

# Use of Dynamically Adaptive Grids for Modeling Three-Dimensional Unsteady Gas Flows with High Gradients

D. V. Rudenko\* and S. V. Utyuzhnikov\*\*

\* Department of Computational Mathematics, Moscow Institute of Physics and Technology,  
 Dolgoprudnyi, Moscow oblast, 141700 Russia

\*\* UMIST, Department of Mechanical, Aerospace & Manufacturing Engineering,  
 PO Box 88, Manchester, M60 1 QD UK

e-mail: drudenko@3ka.mipt.ru, s.utyuzhnikov@umist.ac.uk

Received September 19, 1999

**Abstract**—An algorithm for construction of dynamically adaptive 3D grids is described. Conditions for determining the time step and restrictions on grid transformations are given for the case where Rodionov's UNO scheme is used for solving a system of the Euler equations on a moving curvilinear grid. The efficiency of using the dynamically adaptive grids is demonstrated by solving the problem on gas-dynamics consequences of a cosmic body explosion in the Earth atmosphere.

## 1. INTRODUCTION

The use of computational grids adapted to solutions is efficient when modeling gas flows with high gradients (boundary layers, shock waves, contact discontinuities, and the like). Even in the case of a one-dimensional problem, adaptive grids make it possible to considerably reduce computational expenditures.

The use of adaptive grids is a considerably more complicated task when the zone of high gradients moves in space. In this case, two fundamentally different approaches can be used. The first approach suggests that the computational grid is fixed for one or several time steps (see, e.g., [1, 2]). When the grid is changed, the solution is interpolated anew for the new grid. In the second approach, new spatial variables are introduced in such a way that there are no high gradients in the computational region [3–6]. The transformation of old variables  $x$  in the physical space to new variables  $\xi$  depends on time and satisfies certain restrictions. The problem is solved on a uniform grid in terms of the variables  $\xi(x, t)$ . In this case, the solution is not reinterpolated from one grid to another; however, certain restrictions on the transformation  $\xi(x, t)$  are to be imposed. This approach was usually used in the one-dimensional case and/or for computation of steady, or almost steady, flows.

In the work [7], a two-dimensional algorithm for construction of dynamically adaptive grids in the framework of the second approach is discussed. In this paper, the algorithm described in [7] is generalized for the three-dimensional case. Restrictions imposed on the time and grid steps when using adaptive grids for solving nonstationary problems are examined. The applicability of the algorithm suggested and its efficiency for solving complex three-dimensional problems are illustrated by solving the problem of modeling the gas-dynamics consequences of the explosion of a cosmic body in the Earth atmosphere.

## 2. GRID CONSTRUCTION

As was noted in the Introduction, when solving a problem on a nonstationary grid, the system of governing equations is rewritten in terms of new variables  $\xi(x, t)$ , which are selected from the conditions that the desired function has no regions of high gradients and the adaptive grid is orthogonal and sufficiently smooth. Thus, the problem is to find a transformation of independent variables for which the computational grid satisfies the above requirements.

In this paper, we use the variational approach [8] for construction of the computational grid. The desired transformation possessing the required properties is determined by minimizing certain functionals corresponding to these properties.

In order to construct a grid adapted to the solution, a distribution of the computational nodes that minimizes the functional

$$I_w = \iiint W^2 g^{1/2} dx dy dz = \iiint W^2 g d\xi d\eta d\zeta \quad (2.1)$$



is found. Here,  $W^2 = \sqrt{1 + \alpha |\nabla f(\xi, \eta, \zeta)|^2}$  is a weight function,  $g = \det \|g_{ij}\|$ ,  $g_{ij} = (\mathbf{r}_{\xi^i}, \mathbf{r}_{\xi^j})$ , and  $g^{ij} = (\nabla \xi^i, \nabla \xi^j)$ .

In the expression for the weight function  $W^2$ ,  $f$  is an unknown function to be determined from the solution of the gas-dynamics problem; it may be pressure, density, temperature, etc., or a weighted sum of these, depending on a particular problem under consideration.

As can be seen from (2.1), the transformation that minimizes  $I_W$  redistributes the nodes of the grid from the regions with lesser values of the weight function to those with greater values of  $W^2$ . If the weight function is constant in the entire computational region, the distribution of the grid nodes remains uniform. In the numerical implementation of the algorithm suggested in this paper, the computational grid remains close to uniform if variations of the weight function are less than 1% of its average value in the region. Thus, by means of a parameter  $\alpha$ , the regions where considerable grid adaptation takes place can be confined to those where the solution gradients are greater than a certain value. Based on these considerations, this parameter was chosen as  $\alpha = \Lambda (\max |\nabla f(\xi, \eta, \zeta)|)^2$ , where  $0.1 < \Lambda < 10$ .

The grid smoothness and orthogonality are characterized by  $g^{ii}$  and  $g^{ij}$  ( $i \neq j$ ), respectively. The corresponding functionals have the following form (( $i, j, k$ ) are cyclic):

$$I_S = \iiint \sum g^{ii} dx dy dz = \iiint \left( \sum_{i=1}^3 (g_{jj}g_{kk} - g_{jk}^2) g^{-0.5} \right) d\xi d\eta d\zeta, \quad (2.2)$$

$$I_O = \iiint g^{3/2} \sum_{i=1}^3 (g^{jk})^2 dx dy dz = \iiint \sum_{j=1}^3 (g_{ij}g_{ik} - g_{ii}g_{jk})^2 d\xi d\eta d\zeta. \quad (2.3)$$

The minimization of the linear combination

$$I = I_S + \lambda_O I_O + \lambda_W I_W \quad (2.4)$$

of functionals (2.1)–(2.3) gives rise to a computational grid satisfying all three required conditions.

The grid is found from the solution of the system of Lagrange–Euler equations written for functional (2.4):

$$I = \iiint F d\xi d\eta d\zeta.$$

In terms of the variables  $\xi(\mathbf{x}, t)$ , the system has the form

$$\sum_{j=1}^3 \frac{\partial}{\partial \xi^j} \frac{\partial F}{\partial (x_i)_{\xi^j}} - \frac{\partial F}{\partial x_i} = 0, \quad i = 1, 3.$$

Applying simple transformations, we get the following system of equations in the grid node coordinates:

$$\sum_{j=1}^3 A_{jj} \mathbf{r}_{\xi^j \xi^j} = \mathbf{b}, \quad (2.5)$$

where

$$\mathbf{b} = F_W \nabla W - \left( \sum_{j=1}^3 \sum_{k=1, k \neq j}^3 A_{jk} \mathbf{r}_{\xi^j \xi^k} + \sum_{j=1}^3 \sum_{k=1}^3 A'_{jk} W_{\xi^j} \mathbf{r}_{\xi^k} + \sum_{j=1}^3 \sum_{k=1}^3 \sum_{m=1}^3 \sum_{n=1}^3 A_{jkmn} (\mathbf{r}_{\xi^j \xi^m} \mathbf{r}_{\xi^n}) \mathbf{r}_{\xi^k} \right),$$

$$A_{jk} = \frac{\partial F}{\partial g_{jk}} + \frac{\partial F}{\partial g_{kj}}, \quad A'_{jk} = \frac{\partial F_W}{\partial g_{jk}} + \frac{\partial F_W}{\partial g_{kj}}, \quad A_{jkmn} = \frac{\partial A_{jk}}{\partial g_{mn}} + \frac{\partial A_{jk}}{\partial g_{nm}}, \quad F_W = \frac{\partial F}{\partial W}.$$

The system is solved in the cubic region  $[0, 1] \times [0, 1] \times [0, 1]$ . For boundary conditions on one of the boundaries, we use a fixed node distribution, which is taken from the previous time layer and corrected in accordance with the condition of orthogonality of the coordinate lines to the region boundary. On the other boundaries, the condition of orthogonality of the coordinate lines to the boundary of the computation region is imposed:  $\mathbf{r}_{\xi^i} - \mathbf{n}_i (\mathbf{r}_{\xi^i}, \mathbf{n}_i) = 0$ , where  $\mathbf{n}_i$  is the normal to the  $i$ th boundary.

Since we solve a nonstationary problem, the grid is recalculated in a specified number of steps, which is determined by a particular problem.

## 3. GRID GENERATOR

Since the Lagrange–Euler equations are elliptic, solution of the system of equations for the grid generation is a rather difficult task. The finite-difference analog of the Lagrange–Euler equations is constructed as follows. First, the left-hand side of (2.5) is linearized. The derivatives are approximated by the second-order differences. As a result, we get

$$L(\mathbf{r}) = \mathbf{b}. \quad (3.1)$$

In this equation,

$$L(\mathbf{r}^{(n)}) = \sum_{p=1}^3 A_{pp} L_{pp} \mathbf{r}^{(n)}$$

is the difference operator of the linearized system (2.5),  $\mathbf{b}$  is the right-hand side of (2.5), and  $L_{pp}$  is a one-dimensional difference Laplace operator with respect to the coordinate  $\xi_p$ .

To speed-up the convergence of the solution to (3.1), we use the method of a spectrum-equivalent operator [9]. Since the system is elliptic, the differential Laplace operator is used for a preconditioner. Thus, the linear iterations have the form

$$B(\mathbf{r}^{(n+1)}) = B(\mathbf{r}^{(n)}) - \frac{1}{\lambda_{\max}} [L(\mathbf{r}^{(n)}) - \mathbf{b}], \quad (3.2)$$

where

$$B(\mathbf{r}^{(n)}) = \sum_{p=1}^3 \frac{\theta_p}{\lambda_p} L_{pp} \mathbf{r}^{(n)}$$

is the preconditioning operator,  $\lambda_p$  is the maximum eigenvalue of  $L_{pp}$ ,  $\lambda_{\max}$  is the maximum eigenvalue of  $L$  given by the approximate formula

$$\lambda_{\max} = \max_{i,j,k} \left\{ \sum_{i_0=-1}^1 \sum_{j_0=-1}^1 \sum_{k_0=-1}^1 |a_{i+i_0, j+j_0, k+k_0}| \right\};$$

and  $a_{i,j,k}$  are coefficients of the difference operator  $L$  at the node  $i, j, k$ . The value of the parameter  $\theta_p$  is chosen from the interval  $(0, 1]$  to make spectra of the operator  $B$  and  $L/\lambda_{\max}$  match each other. Note that, by accurately choosing this coefficient, we can considerably speed-up the convergence; however, even without this, the algorithm works efficiently.

On each step of the linear transformations, the solution to equation (3.2) is found by inverting the operator  $B$  by means of the fast Fourier transform along two directions and the sweep algorithm (known as the Thomas algorithm or “progonka”) along the third one.

