ISSN 0965-5425, Computational Mathematics and Mathematical Physics, 2017, Vol. 57, No. 8, pp. 1335–1359. © Pleiades Publishing, Ltd., 2017.
Original Russian Text © I.V. Egorov, A.V. Novikov, A.V. Fedorov, 2017, published in Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 2017, Vol. 57, No. 8, pp. 1347–1376.

Direct Numerical Simulation of the Laminar–Turbulent Transition at Hypersonic Flow Speeds on a Supercomputer

I. V. Egorov[a], A. V. Novikov[b]*, and A. V. Fedorov[b]

[a] *Zhukovsky Central Aerohydrodynamic Institute (TsAGI), National Research Center "Zhukovsky Institute", Zhukovskii, Moscow oblast, 140180 Russia*

[b] *Moscow Institute of Physics and Technology, Dolgoprudnyi, Moscow oblast, 141701 Russia*

**e-mail: AndrewNovikov@yandex.ru*

Received May 18, 2015; in final form, September 26, 2016

Abstract—A method for direct numerical simulation of three-dimensional unsteady disturbances leading to a laminar–turbulent transition at hypersonic flow speeds is proposed. The simulation relies on solving the full three-dimensional unsteady Navier–Stokes equations. The computational technique is intended for multiprocessor supercomputers and is based on a fully implicit monotone approximation scheme and the Newton–Raphson method for solving systems of nonlinear difference equations. This approach is used to study the development of three-dimensional unstable disturbances in a flat-plate and compression-corner boundary layers in early laminar–turbulent transition stages at the free-stream Mach number M = 5.37. The three-dimensional disturbance field is visualized in order to reveal and discuss features of the instability development at the linear and nonlinear stages. The distribution of the skin friction coefficient is used to detect laminar and transient flow regimes and determine the onset of the laminar–turbulent transition.

Keywords: direct numerical simulation, laminar–turbulent transition, hypersonic flows, boundary layer.

DOI: 10.1134/S0965542517080061

INTRODUCTION

The problem of computing the laminar–turbulent transition (LTT) in hypersonic flows over bodies is a major task in high-speed aerodynamics. LTT leads to a significant increase in heat transfer and the skin friction drag of aircraft and affects the performance of propulsion and control units. Solving this problem requires a deep insight into the physical mechanisms of the transition to turbulence. In the case of small external disturbances, which are typical of the flight conditions, the LTT consists of three basic stages [1]: receptivity to external disturbances; the growth of unstable modes, such as first and second Mack ones, cross flow instability, and Görtler vortices; and the nonlinear breakdown of disturbances, which leads to a fully turbulent flow regime. The physical mechanisms of these stages can be investigated experimentally or numerically.

The most suitable approach to systematic studies of linear and especially nonlinear LTT stages is based on "controlled" experiments. However, there are relatively few publications concerning such experiments at super- and hypersonic speeds. Thermoanemometric measurements of wave trains generated by a harmonic point source in a flat-plate boundary layer at a Mach number of 2 were conducted by Kosinov et al. [2, 3] (their results were confirmed by direct numerical simulation [4]). Controlled disturbances produced by a glow discharge on a cone at M = 4 were experimentally studied in [5]. Experimental data obtained on a sharp cone at M = 5.95 with a glow discharge used as a harmonic point source of disturbances were presented in [6, 7]. Wave packets and turbulent spots in a hypersonic nozzle boundary layer within the Boeing/AFOSR Mach 6 Quiet Tunnel (BAM6QT) were investigated in [8, 9]; disturbances were generated by controlled spark and glow discharges. Controlled temperature disturbances produced by a powerful laser in the BAM6QT were analyzed in [10]. All these experiments yield rather limited data on disturbance fields, since high-frequency fluctuations are technically difficult to measure at numerous points under severe super- and hypersonic flow conditions.

In contrast to physical experiments, direct numerical simulation (DNS) provides complete information on spatial and temporal disturbance fields, so that various LTT mechanisms can be identified and studied in detail. Moreover, with the use of DNS, all LTT stages can be computed in a holistic manner by

solving the full unsteady Navier–Stokes equations without imposing any constraints on the basic (unperturbed laminar) flow and the disturbance amplitude. However, at late nonlinear stages of LTT with natural Reynolds numbers, DNS requires fine three-dimensional grids (with cell sizes corresponding to the Kolmogorov scale), which consist of hundreds of million nodes. Due to the intensive development of multiprocessor systems and modern methods for parallel computations, only recently has it become possible to perform such numerical experiments for hypersonic boundary layers and to approach fully turbulent flows. For example, unstable controlled disturbances in simple configurations, namely, a plate and a cone with a zero angle of attack were DNS simulated in [11, 12] by applying specialized numerical approaches. More specifically, the basic method in [11] was based on a shock-capturing finite-difference scheme for computing the steady flow over a leading edge with a separated shock wave, while a higher-order shock-fitting scheme for computing unsteady disturbances was used downstream. It was noted that this approach faces difficulties in the case of disturbances with extremely large amplitudes and high gradients. In [12] a monotone scheme was first used to compute the complete steady flow field, out of which a near-wall domain was then cut off and the propagation of disturbances in it was simulated by applying a hybrid numerical approach with a high-order finite-difference scheme and the spectral method.

In practice, it is convenient for DNS to use a single universal numerical method in order to calculate both steady flow fields with all spatial inhomogeneities (shock waves, separations, high pressure gradients) and unsteady disturbances without imposing amplitude constraints.

In this work, hypersonic flows are computed by applying DNS based on a fully implicit finite volume method (third-order accurate in space and second-order accurate in time) with a quasi-monotone Godunov-type scheme. This approach is especially efficient when the governing equations of the problem are rather stiff as in the case of hypersonic flows, where the computational domain contains systems of shock waves [13] or when nonequilibrium physicochemical processes are simulated [14]. Unfortunately, the monotonicity of the scheme implies that it is highly dissipative, which leads to a nonphysical decrease in the amplitudes of disturbances propagating in the computational domain. However, on fairly fine grids, numerical viscosity overall does not distort the basic physics of instability development. Previously, the method proposed was used to simulate the receptivity [15] and stability [16, 17] of a hypersonic boundary layer flow in the two-dimensional case and to develop passive methods for stabilizing a hypersonic boundary layer with the help of a porous coating [18] and a wavy surface with a series of local separations [19].

In the above-indicated works, the method was implemented on a single processor. Accordingly, the maximum grid sizes were technically limited. In the case of efficient parallel computations, the constraint on the number of grid nodes can be overcome by using more computational units (processors and memory). This can be performed without much difficulty in view of the modern trends in the development of computers. Thus, it seems reasonable to simulate LTT by applying a universal and robust low-order method on a high-performance supercomputer, rather than using high-order specialized methods, which save resources, but require labor-consuming adaptation and apply to a limited class of flows.

In this work, we describe a parallel implementation of the considered numerical method intended for computations on cluster-type multiprocessor supercomputers with distributed memory. The mathematical formulation of the problem of simulating viscous compressible gas flows is presented. We describe a method for the numerical integration of the Navier–Stokes equations and an algorithm for parallel computations and give some results obtained by applying the method to the DNS of an initial LTT stage. The method was implemented using the in-house solver HSFlow (High Speed Flow) with parallel algorithms employed at all levels for efficient computations on cluster supercomputers with a large number of processors and distributed memory. The solver was verified by simulating wave packet propagation in a plate boundary layer at the free-stream Mach number $M_\infty = 5.35$ and comparing the results with those of [20]. As an example of DNS simulation of the nonlinear LTT stage in a more complicated flow, we study the development of three-dimensional disturbances in a separated boundary-layer flow in a compression corner of 5.5° at $M_\infty = 5.373$. We demonstrate the disturbance fields on the body surfaces, three-dimensional disturbance vortex structures, and fluctuation spectra, which are used to reveal and discuss features of the instability development at the linear and nonlinear transition stages. Additionally, the onset of LTT is determined by analyzing the distributions of the skin friction coefficients in the laminar and transient regions. The velocity profiles in the transient boundary layer are also shown.

1. PROBLEM FORMULATION

In continuum mechanics, the gas motion is generally described by the unsteady three-dimensional Navier–Stokes equations, which also serve as a basis for the direct numerical simulation of turbulent flows.

In an arbitrary curvilinear coordinate system ξ, η, ζ, where $x = x(\xi,\eta,\zeta)$, $y = y(\xi,\eta,\zeta)$, and $z = z(\xi,\eta,\zeta)$ are Cartesian coordinates, the Navier–Stokes equations are written in conservation form as

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{E}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} + \frac{\partial \mathbf{F}}{\partial \zeta} = \mathbf{0}. \tag{1}$$

Here, $\mathbf{Q}$ is the vector of conservative dependent variables of the problem and $\mathbf{E}$, $\mathbf{G}$, and $\mathbf{F}$ are the flux vectors in the curvilinear coordinate system. The vectors $\mathbf{E}$, $\mathbf{G}$, and $\mathbf{F}$ are related to the corresponding Cartesian vectors $\mathbf{E}_c$, $\mathbf{G}_c$, and $\mathbf{F}_c$ by the formulas

$$\begin{gathered}\mathbf{Q} = J\mathbf{Q}_c, \quad \mathbf{E} = J\left(\mathbf{E}_c\frac{\partial \xi}{\partial x} + \mathbf{G}_c\frac{\partial \xi}{\partial y} + \mathbf{F}_c\frac{\partial \xi}{\partial z}\right),\\ \mathbf{G} = J\left(\mathbf{E}_c\frac{\partial \eta}{\partial x} + \mathbf{G}_c\frac{\partial \eta}{\partial y} + \mathbf{F}_c\frac{\partial \eta}{\partial z}\right), \quad \mathbf{F} = J\left(\mathbf{E}_c\frac{\partial \zeta}{\partial x} + \mathbf{G}_c\frac{\partial \zeta}{\partial y} + \mathbf{F}_c\frac{\partial \zeta}{\partial z}\right),\end{gathered} \tag{2}$$

where $J = \partial(x,y,z)/\partial(\xi,\eta,\zeta)$ is the Jacobian of the transformation.

The curvilinear coordinate system (ξ,η,ζ) is used for discretization on a uniform grid. Specifically, an arbitrary grid given in Cartesian coordinates is mapped onto a uniform grid in curvilinear coordinates. The Cartesian components of the vectors $\mathbf{E}_c$, $\mathbf{G}_c$, and $\mathbf{F}_c$ for the three-dimensional Navier–Stokes equations are given by

$$\mathbf{Q}_c = \begin{Vmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ \rho e\end{Vmatrix}, \quad \mathbf{E}_c = \begin{Vmatrix}\rho u\\ \rho u^2 + p + \tau_{xx}\\ \rho uv + \tau_{xy}\\ \rho uw + \tau_{xz}\\ \rho uH + I_x\end{Vmatrix}, \quad \mathbf{G}_c = \begin{Vmatrix}\rho v\\ \rho uv + \tau_{xy}\\ \rho v^2 + p + \tau_{yy}\\ \rho vw + \tau_{yz}\\ \rho vH + I_y\end{Vmatrix}, \quad \mathbf{F}_c = \begin{Vmatrix}\rho w\\ \rho wu + \tau_{xz}\\ \rho wv + \tau_{yz}\\ \rho w^2 + p + \tau_{zz}\\ \rho wH + I_z\end{Vmatrix},$$

where ρ is the gas density; u, v, w are the Cartesian components of the velocity $\mathbf{V}$; p is the pressure; $e = h - \frac{p}{\rho} + \frac{1}{2}(u^2 + v^2 + w^2)$ is the total energy per unit volume; $H = h + \frac{1}{2}(u^2 + v^2 + w^2)$ is the total enthalpy; $h = c_pT$ is the static enthalpy; T is the temperature; c_p is the specific heat capacity at constant pressure; and $\boldsymbol{\tau}$ is the viscous stress tensor, which is symmetric and is related to the strain rate tensor $\mathbf{s}$ by the linear dependence $\boldsymbol{\tau} = -\mu\mathbf{s}$. For a compressible gas, the components of $\mathbf{s}$ have the form

$$\mathbf{s}_{xx} = 2\frac{\partial u}{\partial x} - \frac{2}{3}\operatorname{div}\mathbf{V}, \quad \mathbf{s}_{yy} = 2\frac{\partial v}{\partial y} - \frac{2}{3}\operatorname{div}\mathbf{V}, \quad \mathbf{s}_{zz} = 2\frac{\partial w}{\partial z} - \frac{2}{3}\operatorname{div}\mathbf{V},$$

$$\mathbf{s}_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}, \quad \mathbf{s}_{xz} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}, \quad \mathbf{s}_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y},$$

and the heat flux $\mathbf{I}$ is defined by the expression

$$\mathbf{I} = -\lambda\operatorname{grad}(T) + \boldsymbol{\tau}\mathbf{V},$$

where μ and λ are the molecular viscosity and thermal conductivity, respectively.

System (1) is closed by an equation of state and dependences of the transfer coefficients on temperature and pressure. In this study, we use the ideal gas model with the equation of state

$$p = \rho RT/M,$$

where R is the universal gas constant and M is the molecular weight of the gas. The molecular viscosity is assumed to depend only on temperature and is calculated according to Sutherland's law:

$$\frac{\mu}{\mu_\infty} = \frac{1 + \dfrac{T_\mu}{T_\infty}}{\dfrac{T}{T_\infty} + \dfrac{T_\mu}{T_\infty}} \left(\frac{T}{T_\infty}\right)^{\frac{3}{2}},$$

where $T_\mu = 110.4$ K for air. The Prandtl number $\mathrm{Pr} = \mu c_p/\lambda$ is assumed to be constant.

For numerical integration, the Navier–Stokes equations are nondimensionalized. The Cartesian coordinates $x = \bar{x}L$, $y = \bar{y}L$, $z = \bar{z}L$ are normalized by the characteristic length L; the time $t = \bar{t}L/V_\infty$, by the characteristic time L/V_∞; the velocity components $u = \bar{u}V_\infty$, $v = \bar{v}V_\infty$, and $w = \bar{w}V_\infty$, by the magnitude of the free-stream velocity V_∞; the pressure $p = \bar{p}(\rho_\infty V_\infty^2)$, by the doubled free-stream dynamic pressure; and the other gasdynamic variables, by their free-stream values. The overbar denotes dimensionless variables, and the infinity symbol denotes free-stream values. After this nondimensionalization, the following basic similarity parameters appear in the Navier–Stokes equations: the ratio of specific heats $\gamma = c_p/c_v$, the free-stream Mach number $\mathrm{M}_\infty = V_\infty/a_\infty$ (a is the speed of sound), the Reynolds number $\mathrm{Re}_\infty = (\rho_\infty V_\infty L)/\mu_\infty$, and the Prandtl number Pr. Most of the numerical results below are presented in dimensionless variables, and the overbar is omitted for simplicity.

