Modeling Two-Dimensional Unsteady-State Problems on Grids with Dynamic Adaptation

N. A. DAR'IN AND V. I. MAZHUKIN

Keldysh Institute of Applied Mathematics, Moscow

A finite difference method is described for solving two-dimensional Stefan problems and boundary layer problems. A comparison between numerical and analytical solutions has shown that very accurate calculations are possible on grids with small numbers of nodes.

1. INTRODUCTION

Along with construction of high-quality difference schemes, which are a traditional mainstay [1], one of the most important aspects of modeling of problems of mathematical physics is the selection and generation of computational grids optimal for the given solution. Considerable attention has recently been paid to developing computational grids, and there have now developed several trends in their construction, generation, and adaptation (see [2, 3]). Methods for generating grids that automatically adapt to the solution satisfy the requirement for optimal node distribution best. Because the distribution of the nodes in adaptive grids is dynamic, the grid node coordinates are not known beforehand, and must be determined, together with the grid functions, while solving the problem. One technique for grid construction involves imposing additional conditions on the initial problem. There is no unified approach to this problem, since adaptive grid generation methods are still being developed, and it is still impossible to say which approach is preferable.

A key problem in all the approaches to adaptive grid generation is how to choose that characteristic of the numerical solution which will be used as the parameter controlling the node shifts, and to determine the relationship between the procedures for solving the differential problem and constructing the grid. The technique most widely used in choosing the control parameter in multidimensional steady- [3] and unsteady-state [4] problems is the variational principle and its various modifications [5–9]. This principle of adaptive grid generation is based on minimizing the solution characteristics over the entire domain, which involves determining the extrema of certain functionals. In addition to the variational principle, there are some empirical approaches [10–12] that use information on the numerical solution. Thus, the norm of the truncation error was chosen as the control parameter in [10], while in [11, 12] it was the rate of change of the solution.

1061-7590/93/03301-0016$13.00

The relationship between the difference scheme and the adaptive grid appears to be as important. An ideal grid can only be generated if the solution is known beforehand. When generating a grid with a controlled node arrangement, only limited *a priori* information is available about the solution behavior. Therefore, the computation accuracy essentially depends upon the relationship between the difference scheme used for approximating the differential problem and the method for generating the computational grid. There are two opposite viewpoints on the problem. For instance, the processes of determining the grid solution and the node coordinates are separated in [9, 13], but are implemented independently to avoid coupled oscillations of the solution and the grid (which has been observed [11]). As confirmation of the positive effect of independent implementation, solutions of model gas dynamics and other problems are cited in [9, 13], to demonstrate the effectiveness of the method. On the other hand, experience in mathematically modeling hydrodynamic and heat transfer problems [1, 14–16] shows that the accuracy of the difference solution of partial differential equations depends on how well the distribution of computational grid nodes conforms with the specifics of the solution. In most real problems, it is not known beforehand when and where phenomena such as shock waves and large gradients will appear and how they will propagate. For this reason, it is not always possible to fit the grid nodes to these phenomena. We believe that there should be a close connection between the difference schemes and the grid construction method, and that this connection should be introduced on the level of the differential model.

We have suggested [17–19] a method for unsteady-state spatially one-dimensional boundary value problems that generates adaptive grids dynamically coupled to the solution. The idea is to obtain automatic coordinate transformation from a Cartesian frame to a more universal curvilinear coordinate system in which computational grids with fixed nodes can be used to solve the problem numerically. In physical space, these grids are equivalent to grids with the same number but a controlled distribution of nodes, so the nodes can be concentrated in the regions of singularities, large gradients, and other phenomena arising in the solution.

It should be noted that the mapping from a physical space to a computational space with curvilinear coordinates actually yields a different differential problem. The original equations must be modified and their number changed. The mathematical model has additional partial differential equations to describe the computational grid's dynamics. The number of additional equations equals the spatial dimension of the problem. The actual transformation is determined by the solution's features and the researcher is free to choose the node shift control with a minimum of algorithmic and code changes for different problem types.

The adaptation method was used in unsteady-state one-dimensional boundary value problems, such as classical Stefan problems with one [20, 21] or two [22] phase fronts, boundary layer problems [17], and gas dynamics problems with explicit tracking of shock wavefronts [23–25]. A comparison of the solutions with analytical solutions has revealed that dynamic node distribution substantially reduces the total number of nodes. This implies that we can reduce the amount of computations and raise the algorithm's efficiency.

