Journal of Engineering Physics and Thermophysics, Vol. 75, No. 3, 2002

CALCULATION OF THE FLOW OF A WEAKLY COMPRESSIBLE VISCOUS GAS USING A MULTIPROCESSOR COMPUTER SYSTEM

A. M. Nemukhina[a] **and B. N. Chetverushkin**[b]

UDC 519.63:536.24

A method to calculate the flow of a weakly compressible gas on a multiprocessor computer system using the $(\alpha - \beta)$ iteration algorithm has been proposed. The problem of gas flow in a cavity with a moving cover has been considered as an example. The flow is described using Navier–Stokes equations in natural variables; MAC-type spaced grids are employed. Calculations with the use of a monoprocessor version of the $(\alpha - \beta)$ algorithm and using its parallel implementation for the case of four processors have been carried out; the issue of the efficiency of multisequencing has been studied. Certain aspects of organization of remote access to the multiprocessor system have been discussed.

Weakly compressible gas flows provide the basis for technological processes in many industrial installations; description of the processes of heat and mass transfer is based on studying these flows. A mathematical model of viscous gas flow for small Mach numbers can be constructed on the basis of the model of an incompressible viscous fluid which is based on Navier–Stokes equations. Numerous methods, both explicit and implicit, have been developed at present to solve these equations. The use of implicit difference schemes is more preferable since they are much more stable as a rule. However, the employment of them involves a number of difficulties associated with solution of the corresponding systems of grid equations. In such a situation, in selecting the already existing method or developing a new one, particular emphasis must be placed upon the following aspect.

Very often, especially in problems with the processes of heat and mass transfer, detailed modeling requires a narrow grid with a large number of subdivision points, which significantly increases the volume of computations. This circumstance explains why the time of counting even on quite efficient computers is very long and leads to the fact that narrow grids turn out to be difficult to use. In our opinion, the way out is the employment of parallel computer systems with distributed memory, which enables one to substantially decrease the time required to solve the problem. In light of what has been said above, one must give preference to the algorithms of solution of Navier–Stokes equations which make it possible to implement them on parallel computers.

This work seeks to discuss certain issues concerning software of parallel computers and their application to solution of problems of the dynamics of a weakly compressible gas. To solve Navier–Stokes equations in natural variables on a MAC-type spaced grid [1] we employ the scheme of splitting [2] which inherits the properties of the initial differential problem. The equations obtained are solved using the $(\alpha - \beta)$ iteration algorithm, which allows a rather simple parallel implementation. The calculations whose results are presented in the work have been carried out at the Institute of Mathematical Modeling of the Russian Academy of Sciences on a Parsytec multiprocessor system using connection via the Internet, and part of the work is devoted to a discussion of the issues of organization of remote access.

Description of the $(\alpha-\beta)$ Iteration Algorithm. A distinctive feature of the algorithm employed on a multiprocessor computer is the possibility of its simple parallel implementation. We selected the $(\alpha - \beta)$ iteration algorithm [3, 4] that processes this property, which enables one to successfully apply it to solution of different problems [5]. Although its convergence has not been proved theoretically, the practice shows that it converges quite rapidly [6]. Let us consider a two-dimensional rectangular region on which the grid $\boldsymbol{\Omega} = \{\mathbf{x}: \mathbf{x} = (x_i, x_j)\}$, $0 \le i \le N_i$, $0 \le j \le N_j$ is prescribed. At the internal nodes of this grid, we consider the difference elliptical equation

[a]Academy of the Federal Agency of Government Communication and Information, Orel, Russia; email: cto7@orel.ru; [b]Institute of Mathematical Modeling, Russian Academy of Sciences, Moscow, Russia. Translated from Inzhenerno-Fizicheskii Zhurnal, Vol. 75, No. 3, pp. 22–27, May–June, 2002. Original article submitted January 8, 2001; revision submitted October 8, 2001.

