ISSN 0965-5425, Computational Mathematics and Mathematical Physics, 2016, Vol. 56, No. 6, pp. 1065–1074. © Pleiades Publishing, Ltd., 2016.
Original Russian Text © N.G. Burago, I.S. Nikitin, V.L. Yakushev, 2016, published in Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 2016, Vol. 56, No. 6, pp. 1082–1092.

Dedicated to the memory of O.M. Belotserkovskii

Hybrid Numerical Method with Adaptive Overlapping Meshes for Solving Nonstationary Problems in Continuum Mechanics

N. G. Burago[a], I. S. Nikitin[b], and V. L. Yakushev[b]

[a] *Institute for Problems of Mechanics, Russian Academy of Sciences,*
pr. Vernadskogo 101, korp. 1, Moscow, 119526 Russia

[b] *Institute for Computer-Aided Design, Russian Academy of Sciences,*
ul. Vtoraya Brestskaya 19/18, Moscow, 123056 Russia

e-mail: buragong@yandex.ru, i_nikitin@list.ru, yakushev@icad.org.ru

Received November 9, 2015

Abstract—Techniques that improve the accuracy of numerical solutions and reduce their computational costs are discussed as applied to continuum mechanics problems with complex time-varying geometry. The approach combines shock-capturing computations with the following methods: (1) overlapping meshes for specifying complex geometry; (2) elastic arbitrarily moving adaptive meshes for minimizing the approximation errors near shock waves, boundary layers, contact discontinuities, and moving boundaries; (3) matrix-free implementation of efficient iterative and explicit–implicit finite element schemes; (4) balancing viscosity (version of the stabilized Petrov–Galerkin method); (5) exponential adjustment of physical viscosity coefficients; and (6) stepwise correction of solutions for providing their monotonicity and conservativeness.

Keywords: matrix-free finite element method, exponential adjustment of physical viscosity, overlapping adaptive meshes, fluid flow, large elastoplastic deformations.

DOI: 10.1134/S0965542516060105

1. OVERLAPPING MESHES

Mesh generation for a solution domain of complex (and possibly time-varying) geometry is a complicated and expensive problem, which has been extensively addressed since the 1960s [1]. In addition to the difficulties associated with the design of methods intended, for example, for mapping-based mesh generation, much effort is required to prepare input data on the geometry of the domain and to specify meshes on its boundary. This problem can be substantially simplified by applying the method of overlapping meshes. Specifically, we first introduce a structured mesh that covers (possibly with a margin) the domain under consideration. For example, this can be a bordering box filled with smaller box cells. The boundaries are described using additional overlapping meshes. The nodes and cells covered with overlapping meshes and, hence, lying outside the allowed domain of the continuous medium motion are eliminated from the computation. In many cases, this elimination can be performed in a different manner, namely, by setting analytical conditions, by applying time-varying continuous marker functions taking certain values in the allowed domain of motion, or by using Lagrangian discrete markers moving with the medium (see [1]). In this paper, the geometry of the domain is described by overlapping meshes. The origin of the overlapping mesh method can be traced back to half-century old works on numerical methods, for example, to [2, 3]. At that time, the description of boundaries with the help of overlapping meshes failed to provide sufficient accuracy of numerical solutions, which had to be found on coarse meshes because of the low storage capacity and performance of used computers. Attempts to improve the accuracy by applying better solution approximations near boundaries described by the intersection of bordering and overlapping meshes led to substantial complications in algorithms (see, e.g., the set of types of fractional cells introduced in [4]). At present, the situation has changed cardinally. Thanks to the considerably increased power of modern computers (even personal!), high-resolution meshes can be applied in practice. With the use of moving adaptive meshes of high resolution, the overlapping mesh method ensures sufficient accuracy and becomes progressively more attractive.

2. ADAPTIVE ARBITRARILY MOVING MESHES

The idea of using arbitrarily moving adaptive meshes also has a long history, which is described in [1–17]. Such moving meshes are adapted to solution features and, becoming finer, reduce the approximation errors near the outer, contact, and phase boundaries, as well as in high-gradient zones, such as shock waves and boundary layers. To control the adaptation to the solution, a *monitor function* is introduced into the equations for mapping-based mesh generation. The monitor function is defined so that it has peaks (large positive values) in high-gradient areas of the solution and indicates that the spatial mesh size in these areas has to be reduced to suppress the approximation errors. An ideal approach would be to define monitor functions based on approximation error estimates, but, in actual problems, such estimates are rather inaccurate, so monitor functions are most frequently defined relying on an analysis of the solution method as applied to simplified test problems. It was noted in [18] that a gradual improvement on the nonlinear partial differential equations used for mapping-based mesh generation eventually leads to nonlinear elasticity equations:

$$\mathbf{x} = \mathbf{x}(\tilde{\mathbf{x}},t){:}\ \ \min\int_V \phi(\tilde{\boldsymbol{\varepsilon}},\tilde{T})dV, \quad \mathbf{x}\big|_{\tilde{x}\in\tilde{V}} = \mathbf{x}_*(\tilde{\mathbf{x}},t). \tag{2.1}$$

