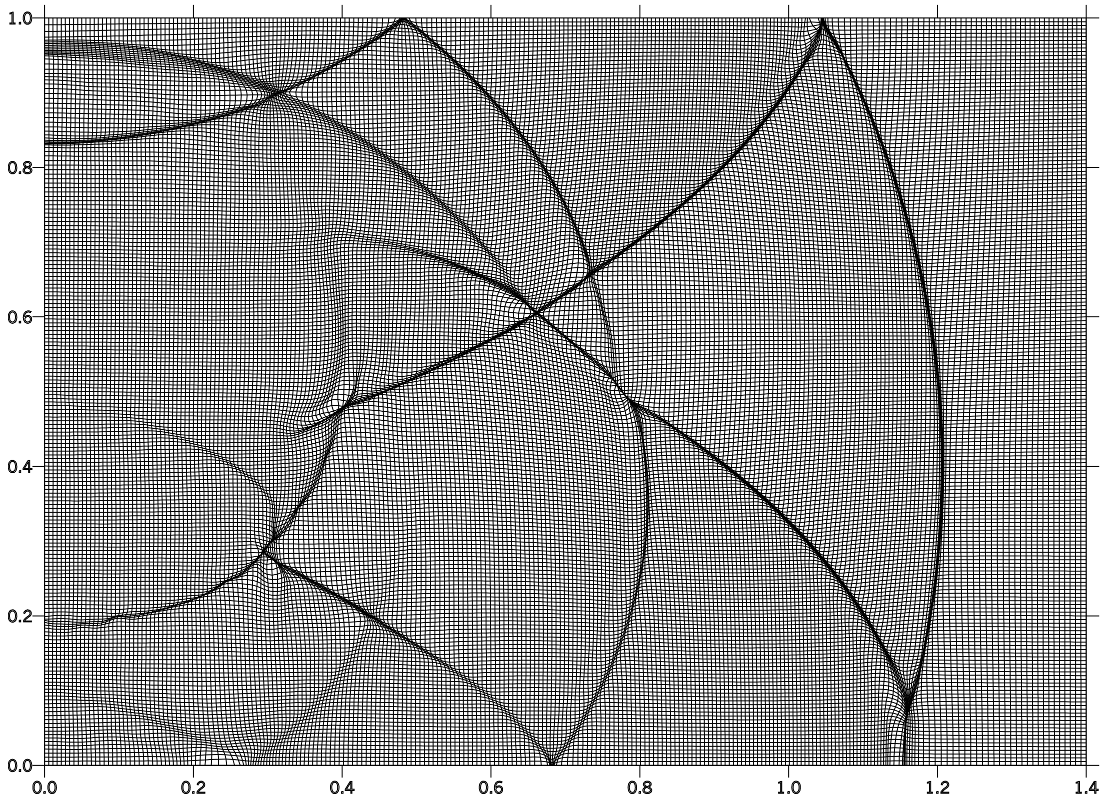

Figure 10: Adapted mesh 200×250, p is a control function, c_a=0.18, D_{max}=5 .