1062-0125/02/7503-0547$27.00

$$A_{i,j}U_{i-1,j} + B_{i,j}U_{i+1,j} - C_{i,j}U_{i,j} + \overline{A_{i,j}}U_{i,j-1} + \overline{B_{i,j}}U_{i,j+1} = -F_{i,j}\,, \quad 0 < i < N_i\,, \quad 0 < j < N_i\,. \tag{1}$$

Boundary conditions are written in the following form:

$$U_{0,j} = \varphi_j^1 U_{1,j} + \xi_j^1\,, \quad U_{N_i,j} = \varphi_j^3 U_{N_i-1,j} + \xi_j^3\,, \quad U_{i,0} = \varphi_i^2 U_{i,1} + \xi_i^2\,, \quad U_{i,N_j} = \varphi_i^4 U_{i,N_j-1} + \xi_i^4\,. \tag{2}$$

It is assumed that the solution satisfies the conditions

$$U_{i,j} = \alpha_{i+1,j}U_{i+1,j} + \beta_{i+1,j}\,, \quad U_{i,j} = \gamma_{i-1,j}U_{i-1,j} + \delta_{i-1,j}\,, \quad U_{i,j} = \overline{\alpha_{i,j+1}}U_{i,j+1} + \overline{\beta_{i,j+1}}\,,$$

$$U_{i,j} = \overline{\gamma_{i,j-1}}U_{i,j-1} + \overline{\delta_{i,j-1}}\,. \tag{3}$$

By employing these assumptions and Eq. (1) we obtain a system of eight equations relative to the coefficients α, γ, $\overline{\alpha}$, $\overline{\gamma}$, β, δ, $\overline{\beta}$, and $\overline{\delta}$; the system of equations for α, γ, $\overline{\alpha}$, and $\overline{\gamma}$ is noninteracting (independent); therefore, the iteration process is constructed in the following manner: first we compute α, γ, $\overline{\alpha}$, and $\overline{\gamma}$ (this cycle of calculations will be called the α process) and then we analogously construct the β process in which the linear equations for β, δ, $\overline{\beta}$, and $\overline{\delta}$ are involved. Thereafter from conditions (2) and (3) we determine the grid function

$$U_{i,N_j} = \frac{\varphi_i^4\,\overline{\beta_{i,N_j}} + \xi_i^4}{1 - \varphi_i^4\,\overline{\alpha_{i,N_j}}}\,, \quad i = N_i - 1, \ldots, 1\,, \quad U_{i,j} = \overline{\alpha_{i+1,j}}\,U_{i+1,j} + \overline{\beta_{i+1,j}}\,, \quad i = N_i - 1, \ldots, 1\,, \quad j = N_j - 1, \ldots, 0\,.$$

The values of the functions $U_{0,j}$ and $U_{N_i,j}$ are easy to determine using conditions (2).

Let us consider the α process. First one constructs upward iterations: on one layer with respect the variable j, the coefficients α are computed from left to right (from the left-hand boundary to the right-hand boundary) and then the coefficients γ are computed from the right-hand boundary to the left-hand one. Thereafter the coefficients $\overline{\alpha}$ on the next layer with respect to j are found using them. The procedure is repeated for all $j = 1, \ldots, N_j - 1$:

$$\alpha_{i+1,j}^{(s+1/2)} = \frac{B_{i,j}}{C_{i,j} - \alpha_{i,j}^{(s+1/2)}A_{i,j} - \overline{\alpha_{i,j}}^{(s-1/2)}\overline{A_{i,j}} - \overline{\gamma_{i,j}}^{(s)}\overline{B_{i,j}}}\,, \quad i = 1, \ldots, N_i - 1\,;$$

$$\gamma_{i-1,j}^{(s+1/2)} = \frac{A_{i,j}}{C_{i,j} - \gamma_{i,j}^{(s+1/2)}B_{i,j} - \overline{\alpha_{i,j}}^{(s-1/2)}\overline{A_{i,j}} - \overline{\gamma_{i,j}}^{(s)}\overline{B_{i,j}}}\,, \quad i = N_i - 1, \ldots, 1\,;$$

$$\overline{\alpha_{i,j+1}}^{(s+1/2)} = \frac{\overline{B_{i,j}}}{C_{i,j} - \overline{\alpha_{i,j}}^{(s+1/2)}\overline{A_{i,j}} - \alpha_{i,j}^{(s+1/2)}A_{i,j} - \gamma_{i,j}^{(s+1/2)}B_{i,j}}\,, \quad i = 1, \ldots, N_i - 1\,.$$