Here, $\tilde{\mathbf{x}} \in \tilde{V}$ is the preimage of the solution domain (arbitrarily moving coordinates), $\mathbf{x}$ is the actual configuration of the solution domain (the current grid in the solution domain), $\mathbf{x} = \mathbf{x}(\tilde{\mathbf{x}},t)$ are the trajectory of moving coordinates, $\tilde{\boldsymbol{\varepsilon}} = 0.5(\tilde{\mathbf{F}}^T \times \tilde{\mathbf{F}} - \mathbf{I})$ is the Green finite strain tensor for the coordinate medium, $\tilde{\mathbf{F}} = \tilde{\nabla} \otimes \mathbf{x}$ is the gradient strain tensor, $\mathbf{I}$ is the unit tensor, $\tilde{\nabla}$ is the operator of spatial differentiation in moving coordinates, $\phi \geq 0$ is the strain energy, $\tilde{T} = \|\tilde{\nabla} \otimes \mathbf{y}\|$ is a monitor function (“antitemperature”), and $\mathbf{y}$ is the vector of unknown functions for the material medium. The moving grid is treated as an isotropic thermoelastic medium. Its deformations are determined by minimizing the energy functional (see (2.1)). In the simplest form, according to the two-constant theory of isotropic nonlinear thermoelastic media, this functional is defined as

$$\min\int_{\tilde{V}} [\tilde{K}[(I_{3\varepsilon} - 1)/2 + I_{1\varepsilon}\tilde{T}] + 2\tilde{\mu}I_{2\varepsilon}]J_*^{-\alpha}d\tilde{V}. \tag{2.2}$$

Here, $\tilde{K}$ is the bulk modulus; $\tilde{\mu}$ is the shear modulus; $I_{1\varepsilon} = \tilde{\boldsymbol{\varepsilon}} : \mathbf{I}$, $I_{2\varepsilon} = (\tilde{\boldsymbol{\varepsilon}} - I_{1\varepsilon}\mathbf{I}/3)^2 : \mathbf{I}$, $I_{3\varepsilon} = J^2$, $J = \det(\tilde{\mathbf{F}})$, and $J_* = \max(J, 10^{-4})$ are strain invariants; and $\alpha \geq 0.5$. Important for successful implementation, the multiplier $J_*^{-\alpha}$ in (2.2) ensures that the generated mesh has no inverted (more precisely, nonconvex) cells even if the initial approximation $\{\mathbf{x}_0\}$ in the iterative solution of problem (2.2) is given with inverted cells. This multiplier creates a barrier on the boundary of the set of meshes with convex cells. When energy functional (2.2) is minimized on the boundary of the solution domain, the constraints on the motion of boundary nodes determined by additional conditions of the problem are taken into account depending on the type of the boundary (contact boundary, phase boundary, free boundary, moving boundary, etc.). The simplest and fairly efficient method for minimizing the energy functional is to transform the nonlinear elliptic elasticity problem for meshes (2.2) into a parabolic problem. For this purpose, additional nonstationary terms with first derivatives with respect to fictitious time are introduced into the grid functional. The resulting parabolic initial–boundary problem is integrated using an explicit time-marching scheme with the initial condition $\mathbf{x}\big|_{t=0} = \mathbf{x}_0$. The variational equation in the auxiliary parabolic problem (the minimum condition for the functional on the grid) becomes

$$\int_{\tilde{V}} [\partial_t\mathbf{x} \cdot \delta\mathbf{x} + (L : \tilde{\nabla}\mathbf{x}) : \tilde{\nabla}\delta\mathbf{x} - f\delta J]d\tilde{V} = 0, \tag{2.3}$$

where L is a fourth-rank tensor and f is a scalar both depending on the deformations and antitemperature in accordance with the expression for ϕ. For the grid problem to be well-posed, the term with a positive definite operator L must be balanced in the norm by the term with multiplier f. It should be stressed that the Jacobian correction term $J_* = \max(\det(\mathbf{F}), 10^{-4})$ is important for successful implementation (see [11, 13, 14]).

The monitor function is called antitemperature, since the grid medium contracts with growing $\tilde{T}$, while a usual elastic medium expands when heated. For example, in the gas dynamics examples considered later, the monitor function is defined to be directly proportional to the divergence of the velocity with a minus sign and has maximum values on shock waves. The above-written nonlinear elasticity equations were successfully used to generate adaptive meshes in [19], where a more detailed description of mesh algorithms of this type and their prehistory can be found.

Below, it will be shown in computations that the joint use of overlapping and adaptive meshes produces good numerical solutions even on relatively coarse meshes.

3. HYBRID SOLUTION METHOD

The hybrid method used relies on the shock-capturing computation of the flow evolution by applying the modified SUPG FEM scheme [20, 21] on arbitrarily moving adaptive and overlapping meshes. The basic features of this method are presented below.

3.1. Conservation laws. These laws (balance equations) are used in a variational form of the Galerkin–Petrov method that is obtained with the use of integration by parts and contains derivatives of the unknown functions of no higher than the first order (see the derivation in [18]):

$$\frac{\partial}{\partial t}\int_V A\delta A dV + \int_V (\mathbf{B} - A\mathbf{\Omega})\cdot\nabla\delta A dV = \int_V f\delta A dV + \int_S (\mathbf{B} - A\mathbf{\Omega})\cdot\mathbf{n}\delta A dS. \tag{3.1}$$

Here, A is the preserved quantity, $\mathbf{B}$ is the flux of A caused by the interaction and chaotic motion of molecules (elasticity, viscosity, and diffusion), $A\mathbf{\Omega}$ is the convective flux of A (caused by ordered motion of the material medium relative to the coordinate medium), f is the source/sink of A, V is the solution domain with boundary S, $\mathbf{\Omega} = \mathbf{u} - \mathbf{w}$ is the convective velocity, and $\mathbf{u}$ and $\mathbf{w}$ are the velocities of the material and coordinate media. The time derivative $\partial/\partial t$ is calculated along the trajectories of arbitrarily moving coordinates (along the trajectories of nodes of the moving grid). Specifically, a given point is a Lagrangian node if $\mathbf{w} = \mathbf{u}$ and an Eulerian node if $\mathbf{w} = 0$.