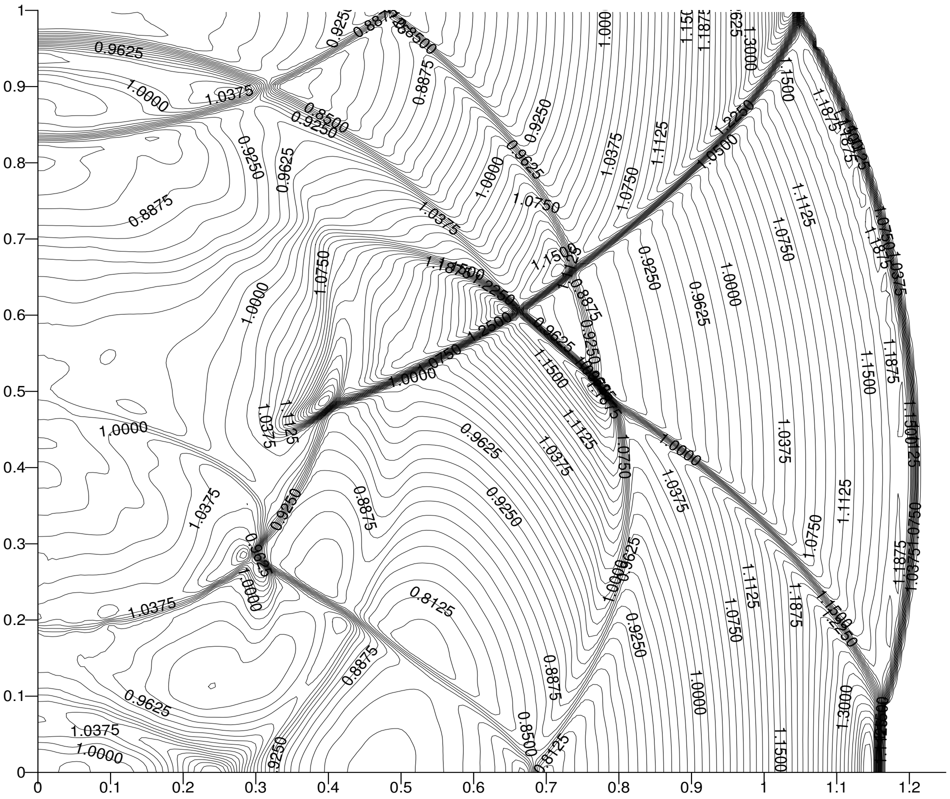

Figure 11: Pressure contours computed on the rectangular mesh 200×250.

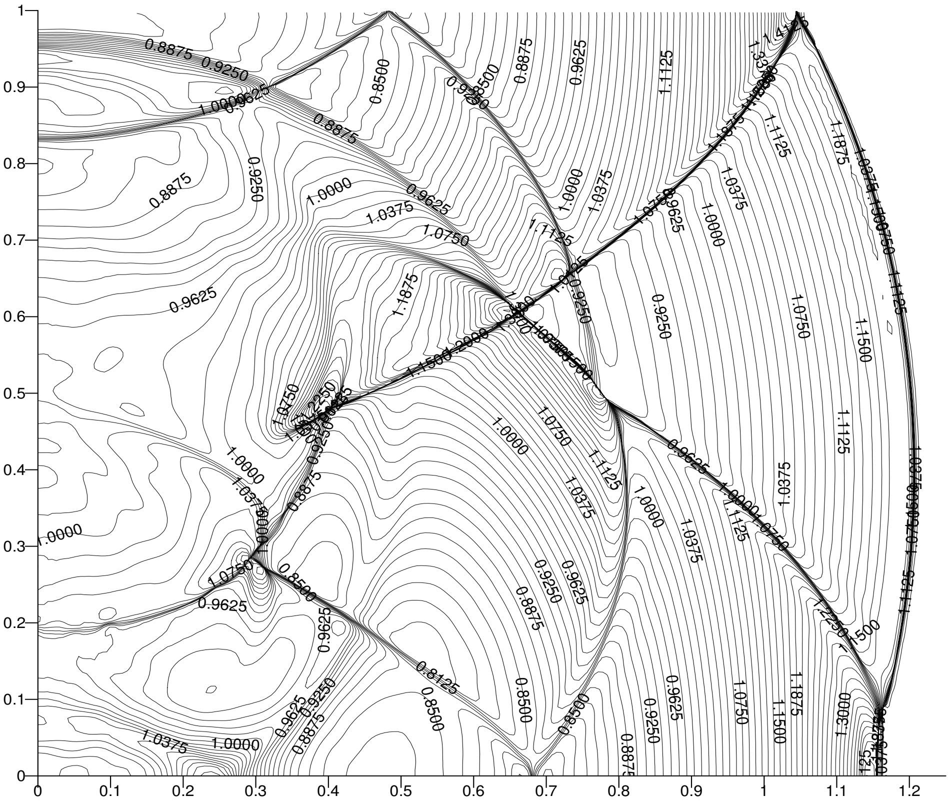

Figure 12: Pressure contours computed on the adapted mesh 200×250.