On the wall boundary of the computational domain, we set the no-slip and impermeability conditions $u = v = w = 0$ and the isothermal condition $T = T_w = \mathrm{const}$. On the outer boundary, we impose either Dirichlet conditions with free-stream values of all dependent variables ($u = 1$, $v = w = 0$, $p = 1/(\gamma M^2)$, $T = 1$) or the dependent variables are extrapolated from inside the computational domain, which corresponds to nonreflecting boundary conditions in the supersonic case.

To obtain a steady flow field, the initial approximation can be specified as a uniform free-stream flow, which develops into a laminar steady flow over the body under study in the course of solving the unsteady problem.

2. APPROXIMATION OF THE EQUATIONS

The initial–boundary value problem formulated above was solved numerically using the finite volume method. Its application to the Navier–Stokes equations (1) yields the following discrete analogues of conservation laws:

$$\frac{3\mathbf{Q}_{i,j,k}^{n+1} - 4\mathbf{Q}_{i,j,k}^{n} + \mathbf{Q}_{i,j,k}^{n-1}}{\Delta t} + \frac{\mathbf{E}_{i+\frac{1}{2},j,k}^{n+1} - \mathbf{E}_{i-\frac{1}{2},j,k}^{n+1}}{h_\xi} + \frac{\mathbf{G}_{i,j+\frac{1}{2},k}^{n+1} - \mathbf{G}_{i,j-\frac{1}{2},k}^{n+1}}{h_\eta} + \frac{\mathbf{F}_{i,j,k+\frac{1}{2}}^{n+1} - \mathbf{F}_{i,j,k-\frac{1}{2}}^{n+1}}{h_\zeta} = 0, \tag{3}$$

where n is the time level number; Δt is the time step; and i, j, k and h_ξ, h_η, h_ζ are the node indices and the step sizes in ξ, η, and ζ, respectively.

In a monotone difference scheme, the fluxes at half-integer nodes are computed by solving the corresponding Riemann problem. Mathematically, this problem is reduced to a nonlinear system of algebraic equations. An approximate method for solving this problem can be based on dimensional splitting (with respect to generalized coordinates) and representing some averaged state of the corresponding Jacobian matrix $\mathbf{A}$ (for example, $\mathbf{A} = \partial\mathbf{E}/\partial\mathbf{Q}$ in the ξ direction) in diagonal form $\mathbf{A} = \mathbf{R}\Lambda\mathbf{R}^{-1}$, where Λ is a diagonal matrix whose elements are six eigenvalues of $\mathbf{A}$.

The convective components of the fluxes $\mathbf{E}$, $\mathbf{G}$, and $\mathbf{F}$ at half-integer nodes are approximated using a monotone Godunov-type scheme [20] and the Roe approximate method [21] for solving the Riemann problem. Since the computational formulas for $\mathbf{E}$, $\mathbf{G}$, and $\mathbf{F}$ are similar, we consider only the vector $\mathbf{E}$. On a cell face (at a half-integer node), we have

$$\mathbf{E}_{i+\frac{1}{2}} = \frac{1}{2}\Big[\mathbf{E}(\mathbf{Q}_\mathrm{L}) + \mathbf{E}(\mathbf{Q}_\mathrm{R}) - \mathbf{B}_\mathrm{LR}\Lambda\big(\varphi(\lambda_{\mathrm{LR}i})\big)\mathbf{B}_\mathrm{LR}^{-1}\big(\mathbf{Q}_\mathrm{R} - \mathbf{Q}_\mathrm{L}\big)\Big].$$

Here, the indices L and R denote quantities calculated using the values from neighboring cells adjacent to the considered face. For example, for face $i + 1/2$, the index L corresponds to cell i (on the left), while the index R, to cell $i + 1$ (on the right). The subscript LR denotes quantities computed using specially averaged values from neighboring cells, and $\Lambda(\varphi(\lambda_{\mathrm{LR}i}))$ is a diagonal matrix with entries $\varphi(\lambda_{\mathrm{LR}i})$, where $\lambda_{\mathrm{LR}i}$ are the

eigenvalues of the operator $\mathbf{A}_{LR} = [\partial\mathbf{E}/\partial\mathbf{Q}]_{LR}$ and $\varphi(\lambda)$ is a smoothing function ensuring that entropy in the numerical solution does not decrease. Finally, $\mathbf{B}_{LR}$ and $\mathbf{B}_{LR}^{-1}$ are matrices whose columns are the right and left eigenvectors of $\mathbf{A}_{LR}$, respectively.

In the computation of the eigenvalues and eigenvectors of $\mathbf{A}_{LR}$, the dependent variables in neighboring cells are averaged according to [21]:

$$u_{LR} = \frac{u_L\sqrt{\rho_L} + u_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}}, \quad v_{LR} = \frac{v_L\sqrt{\rho_L} + v_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}}, \quad w_{LR} = \frac{w_L\sqrt{\rho_L} + w_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}},$$

$$H_{LR} = \frac{H_L\sqrt{\rho_L} + H_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}}, \quad a_{LR}^2 = (\gamma-1)\left(H_{LR} - \frac{1}{2}\left(u_{LR}^2 + v_{LR}^2 + w_{LR}^2\right)\right),$$

where a is the local speed of sound.

To ensure the fulfillment of the entropy condition for choosing a physically correct numerical solution, the function $\varphi(\lambda_i)$ is defined as

$$\varphi(\lambda) = \begin{cases} |\lambda|, & |\lambda| > \varepsilon, \\ \dfrac{\lambda^2 + \varepsilon^2}{2\varepsilon}, & |\lambda| \le \varepsilon, \end{cases}$$

where ε is a parameter responsible for the dissipation properties of the difference scheme (in most of the computations, we used $\varepsilon = 0.1$).

To increase the order of accuracy (up to the third), the dependent variables are interpolated to a face of an elementary cell by applying the weighted essentially nonoscillatory (WENO) scheme [22]

$$\mathbf{Q}_R = \omega_{0R}\mathbf{Q}_{0R} + \omega_{1R}\mathbf{Q}_{1R}, \quad \mathbf{Q}_{0R} = \frac{1}{2}(\mathbf{Q}_{i+1} + \mathbf{Q}_i), \quad \mathbf{Q}_{1R} = -\frac{1}{2}\mathbf{Q}_{i+2} + \frac{3}{2}\mathbf{Q}_{i+1};$$

$$\mathbf{Q}_L = \omega_{0L}\mathbf{Q}_{0L} + \omega_{1L}\mathbf{Q}_{1L}, \quad \mathbf{Q}_{0L} = \frac{1}{2}(\mathbf{Q}_i + \mathbf{Q}_{i+1}), \quad \mathbf{Q}_{1L} = -\frac{1}{2}\mathbf{Q}_{i-1} + \frac{3}{2}\mathbf{Q}_i;$$

$$\omega_{0R} = \frac{\alpha_{0R}}{\alpha_{0R} + \alpha_{1R}}, \quad \omega_{1R} = \frac{\alpha_{1R}}{\alpha_{0R} + \alpha_{1R}}; \quad \alpha_{0R} = \frac{d_0}{(\varepsilon + \beta_{0R})^2}, \quad \alpha_{1R} = \frac{d_1}{(\varepsilon + \beta_{1R})^2};$$

$$\omega_{0L} = \frac{\alpha_{0L}}{\alpha_{0L} + \alpha_{1L}}, \quad \omega_{1L} = \frac{\alpha_{1L}}{\alpha_{0L} + \alpha_{1L}}, \quad \alpha_{0L} = \frac{d_0}{(\varepsilon + \beta_{0L})^2}, \quad \alpha_{1L} = \frac{d_1}{(\varepsilon + \beta_{1L})^2};$$

$$\beta_{0R} = (\mathbf{Q}_{i+1} - \mathbf{Q}_i)^2, \quad \beta_{1R} = (\mathbf{Q}_{i+2} - \mathbf{Q}_{i+1})^2;$$

$$\beta_{0L} = (\mathbf{Q}_i - \mathbf{Q}_{i+1})^2, \quad \beta_{1L} = (\mathbf{Q}_i - \mathbf{Q}_{i-1})^2;$$

$$d_0 = \frac{2}{3}, \quad d_1 = \frac{1}{3}, \quad \varepsilon = 10^{-4}.$$

The diffusion components of the fluxes $\mathbf{E}$, $\mathbf{G}$, and $\mathbf{F}$ on faces of an elementary cell are approximated by applying a second-order accurate central difference scheme. The derivatives are computed using the formulas

$$\left.\frac{\partial f}{\partial\xi}\right|_{i+\frac{1}{2},j,k} = \frac{1}{h_\xi}(f_{i+1,j,k} - f_{i,j,k}),$$

$$\left.\frac{\partial f}{\partial\eta}\right|_{i+\frac{1}{2},j,k} = \frac{1}{4h_\eta}(f_{i+1,j+1,k} + f_{i,j+1,k} - f_{i+1,j-1,k} - f_{i,j-1,k}),$$

$$\left.\frac{\partial f}{\partial\varsigma}\right|_{i+\frac{1}{2},j,k} = \frac{1}{4h_\varsigma}(f_{i+1,j,k+1} + f_{i,j,k+1} - f_{i+1,j,k-1} - f_{i,j,k-1}),$$

where f is any of the nonconservative (primitive) dependent variables u, v, w, p or T of the problem.

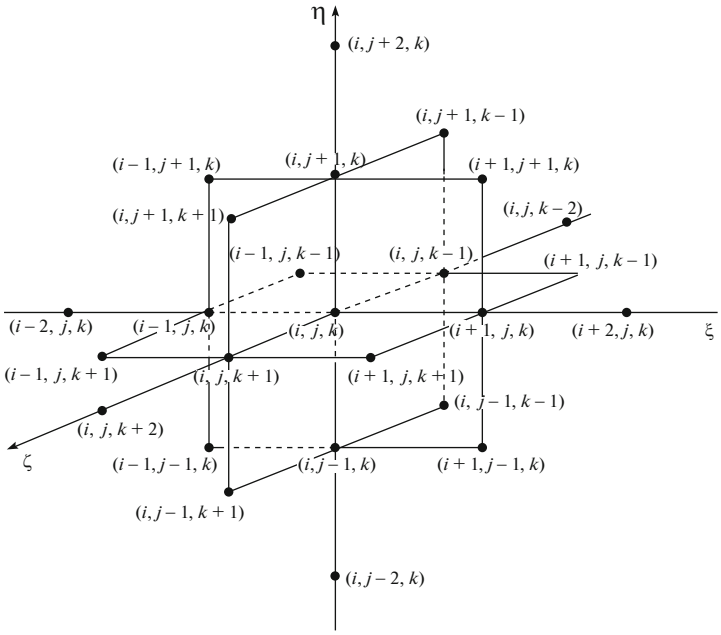

Fig. 1. Stencil of the finite-difference scheme in the three-dimensional case.

The stencil of the finite-difference for approximating the full Navier–Stokes equations consists of 25 points (see Fig. 1). Apparently, the fully implicit nonlinear difference scheme described is unconditionally stable as applied to one-dimensional linear problems. Experience gained from the application of this scheme to the simulation of various two- and three-dimensional flows shows that it is fairly stable in a broad class of practical problems.

3. SOLUTION OF THE NONLINEAR GRID EQUATIONS

After applying the above difference approximation to the Navier–Stokes equations and using the corresponding boundary conditions on a certain grid, the integration of the nonlinear partial differential equations is reduced to solving a system of nonlinear algebraic equations

$$\mathbf{R}(\mathbf{X}) = 0, \tag{4}$$

where $\mathbf{X}$ is the vector of desired dependent variables (the values of the gasdynamic variables at grid nodes, including boundary ones). This system of grid equations can be efficiently solved using the well-known Newton method; its main advantage is that it has a quadratic convergence rate. To solve (4), we used the modified Newton–Raphson method

$$\mathbf{X}^{[k+1]} = \mathbf{X}^{[k]} - \tau_{k+1}\mathbf{D}_{k_0}^{-1}\mathbf{R}(\mathbf{X}^{[k]}),$$

where $\mathbf{D}_{k_0} = (\partial\mathbf{R}/\partial\mathbf{X})_{k_0}$ is the Jacobian matrix, k and k_0 are iteration numbers satisfying $k_0 \le k$, and $\mathbf{R}(\mathbf{X}^{[k]})$ is the residual vector. The expression $\mathbf{D}_{k_0}^{-1}\mathbf{R}(\mathbf{X}^{[k]}) \equiv \mathbf{Y}^{[k]}$ is the solution to the linear system of equations $\mathbf{D}_{k_0}\mathbf{Y}^{[k]} = \mathbf{R}(\mathbf{X}^{[k]})$. In the course of the numerical solution, the regularization parameter of Newton's method with respect to the initial approximation τ_k was determined by the formula (see [23])

$$\tau_{k+1} = \frac{(\mathbf{Y}^{[k]} - \mathbf{Y}^{[k-1]}, \mathbf{Y}^{[k]})}{(\mathbf{Y}^{[[k]]} - \mathbf{Y}^{[k-1]})^2}.$$

With the convergence of the iterative process, we have $\tau_k \to 1$ and the convergence rate theoretically becomes quadratic.

The most effort in the implementation of Newton's method is required by the generation of the matrix $\mathbf{D}_{k_0} = (\partial\mathbf{F}/\partial\mathbf{X})_{k_0}$ and the subsequent solution of the system of linear equations with this matrix.

Since the approximation of the equations in each mesh cell involves only several neighboring nodes (in the 3D case, 25 for the TVD scheme), the computational complexity of generating the Jacobian matrix is $O(N)$, where N is the number of grid nodes. In iteration, the Jacobian matrix is formed via finite increments of the residual vector $\mathbf{R}$ in the vector of desired grid variables $\mathbf{X}$. This technique is universal, since it can easily be extended to an arbitrary system of grid equations (without preliminarily specifying its form). Rather frequently, the approximation of differential equations yields difference equations of complicated form, so the analytical formation of the Jacobian matrix becomes rather labor-consuming. Specifically, this occurs when the Navier–Stokes equations are solved by applying monotonized schemes. Moreover, the number of arithmetic and logical computer operations required for the analytical formation of the Jacobian matrix can generally be greater than that in the case of using the finite increment procedure.

The storage and CPU time required for solving the system of linear algebraic equations in nonlinear iteration,

$$(\partial\mathbf{R}/\partial\mathbf{X})_{k_0}\mathbf{Y}^{[k]} = \mathbf{R}\left(\mathbf{X}^{[k]}\right),$$

depends substantially on the degree of sparseness of the matrix $(\partial\mathbf{R}/\partial\mathbf{X})_{k_0}$. When the Navier–Stokes equations are approximated by the second-order accurate difference scheme described in Section 2, the operator $(\partial\mathbf{R}/\partial\mathbf{X})_{k_0}$ has a sparse 25-diagonal block structure, while its elementary block is a dense 5×5 matrix. Preliminary computations have shown that the convergence of the nonlinear iteration depends substantially on the stencil points used for the convective component and direct derivatives of the dissipative component in the Navier–Stokes equations. For mixed derivatives of the dissipative component, the use of "corner" points in the stencil has only a small effect on the convergence of the nonlinear iteration. Accordingly, the diagonals corresponding to the mixed derivatives were omitted from $(\partial\mathbf{R}/\partial\mathbf{X})$, thus reducing the storage and the total number of arithmetic operations in nonlinear iteration nearly by half. As a result, the operator $(\partial\mathbf{R}/\partial\mathbf{X})$ had a sparse 13-diagonal block structure in the 3D case.