In (3.1), we set $A = \rho$, $\mathbf{B} = \nu_{\text{art}}\nabla\rho$, and $f = 0$ for the mass conservation law; $A = \rho\mathbf{u}$, $\mathbf{B} = \boldsymbol{\sigma} + \nu_{\text{art}}(\nabla\otimes(\rho\mathbf{u}) + (\nabla\otimes(\rho\mathbf{u}))^{\text{T}})/2$, and $f = \rho\mathbf{g}$ for the momentum conservation law; and $A = \rho e = \rho(U + \mathbf{u}\times\mathbf{u}/2)$, $\mathbf{B} = -\mathbf{q} + \nu_{\text{art}}\nabla(\rho e)$, and $f = \nabla\times(\boldsymbol{\sigma}\cdot\mathbf{u}) + \rho r$ for the energy balance equation. Here, $\rho\mathbf{g}$ are the external body forces, ρr are the external sources/heat sinks, T is the temperature, $U = c_V T$ is the internal energy, and c_V is the heat capacity at constant volume. The computation of the artificial viscosity ν_{art} is described later.

3.2. Constitutive relations. In the computations described below for fluid dynamics and elastoplasticity problems, the constitutive relations for stresses $\boldsymbol{\sigma}$ and heat fluxes $\mathbf{q}$ are written in a unified manner with the use of vectors and tensors in the actual configuration $\{\mathbf{x}\}$. A detailed description for various media can be found in [22].

For example, for compressible fluid flows, the constitutive relations have the form

$$\boldsymbol{\sigma} = -p\mathbf{I} + \boldsymbol{\sigma}_v, \quad \boldsymbol{\sigma}_v = 2\mu_v(\mathbf{e} - (\mathbf{e} : \mathbf{I})\mathbf{I}/3), \quad p = (\gamma - 1)\rho U, \quad \mathbf{q} = -k_T\nabla T,,$$

where p is the pressure, $\boldsymbol{\sigma}_v$ is the deviator of viscous stresses, μ_v is the physical viscosity coefficient, $\mathbf{e} = 0.5(\nabla\otimes\mathbf{u} + (\nabla\otimes\mathbf{u})^{\text{T}})$ is the Euler strain rate tensor, $\mathbf{q}$ is the heat flux, and k_T is the thermal conductivity.

3.3. Approximation of the solution and equations. The numerical method relies only on simple piecewise linear, bilinear, and trilinear solution approximations in cells (finite elements). The integrals of the variational equations are calculated using the simplest Gaussian quadrature rules, which, in the one-dimensional case, correspond to the trapezoidal rule for terms with time derivatives and to the rectangle rule for the other terms of the variational equations. Thus, diagonal unmatched mass matrices [23] are used in the approximation of the terms with time derivatives. As a result, the algorithms used are simple analogues of explicit central-difference schemes. Moreover, the algorithms are flexible and easy to code and modify for various problems.

These simple approximations were chosen because their higher order accurate counterparts require much more effort related to programming, debugging, and modifying algorithms, but, in the authors' view, fail to yield a significant improvement in the accuracy.

3.4. Allowance for additional conditions (constraints). In the shock-capturing computation [24–28], any additional conditions (constraints), for example, contact ones, breakage/consolidation, phase transitions, incompressibility, and other conditions are taken into account by modifying the variational equations with the use of the Lagrange multiplier or penalty function method (in various versions).

3.5. FEM schemes. Depending on the rate of the underlying process, evolution problems are computed by applying explicit, explicit–implicit, or implicit two-level schemes of the first and quasi-second orders of accuracy. For convective terms, we always use an explicit central-difference approximation with artificial viscosity introduced to ensure its stability.

3.6. Implementation of explicit–implicit and implicit schemes. An implicit approximation of the flux terms **B** is used when their explicit approximations require severe time-step constraints or in the case of stationary (or quasi-static in the terminology of solid mechanics) problems. Implicit schemes are implemented iteratively, and the computational algorithm for a single iteration step is completely identical to the computation of a time step in the explicit scheme. Accordingly, the implicit algorithms are very short and simple and do not involve the computation or storage of stiffness matrices or any matrix operations. The computer memory requirements are low, while high speeds of the implicit schemes are achieved by applying the conjugate gradient method [29], which requires a finite number of operations proportional to $N^{3/2}$, where N is the number of unknowns. Preconditioning is based on the diagonal components of the stiffness matrix, which is easy to compute and store. The required computer memory for the conjugate gradient method is $5N$ (irrespective of the number of dimensions). A version of the method is used that does not require that the discrete operator be symmetric or positive. A detailed description of the implicit finite-element algorithms and the conjugate gradient method can be found in [27].

3.7. Conservativeness of the algorithms. This property is ensured by integrating by parts the conservation laws written in the variational Galerkin–Petrov form, followed by a finite-element approximation of the solution. Local conservativeness, i.e., the balance of fluxes between the node-surrounding volumes is justified trivially and follows from the variational form of the equations and the fact that the derivatives with respect to a constant are zero. Global conservativeness follows from correctly specified boundary conditions.

3.8. Specification of artificial viscosity. The time-stepping schemes (in physical or fictitious time at iteration) are entirely similar to two-level central-difference schemes. The stability of explicit central-difference schemes is ensured by introducing artificial viscosity. In the variety of methods available for this purpose [30], after performing numerous tests, we chose the stabilized finite-element Galerkin–Petrov scheme with upwind differences [20, 21]. However, this scheme was used in a simplified version without introducing upwind differences. The artificial viscosity was defined so that the contribution made by the diffusion terms on the boundaries between the node-surrounding volumes was balanced in the norm by the contribution of the other terms from the homogeneous part of the equation. Note that "on the boundaries between the node-surrounding volumes" means "at Gaussian points of the quadrature rules used to integrate the variational equation"; moreover, these points are placed at the centers of finite elements. This simplified version of the stabilized Galerkin–Petrov scheme is called the *balancing-viscosity scheme*. The formulas for computing the artificial viscosity $\nu^{(j)}_{\rm art}$ are given by

