ISSN 0965-5425, Computational Mathematics and Mathematical Physics, 2009, Vol. 49, No. 11, pp. 1869–1884. © Pleiades Publishing, Ltd., 2009.
Original Russian Text © T.G. Elizarova, E.V. Shil'nikov, 2009, published in Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 2009, Vol. 49, No. 11, pp. 1953–1969.

Numerical Analysis of a Quasi-Gasdynamic Algorithm as Applied to the Euler Equations

T. G. Elizarova and E. V. Shil'nikov

Institute for Mathematical Modeling, Russian Academy of Sciences, Miusskaya pl. 4a, Moscow, 125047 Russia
e-mail: telizar@mail.ru, shiva@imamod.ru

Received January 19, 2009; in final form, February 2, 2009

Abstract—The computational properties of a quasi-gasdynamic algorithm are analyzed as applied to strong-shock Riemann problems and acoustic disturbance propagation. It is shown that a unified treatment can be given to these problems in the framework of the algorithm. Stability conditions are numerically obtained, and the error and computational complexity of the difference scheme are estimated.

DOI: 10.1134/S0965542509110050

Key words: quasi-gasdynamic algorithm, accuracy, stability, computational complexity, Riemann problem, acoustic disturbances, Euler equations.

INTRODUCTION

As a rule, numerical algorithms for gas flow simulation are constructed by using progressively more complex models. First, an effective method is proposed for solving simplified equations, for example, the advection or Burgers equation. Then the method is extended to one-dimensional gasdynamic equations; next, to a system involving dissipation; then, to multidimensional problems, unstructured meshes, and so on, depending on the objective. When the quasi-gasdynamic (QGD) equations were constructed, a numerical algorithm was first proposed for the general case, namely, for viscous unsteady flows in three dimensions. Moreover, the corresponding system of equations was written out in an invariant form. Later, the numerical algorithm was adapted for the numerical simulation of two- and one-dimensional problems on uniform grids. Due to this approach (from general to specific), one-dimensional flows have received little attention thus far.

In this paper, the stability and accuracy of a numerical algorithm based on the QGD equations are numerically studied as applied to Riemann problems. The computational complexity of the algorithm is estimated. Additionally, the algorithm is applied to a Riemann problem with large initial drops in density and pressure and to acoustic disturbance propagation. These examples are sophisticated tests and demonstrate the wide applicability of the numerical algorithm. A representative set of test computations based on the QGD algorithm can be found in [1].

Note that the QGD algorithm (see, e.g., [2, 3]) and related kinetically consistent difference schemes [4] have been successfully applied to the numerical simulation of a wide variety of viscous compressible gas flows in two and three dimensions.

1. QUASI-GASDYNAMIC EQUATIONS

According to [2, 3, 5], the QGD equations with allowance for external forces F_i and external heat sources Q are written in traditional notation as conservation laws

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x_i} j_{mi} = 0, \tag{1}$$

$$\frac{\partial}{\partial t}\rho u_i + \frac{\partial}{\partial x_j} j_{mj} u_i + \frac{\partial}{\partial x_i} p = \rho_* F_i + \frac{\partial}{\partial x_j}\Pi_{ji}, \tag{2}$$

$$\frac{\partial}{\partial t}\rho\left(\frac{u^2}{2} + \varepsilon\right) + \frac{\partial}{\partial x_i} j_{mi}\left(\frac{u^2}{2} + \varepsilon + \frac{p}{\rho}\right) + \frac{\partial}{\partial x_i} q_i = j_{mi} F_i + \frac{\partial}{\partial x_i}\Pi_{ij} u_j + Q, \tag{3}$$

where

$$j_{mi} = \rho(u_i - w_i), \tag{4}$$

$$w_i = \frac{\tau}{\rho}\left(\frac{\partial}{\partial x_j}\rho u_i u_j + \frac{\partial}{\partial x_i}p - \rho F_i\right), \tag{5}$$

$$\rho_* = \left(\rho - \tau\frac{\partial}{\partial x_k}\rho u_k\right), \tag{6}$$

$$\Pi_{ij} = \Pi_{\mathrm{NS}ij} + \tau\rho u_i\left(u_k\frac{\partial}{\partial x_k}u_j + \frac{1}{\rho}\frac{\partial}{\partial x_j}p - F_j\right) + \tau\delta_{ij}\left(u_k\frac{\partial}{\partial x_k}p + \gamma p\frac{\partial}{\partial x_k}u_k - (\gamma-1)Q\right), \tag{7}$$

$$q_i = q_{\mathrm{NS}i} - \tau\rho u_i\left(u_j\frac{\partial}{\partial x_j}\varepsilon + pu_j\frac{\partial}{\partial x_j}\frac{1}{\rho} - \frac{Q}{\rho}\right). \tag{8}$$

Here and below, summation over repeated indices is implied. The coupling equations have the form

$$p = \rho RT, \quad \varepsilon = \frac{p}{\rho(\gamma-1)}. \tag{9}$$

The heat flux and the Navier–Stokes viscous stress tensor are calculated by the formulas

$$q_{\mathrm{NS}i} = -\kappa\frac{\partial}{\partial x_i}T, \tag{10}$$

$$\Pi_{\mathrm{NS}ij} = \mu\left(\frac{\partial}{\partial x_i}u_j + \frac{\partial}{\partial x_j}u_i - \frac{2}{3}\delta_{ij}\frac{\partial}{\partial x_k}u_k\right). \tag{11}$$

$$\mu = \mu_\infty\left(\frac{T}{T_\infty}\right)^\omega, \quad \kappa = \mu\frac{\gamma R}{(\gamma-1)\mathrm{Pr}}, \quad \tau = \frac{\mu}{p\mathrm{Sc}}, \tag{12}$$

where μ is the dynamic viscosity, κ is the thermal conductivity coefficient, τ is a relaxation parameter having the dimension of time, γ is the adiabatic index, Pr is the Prandtl number, and Sc is the Schmidt number. The well-posedness and physical adequacy of this system and its simplified versions were examined in [2–7].

Note that the additional dissipative terms with the coefficient τ in the QGD equations, namely, those in mass flux (5), viscous stress tensor (7), and heat flux (8), vanish in flow regions where the solution satisfies the stationary Euler equations.

The balance equation for the entropy s in the QGD system has the form

$$\frac{\partial}{\partial t}\rho s + \frac{\partial}{\partial x_i}j_{mi}s = -\frac{\partial}{\partial x_i}\frac{q_i}{T} + \kappa\left(\frac{1}{T}\frac{\partial}{\partial x_i}T\right)^2 + \frac{\Phi}{T}, \tag{13}$$

where the dissipation function Φ can be represented as

$$\begin{aligned}\Phi = {} & \frac{\Pi_{\mathrm{NS}ij}\Pi_{\mathrm{NS}ij}}{2\mu} + \tau\rho\left(u_k\frac{\partial}{\partial x_k}u_i + \frac{1}{\rho}\frac{\partial}{\partial x_i}p - F_i\right)^2 + \frac{\tau p}{\rho^2}\left(\frac{\partial}{\partial x_i}\rho u_i\right)^2 \\ & + \frac{\tau\rho}{\varepsilon}\left(u_i\frac{\partial}{\partial x_i}\varepsilon + \frac{p}{\rho}\frac{\partial}{\partial x_i}u_i - \frac{Q}{2\rho}\right)^2 + Q\left(1 - \frac{(\gamma-1)\tau Q}{4p}\right).\end{aligned} \tag{14}$$

The assumption that the last term in (14) is nonnegative imposes a constraint on the value of τ depending on the intensity of external heat sources.[1]

2. NUMERICAL ALGORITHM FOR ONE-DIMENSIONAL FLOWS

For numerical convenience, system (1)–(3) is reduced to a dimensionless form by using the reference density ρ_0, the reference speed of sound $c_0 = \sqrt{\gamma RT_0}$, and the reference length L. The nondimensionalization procedure does not change the form of the equations.

Below, the QGD equations are solved for one-dimensional flows without allowance for external forces or heat sources ($Q = 0$, $F_i = 0$). In this case, system (1)–(3) substantially simplifies.