This study is an extension to unsteady-state two-dimensional boundary value problems of the method of adaptive grids dynamically coupled to the solution [17–19]. The method's potential can be exemplified by solving model Stefan and boundary layer problems.

2. CONSTRUCTION PRINCIPLE

The principle of constructing finite-difference methods on adaptive grids dynamically coupled to the solution is defined as a method for solving unsteady-state problems in which the grid node coordinates are determined along with the grid functions. The determination of grid functions and node coordinates constitutes a unified self-consistent problem in which some of the equations describe the phenomenon in question in terms of the grid's movement of nodes in the grid, while others describe the computational grid's dynamics as they depend on the solution. In extreme cases these equations entail formulations of the problem in either Euler (no adaptation) or Lagrangian (the grid node and hydrodynamic) velocities.

The generation method is based on an automatic coordinate mapping that uses the solution. It yields a coordinate system (hereinafter referred to as curvilinear), in which lines are mapped to the domain boundaries and are perpendicular. Depending on the problem, the coordinate lines may follow moving boundaries, may become denser in regions of singularities or large gradients, or may move according to some relation that minimizes the viscosity or the nonmonotonicity of the difference scheme.

The Cartesian coordinates x, y, t are used as the initial frame system when the problem is posed mathematically. Then by a general mapping $\xi = \xi(x, y, t)$, $\eta = \eta(x, y, t)$, $\tau = t$, the physical plane is mapped onto a rectangle in the curvilinear coordinate plane ξ, η (see Figure 1). The mathematical formulation of the problem in curvilinear coordinates is more complicated because the metric coefficients x_ξ, x_η, y_ξ, and y_η enter the equations. However, the transition to a more universal reference frame allows orthogonal grids that are either uniform or nonuniform in one or both directions to be constructed for more complex domains in physical space.

The inverse mapping uses two evolutional equations, which describe the trajectories along which the grid nodes move, and form the mechanism of the dynamic distribution of the grid nodes. For two-dimensional unsteady-state problems, this method involves a system of two partial differential equations whose right-hand sides can be represented as two functions Q and P dependent on the solution. Generally, the functions are arbitrary. The type of function is defined by the solution features and determines the mode of adaptation.

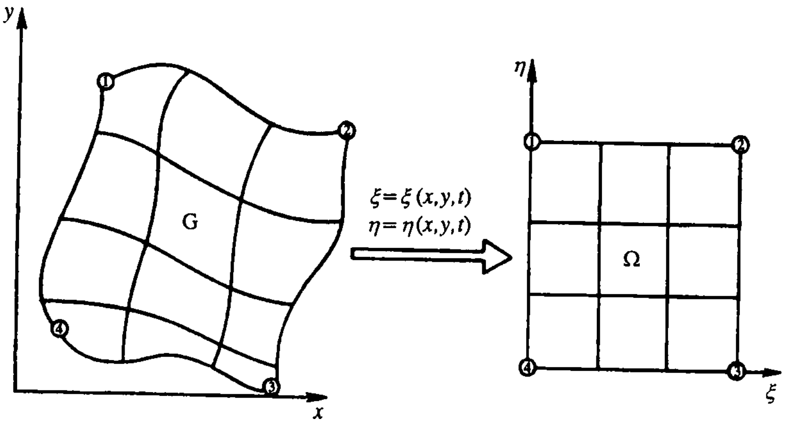

Figure 1.

3. PROBLEM FORMULATION

Let us consider an unsteady-state heat transfer equation in an arbitrary domain $G\{x, y\}$ with a source, *i.e.*,

$$\begin{gathered}\frac{\partial u}{\partial t} = -\frac{\partial W_1}{\partial x} - \frac{\partial W_2}{\partial y} + g,\\ W_1 = -\lambda\frac{\partial u}{\partial x}, \qquad W_2 = -\lambda\frac{\partial u}{\partial y}.\end{gathered} \tag{1}$$