$$[\nu^{(1)}_{\rm art}]_k = \alpha_k \frac{d^{(1)}_{\rm 1k}}{d^{(1)}_{\rm 2k}}, \quad [\nu^{(2)}_{\rm art}]_k = \alpha_k \frac{d^{(2)}_{\rm 1k}}{d^{(2)}_{\rm 2k}}, \quad [\nu^{(3)}_{\rm art}]_k = \alpha_k \frac{d^{(3)}_{\rm 1k}}{d^{(3)}_{\rm 2k}},$$

where k is a finite-element index, the time level index is omitted from all variables to simplify the notation, and the parameter α_k is defined as$\alpha_k = 0.5$ if $[\nabla \cdot \mathbf{u}]_k \Delta t < 0.1$ and $\alpha_k = 1.0$ otherwise. The time step Δt is determined by the stability condition (see below). The coefficients $d^{(j)}_{1k}$ and $d^{(j)}_{2k}$ ($j = 1, 2, 3$) are given by the formulas

$$(d^{(1)}_{1k})^2 = ([\nabla \cdot (\rho \boldsymbol{\Omega})]_k)^2/m, \quad (d^{(1)}_{2k})^2 = \sum_{l=1}^{m} ([\nabla \rho]_k \cdot \nabla_{kl})^2,$$

$$(d^{(2)}_{1k})^2 = [\nabla \cdot (\rho \mathbf{u} \boldsymbol{\Omega}) + \nabla p]_k \cdot [\nabla \cdot (\rho \mathbf{u} \boldsymbol{\Omega}) + \nabla p]_k/m,$$

$$(d^{(2)}_{2k})^2 = \sum_{l=1}^{m} ([\nabla(\rho \mathbf{u})]_k \cdot \nabla_{kl}) \cdot ([\nabla(\rho \mathbf{u})]_k \cdot \nabla_{kl}),$$

$$(d_{1k}^{(3)})^2 = ([\nabla \cdot (\rho e\mathbf{\Omega} + p\mathbf{u})]_k)^2/m, \quad (d_{2k}^{(3)})^2 = \sum_{l=1}^{m}([\nabla(\rho e)]_k \cdot \nabla_{kl})^2.$$

The values of the unknown functions $[A]_k$ at the center of each element k are calculated using the formula

$$[A]_k = \sum_{l=1}^{m} s_{kl}^0 A_{J(k,l)},$$

where $s_{kl}^0 = 1/m$, m is the number of nodes in the element, and $J(k,l)$ is the global index of the node with local index l in element k. The spatial derivatives are calculated using the formula

$$[\nabla A]_k = \sum_{l=1}^{m} \nabla_{kl} A_{J(k,l)}.$$

The property ∇_{kl}: $\sum_{l=1}^{m} \nabla_{kl} = 0$ follows from the fact that the derivative of a constant is zero. In zones of high rarefaction degree caused by the flow acceleration for $[\nabla \cdot \mathbf{u}]_k \Delta t \geq 0.1$, the balancing viscosity coefficient is doubled, which is necessary for the stability of the computation at long times.

3.9. Correction of physical viscosity. To improve the accuracy of the solution in boundary layers, the physical viscosity is corrected (reduced with growing artificial viscosity) by using the exponential adjustment method, which was proposed by Samarskii in 1960s and was developed in [31]. The simplest correction has the form

$$\tilde{\mu}_v = \frac{\mu_v{}^2}{\mu_v + \rho\nu_{\text{art}}^{(2)}}, \quad \tilde{k}_T = \frac{k_T^2}{k_T + \rho\nu_{\text{art}}^{(3)}}.$$

3.10. Elimination of small-scale spurious perturbations. Since all elements of the algorithm are justified only as applied to simple test problems in a loose manner, the solutions of problems based on complete equations exhibit spurious oscillations with a wavelength proportional to the spatial mesh size. Such nonphysical (saw-toothed) perturbations can be immediately detected by changes in the sign of the second derivative in the checked coordinate direction at neighboring nodes belonging to a grid edge. The relative orientations of the edge and the coordinate direction are of no matter. These perturbations do not violate the stability of the method (do not grow), but deteriorate the plots of the solution (the contour line patterns become nonsmooth). These perturbations are eliminated by applying the simplest conservative coordinatewise smoothing procedure, which is applied at a few nodes at the end of every step.

3.11. Determination of second derivatives. In the case of piecewise linear approximations of the solution, the derivatives w_{xx} of some solution component w with respect to x are determined as follows. The obvious variational equation

$$\int_V (w_{xx} - \partial^2 w/\partial x^2)\delta w_{xx} dV = 0$$

for the second derivative is integrated by parts to obtain

$$\int_V w_{xx}\delta w_{xx} dV + \int_V \partial w/\partial x \partial w_{xx}/\partial x dV = \int_S \partial w/\partial x \delta w_{xx} n_x dV,$$

where S is the boundary of the solution domain V. For simplicity, the right-hand side of the equation is set to zero, i.e., either the first or the second derivative with respect to x is assumed to vanish on the boundary. If Gaussian quadrature rules with Gaussian integration points placed at grid nodes are applied to the first integral on the left-hand side, then the second derivatives are explicitly determined by the variational equation, so there is no need to convert nondiagonal matrices. In this way, the second derivatives are quickly calculated using piecewise linear functions of the approximate solution. The second derivatives with respect to y and z are calculated in a similar fashion.