The system of linear algebraic equations obtained in nonlinear iteration was solved using the generalized minimal residual method GMRes [24], which had been recognized as the most reliable and fast in a number of numerical experiments [25].

4. COMPUTATIONAL GRIDS AND THE INTERNAL DATA STRUCTURE

The computations are performed on structured multiblock grids. The grid blocks must join one another over entire faces, node to node.

Inside each grid block, irrespective of the others, we introduce a local curvilinear coordinate system (ξ,η,ζ) in which the grid is uniform in each direction and has stepsizes h_ξ, h_η, and h_ζ, respectively. The curvilinear coordinates are assumed to vary from 0 to 1 within each block. In this curvilinear coordinate system, the gas dynamics equations are discretized as described in (3). The conversion to physical Cartesian coordinates is based on (2) with the use of the inverse metric coefficients $\partial\xi/\partial x$, $\partial\xi/\partial y$, $\partial\xi/\partial z$, $\partial\eta/\partial x$, $\partial\eta/\partial y$, $\partial\eta/\partial z$, $\partial\zeta/\partial x$, $\partial\zeta/\partial y$, and $\partial\zeta/\partial z$, which are determined by the equation

$$\begin{pmatrix} \frac{\partial\xi}{\partial x} & \frac{\partial\xi}{\partial y} & \frac{\partial\xi}{\partial z} \\ \frac{\partial\eta}{\partial x} & \frac{\partial\eta}{\partial y} & \frac{\partial\eta}{\partial z} \\ \frac{\partial\zeta}{\partial x} & \frac{\partial\zeta}{\partial y} & \frac{\partial\zeta}{\partial z} \end{pmatrix} \times \begin{pmatrix} \frac{\partial x}{\partial\xi} & \frac{\partial x}{\partial\eta} & \frac{\partial x}{\partial\zeta} \\ \frac{\partial y}{\partial\xi} & \frac{\partial y}{\partial\eta} & \frac{\partial y}{\partial\zeta} \\ \frac{\partial z}{\partial\xi} & \frac{\partial z}{\partial\eta} & \frac{\partial z}{\partial\zeta} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Here, the direct metric coefficients (the derivatives $\partial x/\partial\xi$, $\partial y/\partial\xi$, $\partial z/\partial\xi$, $\partial x/\partial\eta$, $\partial y/\partial\eta$, $\partial z/\partial\eta$, $\partial z/\partial\xi$, $\partial z/\partial\eta$, $\partial z/\partial\zeta$)) are approximated by finite differences to second-order accuracy.

The grid nodes inside each block have a local three-dimensional index (i, j, k) corresponding to the grid line directions ξ, η, ζ, respectively. Additionally, each node is assigned a global one-dimensional index that is unique over all grid blocks. This global index determines the index of an element in the solution vector $\mathbf{X}$ and the residual vector $\mathbf{R}$ when the system of grid equations (4) is solved.

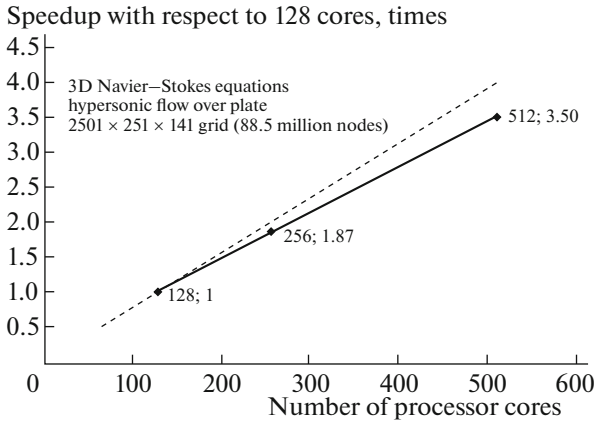

Fig. 2. Speedup as a function of the number of processors for a computer with two 12-core Intel Xeon E5-2670 (2.6 GHz) processors under CentOS (Linux).

To discretize the equations (compute the residual vector $\mathbf{R}$) near a block boundary, we need to know the values of the dependent variables at the nodes of the adjacent blocks. The access to these nodes is based on the concept of shadow values. Specifically, each block is preliminarily extended to several grid surfaces according to the number of points in the discretization stencil, so that these new shadow nodes correspond to the nodes of neighboring block. The dependent variables at the shadow nodes are not calculated in the course of problem solving, but are taken from the adjacent block via a special procedure for scattering shadow values, which is executed before every time iteration. The additional shadow nodes are regarded as belonging to the current block, thus increasing its size, and have local three-dimensional indices (i, j, k), but their global indices remain the same as those of the corresponding "actual" nodes in the adjacent block. Therefore, the total number of nodes in the computational domain remains unchanged.

For each grid block, the mapping between the local three-dimensional indices (including shadow nodes) and the global indices is stored in a special list of indices, which underlies the exchange of shadow values.

5. IMPLEMENTATION OF PARALLEL COMPUTATIONS ON MULTIPROCESSOR SUPERCOMPUTERS

The distributed computations were implemented with the help of parallelization at the block level at all stages of the algorithm, namely, in the discretization procedure and the solution of the system of grid equations.

In the course of the computations, each processor deals with at least one block and stores block data (the grid and the solution at nodes) independently of the other processors. This makes it possible to use multiprocessor computer systems with distributed memory. If the number of processors exceeds the number of initial grid blocks, the grid is additionally divided into the required number of blocks. The numerical results and the grid are stored in the standard international format of CGNS (CFD General Notation System) files [26].

In the case of parallel computations on a distributed-memory supercomputer, the solution vector with a global indexing of elements becomes distributed, i.e., its parts are stored on different processors. Therefore, the residual vector of the nonlinear grid equations and the Jacobian matrix must be distributed as well. In the developed code, operations with such vectors and matrices were implemented with the help of subroutines from the freeware package PETSc (Portable Extensible Toolkit for Scientific Computation) [27], which employs the Message Passage Interface (MPI) for parallel computations.

In each block, the original Navier–Stokes equations are discretized by each processor independently. As a result, a portion of the globally distributed residual vector is formed. The connection between the processors corresponds to the connectivity between the blocks and is provided using the concept of shadow nodes. Exchange of shadow values is implemented via Scatter built-in structures and functions from PETSc. They make use of the mapping list between the global interblock and local intrablock indices, which is formed in the preprocessing of the grid.

In the solution of the grid equations at every iteration step, the Jacobian matrix is formed in a parallel manner by using finite increments of the residual vector in the desired grid variables. The resulting Jacobian matrix is fully distributed. The linear system with this matrix is solved by applying a parallel GMRes version [14] (implemented in PETSc) with a block Jacobi-type preconditioner.

Thus, each stage of the implemented numerical method is executed in parallel in each block of the computational domain. Therefore, the speedup of the computations is close to perfectly linear if the delays for interprocessor communication can be neglected. Preliminary computations have shown that the PETSc parallel procedures efficiently transfer data between the processors, so that the scalability proves to be good. Figure 2 presents the speedup curve for the hypersonic flow over a sharp plate computed on a grid consisting of 88.5 million nodes.

The relative delays for interprocessor communication can be reduced by increasing the time spent by each processor on the processing of its own amount of data irrespective of the others. In other words, the dimensions of grid blocks and/or their number per processor have to be increased.

The method described above was implemented in the original software package HSFlow (High Speed Flow).

6. VERIFICATION OF THE NUMERICAL METHOD

The method was verified using the results of [28] obtained with another numerical method for solving the Navier–Stokes equations. Specifically, hybrid nonmonotone difference schemes of fourth and higher orders of accuracy and the spectral method were applied in [28]. The free-stream parameters in this work were similar to those in [28], namely, $M_\infty = 5.35$, $Re_{\infty,1} = 14.3\times10^6$ 1/m, $\gamma = 1.4$, $Pr = 0.71$, and $T_\infty^* = 64.3163$ K. The length $L = 0.1$ m was used for nondimensionalization; then $Re_\infty = 1.43\times10^6$. These parameters correspond to their experimental values on the edge of the boundary layer on an sharp cone, which was placed in a shock tube at $M_\infty = 6.0$, $Re_{\infty,1} = 11.0\times10^6$ 1/m, and $T_0^* = 433.0$ K.

The computational domain was a rectangular parallelepiped with sides $0.45\times0.903\times0.04$ m in the streamwise, vertical, and spanwise directions, respectively. On the lower boundary of the domain, which coincided with the plate surface, we set the no-slip condition. The wall was assumed to be isothermal with temperature $T_w = 300$ K ($\overline{T}_w = 4.664$). On the boundaries $x = x_{max}$ (right) and $z = z_{max}$ (front), we used linear extrapolation of the dependent variables u, v, w, p, and T (nonreflecting boundary condition in the case of a supersonic flow). The free-stream conditions were specified on the boundaries $x = x_{min}$ (left) and $y = y_{max}$ (upper). The symmetry condition was used on the boundary $z = 0$ (back).

The computations were performed on an orthogonal grid consisting of $2501\times251\times141$ nodes in the streamwise, vertical, and spanwise directions, respectively (overall 88.5 million nodes). Near the plate surface, the grid was refined in the vertical direction, so that 120 nodes fell within the boundary layer. The mesh size in the boundary layer remained unchanged, namely, $\Delta x^* = 1.8\times10^{-4}$ m, $\Delta y^* = 3.01\times10^{-4}$ m, and $\Delta z^* = 2.86\times10^{-4}$ m. For comparison, the mesh size in [28] remained unchanged in the streamwise direction on the initial segment ($\Delta x^* = 2.5\times10^{-4}$ m), varied according to a cubic law starting at $\Delta y^* = 6.46\times10^{-6}$ m in the vertical direction, and pseudospectral discretization was used in the spanwise direction.

The computations were performed on a high-performance distributed-memory computer cluster with up to 512 processor cores. The efficiency of the numerical code was also tested on 128 and 256 cores (see Fig. 2).

Disturbances were introduced into the boundary layer through a small circular hole on the plate surface in the form of forced fluctuations of the vertical velocity. These fluctuations were specified by the usteady boundary condition on the wall:

$$v_w = A\cos^3\left(\pi\frac{x-0.5(x_2+x_1)}{x_2-x_1}\right)\cos^3\left(\pi\frac{z}{z_2-z_1}\right)\sin(-\omega t),$$
$$x_1 \le x \le x_2, \quad z_1 \le z \le z_2, \quad 0 \le t \le t_1,$$

where $x_1 = 1.36$, $x_2 = 1.40$, $z_1 = -0.02$, $z_2 = 0.02$, $t_1 = 2\pi/\omega$, and $\omega = \omega^* L^*/U_\infty^*$ is the dimensionless frequency. To provide the nonlinear development of disturbances, the amplitude was chosen sufficiently large: $A = 0.05$ (5% of the free-stream velocity). The indicated form and parameters of the disturbances completely agree with the nonlinear regime used in [28].

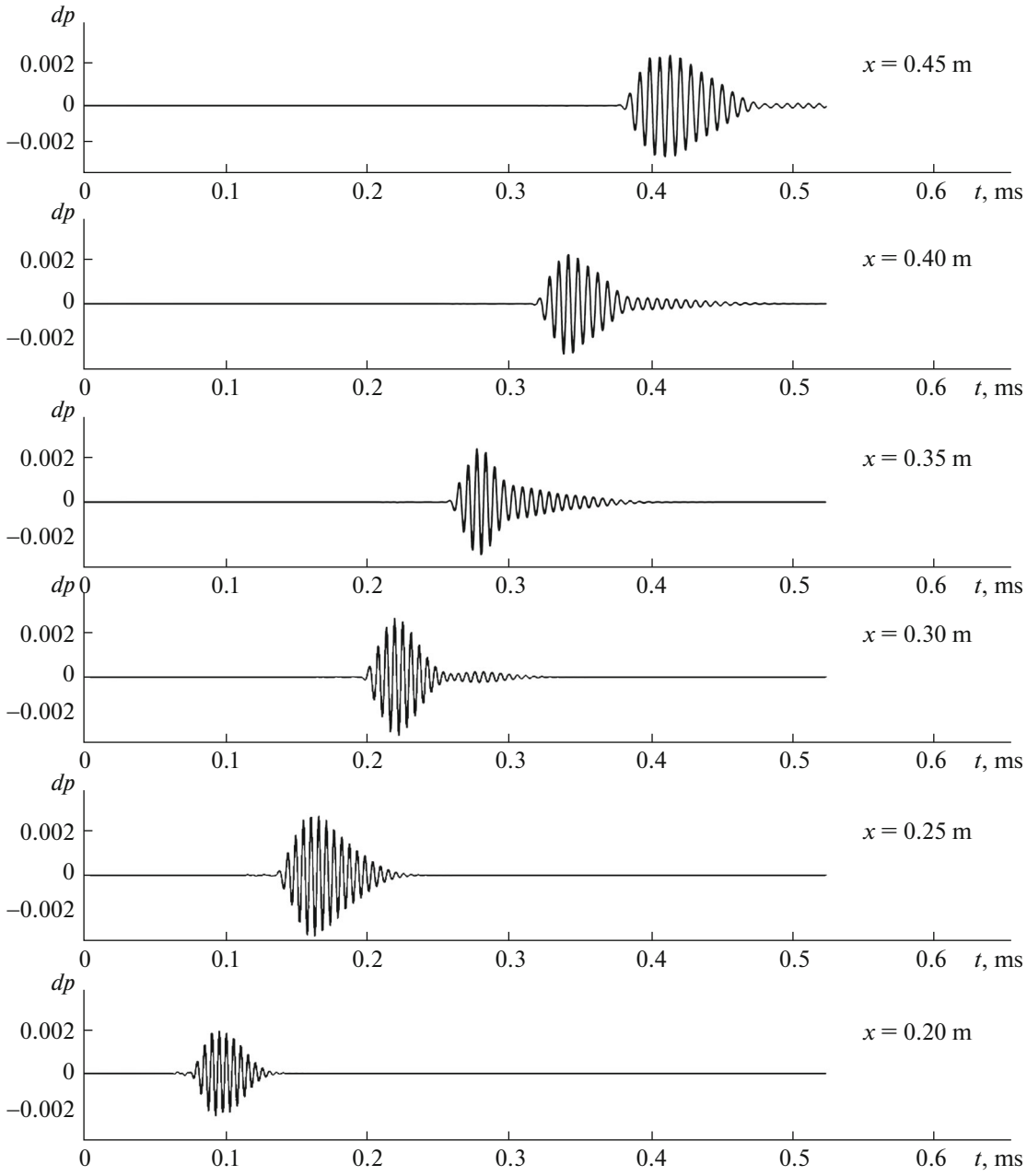

Fig. 3. Surface pressure disturbances at $z = 0$ in various x cross sections for the wave packet simulated according to [28].