Using the general mapping $\xi = \xi(x, y, t)$, $\eta = \eta(x, y, t)$, $\tau = t$, we obtain a space Ω with the curvilinear coordinates ξ, η, τ. Equation (1) in these variables will be written in a rigorously conservative form as

$$\begin{aligned}\frac{\partial(\psi u)}{\partial \tau} =& - \frac{\partial}{\partial \xi}\left[\left(W_1 - u\frac{\partial x}{\partial \tau}\right)\frac{\partial y}{\partial \eta} - \left(W_2 - u\frac{\partial y}{\partial \tau}\right)\frac{\partial x}{\partial \eta}\right]\\ &- \frac{\partial}{\partial \eta}\left[-\left(W_1 - u\frac{\partial x}{\partial \tau}\right)\frac{\partial y}{\partial \xi} + \left(W_2 - u\frac{\partial y}{\partial \tau}\right)\frac{\partial x}{\partial \xi}\right] + \psi g.\end{aligned} \tag{2}$$

We add the equations characterizing the current (instantaneous) coordinates x, y to the above, *i.e.*

$$\frac{\partial x}{\partial \tau} = -Q, \qquad \frac{\partial y}{\partial \tau} = -P. \tag{3}$$

With (3), the initial problem (1) takes the form

$$\begin{aligned}\frac{\partial(\psi u)}{\partial \tau} =& - \frac{\partial}{\partial \xi}\left[(W_1 + uQ)\frac{\partial y}{\partial \eta} - (W_2 + uP)\frac{\partial x}{\partial \eta}\right]\\ &- \frac{\partial}{\partial \eta}\left[-(W_1 + uQ)\frac{\partial y}{\partial \xi} + (W_2 + Pu)\frac{\partial x}{\partial \xi}\right] + \psi g,\\ \frac{\partial x}{\partial \tau} =& - Q, \qquad \frac{\partial y}{\partial \tau} = -P, \qquad \psi = \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi},\end{aligned} \tag{4}$$

where ψ and $\partial x/\partial \xi$, $\partial y/\partial \xi$, $\partial x/\partial \eta$, and $\partial y/\partial \eta$ are the Jacobian and the metric coefficients, respectively; and Q and P are functions dependent on the solution.

Given the functions Q and P, the system of equations (4) can be used to determine the temperature fields and the grid node locations using finite difference methods.

A specific form of the functions Q and P is chosen to construct the computational grid efficiently. The simplest case is a grid that is quasi-uniform in all directions in a domain with moving boundaries. The functions Q and P are therefore chosen in the form

$$Q = -D_0\left(\frac{\partial^2 x}{\partial \xi^2} + \frac{\partial^2 x}{\partial \eta^2}\right), \qquad P = -D_0\left(\frac{\partial^2 y}{\partial \xi^2} + \frac{\partial^2 y}{\partial \eta^2}\right). \tag{5}$$

If in order to solve (1) we need a grid whose nodes become denser where the solution undergoes rapid changes, but are quasi-uniformly distributed elsewhere in the domain, the functions Q and P can be represented [26] as

$$
\begin{aligned}
Q &= -D_0\left(\frac{\partial^2 x}{\partial \xi^2}+\frac{\partial^2 x}{\partial \eta^2}\right) - C_0\left\{\psi\frac{\partial}{\partial \xi}\left[\psi^2\left(\left(\frac{\partial u}{\partial \xi}\right)^2+\left(\frac{\partial u}{\partial \eta}\right)^2\right)^{1/2}\right]\right\},\\
P &= -D_0\left(\frac{\partial^2 y}{\partial \xi^2}+\frac{\partial^2 y}{\partial \eta^2}\right) - C_0\left\{\psi\frac{\partial}{\partial \eta}\left[\psi^2\left(\left(\frac{\partial u}{\partial \xi}\right)^2+\left(\frac{\partial u}{\partial \eta}\right)^2\right)^{1/2}\right]\right\},
\end{aligned}
\tag{6}
$$

where D_0, and C_0 are arbitrary constants of order 1.

The presence of first-order partial derivatives of the solution $\partial u/\partial \xi$, $\partial u/\partial \eta$ in (6) allows us to trace the rate at which the solution changes, and depending upon the rate of change, the grid density can be varied as required. The first terms in the parentheses act as forces that repulse the nodes, *i.e.*, they present the nodes from approaching each other too closely, and tend to make the grid more uniform where the solution changes only gradually. The presence of ψ in the curly brackets with a derivative sign also serves to restrict the mutual approach of the nodes. Note that other choices of Q and P are also possible.