The use of such a combined algorithm for solving equation (3.2) makes it possible to construct an adaptive computational grid considerably faster compared to solving immediately (3.1) by means of an ordinary (e.g., Gauss–Seidel) iteration procedure. Usually, it takes 5–10 iterations to construct the grid. The necessity to carry out iterations on nonlinearity appears usually only when constructing a very fine initial grid. In this case, the right-hand side of equation (3.2) is recalculated with regard to the solution obtained by means of the linear iterations, and, then, the linear iteration process described above is repeated. The time required to renew the adaptive grid is 3–4 times less than that required for computations on one time step.

## 4. SYSTEM OF GAS-DYNAMICS EQUATIONS

We used the above-described algorithm of grid construction for solving a system of the Euler equations. Let us write the Euler equations in the strictly conservative form in terms of the Cartesian coordinates  $\{t, x^i\}$ ,  $i = 1, 3$  (here, and in what follows, repeated indexes mean summation):

$$\mathbf{U}_t + \mathbf{F}^i x^i = \mathbf{Q}, \quad (4.1)$$

where

$$\mathbf{U} = (\rho, \rho u_1, \rho u_2, \rho u_3, E)^T, \quad \mathbf{F}^i = (\rho u_i, \rho u_1 u_i + P \delta_{1i}, \rho u_2 u_i + P \delta_{2i}, \rho u_3 u_i + P \delta_{3i}, (E + P) u_i)^T,$$



$$\mathbf{Q} = (0, \rho g_1, \rho g_2, \rho g_3, \rho \mathbf{g} \mathbf{u})^T, \quad E = \rho(e + 0.5 \mathbf{u}^2), \quad P = P(e, \rho), \quad \delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases}$$

$\rho$  is the gas density,  $\mathbf{u} = (u_1, u_2, u_3)^T$  is the gas velocity in the Cartesian coordinates,  $e$  is the gas specific internal energy per unit volume,  $P$  is the pressure, and  $\mathbf{g} = (g_1, g_2, g_3)^T$  is the acceleration of gravity.

Under the transformation to an arbitrary coordinate system,  $\{t, x^i\} \rightarrow \{\tau, \xi^j\}$ ,  $t = \tau$ ,  $\xi^j = \xi^j(t, x^i)$ ,  $j = 1, 3$ , system (4.1) takes the form

$$\partial \hat{U} / \partial \tau + \partial \hat{F}^j / \partial \xi^j = \hat{Q}, \quad (4.2)$$

where  $\hat{U} = \mathbf{U} / J$ ,  $\hat{Q} = \mathbf{Q} / J$ ,  $\hat{F}^j = \hat{\xi}_i^j \mathbf{U} + \hat{\xi}_x^j \mathbf{F}^i$ ,  $\hat{\xi}_i^j = \xi_i^j / J$ ,  $\hat{\xi}_x^j = \xi_{x^i}^j / J$ , and  $J = \det \|\xi_{x^i}^j\|$ .

The geometric variables  $(J, \hat{\xi}_i^j, \hat{\xi}_x^j)$  must satisfy their own equations—the so-called geometric conservation laws [10, 11]—which can be easily obtained from (4.2) by assuming  $\mathbf{U}, \mathbf{F}^i = \text{const}$ , and  $\mathbf{Q} = 0$ . The latter are conditions of conservation of a particular constant solution on the curvilinear grid:

$$\frac{\partial}{\partial \tau} \left( \frac{1}{J} \right) + \frac{\partial}{\partial \xi^j} (\hat{\xi}_i^j) = 0, \quad (4.3)$$

$$\frac{\partial}{\partial \xi^j} (\hat{\xi}_{x^i}^j) = 0, \quad i = 1, 3. \quad (4.4)$$

Conditions (4.3) and (4.4) imply that the constant solution to equation (4.2) is not changed under the grid transformations. One of the ways to meet condition (4.3) is to find the Jacobian of the transformation to new coordinates immediately from (4.3) (by the same method that is used to solve system (4.3)) rather than by the formula  $J = \det \|\xi_{x^i}^j\|$ .

## 5. DIFFERENCE SCHEME

For the difference scheme when solving (4.2), we use Rodionov's UNO scheme [12], which is a second-order approximation scheme by spatial variables. The second-order approximations by time are obtained by using the predictor–corrector technique.

In terms of variables  $\xi(\xi^1, \xi^2, \xi^3)$ , the rectangular uniform grid  $\xi_{ijk}^l$  is defined as:

$$\xi_{ijk}^1 = i h_1, \quad \xi_{ijk}^2 = j h_2, \quad \xi_{ijk}^3 = k h_3, \quad i = \overline{0, N_1}, \quad j = \overline{0, N_2}, \quad k = \overline{0, N_3},$$

where  $h_l = 1/N_l$ ,  $l = 1, 3$ , and  $N_l$  is the number of computational cells along the coordinate  $\xi^l$ ,  $(i, j, k)$ th computational cell is formed by intersection of the coordinate planes  $\xi^1 = \xi_{i \pm 1, j, k}^1$ ,  $\xi^2 = \xi_{i, j \pm 1, k}^2$ , and  $\xi^3 = \xi_{i, j, k \pm 1}^3$ .

The rectangular uniform grid  $\xi_{ijk}^l$  corresponds to the curvilinear grid  $x_{ijk}^l = x_{ijk}^l(\xi_{ijk}^1, \xi_{ijk}^2, \xi_{ijk}^3, \tau)$ ,  $l = \overline{1, 3}$ , in the Cartesian coordinates, which is determined by solving equation (2.5).

The difference approximation of system (4.2) on the grid  $\xi_{ijk}^l$  has the following form:

$$V_{ijk} \frac{\hat{U}_{ijk}^{n+1} - \hat{U}_{ijk}^n}{\Delta \tau} + (\hat{F}^1 S_{\xi^1})_{i+1/2, j, k} - (\hat{F}^1 S_{\xi^1})_{i-1/2, j, k} + (\hat{F}^2 S_{\xi^2})_{i, j+1/2, k} - (\hat{F}^2 S_{\xi^2})_{i, j-1/2, k} + (\hat{F}^3 S_{\xi^3})_{i, j, k+1/2} - (\hat{F}^3 S_{\xi^3})_{i, j, k-1/2} = \hat{Q}_{ijk}, \quad (5.1)$$

where  $V_{ijk} = h_1 h_2 h_3$  is the volume of the computational cell;  $(S_{\xi^1})_{i \pm 1/2, j, k} = h_2 h_3$ ,  $(S_{\xi^2})_{i, j \pm 1/2, k} = h_1 h_3$ , and  $(S_{\xi^3})_{i, j, k \pm 1/2} = h_1 h_2$  are squares of the corresponding faces of the cell; and  $\Delta \tau$  is the time step. The desired function  $\hat{U}_{ijk}^n$  and the sourcewise term  $\hat{Q}_{ijk}$  are computed at the centers of the computational cells, and the fluxes  $\hat{F}_{i \pm 1/2, j, k}^1$ ,  $\hat{F}_{i, j \pm 1/2, k}^2$ , and  $\hat{F}_{i, j, k \pm 1/2}^3$ , at the centers of the corresponding cell faces.

To simplify the discussion, we consider the case where the solution varies only along one coordinate. After the "predictor," we have

$$\tilde{U}_{ijk}^{n+1} = \hat{U}_{ijk}^n - \frac{\Delta\tau}{h_1} [\hat{F}^1(\hat{U}_{i+1/2,j,k}^-) - \hat{F}^1(\hat{U}_{i-1/2,j,k}^+)],$$

where  $\hat{U}_{i+1/2,j,k}^- = \hat{U}_{ijk}^n + 0.5\delta\hat{U}_{ijk}$  and  $\hat{U}_{i-1/2,j,k}^+ = \hat{U}_{ijk}^n - 0.5\delta\hat{U}_{ijk}$ . Finally, on the "corrector" stage, we get

$$\hat{U}_{ijk}^{n+1} = \tilde{U}_{ijk}^{n+1} - \frac{\Delta\tau}{h_1} [\hat{F}^1(\hat{U}_{i+1/2,j,k}^{n+1,-}) - \hat{F}^1(\hat{U}_{i-1/2,j,k}^{n+1,+})],$$

where  $\hat{U}_{i+1/2,j,k}^{n+1,-} = 0.5(\tilde{U}_{ijk}^{n+1} + 0.5\delta\hat{U}_{ijk} + \hat{U}_{i+1/2,j,k}^-)$  and  $\hat{U}_{i-1/2,j,k}^{n+1,+} = 0.5(\tilde{U}_{ijk}^{n+1} - 0.5\delta\hat{U}_{ijk} + \hat{U}_{i-1/2,j,k}^+)$ , and find  $\delta\hat{U}_{ijk}$  by using the limiters for the UNO scheme in the case of the conservation laws [13].

## 6. RESTRICTIONS ON THE TIME STEP. RESTRICTIONS ON THE SHIFT OF GRID NODES IN THE PROCESS OF ADAPTATION.

For scheme (5.1), the condition on the maximum time step in the  $n$ th layer can be written as

$$\Delta\tau \sum_{l=1}^3 \frac{1}{h_l} \max |\xi_l' + \xi_{x^m}^l u^m + a\delta^l| < 1, \quad (6.1)$$

where  $h_l$  is the grid step along the coordinate  $\xi^l$ ,  $u^m$  are Cartesian components of the velocity vector,  $a$  is the velocity of sound, and  $\delta^l = (\xi_{x^m}^l \xi_{x^m}^l)^{1/2}$  (without summation on  $l$ ).