The problem was solved in two steps. First, we computed an unperturbed steady flow field. Then disturbances were introduced by switching on the unsteady boundary condition modeling an actuator. It should be noted that the numerical error of the unperturbed solution must be much smaller than the amplitude of the unsteady fluctuations. Accordingly, the steady flow field has to be computed to high accuracy.

The forced disturbances introduced into the boundary layer by a short local pulse give rise to a three-dimensional wave packet propagating downstream. The evolution of pressure disturbances on the plate surface along the centerline $z = 0$ is shown in Fig. 3, which presents the amplitude of the wave packet against time in several cross sections in the streamwise coordinate x. This representation is frequently used in experiments measuring oscillograms of fluctuations at several points along the central line of the model. It can be seen that the packet has a pronounced carrier frequency and its extent increases downstream due to the dispersion of the wave components.

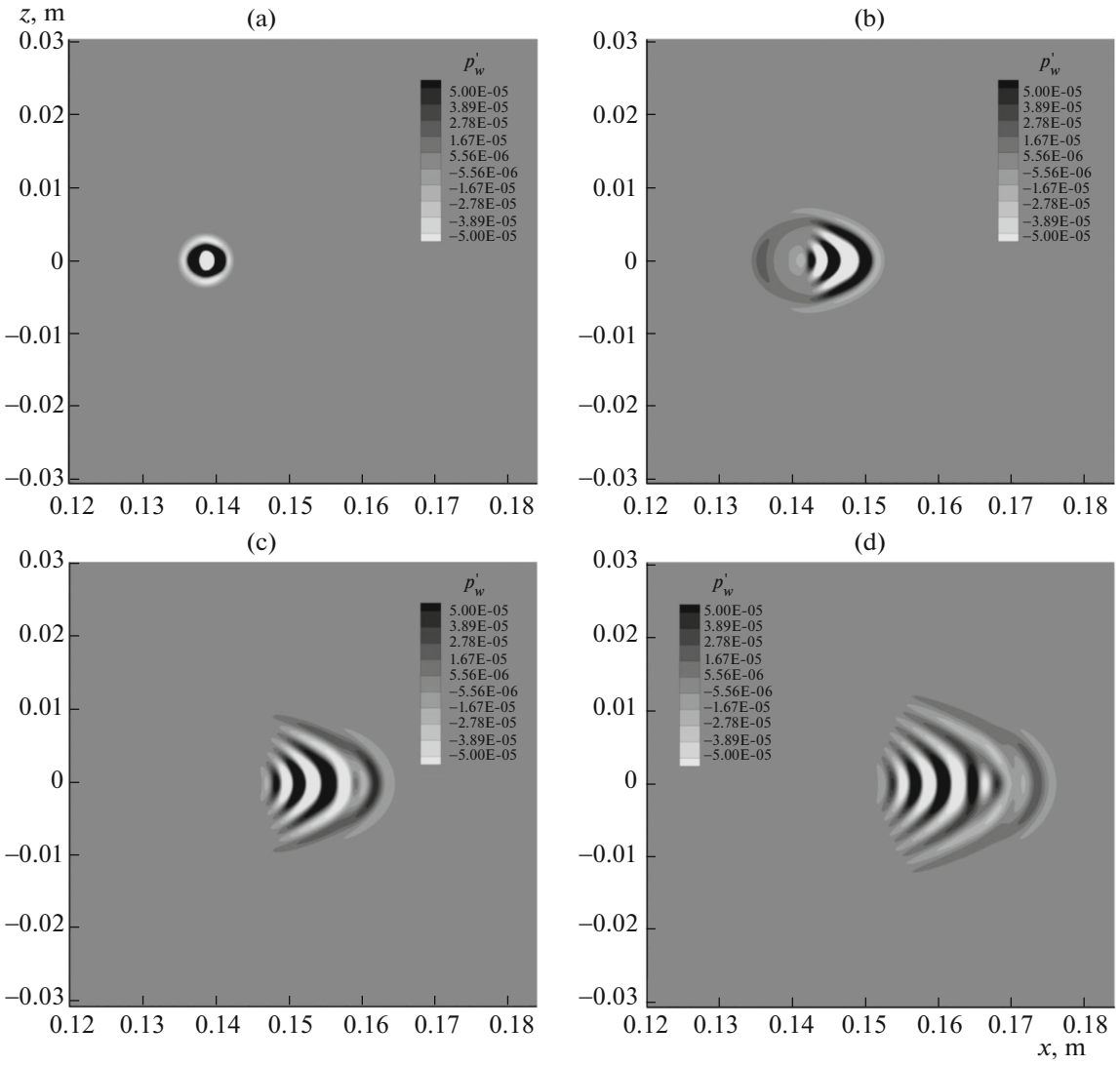

Fig. 4. Instantaneous surface pressure disturbance fields near the actuator at (a) $t = 0.006$, (b) 0.017, (c) 0.029, and (d) 0.041 ms for the wave packet simulated according to [28].

The initial response of the flow to the pulse is demonstrated in Fig. 4, which depicts contour lines of the instantaneous pressure disturbance field on the plate surface near the actuator at several time instants. Hereafter, the disturbance field is calculated by subtracting the basic steady field from the instantaneous perturbed one. First, the wave packet is localized in space around the moving center (see Figs. 4a, 4b). Further downstream, the wavefronts flatten near the central line $z = 0$ (see Figs. 4c, 4d). Plane waves with amplitudes growing downstream begin to dominate. This behavior agrees with the linear stability theory, according to which plane waves (so-called second Mack mode waves) are amplified faster than oblique ones. This tendency persists at later stages of the packet evolution (see Figs. 5, 6). In addition to the central part of the packet with pronounced plane fronts, we can see waves traveling ahead of the main disturbance. They appear at an early stage (Fig. 4d) and propagate somewhat faster than the main wave packet, namely, at speeds corresponding to fast acoustic waves.

In Fig. 6, the wave packet seems to be distorted by nonlinear effects. The interaction between two-dimensional and oblique waves leads to a bottleneck in z observed in the maximum-amplitude area. As the disturbance develops further, the nonlinear effects must be amplified and eventually excite a turbulent spot. This could be seen by executing simulation in a longer computational domain.

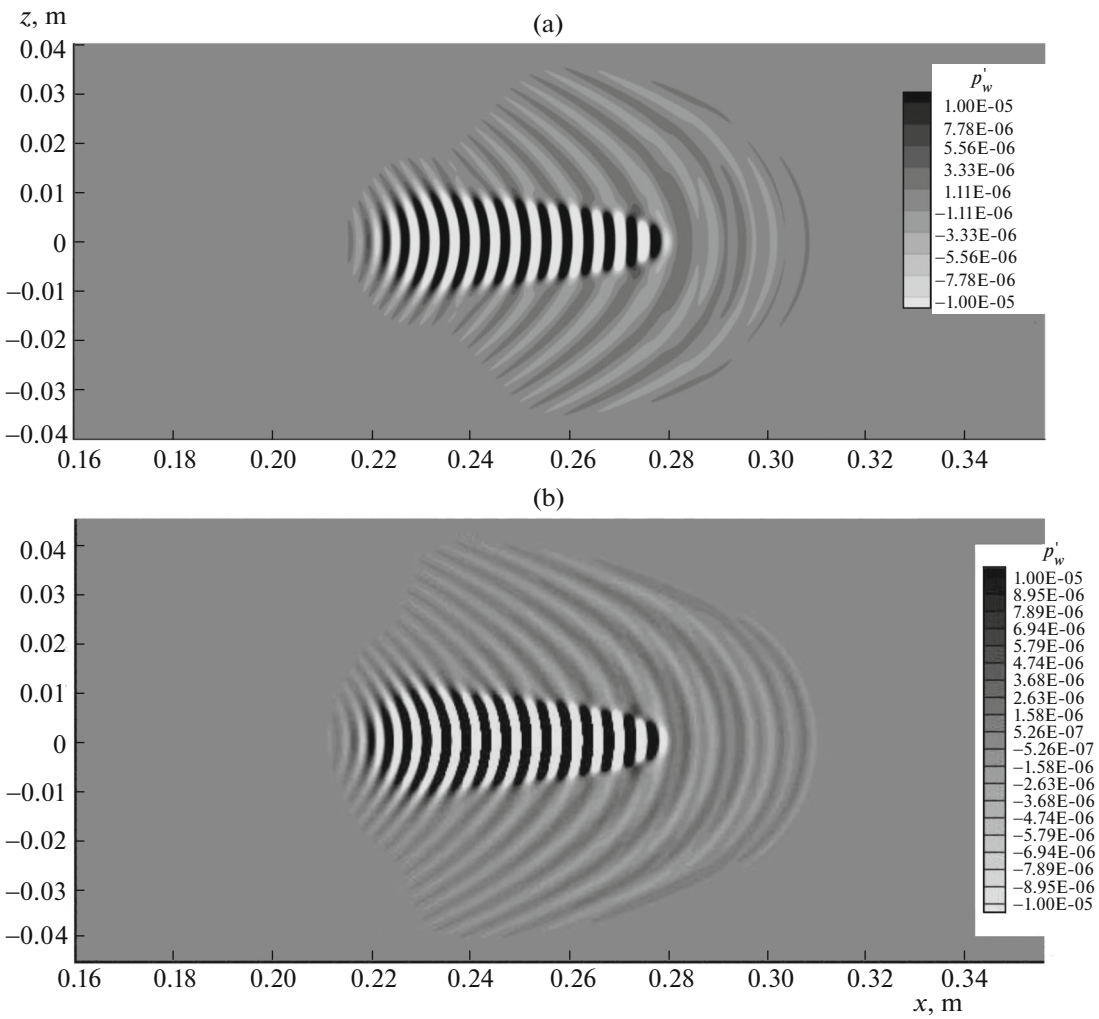

Fig. 5. Instantaneous surface pressure disturbance field far away from the actuator at $t = 0.174$ ms: (a) the present work and (b) [28].

For comparison, Fig. 5 also presents the disturbance field obtained in [28]. It can be seen that the basic structural features of the wave packet are similar in both fields. Specifically, in both cases, the disturbance has the same streamwise and spanwise sizes, the waves involved have similar frontal shapes, and there are 32 extrema along the centerline in both cases, i.e., the dominant wavelengths coincide quantitatively. It should be noted that, by the considered time, the packet has been transformed from initially circular concentric waves into a set of almost plane waves, which is caused by the selective amplification of disturbances in the boundary layer according to the physics of instability. Therefore, the agreement of the packet structure at a later time suggests that the entire physical process is correctly simulated as based on the results of [28].

Figure 7 compares the spectra of the wave packet in the (k_z, f) plane, where k_z is the spanwise wave number. In this work, the spectra were constructed using the discrete Fourier transform of the two-dimensional pressure disturbance $p'_w(t, z)|_{x=\text{const}}$ on the wall for a given x and varying t and z. In [28] k_z was computed directly, since the spectral method was used in the spanwise direction. A comparison of the spectra shows that the range of spanwise wave numbers in [28] is broader than in the present computation, which can presumably be explained by the insufficient grid resolution in the z-direction in the latter case. However, in both cases, the characteristic features of the spectral patterns coincide (shown by dashed ovals in Fig. 7). Thus, the results are in qualitative agreement, though different quantitatively.

It should be noted that, in the present computations based on a low-order method, the spatial grid resolution was approximately 28 nodes in the streamwise direction per dominant wavelength (see Fig. 8),

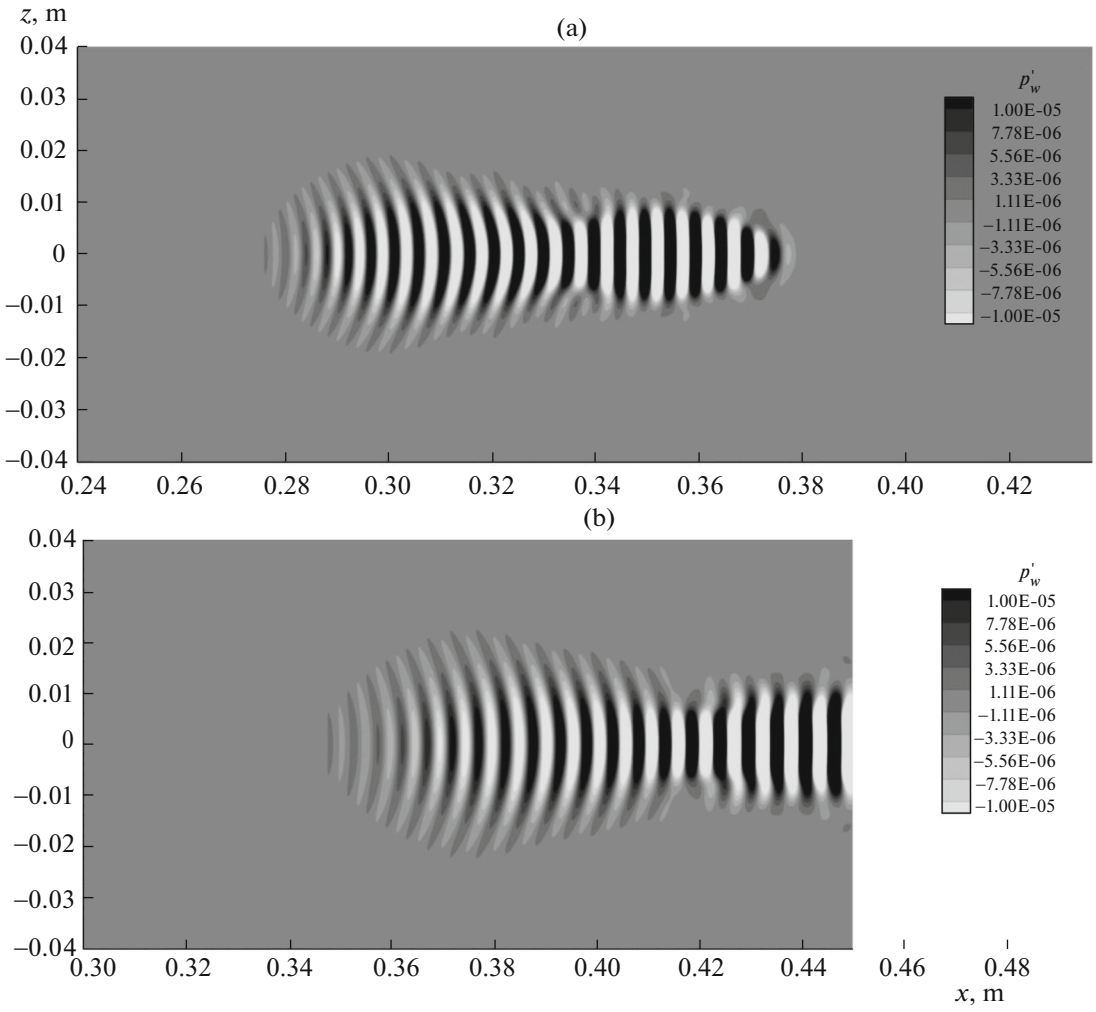

Fig. 6. Disturbance packet at a long distance from the actuator with manifestation of nonlinear effects: instantaneous surface pressure disturbance fields at (a) $t^* = 0.291$ and (b) 0.41 ms.

which was only 38% larger than the resolution in [28], where high-order accurate difference schemes were applied. Moreover, the simulation results obtained for the wave packet were quite comparable, at least at the initial nonlinear stage. This agrees with our previous numerical studies of hypersonic boundary layer stability, which suggest that at least 30 nodes per wavelength are required for unstable waves to be correctly resolved by the present low-order scheme.