4. DIFFERENCE APPROXIMATION

We introduce a computational grid in the $\Omega(\xi,\eta)$ plane with steps h_1 and h_2 that are uniform in all directions, *i.e.*,

$$
\omega_{h_1,h_2} = \left\{(\xi_i,\eta_j);\quad \begin{matrix} \xi_i = \xi_{i-1}+h_1, & i=1,2,\ldots,N, & h_1=\xi_N/N, \\ \eta_j=\eta_{j-1}+h_2, & j=1,2,\ldots,M, & h_2=\xi_M/M \end{matrix}\right\}.
$$

The step for the variable τ is introduced in the usual way, *i.e.*

$$
\omega_\tau = \{\tau_n;\ \tau_{n+1}=\tau_n+\Delta\tau_n,\quad n=0,1\ldots\}
$$

In curvilinear coordinates, the grid steps h_1 and h_2 are time-independent, and the initial orthogonal grid remains intact. In physical space the corresponding node coordinates $(x_i,\ y_i)$ vary with time, depending on the solution's behavior.

The difference approximation of (3)–(5) was done using the integral interpolation technique [17], *i.e.*,

$$
\begin{aligned}
(\psi u)^{n+1}_{i+1/2,j+1/2} =&(\psi u)^n_{i+1/2,j+1/2} - \frac{\Delta\tau}{h_1h_2}[(W_1+uQ)^n_{i+1,j+1/2}\\
&\times(y^n_{i+1,j+1}-y^n_{i+1,j}) - (W_1+uQ)^n_{i,j+1/2}(y^n_{i,j+1}-y^n_{i,j})\\
&-(W_1+uQ)^n_{i+1/2,j+1/2}(y^n_{i+1,j+1}-y^n_{i,j+1})\\
&+(W_1+uQ)^n_{i+1/2,j}(y^n_{i+1,j}-y^n_{ij}) - (W_2+uP)^n_{i+1,j+1/2}(x^n_{i+1,j+1}-x^n_{i+1,j})\\
&+(W_2+uP)^n_{i,j+1/2}(x^n_{i,j+1}-x^n_{ij}) + (W_2+uP)^n_{i+1/2,j+1}(x^n_{i+1,j+1}-x^n_{i,j+1})\\
&-(W_2+uP)^n_{i,j+1/2}(x^n_{i+1,j}-x^n_{ij})] + \Delta\tau(\psi g)^n_{i+1/2,j+1/2},\\
\psi^n_{i+1/2,j+1/2} =&\frac{1}{2h_1h_2}[(x^n_{i+1,j}-x^n_{i,j+1})(y^n_{i+1,j+1}-y^n_{ij})
\end{aligned}
$$

$$
\begin{aligned}
&- (x^n_{i+1,j+1} - x^n_{ij})(y^n_{i+1,j} - y^n_{i,j+1}),\\
W^n_{1,i+1/2,j} &= - \frac{\lambda}{h_1 h_2 \psi^n_{i+1/2,j}}\\
&\times [+(y^n_{i+1/2,j+1/2} - y^n_{i+1/2,j-1/2})(u^n_{i+1,j} - u^n_{ij})\\
&- (y^n_{i+1,j} - y^n_{ij})(u^n_{i-1/2,j+1/2} - u^n_{i+1/2,j-1/2})],\\
W^n_{2,i+1/2,j} &= - \frac{\lambda}{h_1 h_2 \psi_{i+1/2,j}}\\
&\times [-(x^n_{i+1/2,j} - x^n_{i+1/2,j-1/2})(u^n_{i+1,j} - u^n_{ij})\\
&+ (x^n_{i+1,j} - x^n_{ij})(u^n_{i+1/2,j+1/2} - u^n_{i+1/2,j-1/2})],\\
\frac{x^{n+1}_{ij} - x^n_{ij}}{\Delta\tau} &= - Q^n_{ij}, \qquad \frac{y^{n+1}_{ij} - y^n_{ij}}{\Delta\tau} = -P^n_{ij}.
\end{aligned}
\tag{7}
$$

The functions x^n_{ij}, y^n_{ij}, Q^n_{ij}, and P^n_{ij} correspond to the grid nodes. The functions $\psi^n_{i+1/2,j+1/2}$, and $u^n_{i+1/2,j+1/2}$ correspond to the cell centers. The fluxes $W_{1,i+1/2,j}$, $W_{1,i,j+1/2}$,$W_{2,i+1/2,j}$, and $W_{2,i,j+1/2}$ were determined at the midpoints of the cell edges. The necessary interpolation was made by linear interpolation formulas.