[1] The final form of the last two terms was obtained by A.A. Zlotnik (personal communication).

We introduce a uniform grid in x with the mesh size h and a grid in time with the step Δt.

When the Euler equations are solved numerically on the basis of system (1)–(3), all the dissipative terms (i.e., those with the coefficients μ, κ, and τ) are treated as regularizers. In this case, the relaxation parameter, viscosity, and thermal conductivity coefficients are interrelated. In dimensionless form, they are calculated by the formulas

$$\tau = \alpha\frac{h}{c}, \quad \mu = \tau p \mathrm{Sc}, \quad \kappa = \frac{\tau p \mathrm{Sc}}{\mathrm{Pr}(\gamma-1)}, \tag{15}$$

where $c = \sqrt{\gamma p/\rho}$ is the local speed of sound. Moreover, Pr and Sc are treated as numerical coefficients for the tuning of artificial viscosity, if necessary.

The flow parameters (i.e., the velocity, density, pressure) are determined at integer nodes, while the fluxes are calculated at half-integer nodes. According to [3, 6], the problem is approximated by an explicit homogeneous difference scheme on a three-point spatial stencil:

$$\hat{\rho}_i = \rho_i - \frac{\Delta t}{h}(j_{mi+1/2} - j_{mi-1/2}), \tag{16}$$

$$\widehat{\rho_i u_i} = \rho_i u_i + \frac{\Delta t}{h}[(\Pi_{i+1/2} - \Pi_{i-1/2}) - (p_{i+1/2} - p_{i-1/2}) - (j_{mi+1/2}u_{i+1/2} - j_{mi-1/2}u_{i-1/2})], \tag{17}$$

$$\begin{aligned}\hat{E}_i = E_i + \frac{\Delta t}{h}\Bigg[&(\Pi_{i+1/2}u_{i+1/2} - \Pi_{i-1/2}u_{i-1/2}) - (q_{i+1/2} - q_{i-1/2}) \\ &- \left(\frac{j_{mi+1/2}}{\rho_{i+1/2}}(E_{i+1/2} + p_{i+1/2}) - \frac{j_{mi-1/2}}{\rho_{i-1/2}}(E_{i-1/2} + p_{i-1/2})\right)\Bigg],\end{aligned} \tag{18}$$

$$p_i = (\gamma-1)\left(E_i - \frac{\rho_i u_i^2}{2}\right).$$

Here, $E_i = \rho_i u_i^2/2 + p_i/(\gamma-1)$ is the total energy of a unit of volume. The discrete mass flux $j_{mi+1/2}$ is calculate as

$$w_{i+1/2} = \frac{\tau_{i+1/2}}{\rho_{i+1/2}}\frac{1}{h}(\rho_{i+1}u_{i+1}^2 + p_{i+1} - \rho_i u_i^2 - p_i), \tag{19}$$

$$j_{mi+1/2} = \rho_{i+1/2}(u_{i+1/2} - w_{i+1/2}). \tag{20}$$

The discrete expression for $\Pi_{i+1/2}$ is given by the formulas

$$\mu_{i+1/2} = \tau_{i+1/2}p_{i+1/2}\mathrm{Sc}, \tag{21}$$

$$\Pi_{\mathrm{NS}i+1/2} = \frac{4}{3}\mu_{i+1/2}\frac{u_{i+1} - u_i}{h}, \tag{22}$$

$$w^*_{i+1/2} = \tau_{i+1/2}\left(\rho_{i+1/2}u_{i+1/2}\frac{u_{i+1} - u_i}{h} + \frac{p_{i+1} - p_i}{h}\right), \tag{23}$$

$$R_{i+1/2} = \tau_{i+1/2}\left(u_{i+1/2}\frac{p_{i+1} - p_i}{h} + \gamma p_{i+1/2}\frac{u_{i+1} - u_i}{h}\right), \tag{24}$$

$$\Pi_{i+1/2} = (\Pi_{\mathrm{NS}} + uw^* + R)_{i+1/2}. \tag{25}$$

The heat flux $q_{i+1/2}$ is calculated as

$$R^q_{i+1/2} = (\tau\rho)_{i+1/2}\left(\frac{u_{i+1/2}}{\gamma-1}\frac{(p/\rho)_{i+1} - (p/\rho)_i}{h} + (pu)_{i+1/2}\frac{1/\rho_{i+1} - 1/\rho_i}{h}\right), \tag{26}$$

$$q_{i+1/2} = -(\tau p)_{i+1/2}\frac{\gamma\mathrm{Sc}}{\mathrm{Pr}(\gamma-1)}\frac{(p/\rho)_{i+1} - (p/\rho)_i}{h} - u_{i+1/2}R^q_{i+1/2}. \tag{27}$$

The accuracy and stability of the QGD algorithm are analyzed below as applied to a Riemann problem.

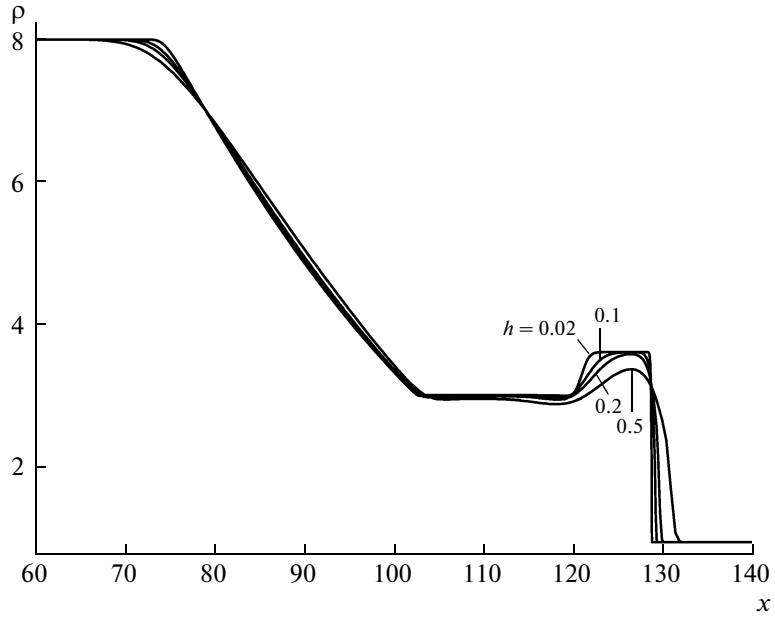

Fig. 1.

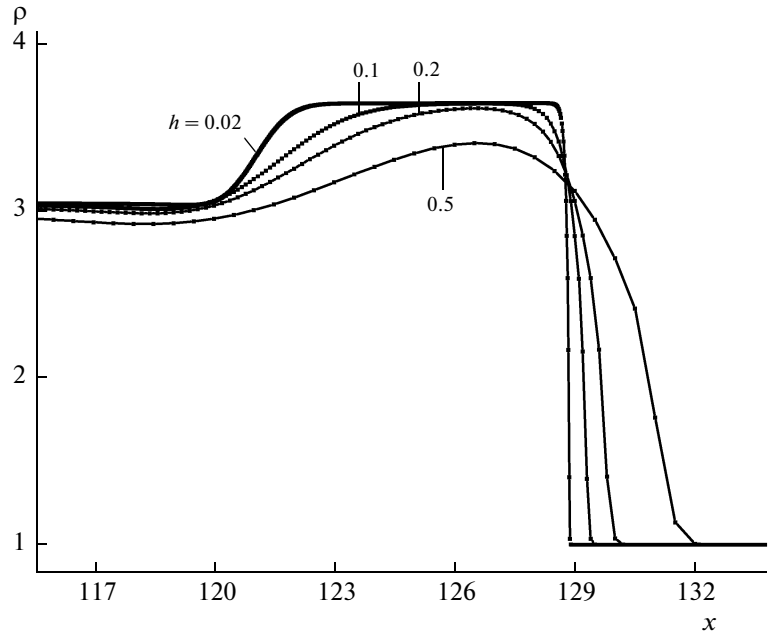

Fig. 2.