Thus, the universal numerical method applied in this work can be used to study the development of three-dimensional disturbances in hypersonic flows.

7. DIRECT NUMERICAL SIMULATION OF THE INITIAL STAGE OF LAMINAR–TURBULENT TRANSITION

As an example of computing the laminar–turbulent transition at the initial stage, we considered laminar flows over a plate and a 5.5° compression corner in which disturbances were artificially introduced by periodic gas blowing–suction through a local hole on the body surface. The free-stream parameters were specified as $M_\infty = 5.373$, $Re_{\infty,1} = 17.9 \times 10^6$ 1/m, and $T_\infty = 74.194$ K; the wall temperature was $T_w = 300$ K. The coordinates were normalized by the characteristic scale $L = 0.3161$ m, i.e., the distance from the leading edge to the kink line on the body surface, which corresponded to the Reynolds number $Re_{\infty,L} = 5.67 \times 10^6$. For both configurations, the blowing–suction was simulated by the following unsteady boundary condition imposed on the vertical mass flux:

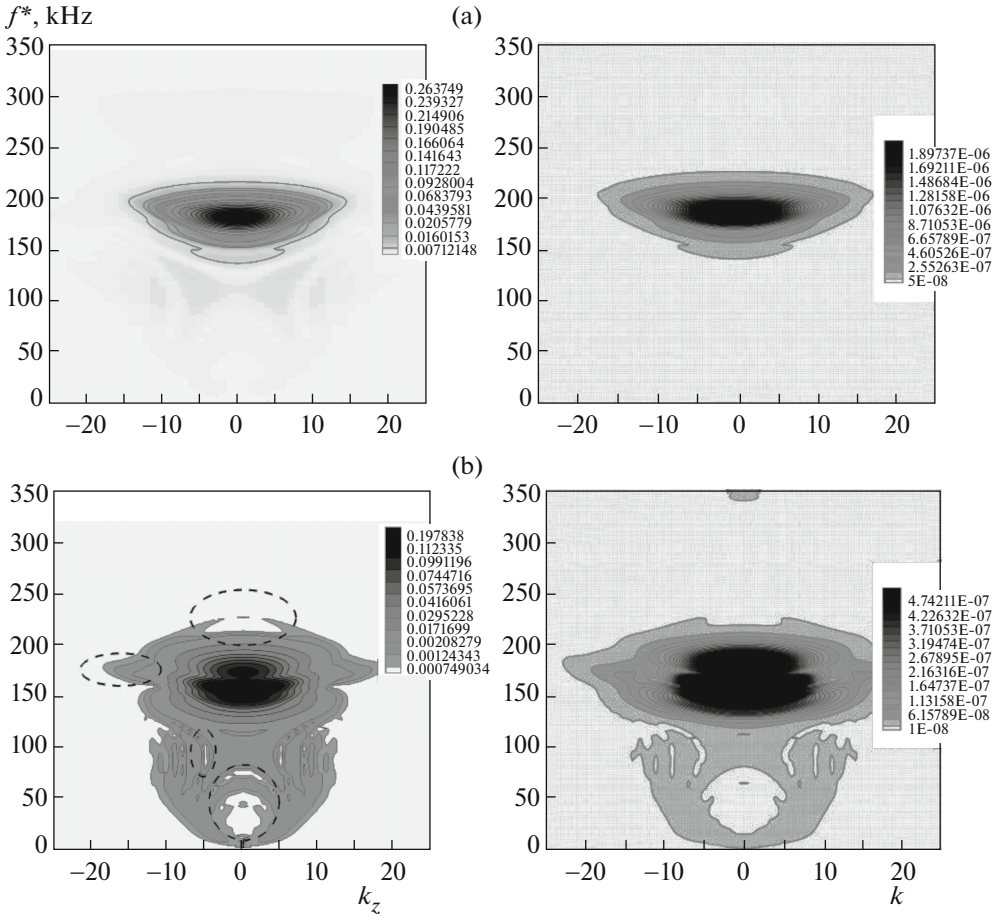

Fig. 7. Spectra of the wave packet in the (k_z, f) plane for (a) $x = 0.25$ and (b) 0.35 m in the present work (left panels) and [28] (right panels). The dashed ovals mark the structural features coinciding in both fields.

$$q_w(x,t) = (\rho v)_w = \varepsilon \sin\left(2\pi \frac{x - x_1}{x_2 - x_1}\right) \sin\left(\pi \frac{z - z_1}{z_2 - z_1}\right) \sin(\omega t),$$
$$x_1 \le x \le x_2, \quad z_1 \le z \le z_2, \quad t \ge 0,$$

where $x_1 = 0.0358$, $x_2 = 0.0521$, and $z_2 = -z_1 = (x_2 - x_1)/4 = 0.004075$ are the boundaries of the blowing–suction actuator. The forcing amplitude $\varepsilon = 0.001$ was chosen so that the initial development of the disturbance was linear. The actuator was placed near the leading edge (the central point $x_0 = (x_1 + x_2)/2 \simeq 0.044$), since, in natural conditions, instabilities in the boundary layer are most effectively excited near the leading edge. The flow parameters and the size and shape of the actuator were taken from [17], where the stability of the flow over a similar configuration was analyzed in two-dimensional setting. In this work, the actuator was switched on at the time $t = 0$ and worked permanently at the frequency $\omega = 125$, which is characteristic of the instability of the first Mack mode (see [1]).

The computations were performed on a curvilinear orthogonal grid consisting of 2801 × 221 × 141 nodes in the x, y, and z directions, respectively (overall 87.3 million nodes). First, a two-dimensional 2801 × 221 grid was constructed by applying a numerical conformal mapping of the rectangle onto the computational domain. This grid was then refined toward the wall. The grid in the streamwise direction was uniform with the mesh size being $\Delta x = 5 \times 10^{-4}$, which corresponded to about 80 nodes per basic wavelength, so that the condition requiring at least 30 nodes per wavelength was satisfied with a margin. In the present study, this criterion was amply fulfilled. Near the wall, mesh refinement was used to ensure an identical step of $\Delta y = 10^{-4}$ within the separation region and the mixing layer. Thus, about 120 nodes fell into the boundary layer, which (as follows from numerical experience) were sufficient for instabilities

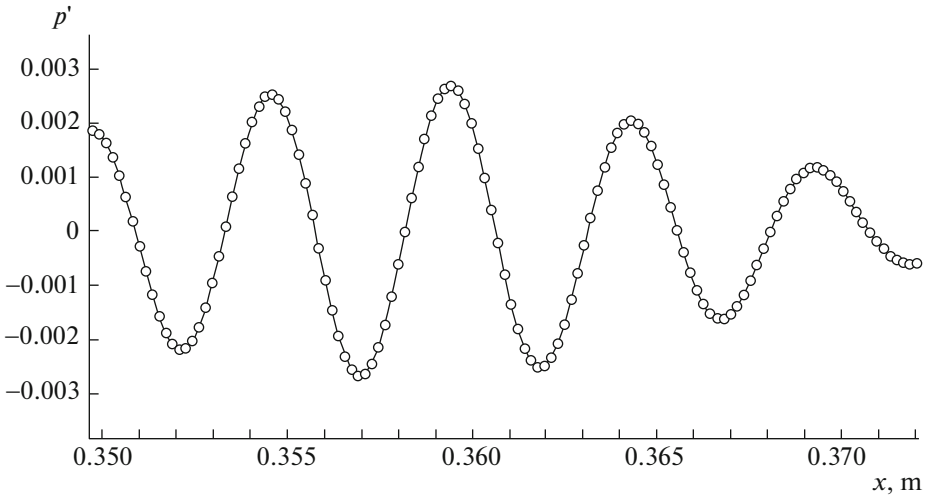

Fig. 8. Grid resolution in the streamwise direction for the surface pressure disturbance along the centerline $z = 0$ at $t^* = 0.291$ ms.

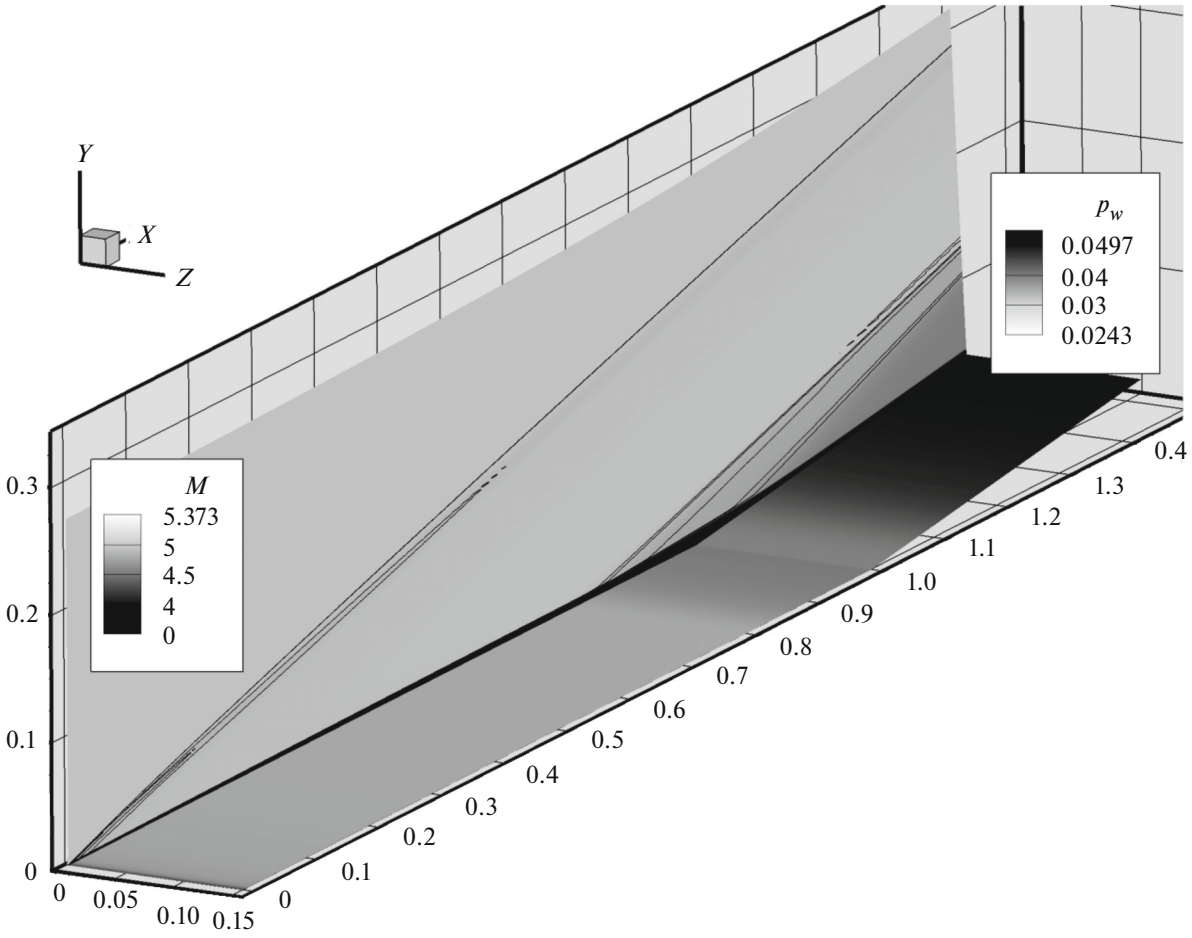

Fig. 9. Steady flow over the compression corner: the Mach number field in the plane of symmetry and the surface pressure distribution.

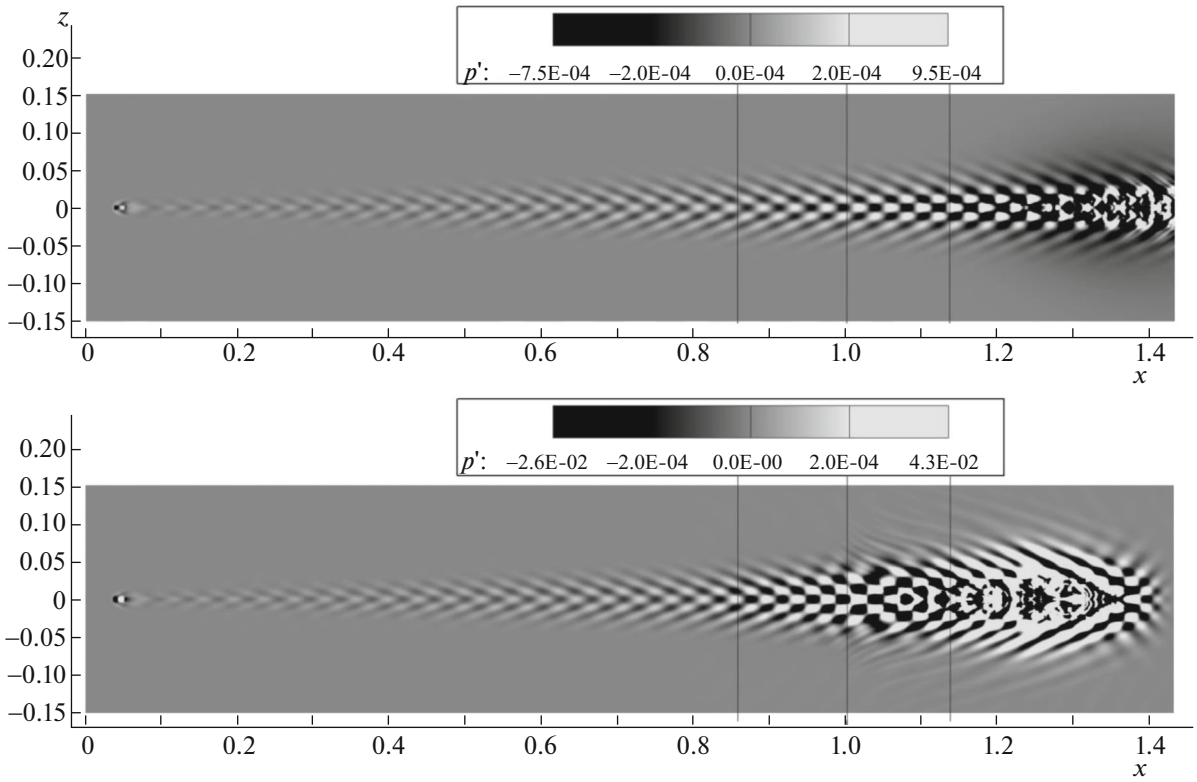

Fig. 10. Surface pressure disturbances for the plate (upper panel) and the compression corner (lower panel) at $t = 1.5$. The vertical lines mark the separation, corner, and reattachment lines for the compression corner.

to be correctly resolved by the proposed numerical scheme. A three-dimensional grid was constructed by adding equidistant points spaced at $\Delta z = 10 \times 10^{-4}$ to the two-dimensional grid in the spanwise direction up to $z = 0.15$. Time integration was performed with the step $\Delta t = 5 \times 10^{-4}$, i.e., we used about 100 steps per a period of the disturbance actuator.