Since

$$\xi_l' = \frac{\Delta\xi_l'}{\Delta\tau}, \quad \Delta\xi_l' = \xi_{n+1}^l - \xi_n^l,$$

where  $\xi_n^l$  and  $\xi_{n+1}^l$  are coordinates of the grid node before and after, respectively, the operation of the grid generator, the computed velocity of the node can take an arbitrary large value as  $\Delta\tau \rightarrow 0$ . In view of this, it seems reasonable to restrict the maximum velocity of the grid nodes by a certain value that agrees with velocity values in the computational region. This can be done if we define  $\xi_{n+1}^l$  as

$$\xi_{n+1}^l = \xi_n^l + \Delta\tilde{\xi}_l', \quad (6.2)$$

where

$$\Delta\tilde{\xi}_l' = \Delta\xi_l' \frac{u_{\text{char}}^l \Delta\tau}{\max |\Delta\xi_l'|} \Lambda, \quad (6.3)$$

$u_{\text{vap}}^l = \min |\xi_{x^m}^l u^m \pm a\delta^l|$ , and  $0 < \Lambda < 1$ . Then,

$$\xi_l' = \Delta\tilde{\xi}_l' / \Delta\tau. \quad (6.4)$$

Thus, the node velocity is given by  $\xi_l' = \alpha^l u_{\text{char}}^l$ , where

$$\alpha^l = \Delta\xi_l' (\max_{i,j,k} |\Delta\xi_l'|)^{-1} \Lambda \leq 1.$$

By virtue of (6.3) and (6.4), we get from (6.1) the following restriction on the time step:

$$\Delta\tau < \left( \sum_{l=1}^3 \frac{1}{h_l} \max |\alpha^l u_{\text{char}}^l + \xi_{x^m}^l u^m \pm a\delta^l| \right)^{-1}. \quad (6.5)$$

In the one-dimensional case, (6.5) takes the form

$$\Delta\tau < \frac{h}{\max |\xi_x(-\dot{x} + (u \pm a))|}, \quad (6.6)$$



where  $\dot{x} = -\xi_r/\xi_x$  is the velocity of the grid node. Let us compare (6.6) with the similar restriction for the case of computation on a stationary grid,

$$\Delta\tau < \frac{h}{\max|\xi_x(u \pm a)|}. \quad (6.7)$$

With regard to (6.3), we have the following estimate for  $\dot{x}$ :

$$|\dot{x}| = \frac{|\alpha|}{|\xi_x|} u_{\text{char}} = \frac{|\alpha|}{|\xi_x|} \min|\xi_x(u \pm a)| < |u \pm a|.$$

Then, instead of (6.6), we can write a stronger condition,

$$\Delta\tau < \frac{h}{\max(|\xi_x||u \pm a|) - \min(|\xi_x||\dot{x}|)} \leq \frac{h}{(1 - \tilde{\alpha}) \max(|\xi_x||u \pm a|)}, \quad (6.8)$$

where

$$\tilde{\alpha} = \alpha \frac{\min|\xi_x| \min(|\xi_x||u \pm a|)}{|\xi_x| \max(|\xi_x||u \pm a|)} < 1.$$

Thus, in the case of computations on adaptive grids, restriction (6.8) on the time step, which is stronger than (6.6), turns out to be considerably weaker than restriction (6.7) for computations on stationary grids.

Similar to condition (6.1) for Eq. (4.2), we can write the following condition for Eq. (4.3):

$$\Delta\tau \sum_{l=1}^3 \frac{1}{h_l} \max|\xi_l^l| < 1. \quad (6.9)$$

Taking into account (6.3) and (6.4), we can obtain from (6.9) the restriction on the maximum shift of nodes of the grid after the  $n$ th time step in the process of its adaptation to the solution as

$$\max|\Delta\tilde{\xi}^l| < h_l/3.$$

To satisfy this condition, we define  $\Lambda$  in (6.3) as

$$\Lambda = \frac{1}{3} \min \left\{ 1, \frac{h}{u_{\text{char}} \Delta\tau} \right\} < 1. \quad (6.10)$$

Indeed, in this case, we have

$$|\Delta\tilde{\xi}^l| = \frac{u_{\text{char}}^l \Delta\tau}{3} \frac{|\Delta\xi^l|}{\max|\Delta\xi^l|} \min \left\{ 1, \frac{h}{u_{\text{char}} \Delta\tau} \right\} \leq h.$$

Thus, by means of (6.5), we find  $\Delta\tau$  for the computation on the  $(n+1)$ th time layer, and (6.2), (6.3), and (6.10) give us  $\xi_{n+1}^l$ .

## 7. EXAMPLES OF COMPUTATIONS. CONSEQUENCES OF THE EXPLOSION OF A COSMIC BODY IN THE EARTH ATMOSPHERE

To illustrate the applicability of the algorithm suggested to the computation of essentially unsteady three-dimensional gas flows with high gradients and to demonstrate its efficiency, we consider the problem on gas-dynamics consequences of the explosion of a cosmic body in the Earth atmosphere.

Under the explosion of a cosmic body, we mean the following process. Since the effect of the incoming flow is assumed insufficient for the separation of the body into large pieces, it is subjected to the avalanche fragmentation into small pieces under the action of the dynamic loads due to the deceleration. The collection of these small pieces behaves like incompressible liquid. The development of the convective instability on the windward side of the region results in its break near the critical point and the loss of simple connectedness. The hot gas from the shock layer penetrates into the region, which results in almost instantaneous evaporation of the pieces. It is this process that is referred to as the explosion [14].

To specify the initial data, we used the problem statement suggested in [15], which is as follows. A region containing a homogeneous gas mixture is placed in the stratified atmosphere at the height  $h_c$  of the explosion



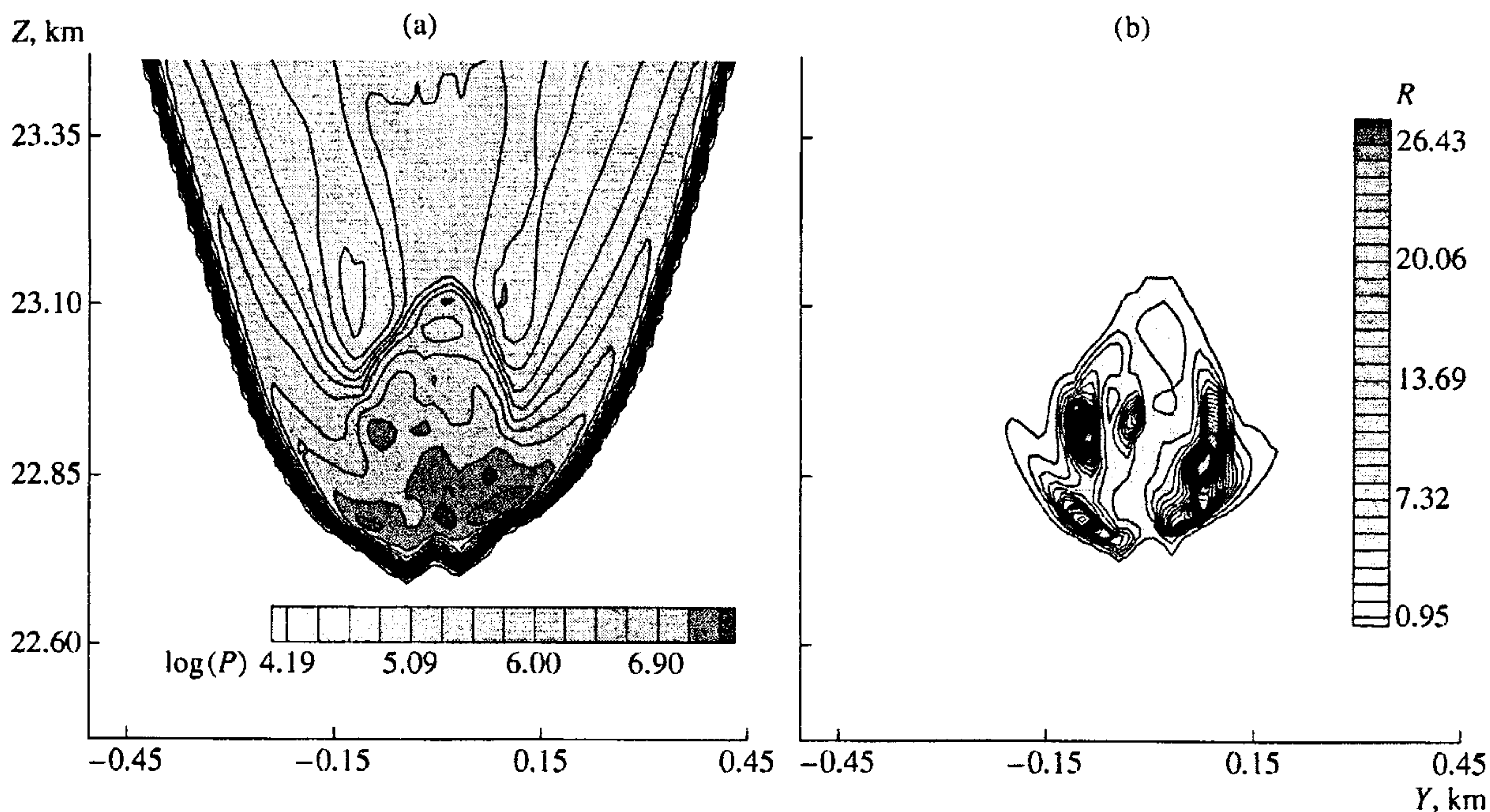


Fig. 1.

(the explosion is meant in the sense of [16]). The gas is described by the following parameters. The mass of the gas is equal to the mass of the cosmic body; the velocity is equal to the velocity of the body  $V_\infty$  at the moment when it enters the atmosphere; the gas density is equal to the density of the body material; the static pressure is equal to the deceleration pressure at the given height,  $P_0 = \rho(h_e) V_\infty^2$ , where  $\rho(h_e)$  is the atmospheric density at the height of the explosion. The only uncertain parameter—the height of the explosion—can be estimated from an approximate solution [16].

We solved the problem on the explosion consequences for an ice cosmic body of density  $\rho_0 = 10^3 \text{ kg/m}^3$ , mass  $m = 216 \times 10^3 \text{ t}$ , and characteristic size  $L = 60 \text{ m}$ . The body entered the atmosphere at the angle  $\Theta = 45^\circ$  with respect to the normal and had the velocity  $V_\infty = 20 \text{ km/s}$ . (The parameters chosen correspond to the generally received parameters of the Tunguska cosmic body [17].) The gas was assumed nonviscous and non-heat-conducting with the equation of state of a real gas.

The height of the explosion was estimated by means of an approximate solution [16] as  $h_e = 3.04H = 21.3 \text{ km}$ , and  $\rho(h_e) \approx 10^{-1} \text{ kg/m}^3$ . Here,  $H = 7 \text{ km}$  is the height scale in terms of the Earth atmosphere density (the scale of the atmospheric irregularity).

The coordinate system  $OXYZ$  was oriented in space such that the axis  $Z$  and the velocity of the body when it enters the atmosphere are directed opposite to each other and the vector  $\mathbf{g}$  belongs to the plane  $XZ$ . For the computational region, we used the rectangular parallelepiped of the initial size  $L_x = L_y = L_z = 300 \text{ m}$  with the edges oriented along the coordinate axes. The computational region moved along the  $Z$ -axis keeping its orientation in space and increased in size such that the flow segment of interest was always inside it.