3. ERROR ESTIMATE FOR THE NUMERICAL ALGORITHM

Formally, the order of scheme (16)–(18) in space is $O(\alpha h)$. The results presented in [1] confirm that a decrease in α is equivalent to spatial mesh refinement.

The accuracy of the scheme is estimated via Riemann problem computations. As initial data, we set the values on left (l) and right (r) intervals of the computational domain: $(\rho_l, u_l, p_l) = (8, 0, 480)$ and $(\rho_r, u_r, p_r) = (1, 0, 1)$ with $\gamma = 5/3$. The problem was solved in the domain (0, 200) with the discontinuity located at the point $x_0 = 100$. The dissipative coefficients were calculated for $\mathrm{Pr} = 2/3$, $\mathrm{Sc} = 1$, and $\alpha = 0.4$. The time step was specified by $\beta = 0.5$.

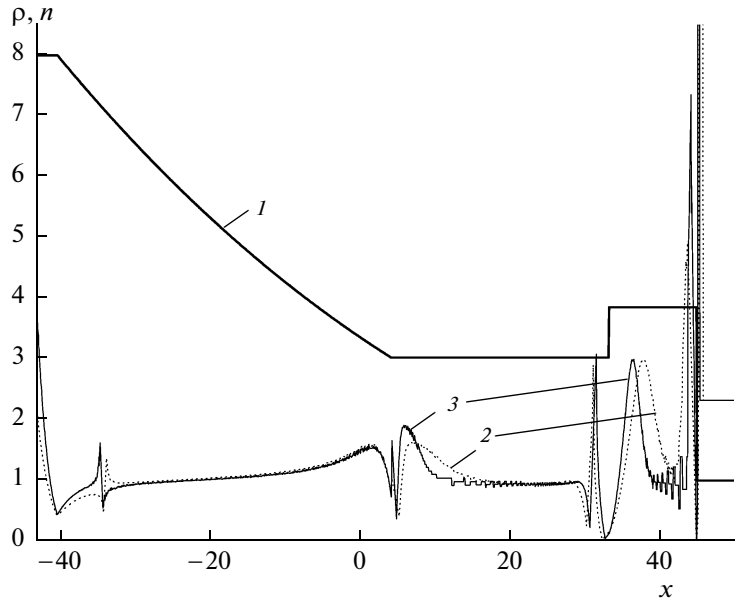

Fig. 3.

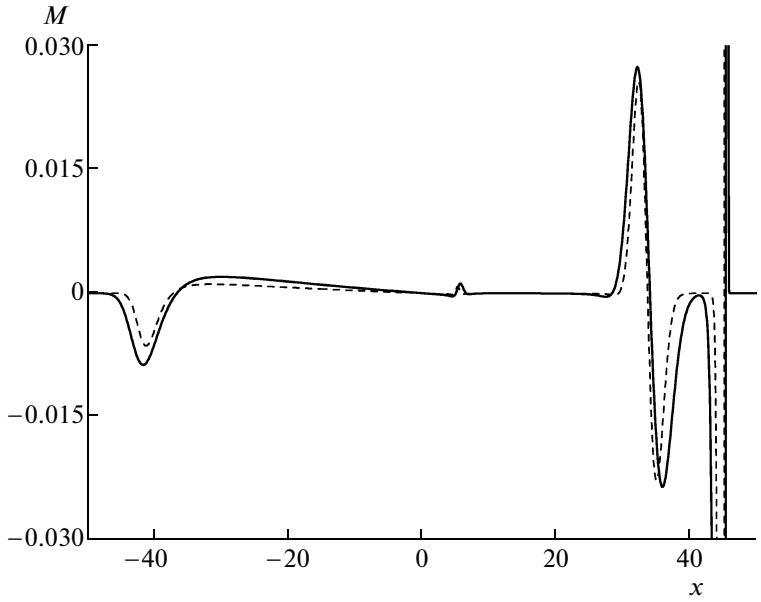

Fig. 4.

Figures 1 and 2 show the density distribution ρ and its fragment at the time $t_{\text{fin}} = 4$. The results were obtained on a sequence of grids with the mesh sizes $h = 0.5$, 0.2, 0.1, and 0.02. It can be seen that the numerical solution converges monotonically as the spatial grid is refined.

The order of accuracy of scheme (16)–(18) can be estimated using the results obtained on a sequence of refined meshes. Let u be the exact solution of the problem. In the case under study, this is a self-similar solution [8]. Let u_1 and u_2 be the discrete solutions obtained on grids with the mesh sizes h and $2h$, respectively. Then, according to [9, 10], if the solution of the problem is sufficiently smooth (i.e., twice differentiable), then the error of in the grid solution has the form

$$u_1 - u = A(2h)^n, \quad u_2 - u = Ah^n, \tag{28}$$

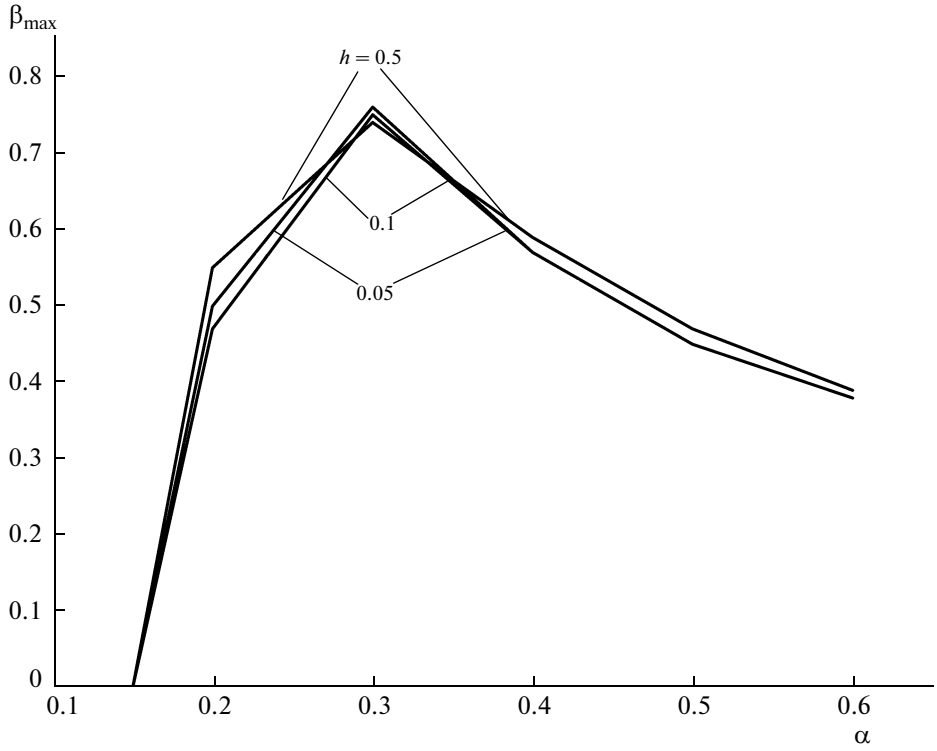

Fig. 5.

where n is the order of accuracy of the difference scheme and A is a constant depending on the derivatives of the solution. Then the order of accuracy is given by the formula

$$n = \log_2 \frac{u_1 - u}{u_2 - u}. \tag{29}$$

Figure 3 displays a self-similar solution of the problem in the case when the initial discontinuity is at the point $x_0 = 0$ (line *1*) and the order of accuracy n of the numerical algorithm is determined according to (29). The order of accuracy was calculated for two sequences of grids: n_1 (curve *2*) for $h = 0.05$ and $2h = 0.1$, and n_2 (curve *3*) for $h = 0.025$ and $2h = 0.05$.

In both cases, the results show that the effective order of accuracy of the QGD algorithm is $0.5 \le n \le 2$ in areas where the solution is smooth, i.e., within the rarefaction wave ($-35 < x < 5$) and behind the contact discontinuity ($5 < x < 30$). In areas where the flow is nearly steady and inviscid, the solution is approximately described by the stationary Euler equations, and the Euler complexes contributing to QGD additional terms are small. In these areas, the accuracy of the scheme increases.