The computations were executed on a cluster-type multiprocessor supercomputer with 768 processor cores (32 computational servers with two processors containing 12 cores each).

Figure 9 shows the computed steady flow field over the compression corner. Due to the viscous-inviscid interaction, a weak shock wave is formed near its leading edge. Further downstream, the corner generates oblique shocks issued from the separation and reattachment regions. With the help of the skin friction coefficient distribution

$$c_f(x) = \frac{\tau_w^*}{\frac{1}{2}\rho^* V_\infty^*} = \left[\mu \frac{\partial u}{\partial n}\right]_w \Big/ \frac{1}{2}\mathrm{Re}_\infty,$$

the coordinates of the separation and reattachment points are found to be $x_{\mathrm{sep}} = 0.857$ and $x_{\mathrm{att}} = 1.136$, respectively. These are the zeros of $c_f(x)$, between which the skin friction becomes negative.

Figure 10 shows the dimensionless pressure disturbance distribution $p'_w(x, z)$ over the plate surface and the compression corner at the time $t = 1.5$, when the head wave packet formed at the moment of switching the actuator on is still within the computational domain. Hereafter, instantaneous disturbance fields are calculated by subtracting the basic field from the perturbed one at different times. Moreover, for illustrative purposes, the fields calculated only for $z \geq 0$ are mirrored with respect to the plane of symmetry $z = 0$. Figure 11 presents the surface disturbance fields at the later time $t = 2.8$, when a spatially constant disturbance—wave train—can be observed in the entire computational domain. Upstream of the separation point, the wave train patterns on the plate and the compression corner are similar to each other, which

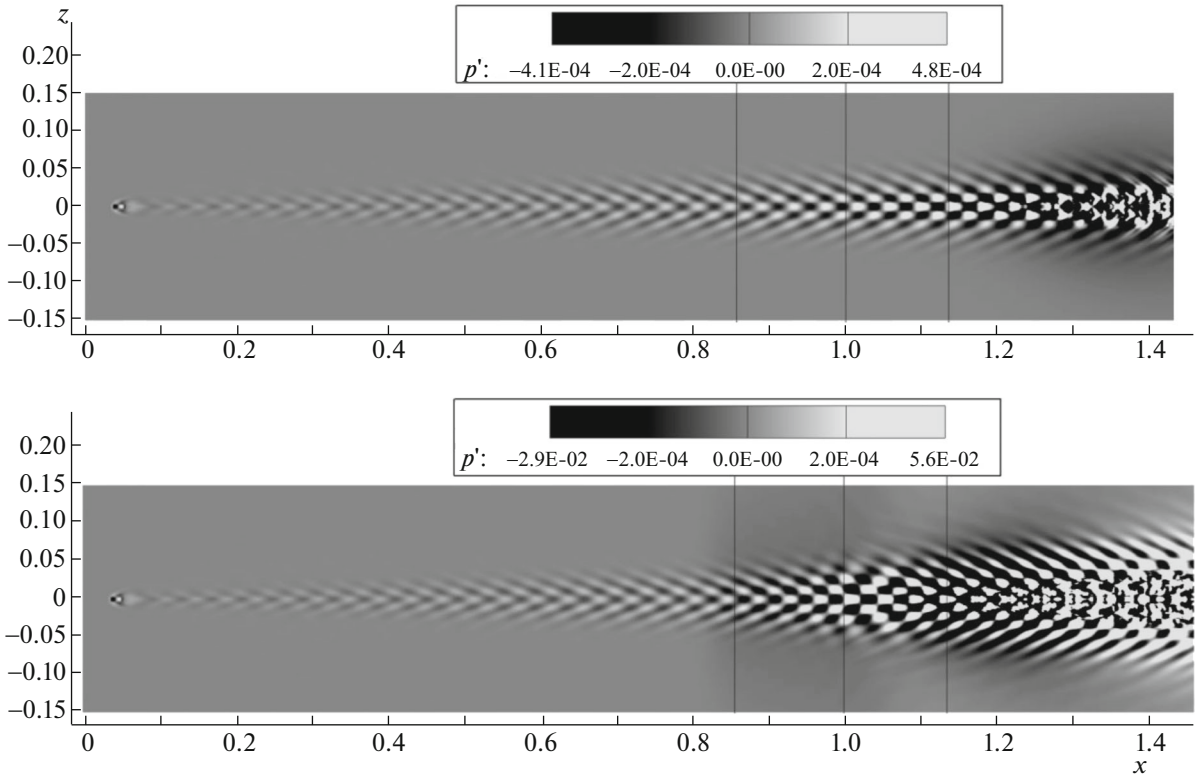

Fig. 11. Surface pressure disturbances for the plate (upper panel) and the compression corner (lower panel) at $t = 2.8$.

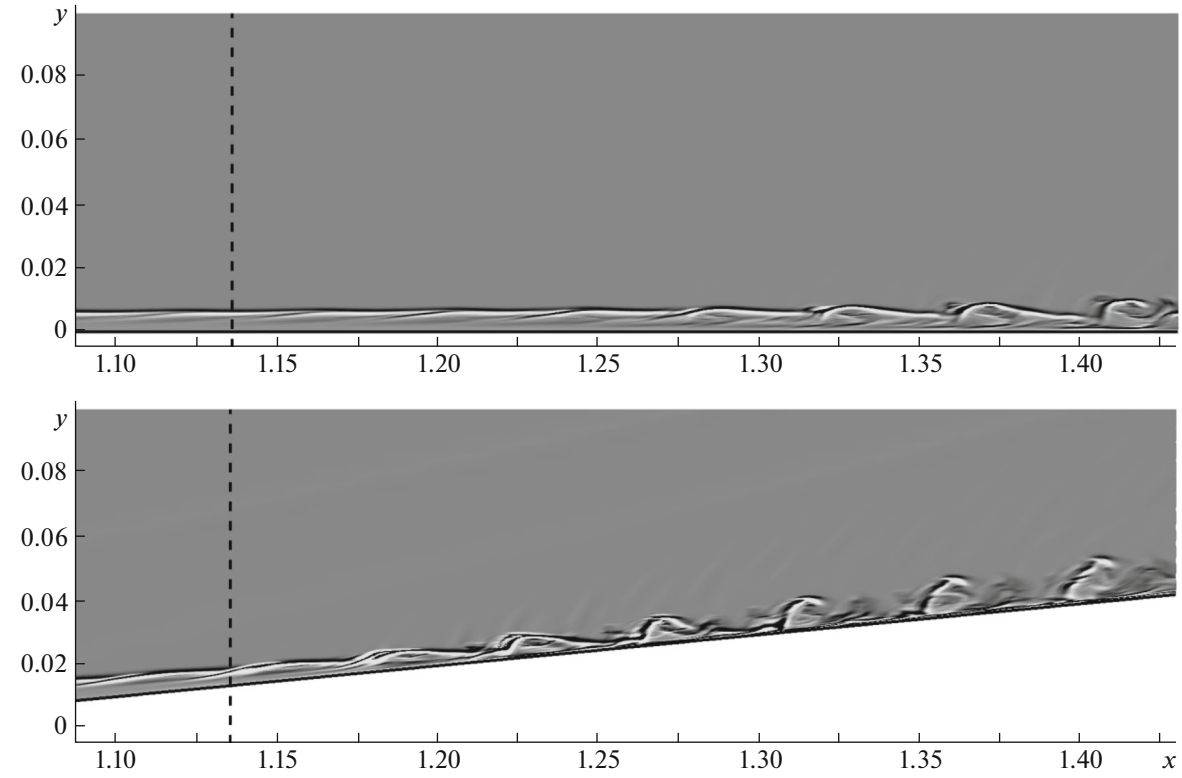

Fig. 12. Shadowgraph in the plane of symmetry on the plate (upper panel) and the compression corner (lower panel) at $t = 2.8$.

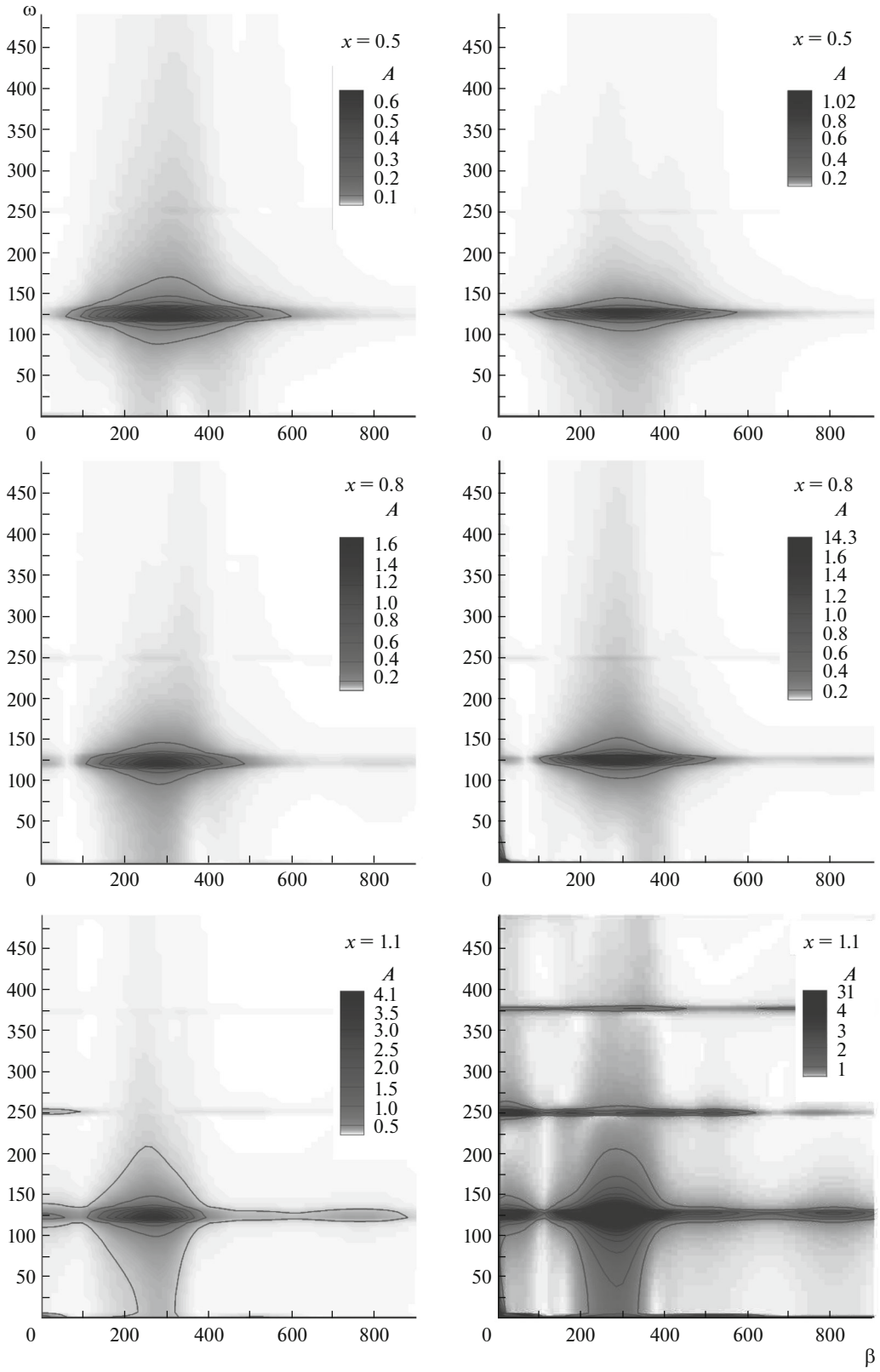

Fig. 13. Spectra in the β–ω plane in various x cross sections for the plate (left panels) and the compression corner (right panels).

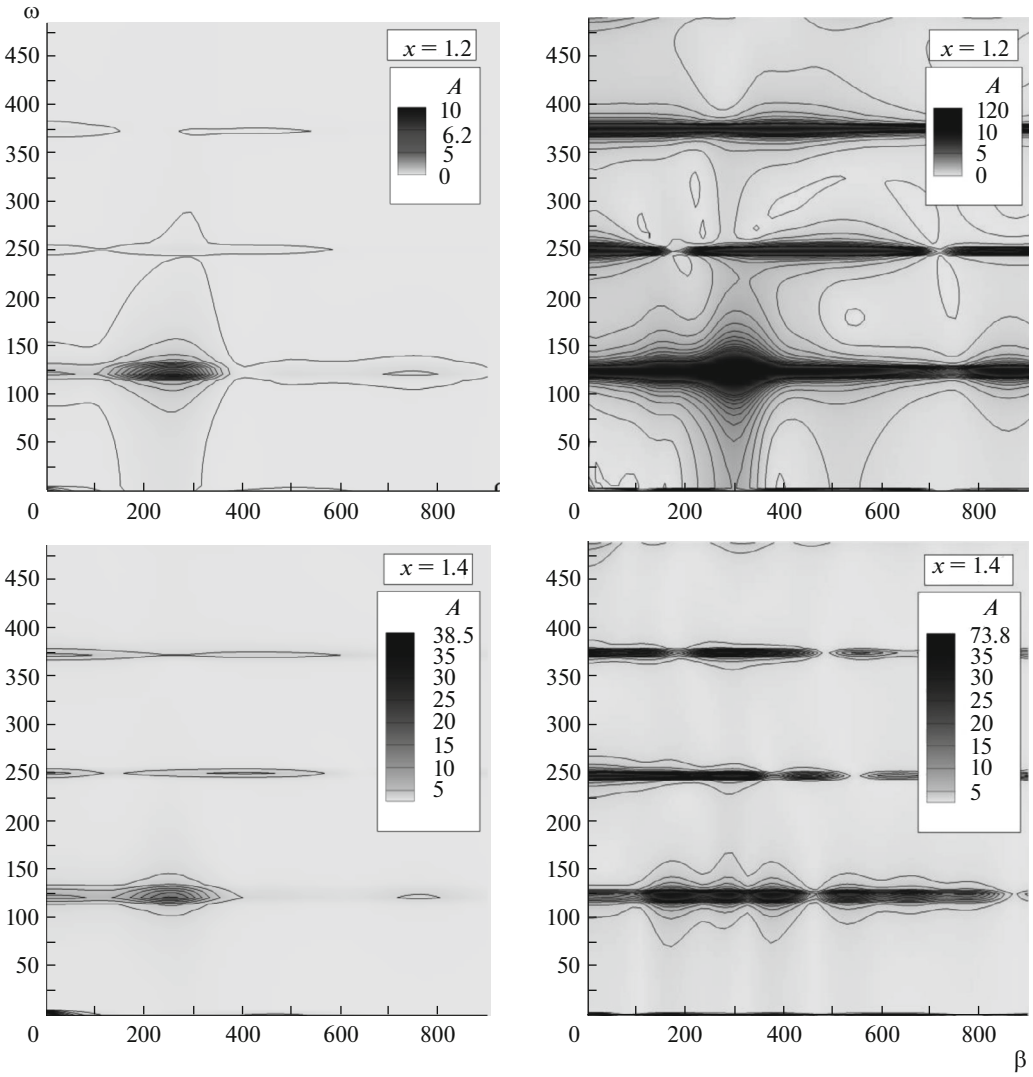

Fig. 13. (Contd.)

indicates a weak influence of the separation in the upstream direction. Immediately downstream of the actuator, they attain a *V* shape typical of the first unstable Mack mode. Near the centerline $z = 0$, the disturbances exhibit a staggered pattern surrounded by oblique waves on both sides. The wave amplitude grows downstream, so that nonlinear breakdown is eventually observed. For the flat plate, this occurs at $x \approx 1.3$. For the compression corner, the wave amplitude grows rapidly in the separation region, so that the disturbance structure is distorted by nonlinear effects ahead of the reattachment point and a "young" turbulent wedge begins to form in the reattached boundary layer for $x > 1.2$.