To understand the effect of the grid adaptation on the results of computations, several series of computations were carried out for an initial stage of the evolution of the gas cloud, which was formed after the explosion. The computations were carried out up to the moment  $0.3 \text{ s}$  after the explosion. The side boundaries of the computational region oriented perpendicular to the axes  $OX$  and  $OY$  moved outside along their normals with the velocity  $1.2 \text{ km/s}$ . The boundaries oriented perpendicular to the axis  $OZ$  moved along  $OZ$  in the negative direction: the lower boundary with the velocity  $20.5 \text{ km/s}$ , and the upper boundary with the velocity  $18 \text{ km/s}$ . Thus, the computational region moved along the axis  $OZ$  and increased in size such that, at the moment  $t = 0.3 \text{ s}$ , we had  $L_x = L_y = 1020 \text{ m}$  and  $L_z = 1050 \text{ m}$ .

The numerical experiments were carried out on the following grids:  $20 \times 20 \times 20$  cells (computations 1 and 4);  $40 \times 40 \times 40$  cells (computations 2 and 5), and  $80 \times 80 \times 80$  cells (computation 3).



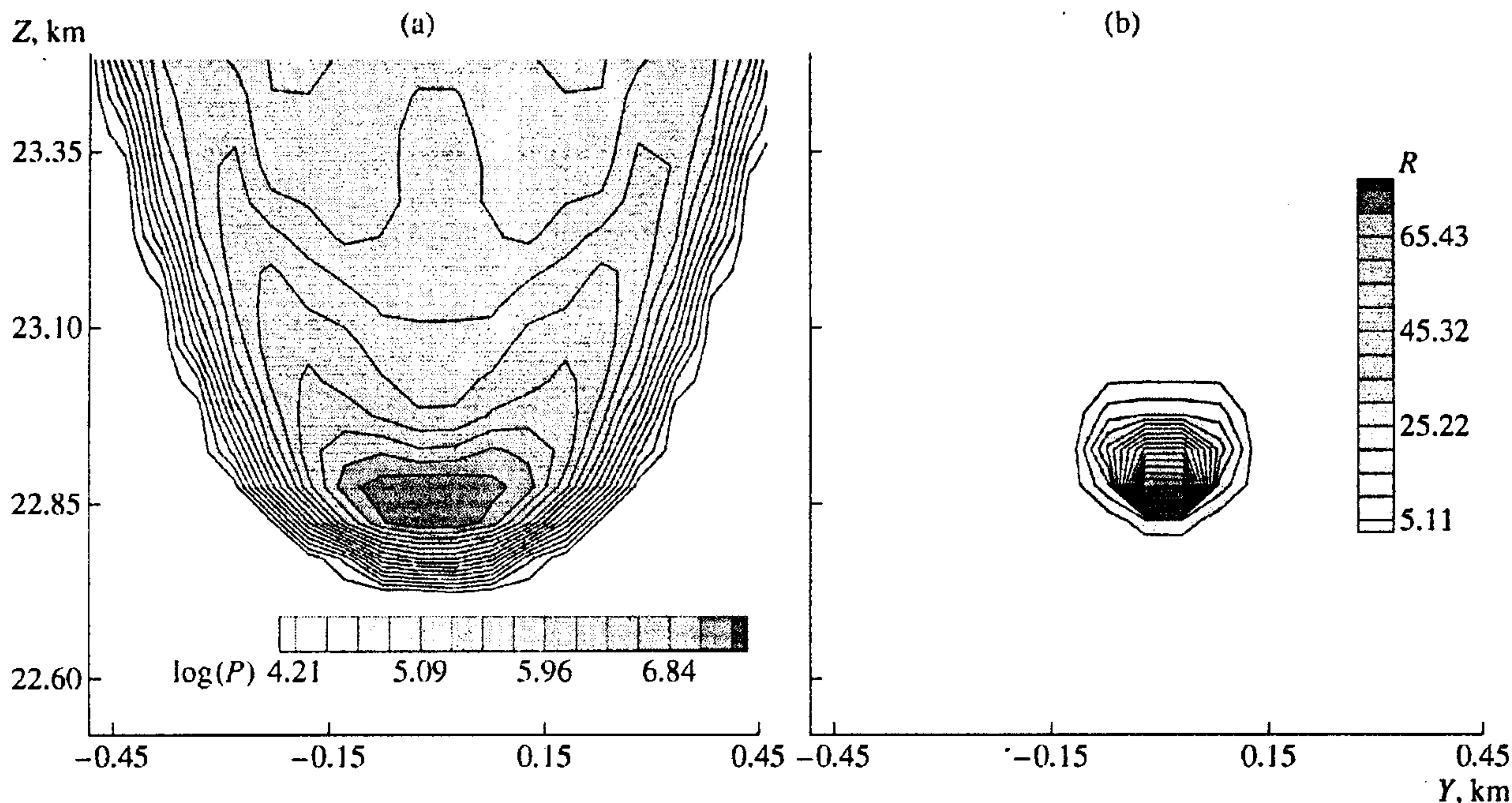


Fig. 2.

In computations 1, 2, and 3, a uniform rectangular grid of the appropriate dimension was used, which was recalculated on each time step.

In computations 4 and 5, we used a dynamically adaptive grid. For the initial approximation, a uniform rectangular grid of the appropriate dimension was used. The procedure of the grid adaptation was invoked on every second time step (which turned out sufficient). The grid was adapted to the solution

$$f = \sigma_p \log\left(\frac{P}{P_0}\right) + \sigma_\rho \log\left(\frac{\rho}{\rho_0}\right), \quad \sigma_p = \sigma_\rho = 0.5,$$

where  $P$  and  $\rho$  are pressure and density, respectively;  $P_0$  and  $\rho_0$  are characteristic values of pressure and density in the computational region; and  $\sigma_p$  and  $\sigma_\rho$  are constants.

Such a choice of the function  $f$  for the given gas-dynamics problem is explained by the necessity to ensure grid adaptation both to the region of the head shock wave, which is characterized by high gradients of  $P$ , and to the wake boundary, for which considerable variations of  $\rho$  are typical. Moreover,  $\rho$  and  $P$  are computed in the course of solving the gas-dynamics part of the problem, which considerably simplifies finding  $f$ . It should be noted that, generally, no unique recipe of construction of the function  $f$  for an arbitrary problem exists; this function is determined by the particular physical problem and the requirements on its solution.

Results of computation 3 seem to be most reliable in what concerns the reconstruction of the real picture of the flow, since, in this case, the finest uniform grid was used. Hence, the results are certainly free of distortions that could be introduced by the grid adaptation. Therefore, we, first, describe the results of computation 3, and, then, compare them with those of the other computations. Basic results of computation 3 are as follows.

Table 1

Parameters	Computation				
	1	2	3	4	5
$P_{\max}$ , Pa	$3.31 \times 10^7$	$5.0 \times 10^7$	$6.4 \times 10^7$	$6.5 \times 10^7$	$5.8 \times 10^7$
$\rho_{\max}$ , kg/m <sup>3</sup>	92.4	164.2	230.1	314.5	333.4
$T_{\max}$ , K	19200	23900	24000	35000	28800



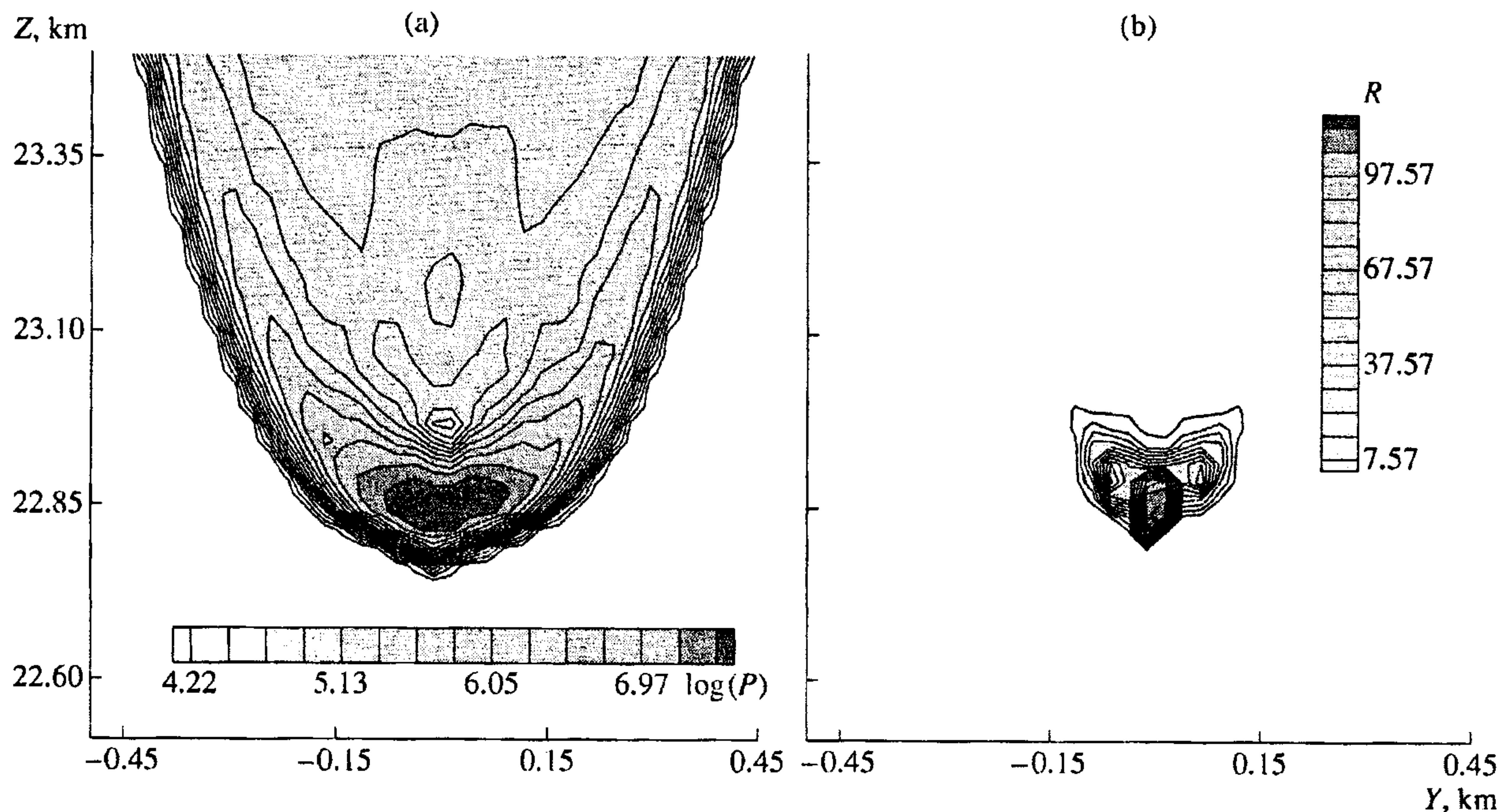


Fig. 3.