In areas where the solution is discontinuous and at points where the discrete solution coincides with the exact one ($u_1 = u$ or $u_2 = u$), this error estimation algorithm is not correct, which is manifested by peaks and negative values of n.

Figure 4 shows the distribution of $M = \partial \rho w/\partial x$ for the continuity equation. In difference form, according to (16), it can be written as

$$M_i = \frac{1}{h}(\rho_{i+1/2} w_{i+1/2} - \rho_{i-1/2} w_{i-1/2}).$$

Here, the discrete values of w are calculated using (19). The computations were performed on grids with $h = 0.05$ (solid curve) and $h = 0.025$ (dashed curve). It can be seen that M_i is small in smooth-solution areas and is close to zero between the rarefaction wave and the contact discontinuity. As a result, the actual order of accuracy of the scheme in the corresponding domains increases and correlates with the distributions of n presented in Fig. 3. The maximum values of M_i in the shock wave region ($\max M_i \sim 40$, $\min M_i \sim 20$, not shown in the figure). The next (in value) peaks are observed within the contact discontinuity, and two additional small peaks of M_i can be seen at the sonic points bounding the rarefaction wave.

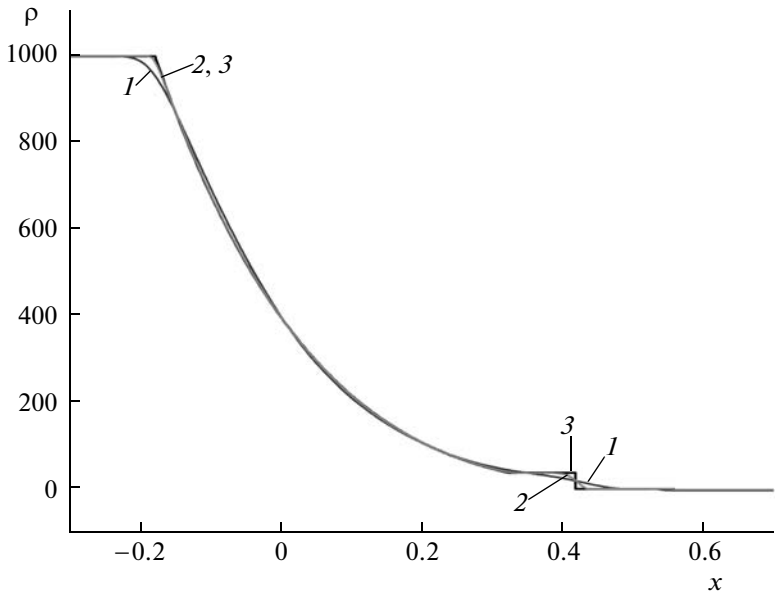

Fig. 6.

Thus, the value of QGD dissipation, which determines the stability of the difference algorithm, automatically adapts to the solution of the problem and depends on its local properties.

3. STABILITY ANALYSIS OF THE NUMERICAL ALGORITHM

Numerical algorithm (16)–(18) is a conditionally stable explicit difference scheme. According to numerical experience and the physics considerations underlying the kinetic-model-based derivation of the QGD equations, the constraint on the time step for these algorithms is determined by the Courant condition

$$\Delta t = \beta \min\left(\frac{h}{|u| + c}\right), \tag{30}$$

where $0 < \beta(\alpha) < 1$ is a numerical coefficient.

In [7] the method of energy inequalities was used to derive a sufficient stability condition for the QGD algorithm and corresponding theorems were proved. A one-dimensional flow was simulated by applying a one-dimensional difference scheme with a constant mesh size based on the Euler equations in the acoustic approximation. The resulting Courant condition was

$$\Delta t \leqslant \beta \frac{h}{c_*}, \tag{31}$$

where $c_* = \sqrt{\gamma R T_*}$ is the spatially averaged speed of sound at the initial time and the coefficient β is given by

$$\beta = \min(\beta_A, \beta_B, \beta_C), \quad \beta_A = \frac{A}{A^*}, \quad \beta_B = \frac{B}{B^*}, \quad \beta_C = \frac{C}{C^*}. \tag{32}$$

Here, A, B, C, A^*, B^*, and C^* are determined by γ, Pr, and α involved in the artificial dissipation formula in (15). For Sc = 1, these parameters are

$$A = \frac{\alpha}{\gamma}, \quad B = \frac{\alpha}{\gamma}\left(\frac{4}{3} + \gamma\right), \quad C = \frac{\alpha}{\Pr(\gamma - 1)}, \tag{33}$$

$$A^* = 2\gamma A^2 + 2(\gamma - 1)AC + \gamma A + B + \frac{1}{2\gamma}, \tag{34}$$

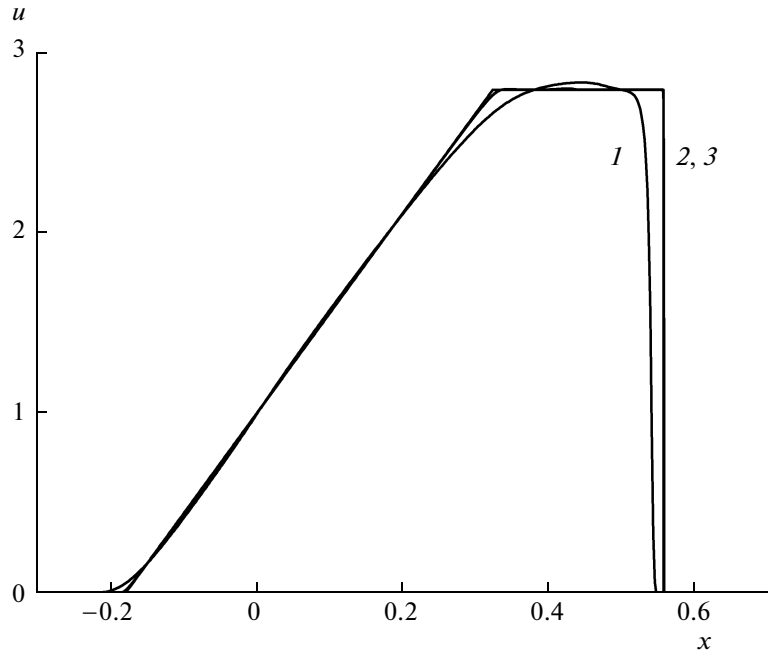

Fig. 7.

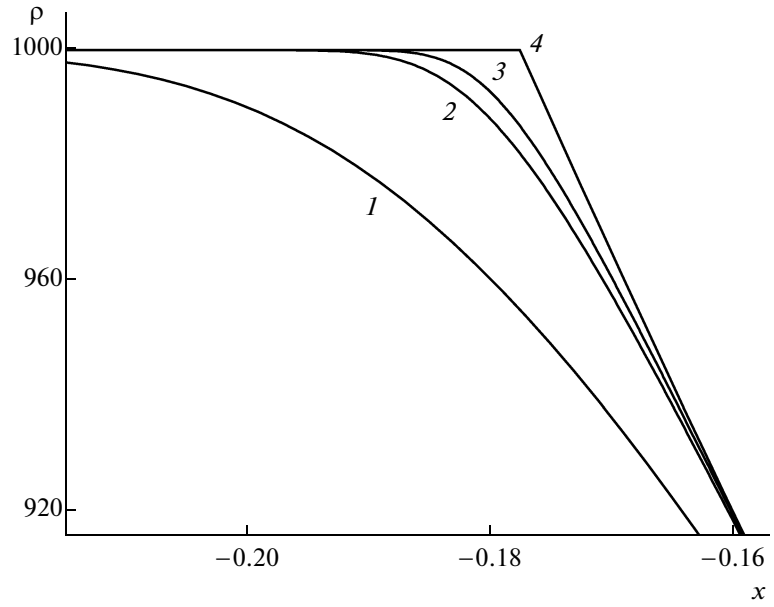

Fig. 8.

$$B^* = 2B^2 + A + \frac{B}{\gamma} + \frac{(\gamma - 1)C}{\gamma} + \frac{1}{2}, \tag{35}$$

$$C^* = (\gamma - 1)C(2A + 2C + 1). \tag{36}$$

The most severe constraint is that determined by β_A. For $\gamma = 5/3$, $\mathrm{Pr} = 2/3$, and $\alpha = 0.5$, the difference algorithms is stable for $\beta \sim 0.12$ (see [7]).