Thereafter one constructs downward iterations: for all $j = N_j - 1, \ldots, 1$, on one j layer, new coefficients α are computed from left to right, then new coefficients γ are computed from right to left, and finally the coefficients $\overline{\gamma}$ are computed on the previous j layer:

$$\alpha_{i+1,j}^{(s+1)} = \frac{B_{i,j}}{C_{i,j} - \alpha_{i,j}^{(s+1)}A_{i,j} - \overline{\alpha_{i,j}}^{(s+1/2)}\overline{A_{i,j}} - \overline{\gamma_{i,j}}^{(s)}\overline{B_{i,j}}}\,, \quad i = 1, \ldots, N_i - 1\,;$$

$$\gamma_{i-1,j}^{(s+1)} = \frac{A_{i,j}}{C_{i,j} - \gamma_{i,j}^{(s+1)}B_{i,j} - \overline{\alpha_{i,j}}^{(s+1/2)}\overline{A_{i,j}} - \overline{\gamma_{i,j}}^{(s)}\overline{B_{i,j}}}\,, \quad i = N_i - 1, \ldots, 1\,;$$

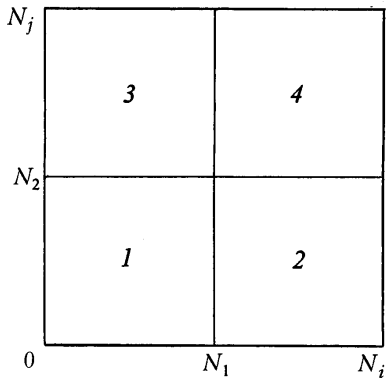

Fig. 1. Subdivision of the calculation region into subregions.

$$\overline{\gamma_{i,j-1}}^{(s+1)} = \frac{\overline{A_{i,j}}}{C_{i,j} - \overline{\gamma_{i,j}}^{(s+1)}\,\overline{B_{i,j}} - \alpha_{i,j}^{(s+1)} A_{i,j} - \gamma_{i,j}^{(s+1)} B_{i,j}}, \quad i = 1, \ldots, N_i - 1 .$$

The upward and downward iterations are repeated until the required accuracy is attained (for example, $\Delta_1 \le 10^{-10}$), where $\Delta_1 = \sqrt{\sum (\overline{\gamma_{i,j}}^{(s+1)} - \overline{\gamma_{i,j}}^{(s)})^2 h_1 h_2}$ (h_1 and h_2 are the grid steps with respect to the variables x_1 and x_2 respectively). When the β process is implemented, after each β iteration one computes the norm

$$\Delta_2 = \sqrt{\Sigma \left(A_{i,j}U_{i-1,j} + B_{i,j}U_{i+1,j} - C_{i,j}U_{i,j} + \overline{A_{i,j}}U_{i,j-1} + \overline{B_{i,j}}U_{i,j+1} + F_{i,j}\right)^2 h_1 h_2} .$$

On attainment of a certain accuracy (for example, $\Delta_2 \le 10^{-4}$), the process is considered to be completed.

Parallel Implementation of the ($\alpha - \beta$) Algorithm. Several methods of multisequencing of the ($\alpha - \beta$) algorithm exist [6] but they all are closely related to the principle of geometric parallelism. The calculation region is subdivided into a certain number of subregions. In this work, consideration is given to the subdivision of the region into four subregions, as is shown in Fig. 1. Counting in such a region is organized as follows. Let us consider the α process.

Upward Iterations. S t a g e 1. In regions 1 and 2, counting occurs from the lower boundary with respect to j, the coefficients α are counted from the left-hand boundary with respect to i to the right-hand boundary, while in region 2 the coefficients γ are counted from the right-hand boundary to the left-hand one. Simultaneously, counting from the upper boundary with respect to j occurs in regions 3 and 4; in region 3, the coefficients α are counted from the left-hand boundary with respect to i to the right-hand boundary, while in region 4 the coefficients γ are counted from the right-hand boundary with respect to i to the left-hand one. At the boundary N_1, we have the exchange: the coefficients γ are transferred to regions 1 and 3 and the coefficients α are transferred to regions 2 and 4.

S t a g e 2. Counting occurs from the lower boundary with respect to j in regions 1 and 2 and from the upper boundary with respect to j in regions 3 and 4; in regions 1 and 3, the values of γ are computed from the right-hand boundary to the left-hand one (on the same layer with respect to j, as at stage 1), while in regions 2 and 4 the values of α are computed from the left-hand boundary to the right-hand one.

S t a g e 3. One computes $\overline{\alpha}$ on the next layer with respect to j in regions 1 and 2 and $\overline{\gamma}$ on the previous layer with respect for j in regions 3 and 4.