5. MODELING STEFAN PROBLEMS

An essential requirement on the algorithms for multidimensional problems is that the grids have a small number of nodes in each direction, but retain the accuracy of the calculation. That is why all the previous studies involved coarse grids.

In some problems, such as those involving the synthesis of amorphous materials, the effect of concentrated energy fluxes on metals and semiconductors, and thermal diffusion (in all of which a phase transition, which is an essentially nonequilibrium phenomenon, can be controlling), and also in problems that require an exact correction for the hydrodynamic phenomena that accompany the phase transitions, the phase front must be carefully extracted, and the processes occurring on it taken into account. Existing numerical algorithms for explicit front extraction (or separation) [27, 28] require, especially in multidimensional cases, complicated grid, constructions and are usually cumbersome and costly. The development of more economical and effective algorithms for solving multidimensional problems with explicit separation of the phase boundary is an important goal. One approach to the problem is the adaptive grid method.

A classical Stefan problem is to determine the temperature fields and the phase front velocities v_{sl}. The main difficulty in a mathematical investigation of such a problem is that the phase boundary $\Gamma_{sl}(t)$, and therefore the subdomains $G_s(x, y, t)$ and $G_l(x, y, t)$ in which the solution is sought, are not known beforehand and have to be determined during the solution. Besides, there are frequently situations when the domains containing emerging or initial phases change in size by large factors. In solving such problems numerically, the computational grids used at the beginning of the process become unsuitable after some time, and the grid must be reconstructed (*e.g.*, by an interpolation procedure) so the computation can be continued.

The main requirement on the computational grids used to solve such a problem is that there should be a mechanism for automatically reconstructing the grid with a fixed total number of cells, their size depending on the flux through the interface, *i.e.* on the solution's behavior. It is also desirable to eliminate moving boundaries.

The computational grid is generated and the problem solved in curvilinear coordinates. With this in mind, equation (1) with $g = 0$ is used in each of the subdomains G_s and G_l. We write the differential Stefan condition on the phase boundary $\Gamma_{sl}(t)$:

$$W_s^n - W_l^n = L_m \varrho v_{sl}^n, \qquad W_s^\tau = W_l^\tau \tag{8}$$

and write the isothermal condition

$$u_s = u_l = u_m, \tag{9}$$

where the superscripts n and τ refer to the normal and tangential components of the heat fluxes; the subscripts s and l refer to the solid and liquid phases; ϱ is the density, and L_m and u_m are the latent heat and temperature of the phase transition.

In curvilinear coordinates, the phase boundary $\Gamma_{sl}(x, y, t)$ is associated with a coordinate line $\Gamma_{sl}(\xi, \eta)$, whose location is fixed. The boundary condition in (9) is used to determine the material flowing from one domain to another. By doing so, we avoid the main difficulty inherent in the Stefan problem. In physical space the grids in both subdomains are reconstructed in accordance with the value and the sign of the mass flux. The number of cells in each subdomain remains constant while their sizes change. Besides, the mapping of a physical plane with a complex shape onto a rectangular domain yields orthogonal grids that are uniform in one or both directions, and have time-invariable steps h_1 and h_2.

The method was tested with a problem that has a self-similar solution in one variable [29]. A Stefan freezing problem with a planar interface $\Gamma_{sl}(t)$ (straight line) between the domains G_s and G_l was considered in the rectangular domain $G = \{0 \le x \le 1, 0 \le y \le 0.5\}$. The coordinate axes are chosen so that the phase boundary $\Gamma_{sl}(t)$, while moving parallel to itself, remains parallel to none of the axes. In the numerical calculations, the following values of the thermal parameters were chosen: $\lambda = c_p = \varrho = 1$, $L_m = 1$, and $u_m = 0$. By rotating the coordinate system through an angle φ, the problem reduces to a one-dimensional problem with space coordinate $z = x\cos\varphi + y\sin\varphi$. A self-similar solution exists for this problem, *i.e.*,

$$u(z,t) = \begin{cases} -1 + \Phi(z/(2\sqrt{t}))/\Phi(\beta), & 0 \le z \le z_\Gamma, \\ 0, & z_\Gamma < z \le 1, \end{cases}$$

where

$$\Phi(z) = \frac{2}{\sqrt{\pi}} \int\limits_0^z e^{-y^2} \mathrm{d}y, \qquad z_\Gamma = 2\beta\sqrt{t}, \qquad \beta = 0.62.$$