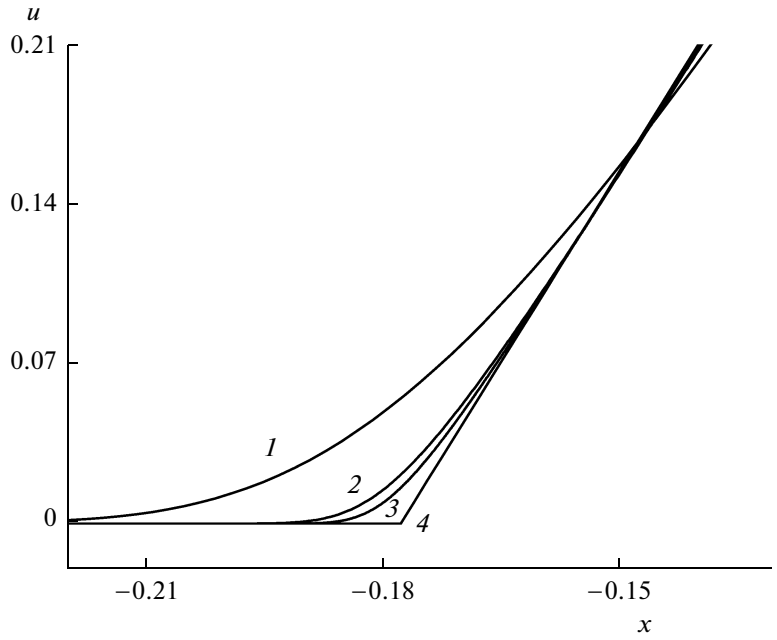

Fig. 9.

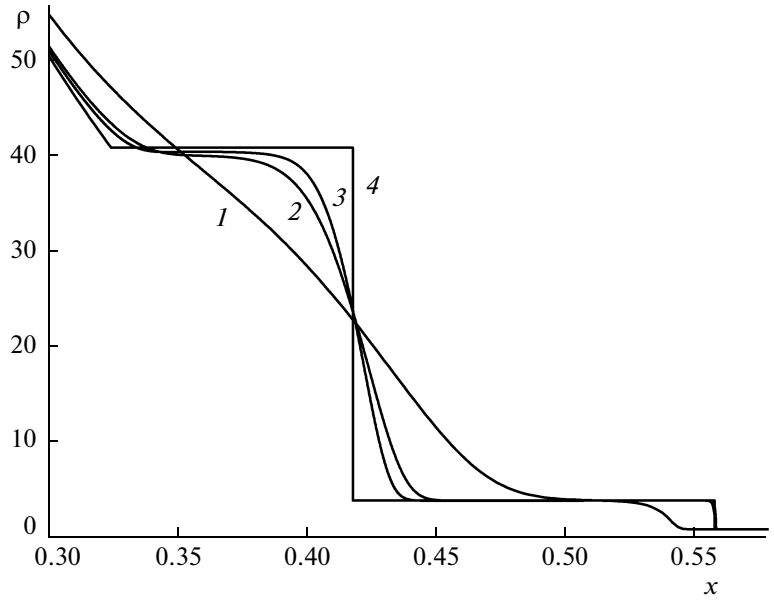

Fig. 10.

The validity of Courant condition (30) was numerically studied in the case of the above problem.

Specifically, for each α in the range 0.2–0.5 in (15), varying the time step, i.e., the coefficient β in (30), we determined the maximum possible time step corresponding to no oscillations behind the shock front and the entropy wake (dip) generated in the density profile behind the rarefaction wave. The resulting maximum possible time step is shown in Fig. 5 for $h = 0.5$, 0.1, and 0.05.

The computations suggest that the stability condition for scheme (16)–(18) has the form of a Courant relation with β nearly independent of the spatial mesh size. The plot clearly shows that the time step depends on the regularization parameter. In this example, the function $\beta(\alpha)$ has a maximum at $\alpha_{max} = 0.3$,

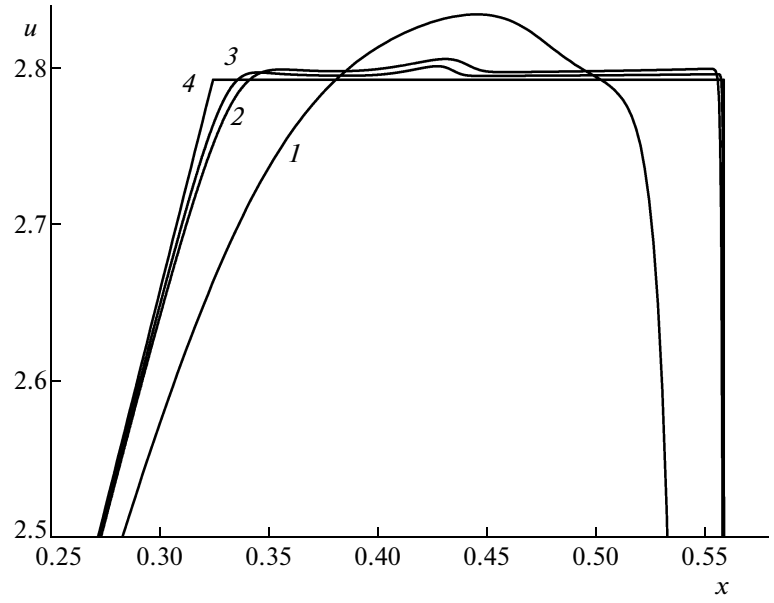

Fig. 11.

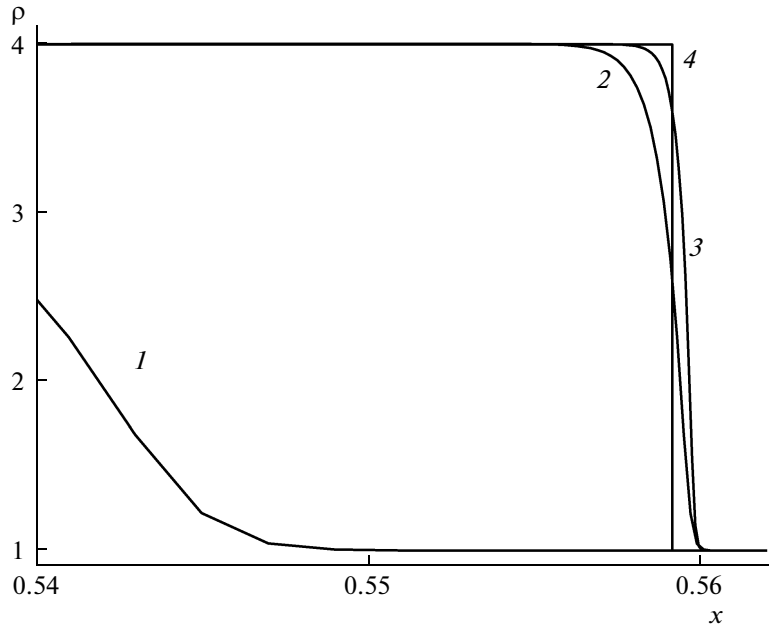

Fig. 12.

which corresponds to $\beta = 0.7$. The use of $\alpha > \alpha_{max}$ in numerical computations is unreasonable. For small $\alpha < 0.15$, the numerical algorithm is unstable.

In other problems, the function $\beta_{max}(\tau)$ can be different, but numerical experience suggests that the qualitatively features still persist.

To conclude, we note that, obtained for the linearized problem, stability condition (32) is more restrictive than that used in practice.