Next, stages 1–3 are repeated until the boundary N_2 is attained at stage 3. Then at this boundary we have the exchange: computed $\overline{\gamma}$ are transferred to regions 1 and 2 and computed $\overline{\alpha}$ are transferred to regions 3 and 4.

Downward iterations are carried out analogously.

Example of Calculation. *Formulation of the Differential Problem.* Let us consider a calculation of the unsteady flow of a weakly compressible gas under the action of a spatially distributed force in a cavity of rectangular cross section with a moving upper cover. The gas is weakly compressible if the relative density change is small, which occurs for a Mach number much less than unity or, what is the same, if the flow velocity is small as compared to the velocity of sound. As is well known, in this case gas flows can approximately be considered as incompressible;

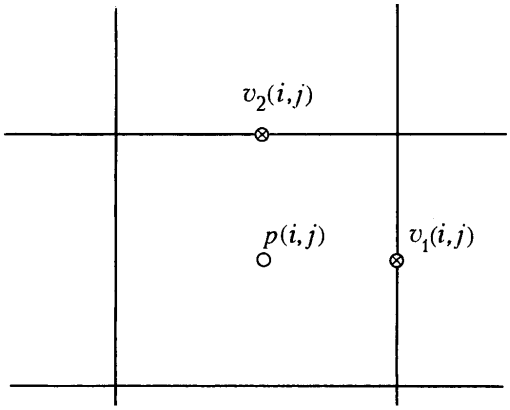

Fig. 2. Spaced grids of the MAC type.

therefore, the model of incompressible liquid flow presented in [2, 7] in detail has been used as the basis for description of the motion of a gas. The additive scheme of splitting with respect to the time coordinate is employed in this method. The motion of a weakly compressible gas in the region $\mathbf{\Omega} = \{\mathbf{x}: \mathbf{x} = (x_1, x_2)\}$, $0 < x_\alpha < l_\alpha$, $\alpha = 1, 2$, is described by the Navier–Stokes equation (in dimensionless form) and the equation of incompressibility of a medium

$$\frac{\partial \mathbf{v}}{\partial t} + C(\mathbf{v})\,\mathbf{v} + \operatorname{grad} p - \nu \Delta \mathbf{v} = \mathbf{f}(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_1, \quad 0 < t < \Theta, \tag{4}$$

$$\operatorname{div} \mathbf{v} = 0, \quad \mathbf{x} \in \Omega_1, \quad 0 < t < \Theta, \tag{5}$$

where $\nu = 1/\mathrm{Re}$ and $\Delta \equiv \operatorname{div} \operatorname{grad}$;

$$C(\mathbf{v}) = \sum_{\alpha=1}^{2} C_\alpha(\nu_\alpha); \quad C_\alpha(\nu_\alpha)\,\mathbf{u} = \frac{1}{2}\left[\frac{\partial}{\partial x_\alpha}(\nu_\alpha \mathbf{u}) + \nu_\alpha \frac{\partial \mathbf{u}}{\partial x_\alpha}\right]; \quad \alpha = 1, 2.$$

Equations (4) and (5) are supplemented with the initial conditions and the boundary conditions (for example, sticking and nonflow conditions):

$$\mathbf{v}(\mathbf{x}, 0) = \mathbf{v}_0(\mathbf{x}), \quad \mathbf{x} \in \Omega_1; \quad \mathbf{v}(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \partial\Omega_1, \quad 0 < t \le \Theta.$$

To unambiguously determine the pressure one must employ an additional condition, for example, $\int_{\Omega_1} p(\mathbf{x}, t)\, d\mathbf{x} = 0$, $0 < t \le \Theta$. We solved a specific problem in which the following conditions were satisfied: the region was a unit square ($l_\alpha = 1$, $\alpha = 1, 2$), the force acting on the gas was absent, the initial conditions were homogeneous, and the boundary conditions were inhomogeneous:

$$\mathbf{v}(\mathbf{x}, t) = \begin{cases} (0, 0), & x_2 \ne 1, \\ (1, 0), & x_2 = 1, \end{cases} \quad \mathbf{x} \in \partial\Omega_1, \quad 0 < t \le \Theta.$$

Numerical Implementation on Spaced Grids. In the calculations, we employed the MAC-type space grids for the pressure and the velocity components [1] (i.e., grids analogous to those employed in the known MAC method) which are given in Fig. 2; the pressure refers to the center of a cell, while the velocity components refer to its sides. To solve the problem formulated we used a purely implicit factorized scheme of splitting [8]. Omitting intermediate calculations (they are presented in [2, 7] in detail), we give just the description of the algorithm which consists of three stages.