3.12. Stability conditions. For explicit schemes, the time step Δt is determined by the stability condition, which has the usual form

$$\Delta t \leq \min_k \beta \frac{h_k}{(c_k + |\mathbf{\Omega}_k| + (D+1)!\max(\mu_v, k_T)/\rho/h_k)},$$

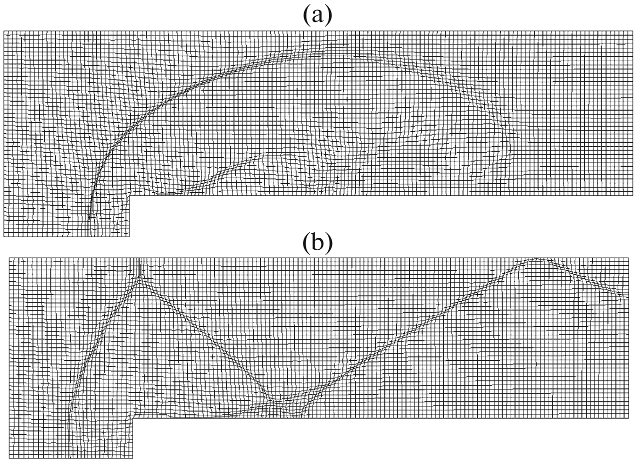

Fig. 1. Adaptive mesh for the ideal gas flow in a step channel with M = 3 and $\gamma = 1.4$ at (a) $t = 0.5$ and (b) $t = 4.0$.

where h_k is the size of the kth finite element; $D = 1, 2, 3$ is the spatial dimension of the problem; c_k is the speed of sound; and β is a safety factor (usually equal to 0.3–0.6). If the diffusion constraint on the time step becomes severe (large coefficients μ_v and k_T) or if the flow in much of the solution domain is subsonic ($|\mathbf{u}| < c_k$), then the stability condition can be substantially relaxed by applying implicit approximations to nonconvective fluxes. Since convective fluxes are explicitly approximated, the convection constraint (the term $|\mathbf{\Omega}_k|$ in the denominator of the stability condition) must be satisfied in any case. In elastoplasticity problems with large deformations, the time step has to be restricted by an additional condition requiring that variations in the norm of the deformations be small as compared with the characteristic deformation corresponding to the yield limit or the tensile strength. For more details, see [27].

The above collection of techniques, which can be justified only by invoking plausible arguments, and overlapping meshes combined with adaptation of an arbitrarily moving grid by applying elastic meshes, produce surprisingly good results. An especially valuable feature of the algorithm is that there is no need to endlessly tune the parameters of the method when the type of problems changes (except for choosing monitor functions).

4. RESULTS

Below, moving adaptive and overlapping meshes are used to solve typical problems with moving material interfaces and discontinuities.

Example 1. Figure 1 shows an adaptive elastic grid at the times $t = 0.5$ and 4.0 in a well-known test problem concerning the ideal gas flow in a channel (3.0 × 1.0) with a step of height 0.4 placed at a distance of 0.6 from the channel inlet. The inlet velocity is characterized by a Mach number of 3.0. The ratio of specific heats is 1.4. The discontinuities are visualized by the refinement of the adaptive grid. The numerical results agree well with the known solution. The only explicit flaw in the adaptation is that the grid is not refined at the contact discontinuity extending from the beginning of the "stem" of the first shock wave reflection from the upper wall. The contact discontinuity could be captured if the velocity divergence term in the expression for the monitor function is added to the norm of the density gradient. However, this has not yet been done.

Fifty years ago, attempts were made to explicitly detect strong discontinuities (jumps) in solutions by using the Rankin–Hugoniot relations for determining their locations and values. These attempts faced insuperable difficulties associated with the fact that, as the discontinuities interacted and reflected from the walls, the number of discontinuities and the number of smooth-solution domains increased sharply, which prevented their explicit consideration. In this respect, the method of adaptive meshes, on which discontinuities are captured with good accuracy in shock-capturing computations, can actually be treated as a good technique for achieving the half-century-standing goal—the numerical computation of flows with multiple discontinuities.

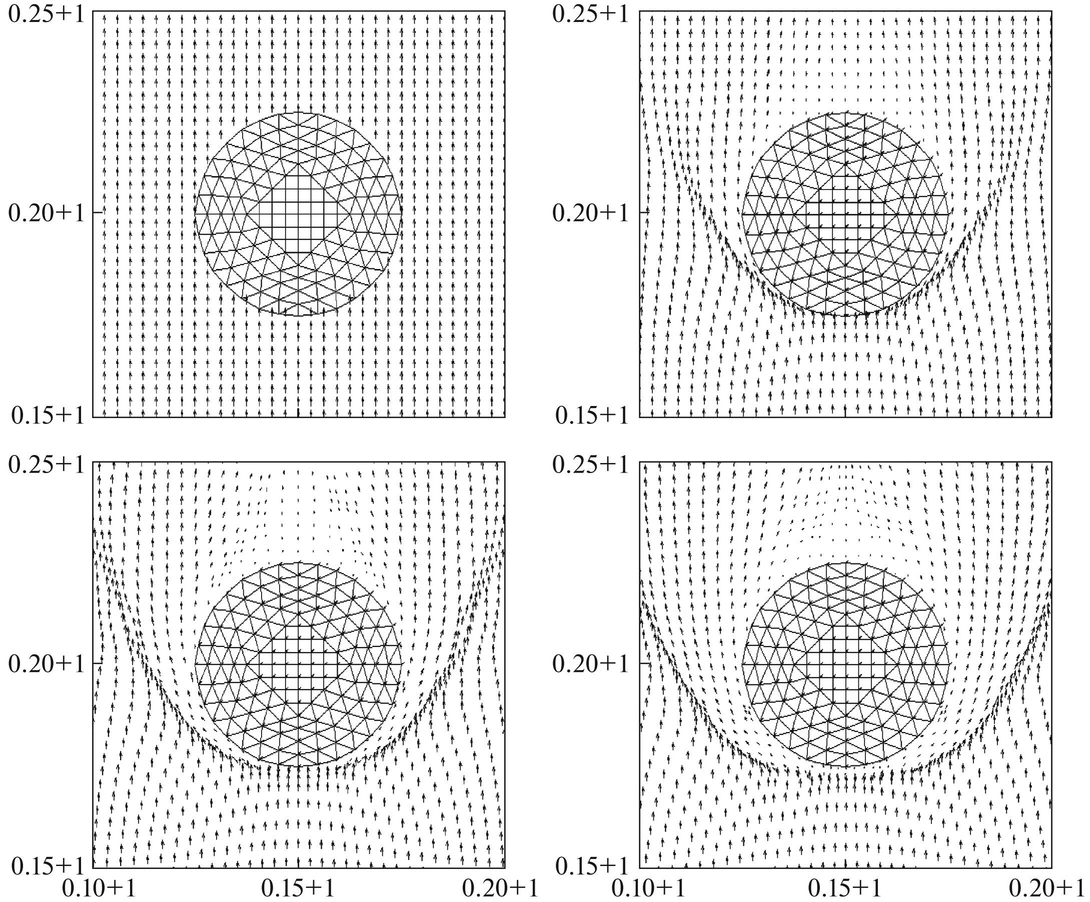

Fig. 2. Velocity field near the overlapped cylinder at $t = 0, 0.1, 0.2$, and 0.3.