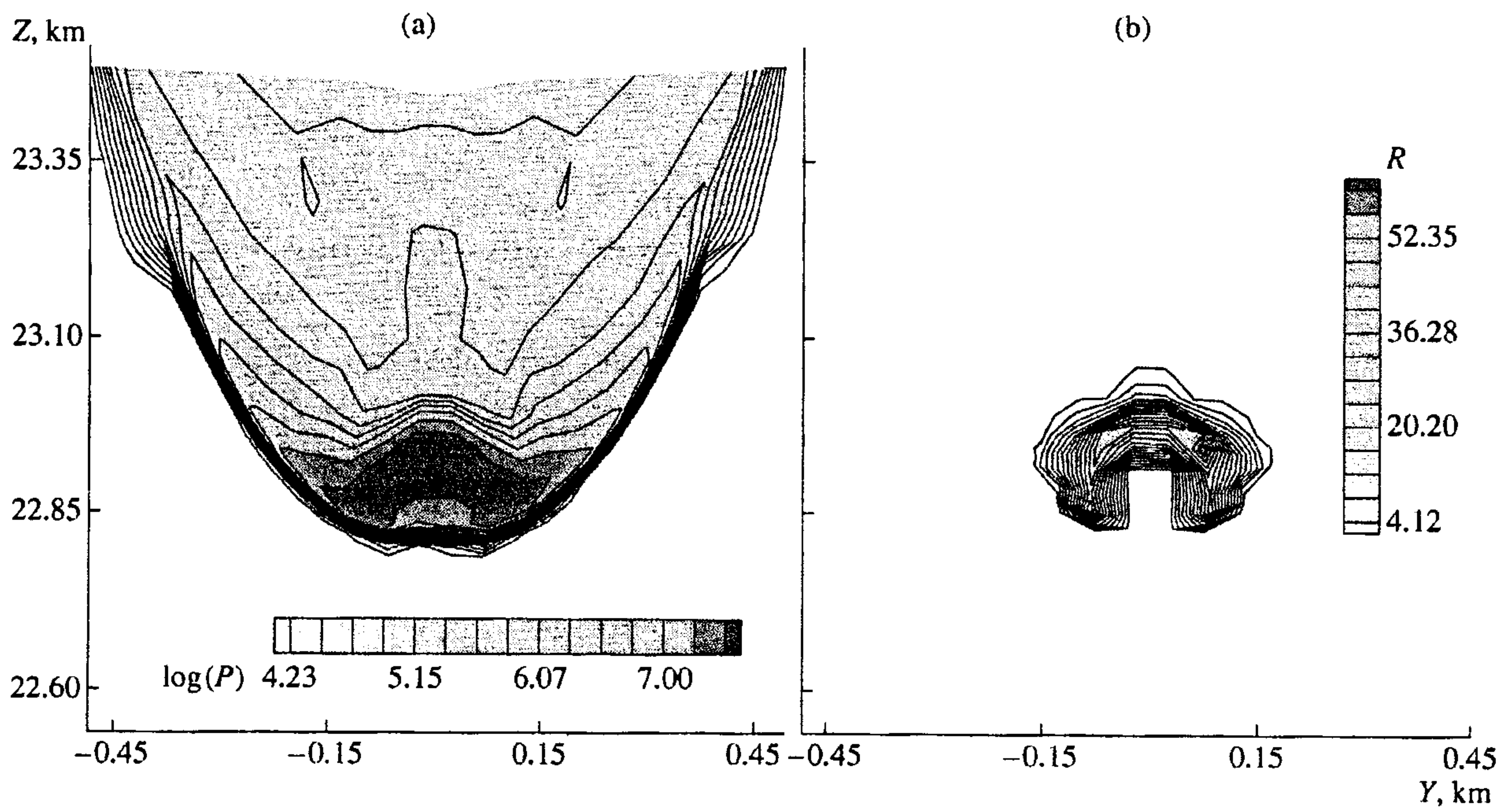


Fig. 4.

From the moment of the explosion until 0.1 s, the observed picture of the flow is characterized by the separation of the fragment and the expansion of the material in the side direction and backward, to the wake region. The maximum pressure is observed behind the front of the shock wave, on the leading edge of the dense gas cloud. The maximum temperature is observed in the shock wave front.

By  $\sim 0.12$  s, the material of the cosmic body is collected in a bowl-like layer of characteristic size  $\sim 200$  m. At approximately 0.13 s, there begins the process of forcing this layer through along the axis under the action of the convective instability, and, by the time  $\sim 0.2$  s, the dense central cloud loses the property of



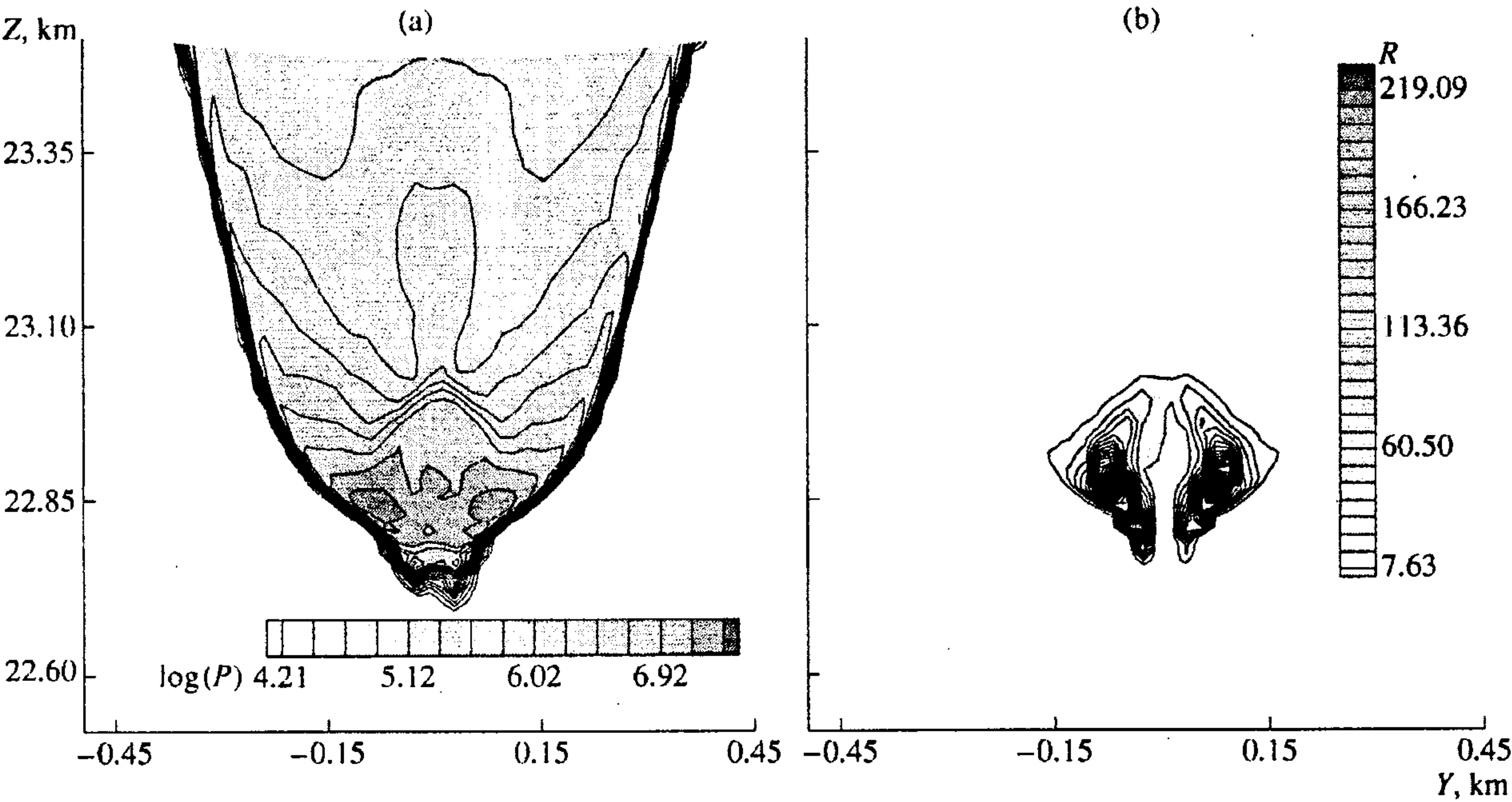


Fig. 5.

being simply connected. On the axis of the cloud, a channel of characteristic cross size ~30 m filled with rarefied gas is formed. By the moment 0.3 s, the dense gas cloud is separated into several fragments, which can easily be seen in Fig. 1 ((a) for log(P) and (b) for  $\rho$  [kg / m<sup>3</sup>], the cross section in the plane  $X = 0$  km).

The flow parameters are as follows. The shock wave is at the height ~16 km ( $Z = 22.7$  km, and the angle between the axis  $OZ$  and the normal to the Earth surface is  $\Theta = 45^\circ$ ); the characteristic size is ~600 m; the wave-front velocity is ~-19 km/s;  $P_{\max}$  is approximately  $7.4 \times 10^7$  Pa; the maximum gas density in the cloud fragments is ~230.1 kg/m<sup>3</sup>; and the maximum temperature ~24000 K is observed in the front edges of the dense fragments. The flow behind the shock wave has complex spatial structure.

The computations on the grids with lesser number of cells (computations 1, 2, 4, and 5), in the whole, recover the same flow picture as that in computation 3. For example, they all give about the same depth of the penetration of the shock wave and the same wave-front velocity: at  $t = 0.3$  s, the shock wave is at the height ~16 km, and the wave-front velocity is ~19 km/s. The values of the basic flow parameters at 0.3 s are presented in Table 1. The pictures of the flow obtained in computations 1, 2, 4, and 5 for the same moment are depicted in Figs. 2-5, respectively. Here, and in what follows, the motion of the gas cloud is described by the isolines of the logarithm of pressure log(P) and density  $\rho$ . The panels (a) and (b) in Figs. 2-5 show the isolines of log(P) and  $\rho$ , respectively, in the plane  $X = 0$  km.