We considered a computational grid with a total of 15×6 nodes in domain G. We plotted the boundary $\Gamma_{sl}(x, y, t_0)$ at an angle of 30 deg. The initial state is chosen to correspond to the self-similar solution at $t_0 = 0.162$ (see Figure 2). Mapping to the curvilinear coordinates ξ, η, τ transformed the domain G into a rectangular domain Ω, in which the initial computational grid became uniform in both dimensions with the same total number of the nodes, and with the steps $h_1 = h_2 = 5 \cdot 10^{-2}$. The plane boundary $\Gamma_{sl}(\xi, \eta)$ coincided with the coordinate line $\xi = \xi_{N_0}$, $N_0 = 11$, and its position, as well as the steps h_1 and h_2, remained unchanged over time.

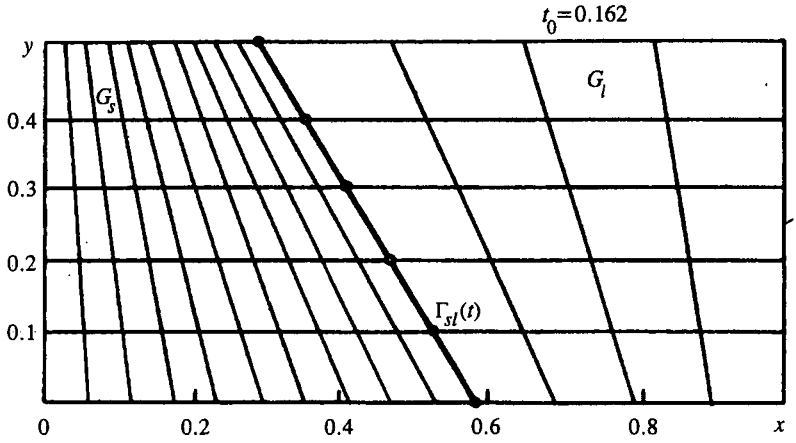

Figure 2.

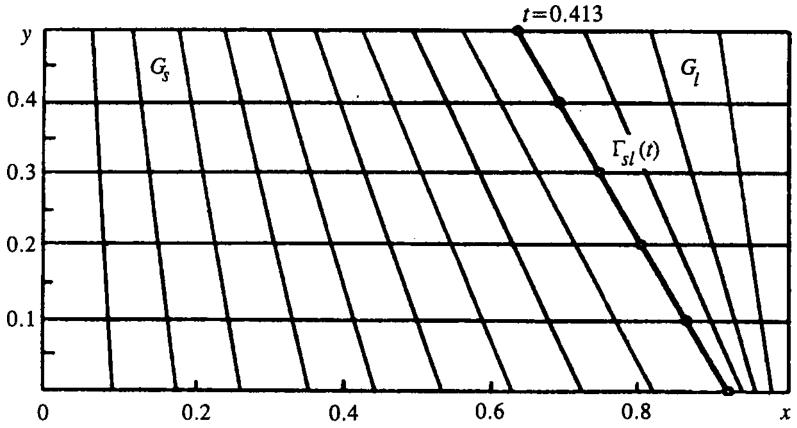

Figure 3.

The time integration step was restricted by the requirement that the explicit scheme be stable. Figure 3 displays the phase front's location and the computational grid in the physical space at $t = 0.413$. The self-similar and numerical solutions were compared along the axis x for three cross-sections with $j = 1, 3, 5$, where j is the row number, and at three different time moments. The relative error δu was under 1% which is quite acceptable for practical purposes. Table 1 lists the values of Δl which characterize the deviation of the phase boundary from planar, where $\Delta l = l_{\text{des}} - l_{\text{ex}}$, l_{ex} is the exact value of the coordinate z_Γ corresponding to the phase boundary, and l_{des} is the computed value of the coordinate. The results show that the location of the phase boundary is accurately determined by the algorithm.

It seemed interesting to test the possibilities of the method experimentally in the following situations:

(a) regions where the phase states changes by orders of magnitude over time;

(b) the initial computational grids in physical space differ significantly from orthogonal grids.