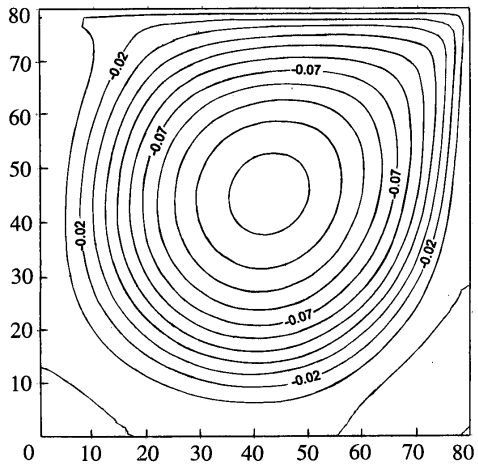

Fig. 3. Lines of the level of a streamfunction (the number of grid points is plotted along the axes).

S t a g e 1. First from the equation

$$\frac{\mathbf{w}^{n+1/2}-\mathbf{w}^{n}}{\tau}+C\,(\mathbf{w}^{n})^{n+1/2}+D\,\mathbf{w}^{n+1/2}+P\,\mathbf{w}^{n}=\mathbf{f}^{\,n}, \tag{6}$$

where the difference operators P, D, and $C(\mathbf{w})$ are determined as follows:

$$C\,(\mathbf{w})=\sum_{\alpha=1}^{2}C_{\alpha}\,(w_{\alpha}),\quad C_{\alpha}\,(w_{\alpha})\,\mathbf{u}=\frac{1}{2}\Big[(w_{\alpha}\,\mathbf{u})^{\circ}_{x_{\alpha}}+w_{\alpha}\,\mathbf{u}^{\circ}_{x_{\alpha}}\Big],\quad \alpha=1,2\,,\ P\,\mathbf{w}=\mathrm{grad}_{h}\,p\,,\quad D\,\mathbf{w}=-\nu\,\Delta\mathbf{w}\,,$$

we calculate the intermediate velocity $\mathbf{w}^{n+1/2}$.

S t a g e 2. Thereafter we determine the pressure on the basis of solution of the problem

$$\mathrm{div}_{h}\,\mathrm{grad}_{h}\,\delta p^{n+1}=\frac{1}{\tau}\,\mathrm{div}_{h}\,\mathbf{w}^{n+1/2}, \tag{7}$$

where the pressure correction is denoted by $\delta p^{n+1}=p^{n+1}-p^{n}$.

S t a g e 3. We compute the velocity and the pressure on a new time layer from the explicit formulas

$$\mathbf{w}^{n+1}=\mathbf{w}^{n+1/2}-\tau\,\mathrm{grad}_{h}\,\delta p^{n+1},\quad p^{n+1}=p^{n}+\delta p^{n+1}\,. \tag{8}$$

The $(\alpha-\beta)$ algorithm is employed to implement stages 1 and 2. To unambiguously solve Eq. (7) one must employ an additional condition, for example, fix the value of δp^{n+1} at one internal point of the grid. Any initial approximation of p^{0} can be taken. We should note that for the $(\alpha-\beta)$ algorithm to be stable it is necessary that the diagonal predominance [9] hold, since the algorithm is constructed on the basis of the method of runs in two spatial directions. This always holds for stage 1, but for stage 2 this holds only on condition $\tau\le\frac{1}{\nu}\{\max(h_1,h_2)\}^{2}$.

Calculation Results. First we considered the solution of problem (6)–(8) using a monoprocessor version of the $(\alpha-\beta)$ algorithm. To compare steady-state flows we employed the data of [7]. To obtain more objective results we used various grids (20×20, 40×40, 80×80, 16×160, and 320×320) and different Reynolds numbers (Re = 100, 400, and 1000); the time step was constant ($\tau=0.01$). The calculations were stopped if $\max\left(\max\limits_{i,j}\dfrac{w_{1i,j}^{n+1}-w_{1i,j}^{n}}{\tau},\max\limits_{i,j}\dfrac{w_{2i,j}^{n+1}-w_{2i,j}^{n}}{\tau}\right)$ was less than or equal to 10^{-5}. It was noted that when the steady-state