Example 2. It shows the application of overlapping meshes to the computation of the flow past two cylinders. An advantage was revealed even prior to the computations. Specifically, the solution domain of complex multiply connected geometry was easily described using overlapping meshes. A fragment of the grid with an overlapping mesh covering one of the cylinders is shown in Fig. 2.

The overlapping meshes show the locations of the cylinders. The locations and sizes of the cylinders can easily be varied by varying several input parameters (the center's location and the radius).

The adaptive grid (see Fig. 3) not only captures the solution discontinuities, but also adapts to the shape of the flow region boundaries described by the overlapping meshes. Figure 4 presents contour lines of the local Mach number at $t = 0.45$ and 7.0 (the number of lines and their range are shown in the figure).

The bordering grid consists of only 160 × 160 square cells. The size of the solution domain is 5.0 × 5.0. The radii of the cylinders are 0.2. The discontinuities and their interaction are perfectly computed.

Example 3. Figure 5 compares (a) our results with (b) those of [21], namely, lines of constant pressure for the ideal gas flow in a channel with a parabolic arch for $M = 1.4$ and $\gamma = 1.4$. The solution domain is 3.0 × 1.0 in size. The results of [21] were obtained on a 184 × 60 grid (without adaptation), while an adaptive 240 × 80 grid was used in our computation. The arch equation is

$$1.0 \le x \le 2.0: \quad y = 0.05(1-(2(x-1.5))^2),$$

where x and y are the horizontal and vertical coordinates in the solution domain.

Example 4. Figure 6 shows the behavior of a moving adaptive mesh in the elastoplastic problem of forging a turbine blade from a cylindrical ingot by using rigid dies. In the entire process, the mesh nodes moved automatically so that the sizes of all mesh cells were held identical whenever possible to avoid a catastrophic decrease in the time step due to the reduced cell size in the compression zones in the computations on Lagrangian meshes moving with the medium. To provide the identity of the volumes of grid cells,

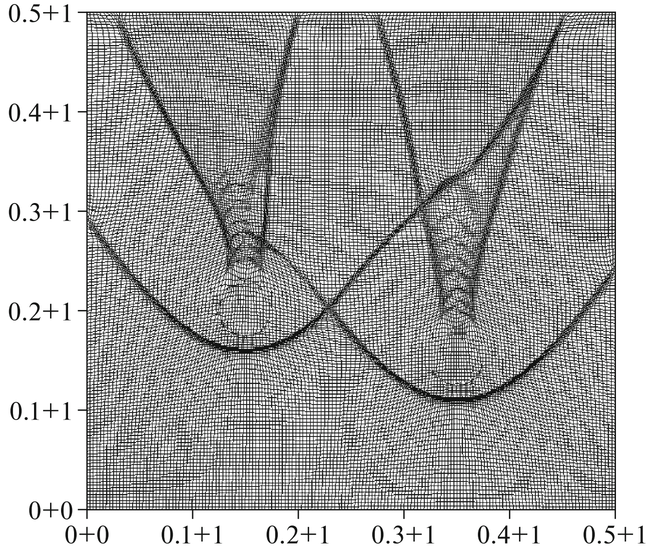

Fig. 3. Bordering adaptive mesh at $t = 7.0$.

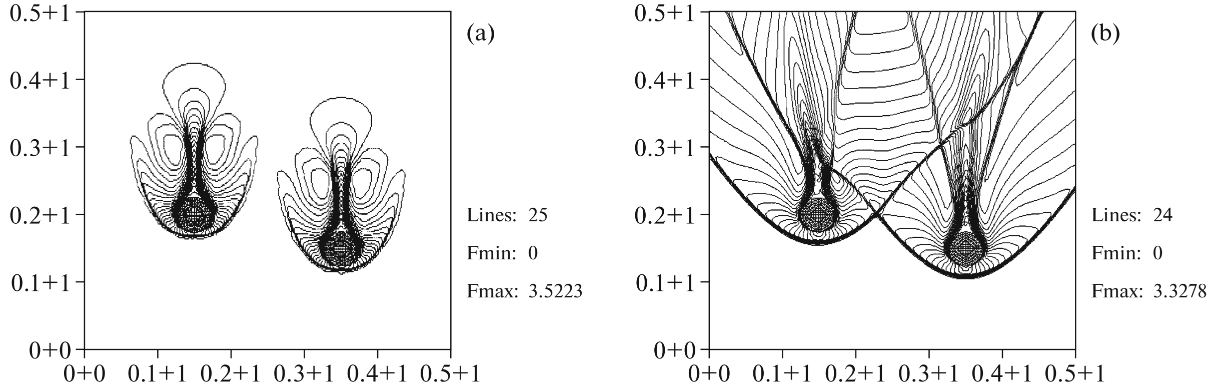

Fig. 4. Contour lines of the local Mach number at (a) $t = 0.45$ and (b) $t = 7.0$.