The beginning of nonlinear breakdown of a wave train is demonstrated in Fig. 12 as a shadowgraph (the Laplacian of density $|\Delta\rho|$) in the plane of symmetry. For the compression corner, the intensity of fluctuations is higher and small-scale vortices are observed near the wall. This agrees with the well-known fact that stochastization in a transient boundary layer begins near the wall.

Variations in the spectral content of the disturbance in its downstream motion and in the growth of the amplitude can be observed in the two-dimensional spectra shown in Fig. 13. The spectra were constructed in various x cross sections by calculating the discrete Fourier transform of the wall pressure fluctuations $p'_w(t, z)|_{x=\text{const}}$ with varying t and z. It can be seen that, at $x = 0.5$ and $x = 0.8$, the series of waves consists basically of oblique waves with frequency $\omega_0 = 125$ and spanwise wave number $\beta_0 \approx 300$, which corre-

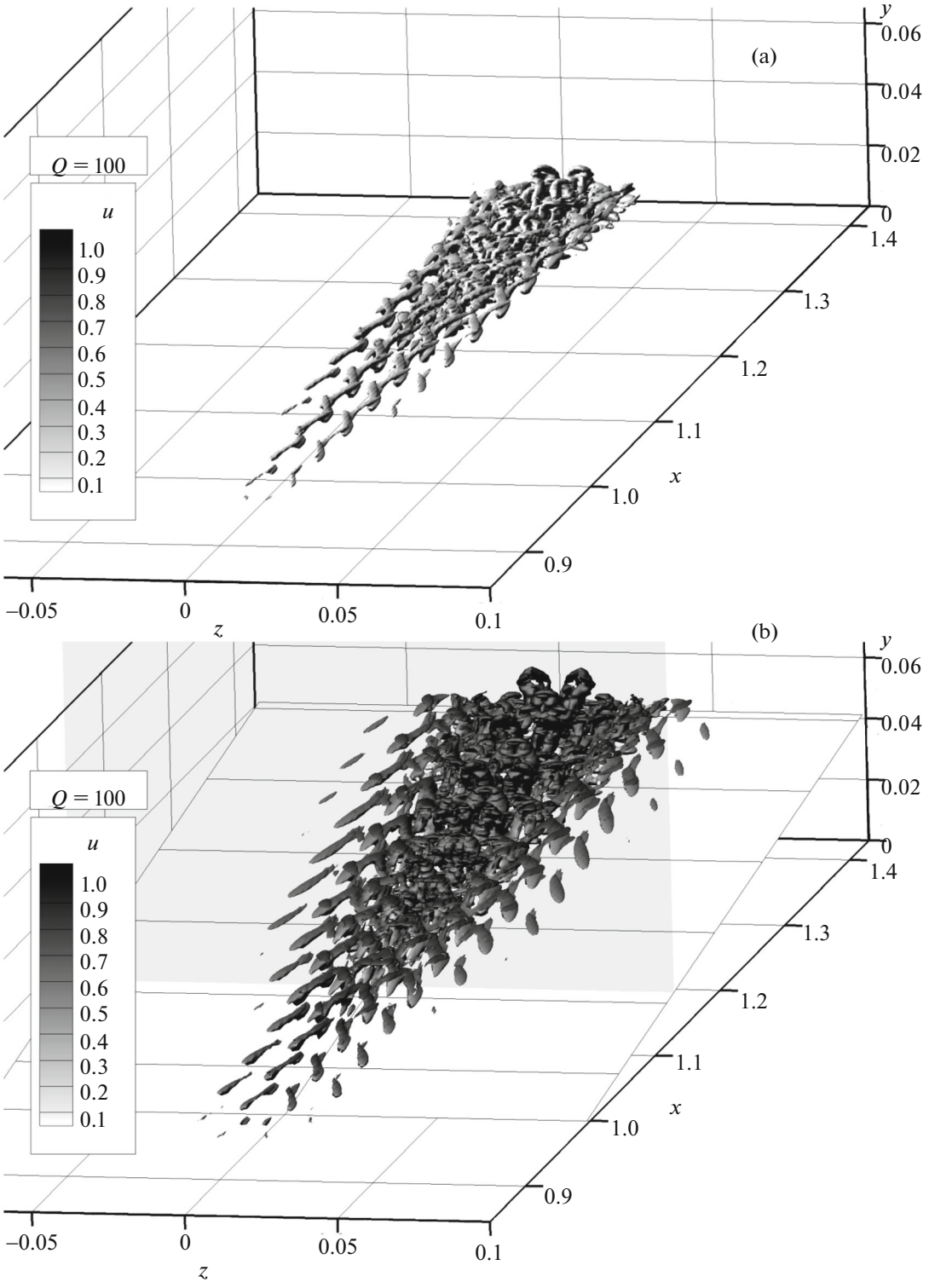

Fig. 14. Surface of the criterion $Q = 100$ colored according to the horizontal velocity on the (a) plate and (b) compression corner at $t = 2.8$.

sponds to a unstable first mode. For $x \geq 1.1$, upper harmonics with frequencies being a multiple of the fundamental one begin to appear in the spectrum: $\omega = 2\omega_0 = 250$, $\omega = 3\omega_0 = 375$, i.e., a fundamental resonance and/or oblique breakdown develop. The upper harmonics in the compression corner are more

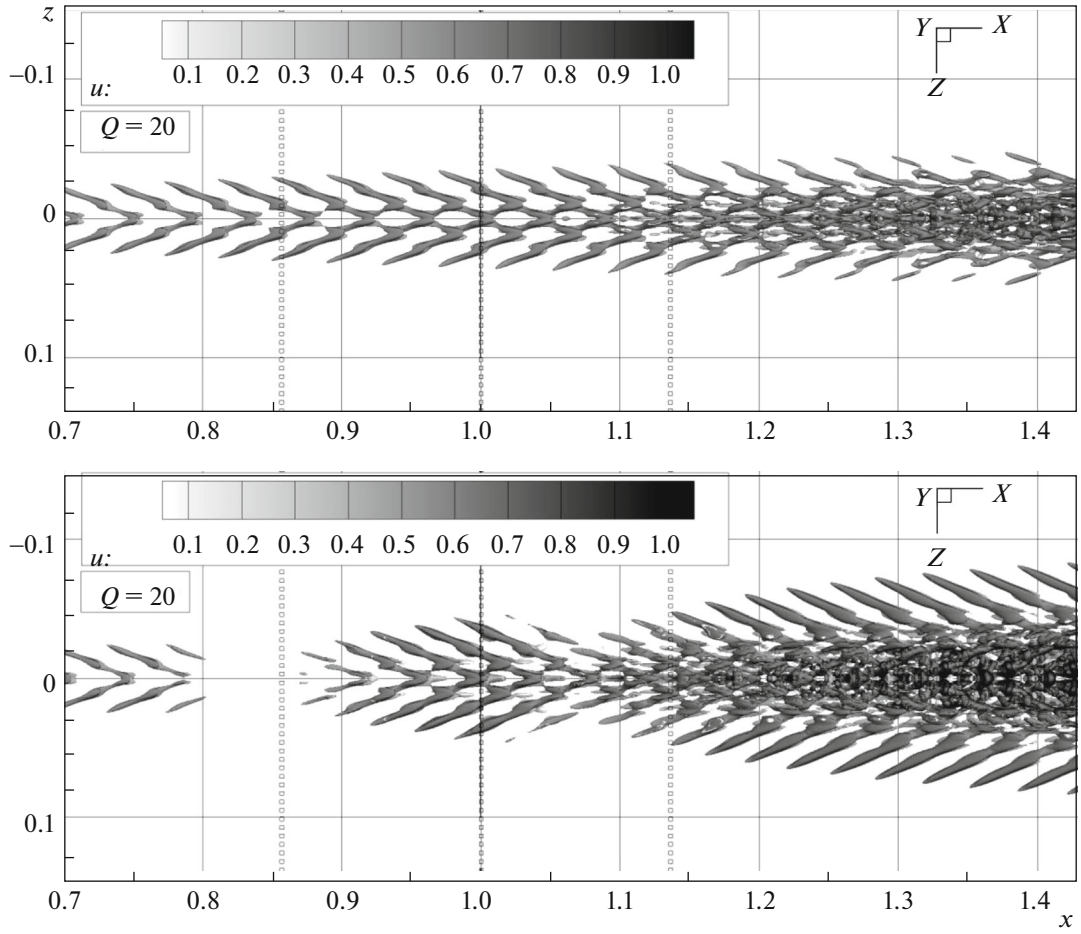

Fig. 15. Surface of the criterion $Q = 20$ colored according to the horizontal velocity on the plate (upper panel) and the compression corner (lower panel) at $t = 2.8$ (top view). The dotted curves mark the separation, corner, and reattachment lines for the compression corner.

intense than those on the flat plate. Additionally, subharmonics appear at later stages. In the case of the compression corner, noticeable intensity is exhibited by harmonics with a zero frequency, which indicates changes in the mean flow field: the boundary layer is transformed from laminar to turbulent regimes.

Figures 14 and 15 show spatial vortex structures of wave trains in the nonlinear breakdown region. Visualization is based on the Q-criterion $Q > 0$, where Q is the second scalar invariant of the strain rate tensor $Q = [|\Omega|^2 - |\mathbf{S}|^2]/2$ with $\mathbf{S} = [\nabla\mathbf{V} + (\nabla\mathbf{V})^{\mathrm{T}}]/2$ and $\Omega = [\nabla\mathbf{V} - (\nabla\mathbf{V})^{\mathrm{T}}]/2$. Lambda structures and their breakdown giving rise to small-scale disturbances can be observed in the figures. In the case of the compression corner, hairpin-shaped vortices rising from the surface are observed along the centerline at late stages of the breakdown. This is a feature characteristic of the onset of LTT.

Figure 16 presents the skin friction coefficient $c_f(x, z)$ over the plate surface and the compression corner for a developed flow. Streaky structures are well seen in the friction field. In the case of the compression corner, friction grows quickly immediately behind the reattachment line and the disturbance elongates in the spanwise direction.

Figure 17 shows the streamwise distributions of c_f on the central line $z = 0$ and averaged in z within the spanwise boundaries of the disturbance. Additionally, the curve for a compressible fully turbulent boundary layer according to the van Driest correlation is depicted for the flat plate. At an early stage of instability development, the distribution of $c_f(x)$ on the plate and in the corner corresponds to a laminar

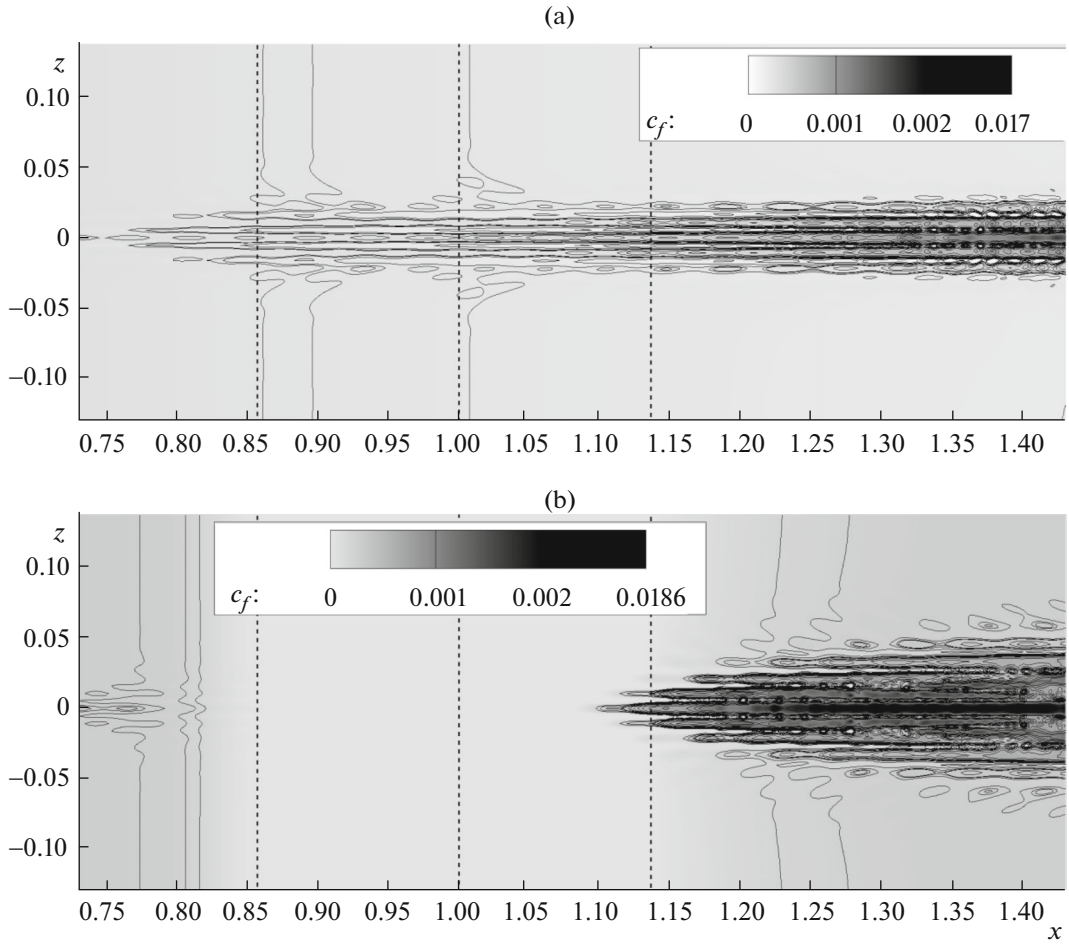

Fig. 16. Skin friction coefficient field for the (a) plate and (b) compression corner at $t = 2.8$. The dashed lines mark the separation, corner, and reattachment lines for the compression corner.

boundary layer. As instability develops, c_f deviates from the laminar value starting at $x \approx 0.9$. In the case of the compression corner, friction grows more sharply and by a larger value, so that a "young" turbulent wedge seems to form in the reattachment region. For the flat plate, the local friction on the centerline exceeds its turbulent value (see Fig. 17a), i.e., there is an overshoot in heat transfer, which is frequently observed at early LTT stages in hypersonic flows [29]. The averaged friction coefficient is considerably lower than its value in the plane of symmetry, which is explained by the inhomogeneity of friction in z exhibited as streaky structures in Fig. 16. Moreover, the averaged friction is lower than the turbulent one (for the flat plate), which indicates a transient flow regime, while complete turbulization occurs downstream outside the computational domain.