5. RIEMANN PROBLEM WITH A SUPERSTRONG DISCONTINUITY

To demonstrate further capabilities of the QGD algorithm, consider a Riemann problem with large drops in density and pressure. The initial conditions were specified as $(\rho_l, u_l, p_l) = (1000, 0, 1000)$

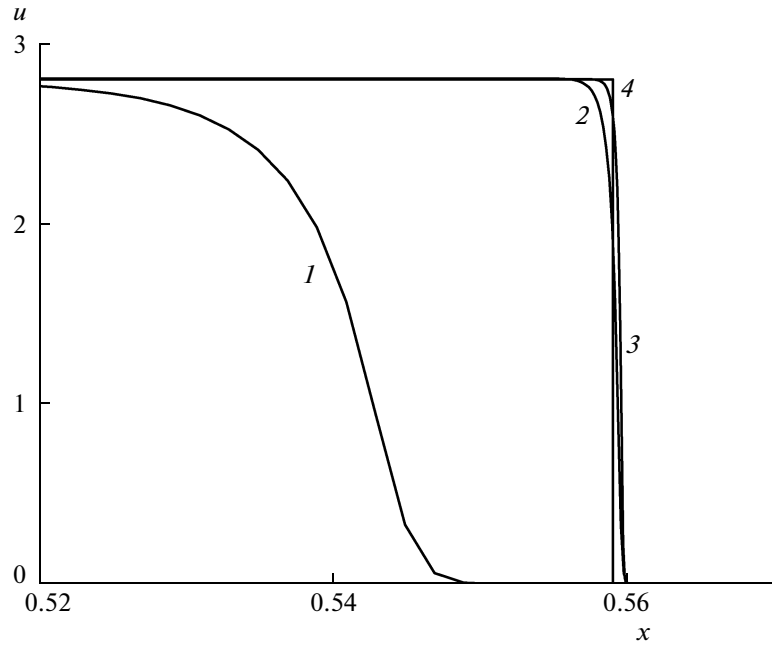

Fig. 13.

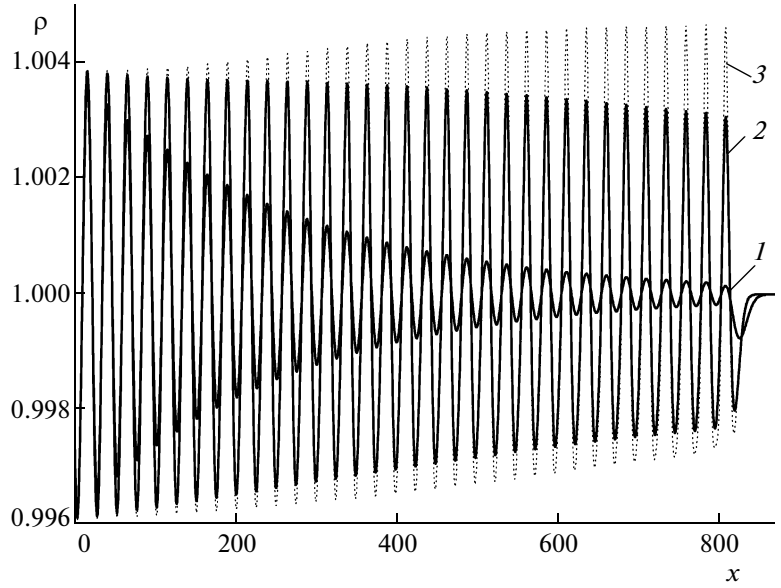

Fig. 14.

for $-0.3 < x < 0$, $(\rho_r, u_r, p_r) = (1, 0, 1)$ for $0 < x < 0.7$, and $\gamma = 1.4$. The computations were performed until the time $t_{\text{fin}} = 0.15$. This problem was used in [11] to compare in detail the capabilities of eight most popular high-order accurate difference schemes. Specifically, Godunov-type schemes and various high-order schemes with splitting and flux-corrected transport were considered.

The resulting optimal parameters for the QGD algorithm were $\alpha = 0.3$ and $\beta = 0.05$. The computations were performed with $\text{Pr} = 1$ and $\text{Sc} = 1$.

Figures 6 and 7 illustrate the convergence of ρ and u distributions with a refined grid in the entire computational domain for (*1*) $h = 0.002$ and (*2*) $h = 0.0001$. Curve *3* depicts the exact solution. Due to the large drops in the flow parameters, the solution near a sonic point, the contact discontinuity, and the shock

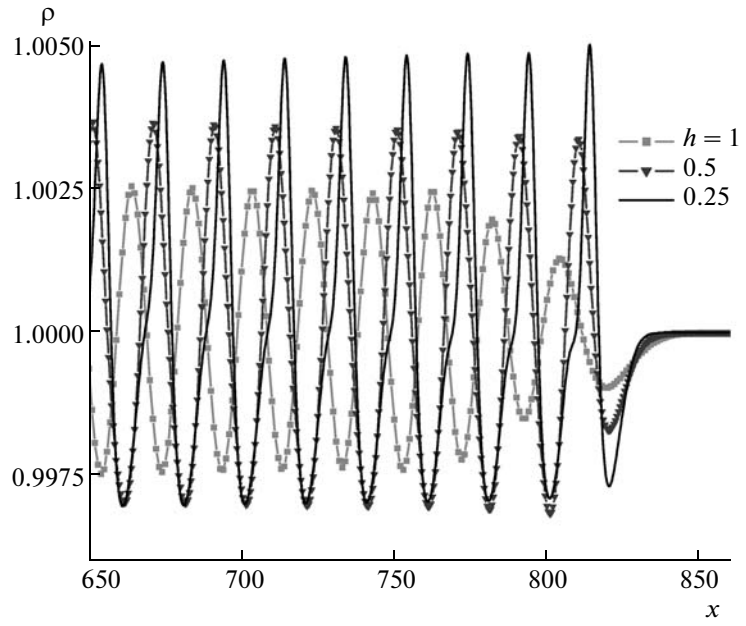

Fig. 15.

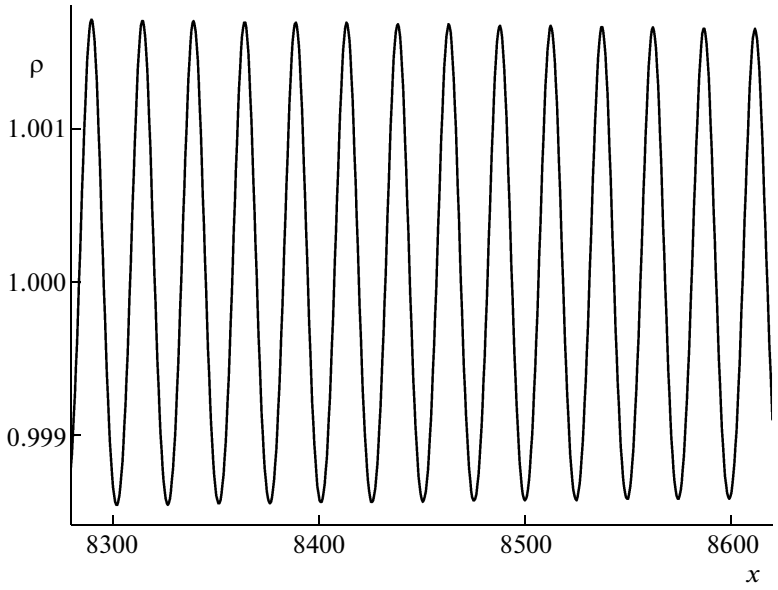

Fig. 16.

waves are shown separately. Specifically, Figs. 8 and 9 display fragments of the density and velocity distributions at the left sonic point. The same quantities are presented in the middle of the computational domain in Figs. 10 and 11 and across the shock wave in Figs. 12 and 13. In all these figures, curve *1* corresponds to $h = 0.002$; *2*, to $h = 0.0002$; *3*, to $h = 0.0001$; and *4* depicts the exact solution. These results clearly show that the discrete solution rapidly converges to the self-similar one as the spatial grid is refined.