As can be seen from Table 1, in computations on uniform grids, the maximum values of pressure, density, and temperature grow as the cell size decreases through the reduction of the scheme viscosity. The computations on adaptive grids yield results close to those of computation 3 with the use of a considerably lesser number of cells through the node concentration (crowding) in the regions where the solution has high gradients. In view of complexity of the problem under consideration and lack of information about the accurate solution, the conclusion about the quality of the results obtained on the adaptive grids rely, in the first turn, on the comparison of the qualitative pictures of the flow obtained in computations 4 and 5 with those in computation 3.

Table 2

Timing	The number of cells	20 × 20 × 20	40 × 40 × 40	80 × 80 × 80
Timing for one step (without adaptation/with adaptation)		1/1.4	8/11.2	37/-
Timing for one variant, h (without adaptation)		~3	~24	~120



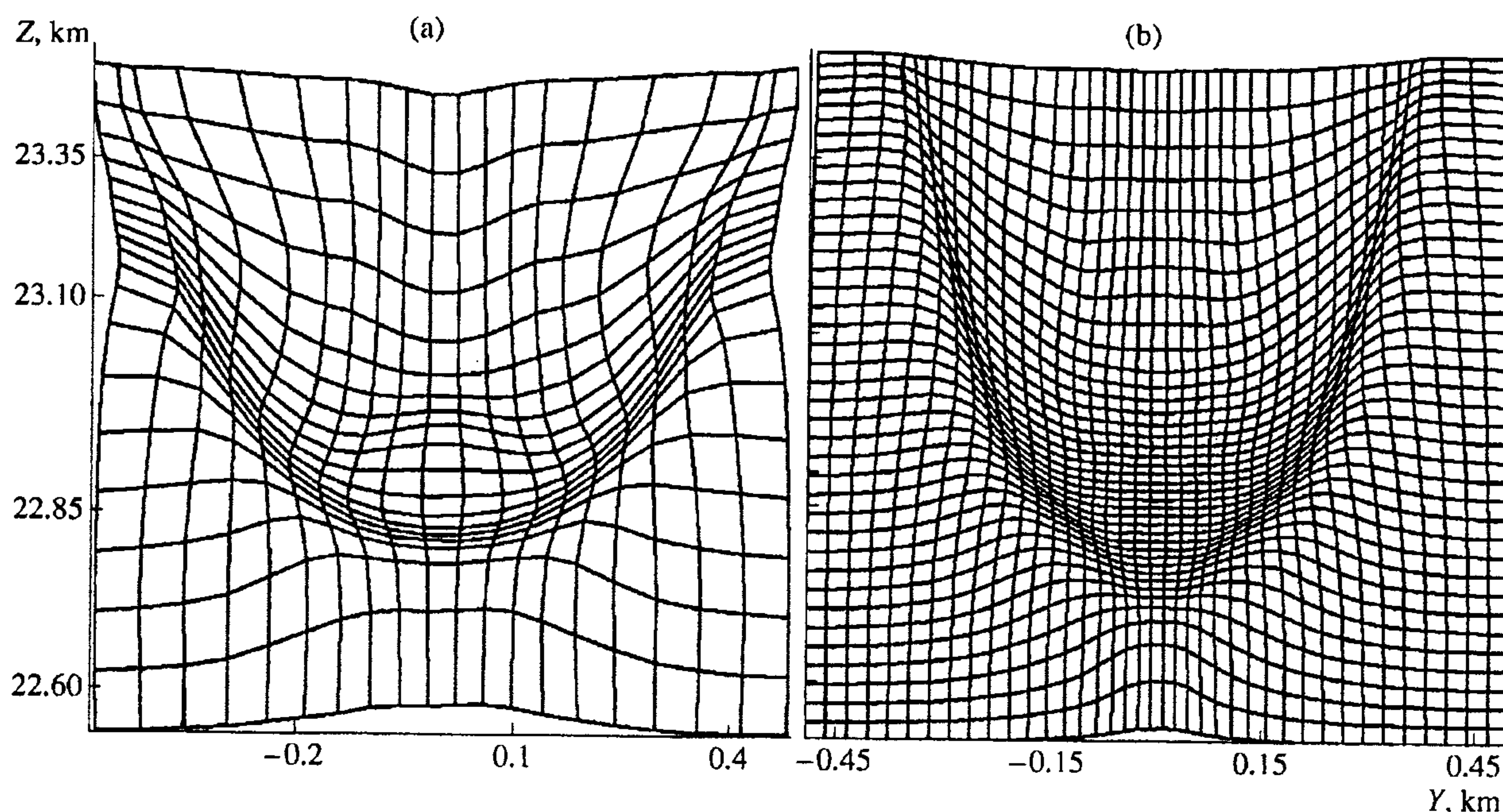


Fig. 6.

As can be seen from the figures, the use of the adaptive grid makes it possible to observe the phenomenon of fragmentation of the dense cloud even on the grid with  $20 \times 20 \times 20$  computational cells, although the picture obtained looks rather schematic compared to those in computations 3 and 5, and the values of pressure and temperature are overestimated (see Table 1). It should be noted that this phenomenon is not obtained on the uniform grid with the same number of cells  $20 \times 20 \times 20$  (computation 1), and even on the uniform grid with  $40 \times 40 \times 40$  cells (computation 2), which brings us to the conclusion that the adaptation of the grid to the solution was quite sufficient. In addition, the use of the adaptive grid made it possible to more clearly separate the front of the shock wave (the difference is especially noticeable when comparing Figs. 2 and 3).

The computational grids for variants 4 and 5 at  $t = 0.3$  s are depicted in Figs. 6a ( $20 \times 20 \times 20$  grid) and 6b ( $40 \times 40 \times 40$  grid), respectively, with the cross section being in the plane  $X = 0$  km. The degree of the computational cell contraction can be represented by the number  $\gamma = V_{un}/V_{min}$ , where  $V_{min}$  is the minimum volume of the grid cell and  $V_{un}$  is the cell volume for the uniform grid with the same number of nodes constructed in the same computational region. For uniform grids,  $\gamma = 1$ . For the grids, shown in Fig. 6,  $\gamma = 6.3$  (the grid in computation 4) and  $\gamma = 6.2$  (the grid in computation 5).

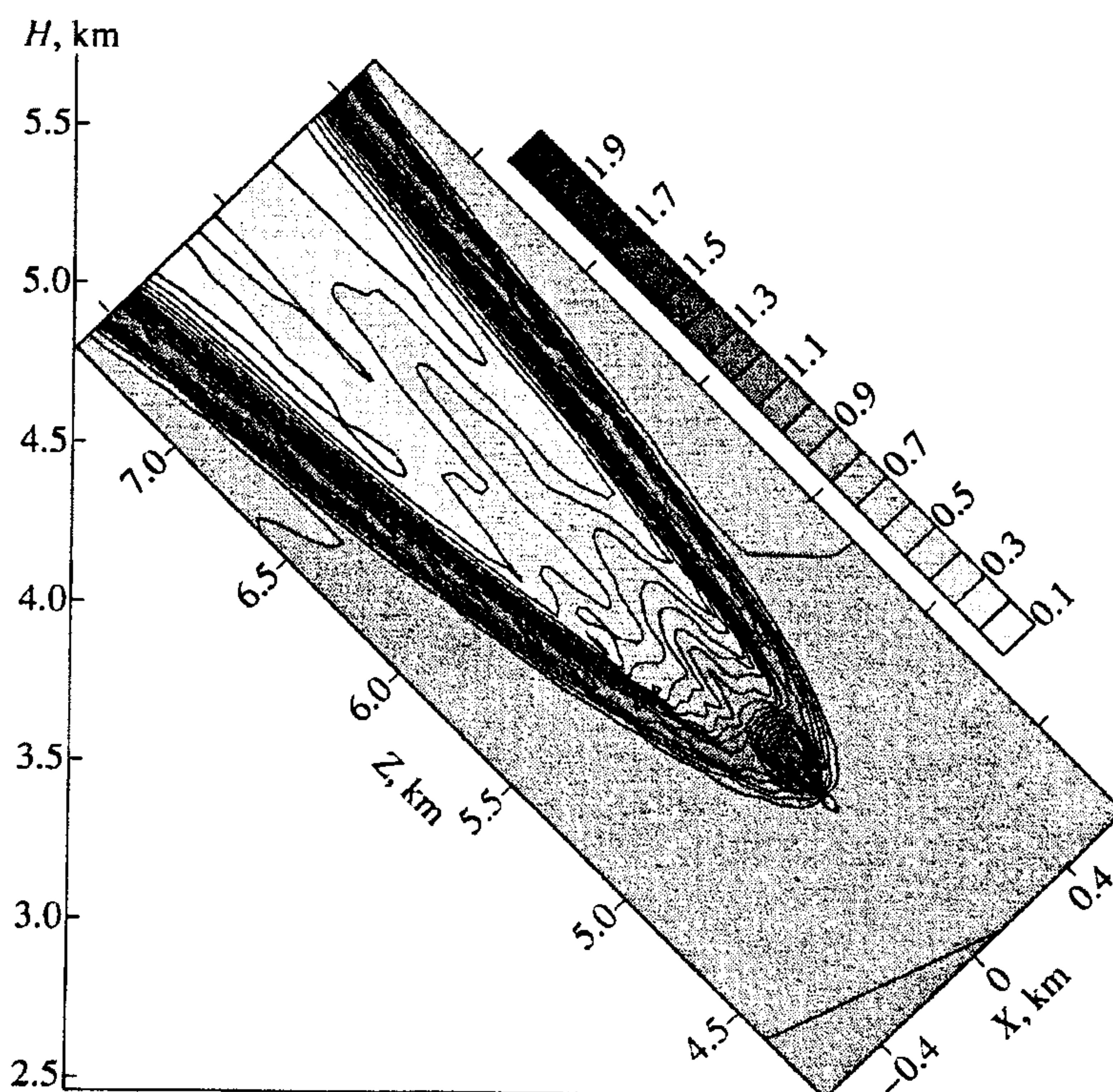


Fig. 7.