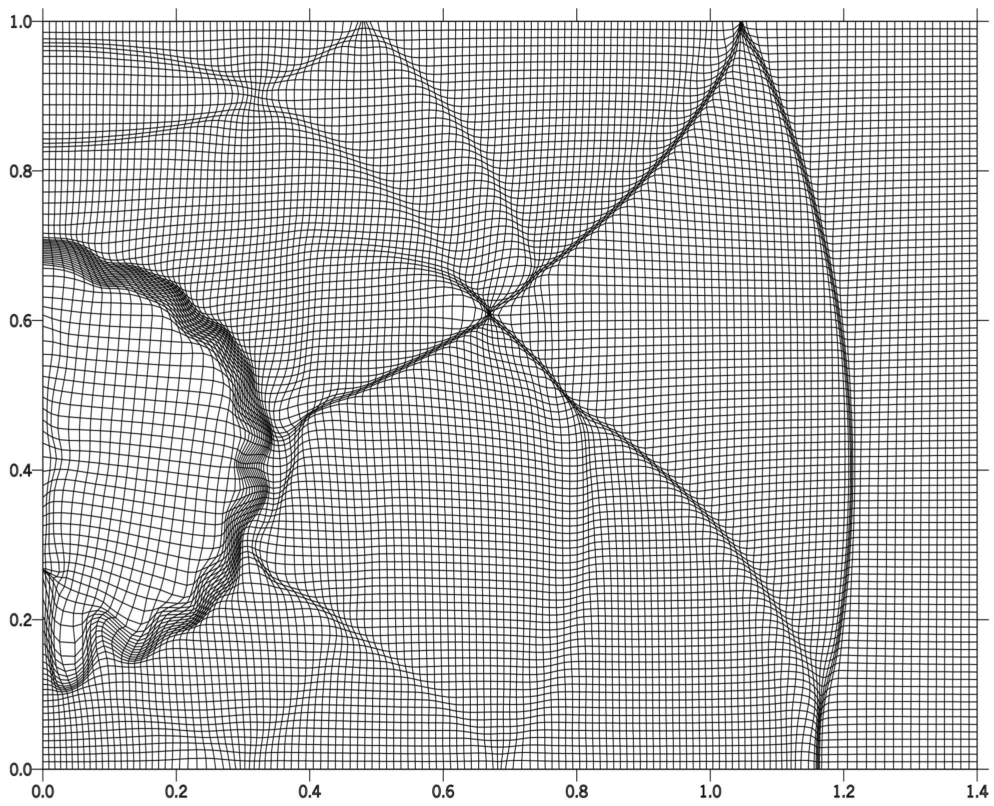

Figure 13: Adapted mesh 100×140, ρ is a control function, c_a=0.4, D_{max}=6 .

vicinity of discontinuities. Following this methodology within the time interval $(0.45, 0.65)$ we set rather a small coefficient of adaptation c_a=0.07 to 0.1. Here it is sufficient to perform 3 mesh iterations at every time step. The iterative parameter τ in quasi-Newton procedure (28) is 0.9. The admissible time step $\triangle t$ is decreased by 5 to 10 times in comparison with $\triangle t$ on the rectangular mesh.

Further, within the final interval $(0.65, 0.7)$, the coefficient c_a is increased up to 0.18 to 0.4 and then time step falls down to 0.05 to 0.005 of $\triangle t$ on the rectangular mesh. It is not a constant value during computation because the mesh constantly "breaths" and accordingly the value of $\triangle t$ periodically first decreases then increases, etc.. At this stage it is sufficient to perform 1 to 2 mesh iterations at every time step. The maximal gradient of the control function in (22) is set D_{max}=5 to 6. Note that the larger parameter D_{max} we can attain the thinner smeared shock we get and, therefore, the higher accuracy we obtain. In the steady flows we do not use this paprameter at all [3] and in some cases of unsteady flow with simple shocks strucure [17] the parameter D_{max} is about 100. However, in the present modeling we have complicated structure of many interacting shocks and should not define very large D_{max}.

In the first series of computations with adaptation the pressure p is used as a control function f. In Figs.8,9 the pressure contours and adapted mesh are shown, respectively. We see that the shocks thickness is decreased significantly in comparison with the fixed mesh solution presented in Fig.7. However the grid lines can be strongly condensed to the outer shock and two more