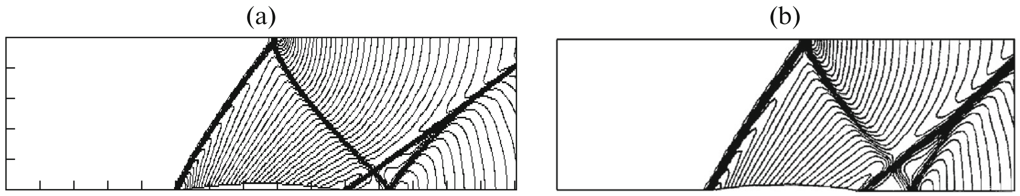

Fig. 5. Contour lines of pressure for the ideal gas flow in a channel with a parabolic arch: (a) present computation and (b) computation from [21] for $M = 1.4$, $\gamma = 1.4$, and $t = 8.0$.

in the equations for elastic meshes the bulk modulus was set 100 times larger than the shear modulus and no monitor function was used. The constitutive relations were specified according to the Prandtl–Reuss theory of plastic flow:

$$\boldsymbol{\sigma} = -pI + \boldsymbol{\sigma}', \quad d\boldsymbol{\sigma}'/dt = 2\mu(\mathbf{e}' - H(F)\boldsymbol{\lambda} : \boldsymbol{\sigma}'), \quad p = K\rho/\rho_0(\rho/\rho_0 - 1), \quad \mathbf{q} = 0,$$

where μ and K are the shear and bulk elastic moduli, H is the Heaviside step function, the tensor $\boldsymbol{\lambda}$ is determined by the von Mises yield condition $F = \boldsymbol{\sigma}' : \boldsymbol{\sigma}' - k_\sigma^2 = 0$, k_σ is the yield surface radius, and $\boldsymbol{\sigma}'$ and $\mathbf{e}'$ are the deviatoric stress and strain rate tensors.

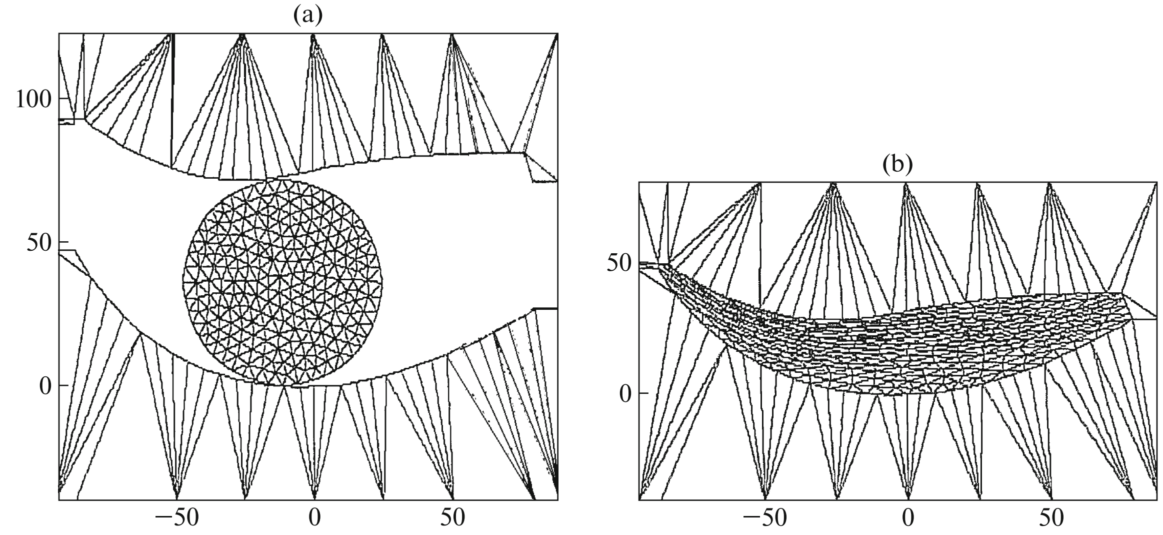

Fig. 6. Elastoplastic problem of forging a turbine blade with dies: adaptive meshes at the (a) beginning and (b) end of the process.

Since the dies were rigid, the meshes in them could be specified as coarse but sufficient for the working boundary of the dies be accurately given. To prevent the cylindrical ingot from slipping from under the dies, low friction proportional to the contact pressure was introduced on the contact surfaces. The problem was solved in dimensionless variables. The stresses, elastic moduli, and yield surface radius were divided by their yield point values. The velocities were divided by the speed of sound. The dimensionless values of the spatial coordinates are clear from the figures of the grid used. In the computations, we used $K = 833$, $\mu = 384$, and $k_\sigma = 0.816$. The speed of the upper die was one-hundredth of the speed of sound, while the lower die was fixed.

Note that the additional terms corresponding to the Jaumann derivatives (objective tensor time derivatives) [16], which are usually used in the constitutive relations for stress, have no noticeable effect on the solution in the given problem.

5. CONCLUDING REMARKS

All components of the numerical approach presented have been repeatedly discussed by various authors (see the overview in [1]). Here, we only showed how attractive the combined use of the above-described collection of computational techniques is for improving the quality of numerical solutions. The numerical algorithms presented are not intended to be optimal, but they were shown to have serious positive qualities. Numerical results for two-dimensional problems were demonstrated. Debugging for three-dimensional problems has been performed, but results worthy of mention have not yet been obtained because of the insufficient speed of the personal computers used.

ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Basic Research, project no. 15-08-02392-a.

REFERENCES

1. N. G. Burago and V. N. Kukudzhanov, “A review of contact algorithms,” Mech. Solids **40** (1), 35–71 (2005).
2. *Methods in Computational Physics: Advances in Research and Applications*, Vol. 3: *Fundamental Methods in Hydrodynamics*, Ed. by B. Alder, S. Fernbach, and M. Rotenberg (Academic, New York, 1964; Mir, Moscow, 1967).
3. *Numerical Methods in Fluid Mechanics*, Ed. by O .M. Belotserkovskii (Mir, Moscow, 1973) [in Russian].
4. O. M. Belotserkovskii and Yu. M. Davydov, *Large-Particle Method in Gas Dynamics* (Nauka, Moscow, 1982) [in Russian].
5. S. K. Godunov and G. P. Prokopov, “The use of moving meshes in gas-dynamical computations,” USSR Comput. Math. Math. Phys. **12** (2), 182–195 (1972).