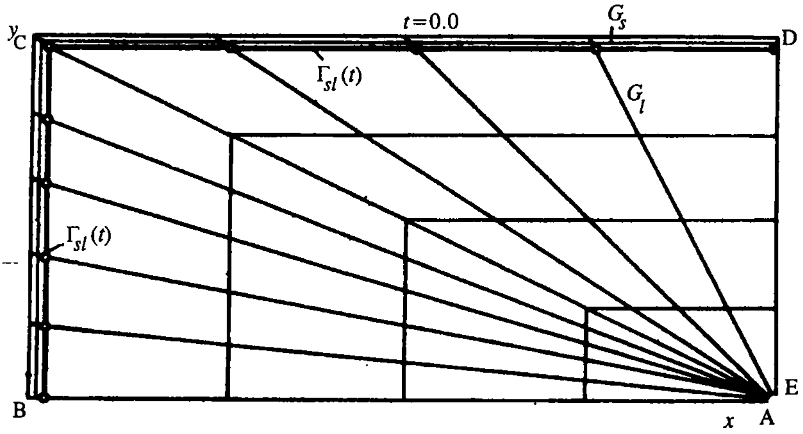

Figure 4.

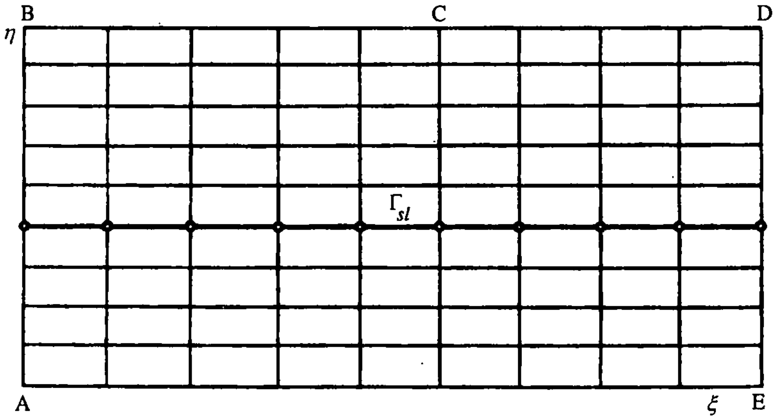

Figure 5.

Table 1.

Row number	Computed values l_{des}					
	$t = 0.2623$		$t = 0.3375$		$t = 0.4123$	
	$l_{\text{ex}} = 0.6357$		$l_{\text{ex}} = 0.7209$		$l_{\text{ex}} = 0.7970$	
	l_{des}	Δl	l_{des}	Δl	l_{des}	Δl
1	0.6370	$1.3 \cdot 10^{-3}$	0.7224	$1.6 \cdot 10^{-3}$	0.7986	$1.6 \cdot 10^{-3}$
2	0.6365	$8 \cdot 10^{-4}$	0.7220	$1.1 \cdot 10^{-3}$	0.7983	$1.3 \cdot 10^{-3}$
3	0.6353	$-4 \cdot 10^{-4}$	0.7213	$4 \cdot 10^{-4}$	0.7978	$8 \cdot 10^{-4}$
4	0.6345	$-1.2 \cdot 10^{-4}$	0.7205	$-4 \cdot 10^{-4}$	0.7972	$2 \cdot 10^{-4}$
5	0.6349	$-8 \cdot 10^{-4}$	0.7207	$-2 \cdot 10^{-4}$	0.7981	$1.1 \cdot 10^{-4}$
6	0.6360	$3 \cdot 10^{-4}$	0.7217	$8 \cdot 10^{-4}$	0.7980	$1 \cdot 10^{-4}$

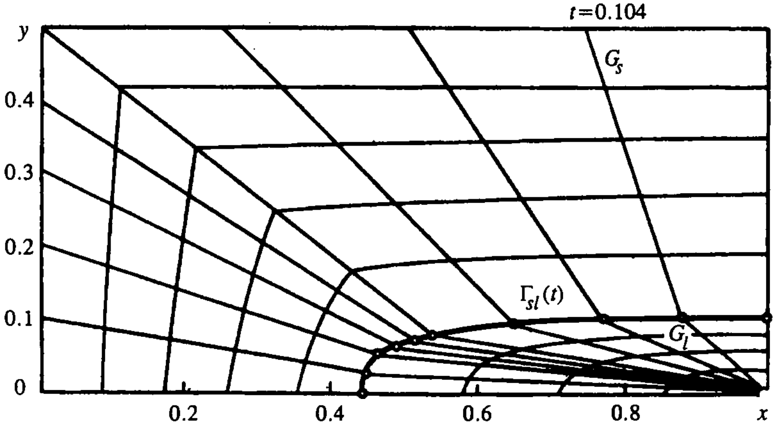

Figure 6.