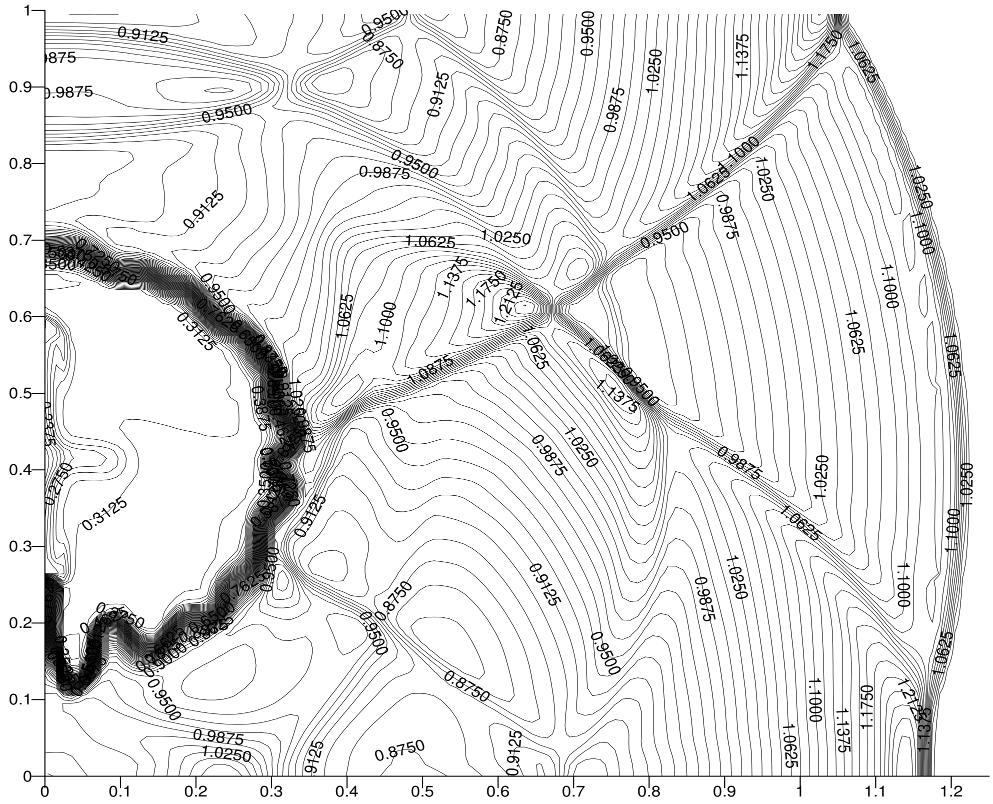

Figure 14: Density contours from 0.2 to 1.3 with increment 0.0125 computed on the rectangular mesh 100×140.

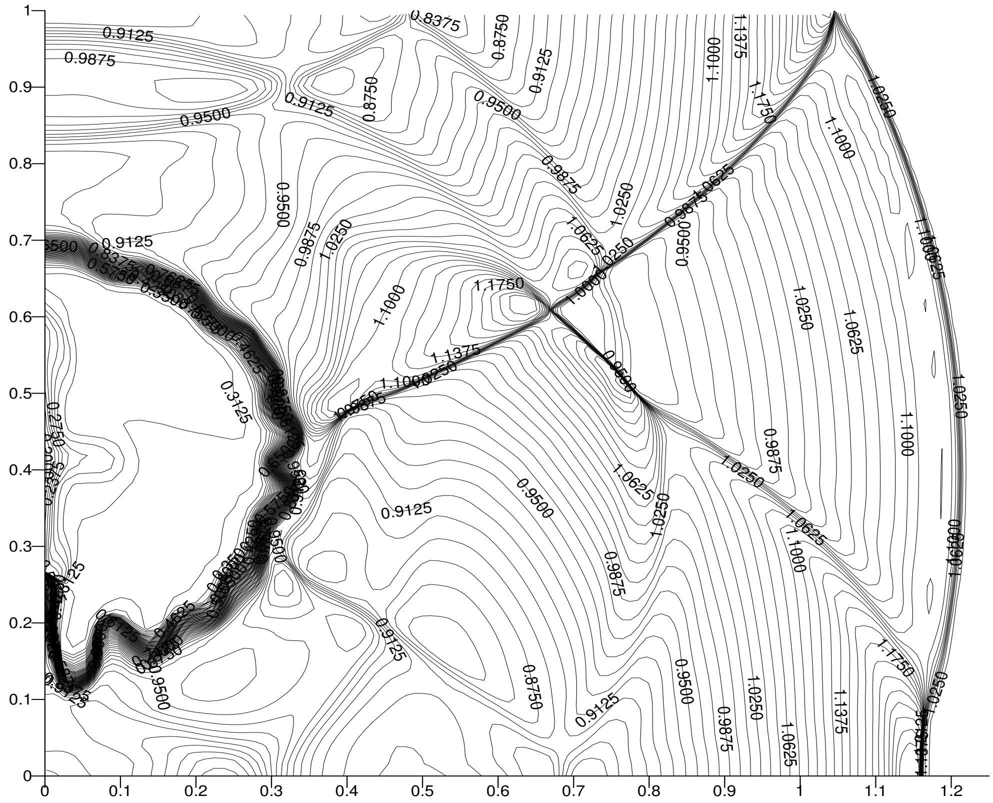

Figure 15: Density contours computed on the adapted mesh 100×140.