An indicative characteristic of the solution of this problem is the velocity distribution across the shock wave. A comparison of Fig. 13 with the results presented in [11] for the same spatial grid with $h = 0.002$ suggests that the QGD algorithm on this grid is similar in accuracy to the difference schemes considered in [11]. Moreover, the QGD solution is to the left of the self-similar one, while the velocity profile produced by all the algorithms in [11] is to the right of the self-similar distribution. As the spatial mesh size

decreases, the accuracy of the QGD algorithm increases sharply and exceeds that of the methods analyzed in [11].

The numerical results, combined with those in [1], suggest that the QGD algorithms are superior when applied to flows simulation on fine grids, which is especially important in the context of wide use of high-performance computer systems.

6. ACOUSTIC DISTURBANCE PROPAGATION

A theoretical analysis and experience of using the QGD equations suggest that the numerical algorithms based on these equations are effective when as applied to the computation of unsteady and pulsating viscous compressible gas flows. Examples are the unsteady supersonic flow over a spiked body [12]; pulsating flows around a hollow cylinder [13]; oscillatory flows in cavities [14]; subsonic flows in the wake behind a cylinder, or von Karman vortex streets [15]; and chaotic flows over a backward-facing step, which simulative turbulent wakes near the bottom [16] and in the wake behind an obstacle [17]. Flows of these kinds are accompanied by the generation of sound oscillations, which are an important object of study in aeroacoustics. Traditionally, such problems are solved using special high-order accurate numerical methods, for example, fourth- or higher order accurate schemes in time and space (see [18] and, e.g., [19], respectively).

In view of what was said above, an interesting task is to examine the capabilities of the QGD algorithm as applied to the direct and homogeneous numerical simulation of acoustic disturbance generation in pulsating or turbulent flows and acoustic disturbance propagation at long distance away from the generation zone. An adequate treatment of acoustic disturbance propagation by using the QGD model in the acoustic approximation was demonstrated in [20].

The numerical results presented below for the propagation of weak sound oscillations were produced by the QGD algorithm as applied to the Euler equations.

As initial data, we used an unperturbed flow field with $\rho_0 = 1$, $p_0 = 1$, and $u_0 = 0$. The velocity of disturbance propagation was $c_0 = \sqrt{\gamma}$.

The condition on the left boundary of the domain was specified as a harmonic disturbance:

$$u(t, 0) = -\frac{A_0}{\sqrt{\gamma}}\sin(2\pi c_0 t/\lambda),$$
$$\rho(t, 0) = 1 - A_0\sin(2\pi c_0 t/\lambda), \quad p(t, 0) = 1 - A_0\sin(2\pi c_0 t/\lambda). \tag{37}$$

Mild (drift) boundary conditions were set on the right boundary of the domain: $\partial f/\partial x = 0$, where $f = (\rho, u, p)$. The wavelength of the acoustic disturbance was $\lambda = 20$, and the amplitude A_0 varied from 0.1 to 0.005. The parameters were specified as $\alpha = 0.2$, $\beta = 0.4$, and $\mathrm{Pr} = 1$. To be definite, we used $\gamma = 7/5$.

Figure 14 shows the numerical results obtained for a small disturbance with the amplitude $A_0 = 0.005$ depending on viscosity, i.e., for (*1*) $\mathrm{Sc} = 1$, (*2*) $\mathrm{Sc} = 0.1$, and (*3*) $\mathrm{Sc} = 0$. The last value corresponds to an inviscid flow. The computations were performed on a grid with $h = 0.5$. An increase in the viscosity and thermal conductivity coefficients (an increase in Sc) led to more intense damping of the oscillation amplitude. However, the oscillation frequency remained unchanged.

The influence of the spatial mesh size on the solution of the problem is shown in Fig. 15, which presents fragments of the density distribution for $\mathrm{Sc} = 0$ at $t = 700$. It can be seen that the damping of the wave amplitude is enhanced with increasing mesh size. On fine grids with $h = 0.5$ and 0.25, the shape of the original disturbance is distorted, which seems natural for the treatment of wave propagation based on the Euler equations (see [21]). A comparison of the figures shows that the phase error is not affected by the viscosity (determined by Sc) but depends on h.

The damping of acoustic density oscillations at long distances from the source can be evaluated by inspecting Fig. 16. The computations were performed with $\mathrm{Sc} = 0.01$ on a grid with $h = 0.5$ up to a distance in x of about 500 wavelengths and correspond to $t = 8000$.

The numerical results suggest that the QGD algorithm, which is first-order accurate in time and space, can be used to simulate acoustic disturbance propagation, including at long distances from the source. Due to the last circumstance, this algorithm is promising as applied to aeroacoustic problems.

7. ESTIMATION OF THE COMPUTATIONAL COMPLEXITY OF THE ALGORITHM

CPU time is an important characteristic to be considered when a numerical solution method is chosen for a particular problem. However, a direct comparison of CPU times taken by different numerical algo-

rithms as applied to the same problem is frequently not very instructive, which is caused by the variety of processors, operating systems, and optimization codes and by the degree of user-based optimization of the formulas involved. Therefore, the estimation of the computational complexity of an algorithm and its comparison to similar characteristics of alternative numerical approaches is an important aspect in choosing a numerical method.

Objective characteristics of the method's computational complexity are the number of arithmetic operations and the time required for data transfer from memory. These characteristics are used to estimate the absolute performance of modern computing systems, including single-processor single-core and multiprocessor multicore ones. To compare different algorithms, these characteristics can be evaluated without resorting to numerical implementations.

A computing unit can be schematically represented as consisting of main memory, fast-access cash memory, and a CPU (processor). Typical data determining the performance of modern computing units can be found, for example, in [22] (see also the references therein).

According to modern views, a processor executes one to four arithmetic operations (additions/subtraction, multiplication, or division) per cycle. The transfer rate of a word between the processor and the cash is about 15 cycles. Preparation for word transfer from the main memory to the cash (so-called latency) takes about 1000 cycles. When data are accessed not at random but rather as a beforehand arranged sequence or array, the data transfer rate between the main memory and the processor is close to the transfer rate of a word between the processor and the cash.

In gas flow simulation on structured meshes, data are regularly arranged in storage, they are transferred in large arrays, and the storage latency does not restrict the performance of the processor. Moreover, if the time required for processing a word in the CPU exceeds the time of its access from the cash, which takes 15 cycles, then the CPU time is directly determined by the number of arithmetic operations per grid point. By estimating these characteristics, we can compare the computational complexity of different algorithms.

While estimating the computational complexity of the QGD algorithm, we took into account the arithmetic operations of addition/subtraction, multiplication, division, and square root extraction involved in formulas (16)–(27), i.e., the operations involved un this algorithm. These operations were taken into account directly without optimization. Index-related and other auxiliary operations were ignored.

Consider the solution of the first equation, i.e., the computation of ρ_i at a new level. Specifically, the additional term $w_{i+1/2}$ to $u_{i+1/2}$ (computed by formula (19)) requires three additions, six multiplications, and one division. The computation of $\tau_{i+1/2}$ (by formula (15)) requires one multiplication, one division, and one root extraction. The computation of $j_{mi+1/2}$ (by formula (20)) requires one addition and one multiplication. The computation of ρ_i (by formula (16)) at a new time level requires two additions, one multiplication, and one division. Thus, the entire procedure for determining the density at a single grid point at a new level requires 10 additions, 17 multiplications, 5 divisions, and 2 square extractions, i.e., 34 arithmetical operations, or cycles.

The computation of $\Pi_{i+1/2}$ (according to scheme (21)–(25)) needs 9 additions, 13 multiplications, and 5 divisions. Therefore, the computation of $\widehat{\rho_i u_i}$ by formula (17) requires 24 additions, 30 multiplications, and 10 divisions, i.e., 64 cycles.

For the third equation, namely, Eq. (18), the computation of $q_{i+1/2}$ (by formulas (26) and (27)) requires 14 additions, 18 multiplications, and 20 divisions. The total energy computation at the point i at a new level requires 22 additions, 23 multiplications, and 22 divisions, i.e., altogether 67 cycles.