Figure 18 shows the vertical velocity profiles near the outlet boundary of the computational domain (maximally far away from the onset of instability development). The profiles of the perturbed boundary layer differ from the laminar ones for both the plate and the compression corner. However, they also differ in shape from the typical parabolic profile for a fully turbulent boundary layer. This confirms that the flow is in a transient regime.

Thus, DNS in the considered example yields a detailed disturbance field at every time instant. As a result, we can investigate the spatiotemporal spectra of disturbances and various averaged characteristics at the initial stage of LTT.

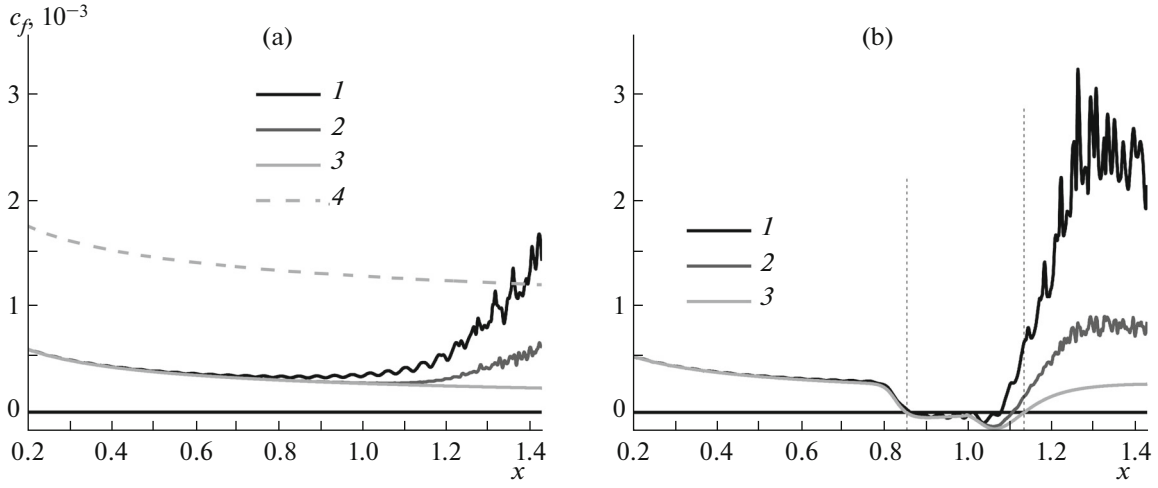

Fig. 17. Streamwise distributions of the skin friction coefficient for the (a) plate and (b) compression corner at $t = 2.8$: (*1*) perturbed flow in the plane of symmetry $z = 0$, (*2*) z-averaged perturbed flow, (*3*) laminar flow, and (*4*) correlation for the turbulent boundary layer.

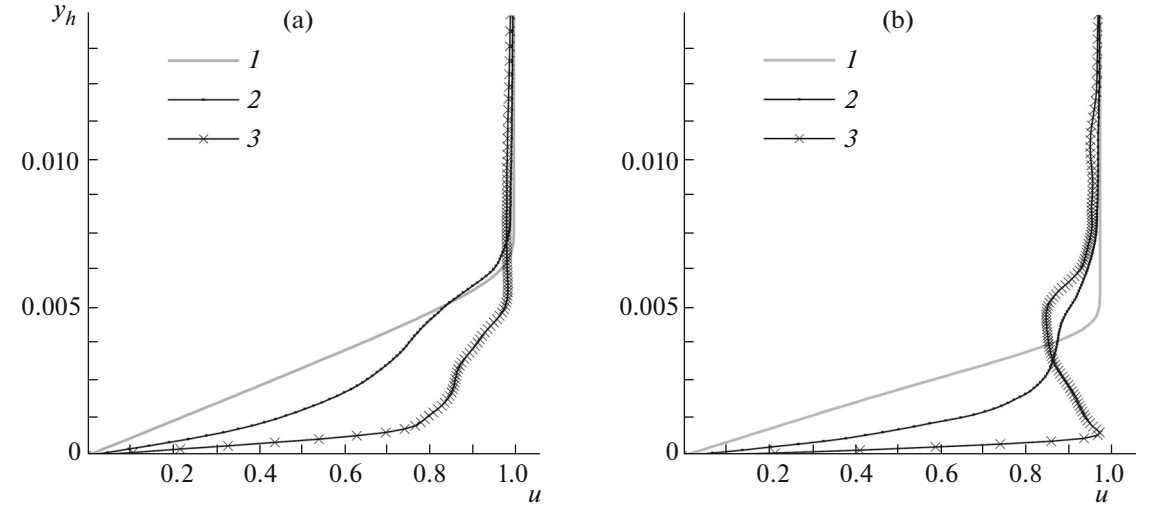

Fig. 18. Velocity profiles in the cross section $x = 1.4$ at $t = 2.8$ for the (a) plate and (b) compression corner: (*1*) laminar unperturbed boundary layer, (*2*) z-averaged profile, and (*3*) flow in the plane of symmetry ($z = 0$).

CONCLUSIONS

A DNS method for computing three-dimensional unsteady disturbances leading to a hypersonic laminar–turbulent transition was proposed and implemented. The simulation relies on solving the full three-dimensional unsteady Navier–Stokes equations. The numerical technique is intended for distributed-memory multiprocessor supercomputers and is based on a second-order fully implicit monotone approximation scheme and a modified Newton–Raphson method for solving nonlinear difference equations. The implemented technique was shown to be well scalable on 128, 256, and 512 processor cores.

The DNS method was used to simulate three-dimensional disturbances propagating in a plate boundary layer at a free-stream Mach number of 5.35. Disturbances were introduced into the boundary layer by setting an unsteady boundary condition on the vertical velocity in the form of a short pulse propagating through a circular hole. This pulse gave rise to a three-dimensional wave packet propagating downstream in the boundary layer. It was shown that plane waves corresponding to the second unstable Mack mode dominate in the packet. Weak nonlinear effects related to the interaction plane and oblique waves were demonstrated. The results were found to agree with a similar numerical study [28], which validates the

present method as applied to the study of the development of three-dimensional disturbances in hypersonic boundary layers. The grid resolution used was 30 nodes per wavelength and 120 nodes across the boundary layer, which can be regarded as the necessary minimum for the proposed universal low-order method.

This method was used to study the development of three-dimensional disturbances in a flat-plate boundary layer and a compression-corner boundary layer at the free-stream Mach number $M = 5.37$. Characteristic structures of the disturbance field during the linear and nonlinear laminar–turbulent transition stages were detected. The viscous friction coefficient distributions on the laminar and transient segments of the body surface were obtained, which make it possible to determine the onset of LTT. Spectral characteristics were constructed in various cross sections, and a broadening of the spectrum corresponding to the LTT onset was demonstrated.

In the case of hypersonic flows, the proposed approach outperforms specialized high-order schemes, since it provides a tool for shock-capturing computations of complex-geometry flows with resolving all strong spatial inhomogeneities and avoids labor-consuming tuning. The increased resource requirements for the method are balanced by its efficient parallel algorithm and the availability of high-performance supercomputers.

ACKNOWLEDGMENTS

This work was performed at the Moscow Institute of Physics and Technology and was supported by the Russian Science Foundation (project no. 14-19-00821, numerical study and analysis of the results) and by the Russian Foundation for Basic Research (project no. 17-08-00969, the development of the algorithm and codes for numerical simulation). The computations were carried out using high-performance computational resources of the Federal Computing Center at the National Research Center "Kurchatov Institute" (http://computing.kiae.ru/).

REFERENCES

1. A. Fedorov, "Transition and stability of high-speed boundary layers," Annu. Rev. Fluid Mech. **43**, 79–95 (2011).
2. A. D. Kosinov, N. V. Semionov, S. G. Shevelkov, and O. I. Zinin, "Experiments on the nonlinear instability of supersonic boundary layers," in *Nonlinear Instability of Nonparallel Flows,* Ed. by T. Valentine, S. P. Lin, and R. C. Philips (Springer, Berlin, 1994), pp. 196–205.
3. A. D. Kosinov and A. Tumin, "Resonance interaction of wave trains in supersonic boundary layer," in *Nonlinear Instability and Transition in Three-Dimensional Boundary Layers*, Ed. by P. W. Duck and P. Hall (Kluwer Academic, Berlin, 1996), pp. 379–388.
4. C. S. J. Mayer and H. F. Fasel, "Investigation of asymmetric subharmonic resonance in a supersonic boundary layer at Mach 2 using DNS," AIAA Paper, No. 2008-0591 (2008).
5. D. W. Laddon and S. P. Schneider, "Measurements of controlled wave packets at Mach 4 on a cone at angle of attack," AIAA Paper, No. 1998-0436 (1998).
6. A. N. Shiplyuk, D. A. Bountin, A. A. Maslov, and N. Chokani, "Nonlinear mechanisms of the initial stage of the laminar–turbulent transition at hypersonic velocities," J. Appl. Mech. Tech. Phys. **44** (5), 654–659 (2003).
7. D. Bountin, A. Shiplyuk, and A. Maslov, "Evolution of nonlinear processes in a hypersonic boundary layer on a sharp cone," J. Fluid Mech. **611**, 427–442 (2008).
8. K. M. Casper, S. J. Beresh, and S. P. Schneider, "Pressure fluctuations beneath turbulent spots and instability wave packets in a hypersonic boundary layer," AIAA Paper, No. 2011-372 (2011).
9. K. Casper, S. Beresh, and S. Schneider, "Characterization of controlled perturbations in a hypersonic boundary layer," AIAA Paper, No. 2012-281 (2012).
10. A. Chou and S. P. Schneider, "Time-frequency analysis of boundary layer instabilities generated by free-stream laser perturbations," AIAA Paper, No. 2015-3076 (2015).
11. X. Zhong and X. Wang, "Direct numerical simulation on the receptivity, instability, and transition of hypersonic boundary layers," Annu. Rev. Fluid Mech. **44**, 527–561 (2012).
12. H. F. Fasel, "Numerical simulation of transition in hypersonic boundary layers," DTIC Document., No. ADA563832 (2011).
13. I. V. Yegorov and O. L. Zaitsev, "An approach to the numerical solution of the bidimensional Navier–Stokes equations using the direct calculation method," USSR Comput. Math. Math. Phys. **31** (2), 80–89 (1991).
14. I. V. Yegorov and D. V. Ivanov, "The use of fully implicit monotone schemes to model plane internal flows," Comput. Math. Math. Phys. **36** (12), 1717–1730 (1996).

15. I. V. Egorov, A. V. Fedorov, and V. G. Soudakov, "Direct numerical simulation of supersonic boundary-layer receptivity to acoustic disturbances," AIAA Paper, No. 2005-97 (2005).
16. I. V. Egorov, A. V. Fedorov, and V. G. Soudakov, "Direct numerical simulation of disturbances generated by periodic suction-blowing in a hypersonic boundary layer," Theor. Comput. Fluid Dyn. **20** (1), 41–54 (2006).
17. I. V. Egorov, A. V. Novikov, and A. V. Fedorov, "Numerical modeling of the disturbances of the separated flow in a rounded compression corner," Fluid Dyn. **41** (4), 521–530 (2006).
18. I. V. Egorov, A. V. Novikov, and A. V. Fedorov, "Numerical simulation of stabilization of the boundary layer on a surface with a porous coating in a supersonic separated flow," J. Appl. Mech. Tech. Phys. **48** (2), 176–183 (2007).
19. D. Bountin, T. Chimitov, A. Maslov, A. Novikov, I. Egorov, A. Fedorov, and S. Utyuzhnikov, "Stabilization of a hypersonic boundary layer using a wavy surface," AIAA J. **51** (5), 1203–1210 (2013).
20. S. K. Godunov, A. V. Zabrodin, M. Ya. Ivanov, A. N. Kraiko, and G. P. Prokopov, *Numerical Solution of Multidimensional Problems in Gas Dynamics* (Nauka, Moscow, 1976) [in Russian].
21. P. L. Roe, "Approximate Riemann solvers, parameter vectors, and difference schemes," J. Comput. Phys. **43**, 357–372 (1981).
22. G. S. Jiang and C. W. Shu, "Efficient implementation of weighted ENO schemes," J. Comput. Phys. **126**, 202–228 (1996).
23. T. Kh. Karimov, "On some iteration methods for solving nonlinear equations in a Hilbert space," Dokl. Akad. Nauk SSSR **269** (5), 1038–1046 (1983).
24. Y. Saad and M. H. Shultz, "GMRes: A generalized minimal residual algorithm for solving nonsymmetric linear systems," SIAM J. Sci. Stat. Comput. **7** (3), 856–869 (1986).
25. I. Yu. Babaev, V. A. Bashkin, and I. V. Egorov, "Numerical solution of the Navier–Stokes equations using variational iteration methods," Comput. Math. Math. Phys. **34** (11), 1455–1462 (1994).
26. CFD General Notation System (2016). http://cgns.org/. Accessed November 15, 2016.
27. S. Balay, J. Brown, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, PETSc Web page (2016). http://www.mcs.anl.gov/petsc. Accessed November 15, 2016.
28. J. Sivasubramanian and H. F. Fasel, "Transition initiated by a localized disturbance in a hypersonic flat-plate boundary layer," AIAA Paper, No. 2011-374 (2011).
29. K. J. Franko, R. Bhaskaran, and S. K. Lele, "Direct numerical simulation of transition and heat-transfer overshoot in a Mach 6 flat plate boundary layer," AIAA Paper, No. 2011-3874 (2011).

Translated by I. Ruzanova