ones adjoining to it. This is due to insufficient number of grid points. In the next calculations we use the mesh 200×250, see Fig.10. Here both the intensive shocks and compression waves are depicted by the condensed grid lines rather fine. Pressure contours, computed on the rectangular and adapted meshes, are presented in Figs.11,12.

Since p is continuous across the tangential discontinuity the mesh does not feel it. In the next calculations the density ρ it is used as a control function f. Here the mesh 100×140 depicts the tangential discontinuity very clear, see Fig.13. Density contours computed on the quasiuniform and adaptive meshes are presented in Figs.14,15. Comparing the grids with the control function f=p, Fig.9, and f=ρ, Fig.13, one can note that in spite of the fact that in the second case c_a is twice larger, in the first case the shocks are indicated by the grid lines a bit more clear. This occurs due to influence of the contact discontinuity on the mesh in the second case.

6 Concluding remarks

Results of computations presented in this report show that the method of adaptive-harmonic grid generation can be successfully applied to unsteady problems of gas dynamics. Grid is clustered in such a way that we can obtain quite a satisfactory resolution of the shocks with not a great number of grid nodes.

As in the case of a uniform mesh, here the shock wave is smeared within 2 or 3 cells and the mesh with strong clustering looks like a set of blocks with the boundaries defined automatically as the lines of grid nodes clustering. Inside every "block" the mesh is quasiuniform and the solution is smooth. 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 (1), we get in the limit the Rankine-Hugoniot conditions across the shock [28, 13]. 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. In real world 2D computations 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.

This approach may be considered as a further development of the moving block technology presented in the monograph [13], where the flow domain is devided into blocks with boundaries being the shock waves or contact discontinuities. Advantage of the moving adaptive grid technology is that such a "cutting" is performed automaticaly. In real-world computations we can obtain such a complex nonstationary configuration of the blocks that even if we would be able to capture their boundaries "by hand", we will have serious problems on joining those blocks.

At a first glance, it seems that for unsteady problems the grid adaptation increases significantly the total time of computation, in our 2D calculations approximately by factor of 10 to 20, due to additional updating the flow parameters after every mesh iteration and reduction of the admissible time step. A detail analysis removes this impression. Suppose we want to reduce the thickness of shock wave smearing by a factor of 20. With the moving mesh method we attain such a cell aspect ratio in the shock zone. However, with the rectangular mesh we need to increase number of grid points by 20 times in not only $x-$ and y-direction but also the time direction, since the time step falls as well. As a result we need computer memory about 400 times larger and the running time will increase by a factor of 20^3=8000. With the moving mesh approach, the computer cost is approximately the number of iterations at each time step multiplying by the ratio of the admissible time steps. So in this case the running time increases by a factor of $20m$ (in reality by less factor since we perform calculations with smallest $\triangle t$ only within a part of time t), where m is the number of iterations at each time step which is about 2 to 3 in the 2D computations. This comparison demonstrates the great saving in computer memory and the gaining in computer time for the use of the proposed moving mesh methods. The advantage of the adaptive procedure seems more impressive in 3D calculations.

References

[1] 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.

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

[3] B.N. Azarenok, Variational barrier method of adaptive grid generation in hyperbolic problems of gas dynamics, *SIAM J. Numer. Anal.*, 40 (2002), pp. 651-682. (Also available online from http://www.math.ntnu.no/conservation/2001/042.html)

[4] M.Ben-Artzi and J.Falcovitcz, A second order Godunov-type scheme for compressible fluid dynamics, J. Comput. Phys. 55(1984), pp. 1–32.

[5] W.M. Cao, W.Z. Huang and R.D. Russel, An r-adaptive finite element method based upon moving mesh PDEs, J. Comp. Phys. 149(1999), pp. 221–244.