TABLE 1. Acceleration and Efficiency of the Parallel Algorithm for Different Re

Grid	20×20	40×40	80×80	160×160	320×320
			Re = 400		
t_1	3.0	54.8	935.9	16426.2	154360.6
t_4	14.3	109.4	939.0	8089.8	56118.2
S_4	0.21	0.50	1.00	2.03	2.75
E_4, %	5.24	12.52	24.92	50.76	68.77
			Re = 1000		
t_1	1.8	43.8	773.7	11123.8	91671.4
t_4	12.1	96.2	781.9	6685.5	39218.7
S_4	0.15	0.46	0.99	1.66	2.34
E_4, %	3.72	11.38	24.74	41.59	58.44

regime was reached an additional vortex appeared at the lower corner of the cavity. The intensity of mass exchange increased with increase in the Reynolds number, although a slight extension of stagnation zones was observed. Figure 3 shows the streamlines for Re = 1000 and an 80×80 grid. At the second stage of calculations, to solve problem (6)–(8) we employed parallel implementation of the $(\alpha - \beta)$ algorithm and investigated the efficiency of multisequencing. Unfortunately, because of the distinctive features of the computer system employed, it seemed impossible to measure the total time of solution of the problem; therefore, in the calculations we employed the time of realization of one iteration measured in special conventional units. Below in Table 1 we give the acceleration $\left(S_4 = \frac{t_1}{t_4}\right)$ and efficiency $\left(E_4 = \frac{S_4}{4} \cdot 100\%\right)$ of the parallel algorithm for Re = 400 and Re = 1000 respectively. It is seen that for coarse grids the use of multiprocessor devices gives no gain in the acceleration because of the comparatively long time of data exchange between the processors, whereas for narrow grids the efficiency of multisequencing is rather high. In general, we can draw a conclusion on the decrease in the efficiency with decrease in the number of points per processor (due to both the coarsening of the grid and the increase in the number of processors).

Carrying Out Computations Using Remote Access. Multiprocessor systems have remained unique until the present time; therefore, their capabilities could be evaluated by only a few scientists. However in recent times, the number of researchers who are able to utilize the resources of multiprocessor systems has sharply increased owing to the growing popularity of global computer networks; in so doing, the topicality of the developments in the field of multisequencing of computational algorithms has increased. The calculations whose results are presented in this work were carried out at the Institute of Mathematical Modeling of the Russian Academy of Sciences on a Parsytec CC multiprocessor computer system under the control of an AIX operating system and a PARIX parallel medium. The system is incorporated in the local network of the Institute which in turn has access to the global Internet. This provides good possibilities for their remote use, since the systems can be accessed from any node of the Internet.

The computations were carried out using remote access from Orel with the use of standard facilities offered by Unix and Windows95 operating systems. The facility telnet is the most important of them for our purposes. Unfortunately, the use of telnet has a number of drawbacks. One drawback is the impossibility of employing modern convenient editors for introduction of alterations into the initial text of a program (the integrated user environment is not incorporated into the PARIX compiler, and in the case of remote access it seemed impossible to employ any window system developed for Unix). Another drawback is as follows: because of the not very high capacity of the communication channel and its heavy loading in certain hours, the time of debugging of the program increases severalfold as compared to direct operation on a personal computer. In certain cases (for example, where it is necessary to revise a large portion of the initial text), the way out is the use of the facility ftp that enables one to transfer files from one remote system to another. The necessary file is transferred from a multiprocessor system to a personal computer, edited in any standard editor, and then transferred back to the multiprocessor system. When ftp is employed the time of editing of the file decreases but it takes additional time to transfer the file in two directions. Therefore, in the case

where a small portion of the program's text must be corrected it is inexpedient to employ this facility. The conclusion is as follows: the largest effect in the saving of time is attained in the case of employment of a combined version, where ftp is employed for a large volume and telnet is employed for a small volume.

The problems arising in the case of employment of multiprocessor systems (in particular, processing and visualization of large amounts of data) are aggravated if the employment of them is implemented using remote access. Since data files can be huge in size in detailed modeling and the capacity of communication channels is rather low, one must, where possible, decrease the amount of data transferred from a multiprocessor system to a remote user. Specially developed algorithms of compression of grid functions different from standard archiving utilities are suitable for this purpose. Another way-out is the employment of "client–server" technology in which all data are stored in the multiprocessor system itself, and the user is operating only with visual presentation of the results of his calculations.