6. V. M. Kovenya and N. N. Yanenko, *Splitting Methods in Gas Dynamics Applications* (Nauka, Novosibirsk, 1981) [in Russian].

7. L. V. Kruglyakova, A. V. Neledova, V. F. Tishkin, and A. Yu. Filatov, "Unstructured adaptive meshes for problems in mathematical physics," Mat. Model. **10** (3), 93–116 (1997).

8. A. N. Gil'manov, *Adaptive Grid Methods in Gas Dynamics Applications* (Fizmatlit, Moscow, 2000) [in Russian].

9. G. P. Prokopov, Preprint IPM RAN (Keldysh Inst. of Applied Mathematics, Russian Academy of Sciences, Moscow, 2006).

10. V. D. Liseikin, "The construction of structured adaptive grids—a review," Comput. Math. Math. Phys. **36** (1), 1–32 (1996).

11. S. A. Ivanenko, "Variational methods for adaptive grid generation," Comput. Math. Math. Phys. **43** (6), 793–806 (2003).

12. V. G. Bazhenov, V. L. Kotov, and S. V. Zefirov, "Matching of various difference schemes in nonstationary continuum mechanics problems by the method of overlapping meshes," Vestn. Nizhegor. Univ. Ser. Mekh. **7** (1), 134–140 (2006).

13. B. N. Azarenok, *On Adaptive Moving Mesh Generation* (Vychisl. Tsentr Ross. Akad. Nauk, Moscow, 2007) [in Russian].

14. *Advances in Grid Generation*, Ed. by O. V. Ushakova (Nova Science, New York, 2007).

15. A. V. Minakov, A. A. Gavrilov, and A. A. Dekterev, "Numerical algorithm for solving three-dimensional fluid dynamics problems with moving solid bodies and free surface," Sib. Zh. Ind. Mat. **11**, (4), 94–104 (2008).

16. V. N. Kukudzhanov, *Numerical Continuum Mechanics* (Fizmatlit, Moscow, 2008; Walter de Gruyter, Berlin, 2013).

17. V. D. Liseikin, *Grid Generation Methods* (Springer, Berlin, 2010).

18. N. G. Burago, "Formulation of basic equations for continuum mechanics in moving adaptive coordinates, in *Numerical Methods in Solid Mechanics*, Ed. by G. I. Pshenichnov (Vychisl. Tsentr Akad. Nauk SSSR, Moscow, 1984), pp. 32–49 [in Russian].

19. N. G. Burago and S. A. Ivanenko, "Application of elasticity equations to adaptive grid generation," *Proceedings of All-Russia Conference on Applied Geometry, Grid Generation, and High-Performance Computations, Moscow, Computing Center of the Russian Academy of Sciences, June 28–July 1, 2004*, Ed. by V. A. Garanzha (Vychisl. Tsentr Ross. Akad. Nauk, Moscow, 2004), pp. 107–118.

20. A. N. Brooks and T. J. R. Hughes, "Streamline upwind Petrov–Galerkin formulation for convection dominated flows," Comput. Methods Appl. Mech. Eng. **32**, 199–259 (1982).

21. G. J. Le Beau, S. E. Ray, S. K. Aliabadi, and T.-E. Tezduyar, "SUPG finite element computation of compressible flows with the entropy and conservation variables formulations," Comput. Methods Appl. Mech. Eng. **104**, 397–422 (1993).

22. N. G. Bourago, A. I. Glushko, and A. N. Kovshov, "Thermodynamic method for constructing constitutive relations for models of continuous media," Mech. Solids **35** (6), 1–9 (2000).

23. G. Strang and G. J. Fix, *An Analysis of the Finite Element Method* (Prentice Hall, Englewood Cliffs, N.J., 1973; Mir, Moscow, 1977).

24. N. G. Burago, "Modeling of elastoplastic fracture," Vychisl. Mekh. Sploshn. Sred **1** (4), 5–20 (2008).

25. N. G. Burago and I. S. Nikitin, http://engjournal.ru/catalog/mathmodel/hidden/883.html

26. N. G. Burago, I. S. Nikitin, and V. L. Yakushev, "Numerical modeling of sintering of powder materials under moving laser actions," Vestn. Kibern., No. 3, 12–23 (2014).

27. N. G. Burago and V. N. Kukudzhanov, "Numerical solution of elastic plastic problems by FEM: ASTRA software package," in *Computational Solid Mechanics* (Nauka, Moscow, 1991), Vol. 2, pp. 78–122 [in Russian].

28. A. A. Samarskii and B. D. Moiseenko, "An economic continuous calculation scheme for the Stefan multidimensional problem," USSR Comput. Math. Math. Phys. **5** (5), 43–58 (1965).

29. J. H. Wilkinson and C. Reinsch, *Handbook for Automatic Computation*, Vol. 2: *Linear Algebra* (Springer-Verlag, Berlin, 1971; Mashinostroenie, Moscow, 1976).

30. A. G. Kulikovskii, N. V. Pogorelov, and A. Yu. Semenov, *Mathematical Aspects of Numerical Solution of Hyperbolic Systems* (Fizmatlit, Moscow, 2001; Chapman and Hall/CRC, London, 2001).

31. E. P. Doolan, J. J. H. Miller, and W. H. A. Schilders, *Uniform Numerical Methods for Problems with Initial and Boundary Layers* (Boole, Dublin, 1980; Mir, Moscow, 1983).

Translated by I. Ruzanova