[6] H.D. Ceniceros and T.Y. How, An efficient dynamically adaptive mesh for potentially singular solutions, J. Comput. Phys., 172(2001), pp. 609–639.

[7] 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.

[8] A.A. Charakhch'yan and S.A. Ivanenko, A variational form of the Winslow grid generator, J. Comput. Phys. 136(1997), pp. 385–398.

[9] P. Colella and P.R. Woodward, The piecewise parabolic method (PMM) for gas-dynamical simulations, J.Comput.Phys., 54 (1984), pp. 174-201.

[10] R.Courant and K.O. Friedrichs, Supersonic Flow and Shock Waves, Springer-Verlag, 1985.

[11] J.K. Dukowicz, Efficient volume computation for three-dimensional hexahedral cells, J. Comput. Phys., 74(1988), pp. 493–496.

[12] S.K. Godunov, A finite difference method for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics. Mat. Sb., 47 (1959), pp. 271–306, (in Russian).

[13] S.K. Godunov (Ed.), et al., Numerical Solution of Multi-Dimensional Problems in Gas Dynamics, *Nauka press, Moscow*, (1976), (in Russian).

[14] C.L. Ingram, D.S. McRay. Extension of a dynamic solution - adaptive grid algorithm and solver to general structured milti-block configurations, AIAA 96-0294, AIAA 34th Aerospace Science Meeting, Reno, NV, Jan. 1996.

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

[16] S.A. Ivanenko , Harmonic mappings, *Chapt. 8 in: Handbook of Grid Generation*, J.F. Thompson et al eds., CRC Press, Boca Raton, Fl, (1999).

[17] S.A. Ivanenko and B.N. Azarenok, 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 (Also available online from http://www.math.ntnu.no/conservation/2001/043.html).

[18] P. Knupp, L. Margolin and M. Shashkov, Reference Jacobian optimization-based rezone strategies for arbitrary Lagrangian Eulerian methods, J. Comput. Phys., 176(2002), pp. 93–128.

[19] V.P. Kolgan, Application of the pinciple of minimal derivatives for constructing finite difference schemes in computations of discontinuous solutions in gas dynamics, Scientific Notes of Central Aerohydrodynamical Institute, Moscow, 3 (1972), N 6, pp. 68–77, (in Russian).

[20] A.G. Kulikovskii, N.V. Pogorelov and A.Yu. Semenov, Mathematical Aspects of Numerical Solution of Hyperbolic Systems, Monographs and Surveys in Pure and Applied Mathematics **188**, Chapman and Hall/CRC, Boca Raton, FL, 2001.

[21] J.O. Langseth and R.J. LeVeque, A wave propagation method for 3D hyperbolic conservation laws, J. Comp. Phys. 165(2000), pp. 126–166.

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

[23] V.D.Liseikin, Grid Generation Methods, Springer-Verlag: New York, 1999.

[24] 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.

[25] 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.

[26] D.S. McRay and Kelly R. Laflin, Dynamic Grid Adaptation and Grid Quality, Chapt. 34 in Handbook of Grid Generation. (CRC Press, Boca Raton, Fl, 1999).

[27] A.V. Rodionov, Increase of accuracy of the Godunov's scheme, USSR Computational Math. and Math. Phys., 27 (1987), No. 12, pp. 1853–1860.

[28] 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.

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

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

[31] G.A. Sod, A survey of finite difference methods for the systems of nonlinear hyperbolic conservation laws, J. Comp. Phys. 27(1978), pp. 1–31.

[32] H. Tang and T. Tang, Moving mesh methods for one- and two-dimensional hyperbolic conservation laws, SIAM J. Numer. Anal., to appear.

[33] J.F.Thompson. A survey of dynamically-adaptive grids in the numerical solution of partial differential equations, Appl. Numer. Maths, 1985, Vol.1, 3-27.

[34] O.V. Ushakova, Conditions of nondegeneracy of three-dimensional cells. A formula of a volume of cells. SIAM J. Sci. Comput., 23(2001), pp. 1273–1289.

[35] E.V. Vorozhtsov and N.N. Yanenko, Methods for Localization of Singularities in Numerical Solution of Gasdynamics Problems, Springer series in Computational Physics, 1990.

[36] P.A.Zegeling. Moving Grid Techniques, Chapt. 37 in Handbook of Grid Generation. (CRC Press, Boca Raton, Fl, 1999).