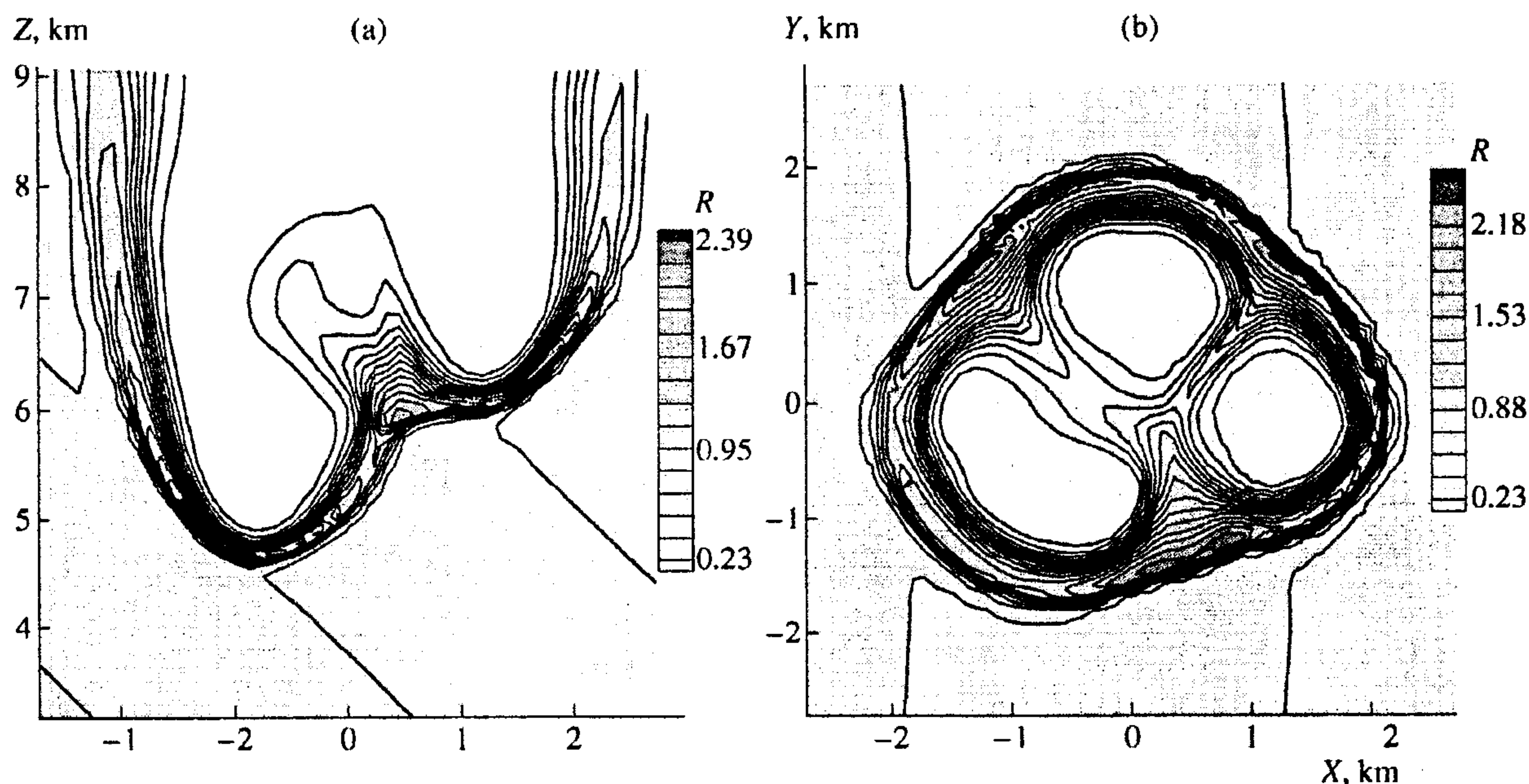


Fig. 8.

Table 2 shows computation time per one time step for different grid dimensions (here, it is measured in terms of the time required for the computation on the grid  $20 \times 20 \times 20$  without adaptation) and time (in hours) required for the whole variant in the case of computation on a uniform grid. The experiments were carried out on a PC Pentium Pro 200.

Thus, the comparison of the results of computations 1–5 allows us to conclude that the algorithm suggested can efficiently be used for modeling spatial unsteady gas flows with high gradients.

The importance of the phenomenon of separation of the dense gas cloud into fragments and its effect on the further development of the flow are illustrated by results obtained in computations 6 and 7, which were carried out on the grid consisting of  $40 \times 40 \times 40$  cells. The motion of the computational region was similar to that in computations 1–5, and the velocities of the boundaries of the computational region were selected such that the head part of the shock wave was always inside the region. The computations started from 0.3 s after the explosion and lasted until the shock wave reached the height of  $\sim 3$  km. In computation 6, the uniform rectangular grid was used, which was recalculated on each time step. For the initial data, the results of computation 3 were used. In computation 7, the adaptive grid was used, with the adaptation procedure being invoked on every second time step. For the initial data and grid approximation, the results of computation 5 were used.

Computation 6 gave the following results. As the dense gas cloud spreads out and decelerates, the entire picture of the flow changes. Starting from  $\sim 0.8$  s, the shock wave takes the shape of a blunted cone with the rounding radius  $\sim 150$  m; the wave-front velocity is  $\sim 14$  km/s;  $P_{\max} \sim 8.0 \times 10^6$  Pa; and  $T_{\max} \sim 10^3$  K. A flow of hot rarefied gas of density  $\sim 0.2$ – $0.3$  kg/m<sup>3</sup> and temperature  $\sim 4000$  K follows the shock wave at the velocity  $\sim 17$  km/s. The shock wave reaches the height of  $\sim 3.4$  km ( $Z \sim 4.8$  km) at  $\sim 1.5$  s; at this moment, its velocity is  $\sim 8.2$  km/s, and  $P_{\max} \sim 8.0 \times 10^6$  Pa (Fig. 7 for  $\rho$  [kg/m<sup>3</sup>], where the cross section belongs to the plane  $Y = 0$  km).

The results of computation 7 considerably differ from those of computation 6. By  $\sim 0.4$  s, the dense cloud is completely separated into several fragments, which considerably increases the cross section of the region occupied by the flow and results in a noticeable deceleration of the fragments in the atmosphere. The picture of the flow is now not symmetric with respect to the plane  $X = 0$ . As the cloud spreads out, the dense fragments generate a system of shock waves followed by hot rarefied gas. The shock wave reaches the height of  $\sim 3.4$  km later, at  $t \sim 3.6$  s, when it has velocity  $\sim 1$  km/s, characteristic cross size  $\sim 4$  km, and  $P_{\max} \sim 6.0 \times 10^6$  Pa. The flow of hot rarefied gas of density  $0.1$ – $0.2$  kg/m<sup>3</sup> and temperature  $\sim 8 \times 10^3$  K follows the shock wave at the velocity  $3$ – $4$  km/s. The observed picture of the flow is represented in Figs. 8a ( $\rho$  [kg/m<sup>3</sup>], the cross section in the plane  $Y = 0$ ) and 8b ( $\rho$  [kg/m<sup>3</sup>], the cross section in the plane  $Z = 6.78$  km). The computational grids for the cross sections in the planes  $Y = 0$  and  $Z = 6.78$  km are depicted in Fig. 9.



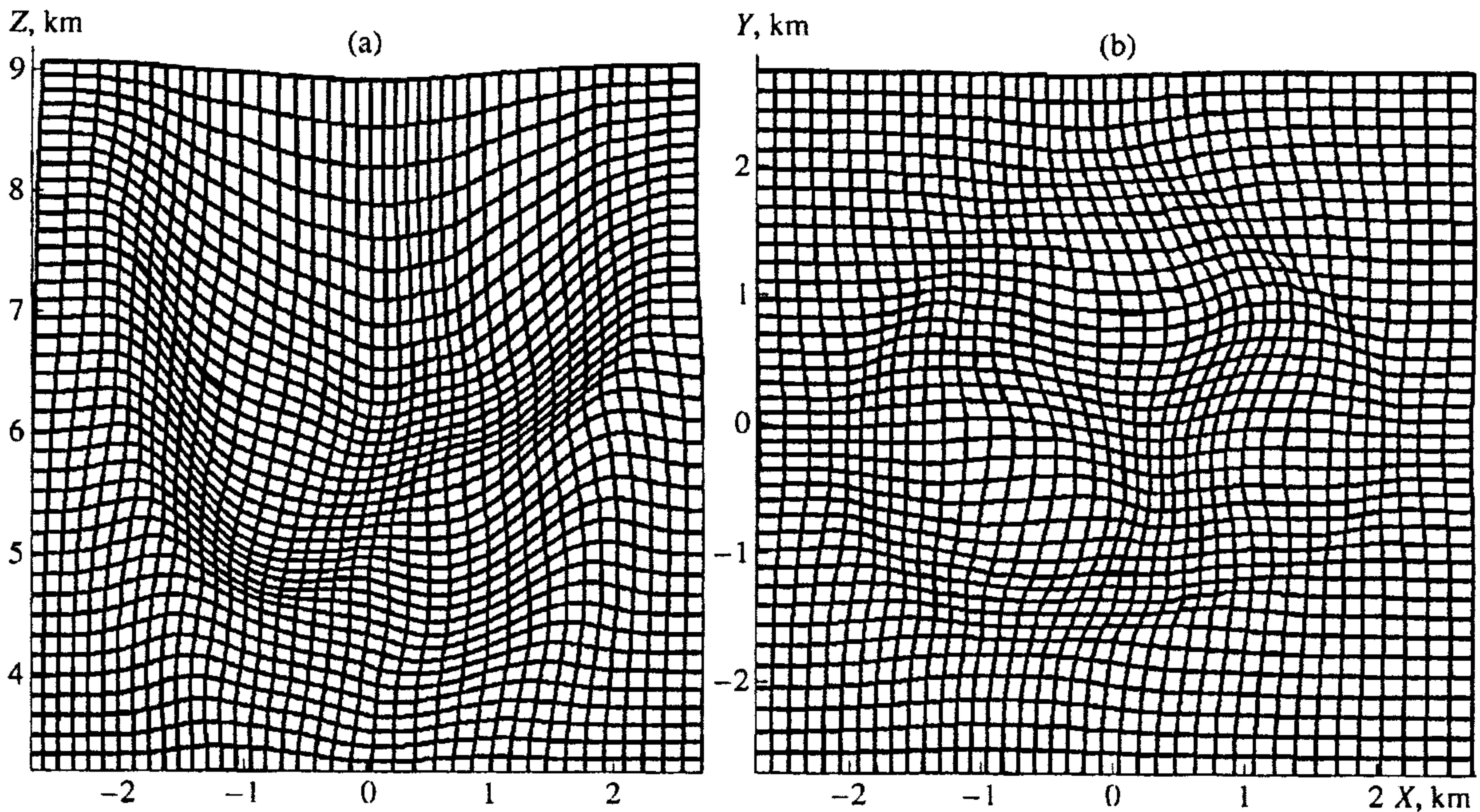


Fig. 9.