CONCLUSIONS

1. The multiprocessor method to solve problems of the dynamics of a weakly compressible gas which employs splitting by space variables and is based on the principle of geometric parallelism has been proposed and tested.

2. The efficiency of multisequencing has been investigated (it turned out to be rather high).

3. The possibility of extending the range of application of multiprocessor devices using remote access and the problems involved have been considered.

This work was carried out with financial support from the Russian Foundation for Basic Research (project 00–01–81057 Bel 2000_a).

NOTATION

Ω, two-dimensional grid; $\mathbf{x} = (x_i, x_j)$, points of this grid; i, j, coordinates of the point; N_i and N_j, maximum value of the coordinates i and j respectively; $A_{i,j}$, $B_{i,j}$, $C_{i,j}$, $\overline{A_{i,j}}$, $\overline{B_{i,j}}$, and $\overline{F_{i,j}}$, coefficients of the elliptical equation; φ_j^1, ξ_j^1, φ_j^3, ξ_j^3, φ_i^2, ξ_i^2, φ_i^4, and ξ_i^4, constants involved in the boundary conditions for the elliptical equation; $U_{i,j}$, arbitrary grid function; $\alpha_{i,j}$, $\beta_{i,j}$, $\gamma_{i,j}$, $\delta_{i,j}$, $\overline{\alpha_{i,j}}$, $\overline{\beta_{i,j}}$, $\overline{\gamma_{i,j}}$, and $\overline{\delta_{i,j}}$, coefficients of the $(\alpha - \beta)$ algorithm; $\alpha_{i,j}^{(s)}$, $\alpha_{i,j}^{(s+1/2)}$, and $\alpha_{i,j}^{(s+1)}$, initial, intermediate, and final values of the coefficient $\alpha_{i,j}$ (the notation for the remaining coefficients of the $(\alpha - \beta)$ algorithm is analogous); Δ_1 and Δ_2, norms employed to check accuracy; h_1 and h_2, grid steps with respect to the variables x_i and x_j respectively; $\mathbf{v} = (v_1, v_2)$ and p, dimensionless time and pressure; $C(\mathbf{v})$, operator of convective transfer; Δ, Laplace operator; P and D, auxiliary operators; t, time; τ, time step; $\mathbf{f}(\mathbf{x}, t)$, vector of bulk forces; Ω_1, region consisting of the internal points of the cavity; $\partial\Omega_1$, boundary of Ω_1; Θ, maximum value of the time; ν, kinematic viscosity; Re, Reynolds number; $\mathbf{v}_0(\mathbf{x})$, initial value of the velocity; $\mathbf{w}^n$, p^n, and $\mathbf{f}^n$, values of the velocity, the pressure, and the bulk-force vector on the nth time layer; S_4 and N_4, acceleration and efficiency of the parallel algorithm; t_1 and t_4, time of computation on one processor and four processors.

REFERENCES

1. F. H. Harlow and J. E. Welch, *Phys. Fluids*, **8**, No. 12, 2182–2189 (1965).
2. P. N. Vabishchevich, A. N. Pavlov, and A. G. Churbanov, *Difference Methods of Solution of Navier–Stokes Equations in Natural Variables on Unspaced Grids*, Preprint No. 34 of the Institute of Mathematics and Mechanics of the Russian Academy of Sciences [in Russian], Moscow (1993).
3. B. N. Chetverushkin, *Mathematical Modeling of Problems of the Dynamics of a Radiating Gas* [in Russian], Moscow (1985).
4. B. N. Chetverushkin, *Zh. Vychisl. Mat. Mat. Fiz.*, **16**, No. 2, 519–524 (1976).
5. B. N. Chetverushkin, *Kinetically Matched Schemes in Gas Dynamics* [in Russian], Moscow (1999).
6. B. N. Chetverushkin and N. G. Churbanova, *Mat. Modelir.*, **3**, 123–129 (1991).
7. P. N. Vabishchevich, A. N. Pavlov, and A. G. Churbanov, *Mat. Modelir.*, **8**, No. 7, 81–108 (1996).
8. A. A. Samarskii, *Theory of Difference Schemes* [in Russian], Moscow (1988).
9. A. A. Samarskii and E. S. Nikolaev, *Methods of Solution of Grid Equations* [in Russian], Moscow (1978).