Thus, the solution of three equations (16)–(18) at a grid point requires 60 additions, 70 multiplications, 41 divisions, and 2 square root extractions, while the computation of a single time step at a spatial point takes about 180 cycles. Assuming that four arithmetic operations are executed per cycle, this result decreases by a factor of 4. However, when a sequence of points is computed, the amount of operations is halved, since all the fluxes for the point $i + 1/2$ are calculated twice: as the flux from the right at the point i and as the flux from the left at the point $i + 1$. Therefore, the number of operations can be reduced by optimizing the numerical code. The main loop in the algorithm involves no logical operations, operations at distant stencil points, or operations of access to single words in main memory, which substantially slow down the performance of the computing unit as a whole.

In the QGD algorithm, the computations at the point i at a new level consist of all the above operations executed for four variables u, ρ, p, and E on a three-point stencil, i.e., the values of these variables are computed at the points $i - 1$, i, and $i + 1$. When the next point $(i + 1)$ is computed, it suffices to extract from storage only four new values at the point $i + 2$, since the values at i and $i + 1$ are already known. The time taken for their extraction is $4 \times 15 = 60$ cycles, which does not exceed the time required for the transition

for the next time level (about 90 cycles). These two values determine the computational complexity of the QGD algorithm.

Thus, the time required for transferring a word from the cash is comparable to the time required for its processing in the CPU. Therefore, the QGD algorithm is efficient from the viewpoint of the performance of the computing unit.

Similar relations between the times required for data transfer and their processing can be obtained for the QGD algorithm in three dimensions and for its implementation on unstructured meshes.

CONCLUSIONS

The numerical results obtained for the Riemann problem suggest that, in smooth-solution regions, the QGD algorithm has an actual order of accuracy ranging from 0.5 to 2 depending on the local properties of the solution. Artificial dissipation inherent in the QGD algorithm is automatically adapted to the solution and is close to zero in regions where the desired solution has no singularities.

The computations show that the QGD algorithm represents a conditionally stable difference scheme obeying the Courant condition. The basic tuning parameters of the algorithm is the numerical coefficient α involved in the regularization parameter τ and the coefficient β determining the time step. Moreover, the function $\beta(\alpha)$ has an extremum corresponding to the optimal time step.

The results obtained for the Riemann problem with a superstrong discontinuity reveal that the numerical solution monotonically converges to a self-similar one as the spatial grid is refined. It is shown that, on fine grids, the QGD algorithm is superior in accuracy to high-order accurate methods with flux-corrected transport or flux splitting.

The simulation of weak disturbance propagation shows that, on sufficiently fine grids, hundreds of harmonic oscillation periods can be calculated using the QGD algorithm. Thus, along with high-order accurate schemes, this algorithm seems promising as applied to acoustic problems.

The estimated computational complexity of the method shows that the time required for executing arithmetic operations is balanced with the time for data transfer from memory per grid point. As a result, the method exhibits high performance when implemented on modern computing systems.

The estimates obtained for the accuracy, stability, computational complexity, and applicability range of the QGD algorithm as applied to one-dimensional flow simulation provide guiding lines for practical computations of multidimensional problems on grids of various structures.

REFERENCES

1. T. G. Elizarova and E. V. Shil'nikov, "Capabilities of a Quasi-Gasdynamic Algorithm as Applied to Inviscid Gas Flow Simulation," Zh. Vychisl. Mat. Mat. Fiz. **49**, 549–566 (2009) [Comput. Math. Math. Phys. **49**, 532–548 (2009)].
2. Yu. V. Sheretov, *Mathematical Modeling of Fluid Flows Based on Quasi-Hydrodynamic and Quasi-Gasdynamic Equations* (Tversk. Gos. Univ., Tver, 2000) [in Russian].
3. T. G. Elizarova, *Quasi-Gasdynamic Equations and Methods for Computing Viscous Flows* (Nauchnyi Mir, Moscow, 2007) [in Russian] Engl. transl.: *Quasi-Gas Dynamic Equations* (Springer, 2009.).
4. B. N. Chetverushkin, *Kinetic Schemes and Quasi-Gasdynamic System of Equations* (Maks Press, Moscow, 2004) [in Russian].
5. T. G. Elizarova and A. A. Khokhlova, "Quasi-Gasdynamic Equations for Gas Flows with External Heat Sources," Vestn. Mosk. Gos. Univ. Ser. 3: Fiz. Astron., No. 3, 10–13 (2007).
6. Yu. V. Sheretov, "On Difference Approximations of Quasi-Gasdynamic Equations for Axisymmetric Flows," in *Application of Functional Analysis in Approximation Theory* (Tversk. Gos. Univ., Tver, 2001), pp. 191–207 [in Russian].
7. Yu. V. Sheretov, "Stability Analysis of Modified Kinetically Consistent Difference Scheme in the Acoustic Approximation," in *Application of Functional Analysis in Approximation Theory* (Tversk. Gos. Univ., Tver, 2004), pp. 147–160 [in Russian].
8. B. L. Rozhdestvenskii and N. N. Yanenko, *Systems of Quasilinear Equations* (Nauka, Moscow, 1978) [in Russian].
9. N. N. Kalitkin, *Numerical Methods* (Nauka, Moscow, 1978) [in Russian].
10. A. A. Samarskii, *Theory of Difference Schemes* (Nauka, Moscow, 1989) [in Russian].
11. H. Tang and T. Liu, "A Note on the Conservative Schemes for Euler Equations," J. Comput. Phys., No. 218, 451–459 (2006).
12. A. N. Antonov, T. G. Elizarova, A. N. Pavlov, and B. N. Chetverushkin, "Mathematical Modeling of Oscillatory Modes in the Flow past a Spiked Body," Mat. Model. **1** (1), 14–23 (1989).

13. A. N. Antonov, T. G. Elizarova, B. N. Chetverushkin, and Yu. V. Sheretov, "Numerical Modeling of Pulsation Modes in the Supersonic Flow around a Hollow Cylinder," Zh. Vychisl. Mat. Mat. Fiz. **30**, 548–556 (1990).
14. M. A. Antonov, I. A. Graur, L. V. Kosarev, and B. N. Chetverushkin, "Numerical Modeling of Pressure Fluctuations in Three-Dimensional Cavities," Mat. Model. **8** (5), 76–90 (1996).
15. T. G. Elizarova, A. A. Khokhlov, and Yu. V. Sheretov, "Quasi-Gasdynamic Numerical Algorithm for Gas Flow Simulations," Int. J. Numer. Methods Fluids **56**, 1209–1215 (2008).
16. T. G. Elizarova and P. N. Nikol'skii, "Numerical Modeling of the Laminar–Turbulent Transition in the Flow over a Backward-Facing Step," Vestn. Mosk. Gos. Univ. Ser. 3: Fiz. Astron., No. 4, 14–17 (2007).
17. B. N. Chetverushkin and E. V. Shil'nikov, "Software Package for 3D Viscous Gas Flow Simulation on Multiprocessor Computer Systems," Zh. Vychisl. Mat. Mat. Fiz. **48**, 118–129 (2008) [Comput. Math. Math. Phys. **48**, 295–305 (2008)].
18. C. K. W. Tam and J. C. Webb, "Dispersion-Relation-Preserving Finite Difference Schemes for Computational Acoustics," J. Comput. Phys. **107**, 262–281 (1993).
19. I. V. Abalakin and T. K. Kozubskaya, "Multiparameter Family of High-Order Accurate Schemes for the Linear Advection Equation," Mat. Model. **19** (7), 56–66 (2007).
20. Yu. V. Sheretov, "Analysis of the Sound Propagation Problem for Linearized QGD Systems," in *Application of Functional Analysis in Approximation Theory* (Tversk. Gos. Univ., Tver, 2001), pp. 178–191 [in Russian].
21. L. D. Landau and E. M. Lifshits, *Fluid Dynamics* (Nauka, Moscow, 1986) [in Russian].
22. S. Williams, J. Shalf, L. Oliker, et al., "Scientific Computing Kernels on the Cell Processor," Int. J. Parallel Program. **35**, 263–298 (2007).