Computation 8 illustrates the capabilities of the algorithm to model large-scale irregularities, such as, for example, the wake formed by the passage of a dense gas cloud consisting of explosion products. This computation was carried out on the dynamically adaptive grid consisting of  $40 \times 40 \times 40$  cells. The boundaries of the computational region moved in space such that to hold the head shock wave and the whole trajectory of the explosion products inside the computational region. The initial data were taken the same as those for computation 5.

In spite of the considerable difference in the characteristic dimensions (at the moment  $\sim 3.6$  s, the characteristic size of the head part of the shock wave along the  $OZ$  axis is  $\sim 5$  km, whereas the characteristic size of the wake is about 25 km), the use of the dynamically adaptive grid allowed us to reconstruct the basic features of the gas cloud evolution and the behavior of the head shock wave obtained in computation 7. For example, by  $\sim 3.6$  s, the head of the shock wave is on the height  $\sim 5$  km, has velocity  $\sim 1$  km/s, and  $P_{\max} \sim 3.8 \times 10^6$  Pa. The function  $\log(P)$  is shown in Fig. 10a, and the cross section of the computational grid in the plane  $Y = 0$  km is depicted in Fig. 10b. Thus, the use

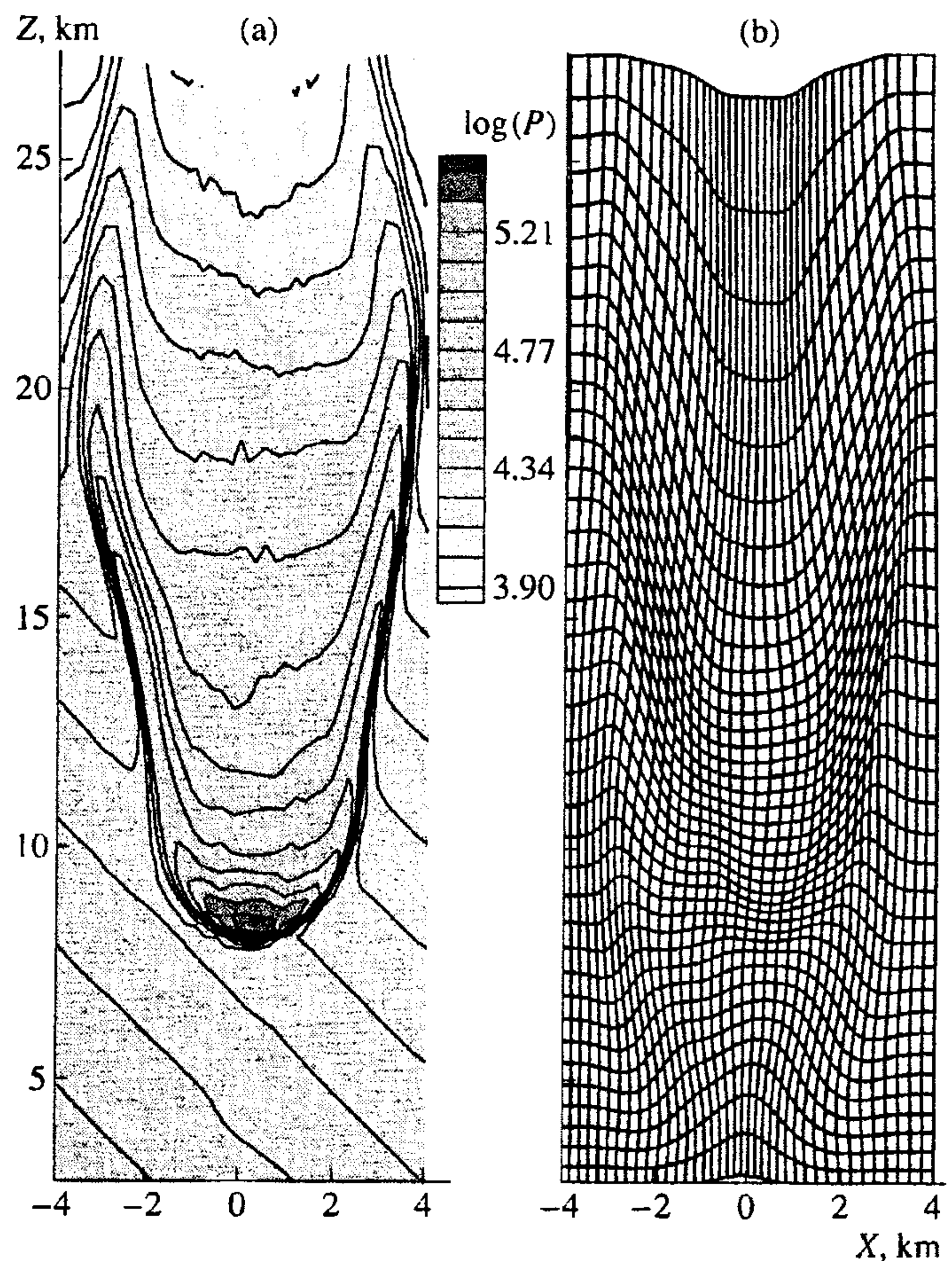


Fig. 10.



of the adaptive grid makes it possible to study both the wake formation and the motion of the gas cloud formed by the explosion products in the framework of one computation.

## 8. CONCLUSIONS

The numerical algorithm for construction of three-dimensional dynamically adaptive grids has been suggested. Based on the results of solving the problem on the consequences of the explosion of a cosmic body in the Earth atmosphere, the conclusion is made that this algorithm can efficiently be used for computation of essentially unsteady three-dimensional gas flows with high gradients.

The use of dynamically adaptive grids makes it possible to take into account specific features of the solution, which affect the qualitative picture of the flow, in computations on grids with relatively small number of computational nodes. Dynamically adaptive grids allow us to examine flows generated by the motion of high-gradient zones inside the computational region in the framework of one computation and in the same computational region, even in the case where the characteristic size of the high-gradient zone is considerably less than the size of the computational region. The use of the adaptive grids considerably reduces the required computational resources, since, in the case of a uniform grid, in order to get the same step in a region of high solution gradients as that used in the adaptive grids, we have to increase the number of nodes in each direction by several times.

## ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Basic Research, project nos. 00-01-00321 and 00-15-96030.

## REFERENCES

1. Godunov, S.K., Zabrodin, A.V., Ivanov, M.Ya., Kraiko, A.N., and Prokopov, G.P., *Chislennyye resheniya mnogomernykh zadach gazovoi dinamiki* (Numerical Solutions of Multidimensional Gas Dynamics Problems), Moscow: Nauka, 1976.
2. Ivanenko, S.A. and Charakhch'yan, A.A., Curvilinear Grids Constructed from Convex Quadrangles, *Zh. Vychisl. Mat. Mat. Fiz.*, 1988, vol. 28, no. 4, pp. 503–514.
3. Dar'in, N.A., Mazhukin, V.I., and Samarskii, A.A., A Finite-Difference Method for Solving Gas-Dynamics Equations with the Use of Adaptive Grids Dynamically Related to Solutions, *Zh. Vychisl. Mat. Mat. Fiz.*, 1988, vol. 28, no. 8, pp. 1210–1225.
4. Mazhukin, V.I., Samarskii, A.A., Ornal'do Kastel'yanos, and Shapranov, A.V., A Dynamic Adaptation Method for Nonstationary Problems with High Gradients, *Mat. Model.*, 1993, vol. 5, no. 4, pp. 32–55.
5. Anderson, D.A., Adaptive Mesh Schemes Based on Grid Speeds, *AIAA Paper*, 1983, pp. 311–318.
6. Gan'zha, D.Kh., Muzafarov, I.F., and Utyuzhnikov, S.V., Use of Moving Adaptive Grids in Algorithms with Compact Approximations, *Zh. Vychisl. Mat. Mat. Fiz.*, 1995, vol. 35, no. 8, pp. 1184–1194.
7. Utyuzhnikov, S.V., Gan'zha, D.H., and Polukhin, V.V., Numerical Algorithms on Moving Adaptive Grids for Modeling of Penetration into the Atmosphere of a Planet, in *Constitutive Laws: Theory, Experiments and Numerical Implementation*, Barcelona: CIMNE, 1995, pp. 290–305.
8. Brackbill, J.U. and Saltzman, J.S., Adaptive Zoning for Singular Problems in Two Dimensions, *J. Comput. Phys.*, 1982, vol. 46, pp. 342–368.
9. Bakhvalov, N.S., Zhidkov, N.P., and Kobel'kov, G.M., *Chislennyye metody* (Numerical Methods), Moscow: Nauka, 1987.
10. Anderson, D.A., Tannehill, J.C., and Pletcher, R.H., *Computational Fluid Mechanics and Heat Transfer*, New York: Hemisphere, 1984, vol. 2. Translated under the title *Vychislitel'naya gidromekhanika i teploobmen*, Moscow: Mir, 1990, vol. 2.
11. Thomas, P.D. and Lombard, C.K., Geometric Conservation Law and Its Application to Flow Computations on Moving Grids, *AIAA J.*, 1979, vol. 17, no. 10, pp. 1030–1037.
12. Rodionov, A.V., Monotonous Second-Order Approximation Scheme for Through Computation of Unsteady Flows, *Zh. Vychisl. Mat. Mat. Fiz.*, 1987, vol. 27, no. 4, pp. 585–593.
13. Hirsch, C., *Numerical Computation of Internal and External Flows*, London: Wiley, 1988, vol. 2.
14. Korobeinikov, V.P., Chushkin, P.I., and Shurshalov, L.V., In-Flight Explosion, in *Modelirovanie v mekhanike* (Modeling in Mechanics), 1988, vol. 2(19), no. 5, pp. 104–1210.
15. Kondaurav, V.I., Konyukhov, A.V., Polukhin, V.V., and Utyuzhnikov, S.V., Mathematical Simulation of Gas Cloud Motion Following the Atmospheric Explosion of a Meteoroid, *J. Fluid Dynamics*, 1998, vol. 33, no. 1, pp. 24–30.
16. Grigoryan, S.S., Motion and Destruction of Meteorites in Planet's Atmosphere, *Kosm. Issled.*, 1979, vol. 17, no. 2, pp. 875–893.
17. Korobeinikov, V.P., Gusev, S.B., Chushkin, P.I., and Shurshalov, L.V., Flight and Fracture of the Tunguska Cosmic Body into Earth's Atmosphere, *Comput. Fluids*, 1992, vol. 21, no. 3, pp. 323–330.