We therefore studied the following model problem. The subdomain G_s in the rectangular domain $G = \{0 \le x \le 1,\ 0 \le y \le 0.5\}$ was bounded at the (initial) time $t = 0$ by $u = -1$, so it was a corner within the lines $x = 0$ and $y = 0.5$ with a width of $1.25 \cdot 10^{-2}$ in both directions. The remainder of the plane was occupied by the domain G_l with $u = 0$ (see Figure 4). Over the course of time, the phase front $\Gamma_{sl}(t)$ travelled to the node $x = 1$, $y = 0$, from the subdomain G_l to the subdomain G_s.

Although this case is similar to the first problem, a rather coarse grid was chosen with a total 10×10 nodes for the domain G. The subdomains G_s and G_l were assigned 10×6 and 10×5 nodes, respectively. Initially the space steps in the subdomains G_s and G_l differed by a factor of more than 20. The grid in the physical space was defined by the intersection of radial rays originating from the lower right corner, which is represented by the arc AE of length $l \le h_1, h_2$, and lines parallel to the coordinate axes (Figure 4). In curvilinear coordinates, the mapping $\xi = \xi(x, y, t)$, $\eta = \eta(x, y, t)$, $\tau = t$ takes the computational grid onto a rectangular one, while the phase boundary location Γ_{sl} becomes associated with the coordinate line $\eta = \eta_{M_0}$, $M_0 = 5$ (Figure 5).

Note that all the boundary conditions were approximated using finite difference relationships written in conservative form. This procedure is described in detail in [30].

Figure 6 shows the form of the computational grid and the phase-front position at time $t = 0.104$. The calculations showed that by time $t = 0.122$ the domain G_s had almost completely moved into domain G_l. The use of such an "exotic" grid with a small number of nodes, and the significant change of the spatial steps in each subdomain does not hamper the calculations in any way.

6. MODELING BOUNDARY LAYER PROBLEMS

The mathematical model of (3) and (5) with $g = -u$ was used to model two versions of the problem that admit a boundary layer solution. The investigations were to show that a two-dimensional adaptation of the computation grid is feasible.

In the first version, the following boundary-value problems were considered in the rectangular domain $G = \{0 \le x \le 1,\ 0 \le y \le 0.5\}$:

$$u(0,y,t) = 1, \qquad u(x,0,t) = 0,$$
$$u(x,0.5,t) = 1, \qquad u(1,y,t) = 0, \qquad u(x,y,0) = 0.$$

The physical and computational domains coincided in this problem. At $t = 0$, the computational grid were also same, with the total number of nodes set equal to (10×10) (Figure 7). The small parameter λ was made 10^{-3}. The characteristic size of the boundary layer was estimated to be $l \approx (\lambda)^{1/2} = 3.33 \cdot 10^{-2}$. Thus, we were concerned with the case in which the spatial steps exceeded the boundary layer's characteristic size $h_1, h_2 > l$ in each direction.

By setting nonzero boundary values of u different and along a corner, the domain of the essentially non–one-dimensional solution was fitted into the upper left corner of the computational domain. Motion to the right and down along the boundaries converts the solution in the adjacent region to the one-dimensional type. In regions where there is only

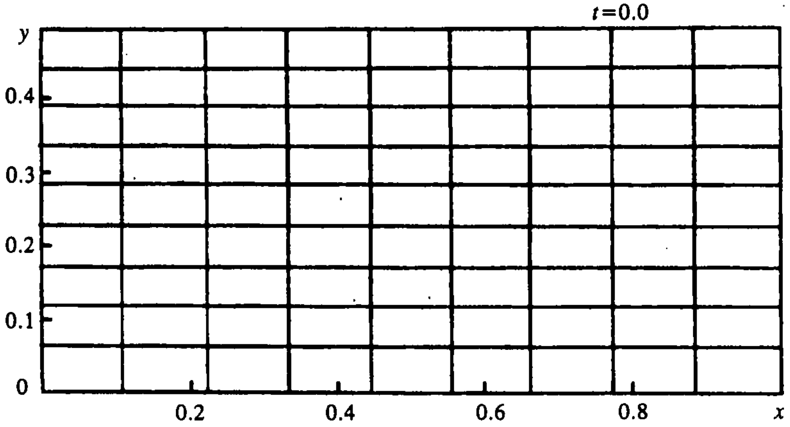

Figure 7.

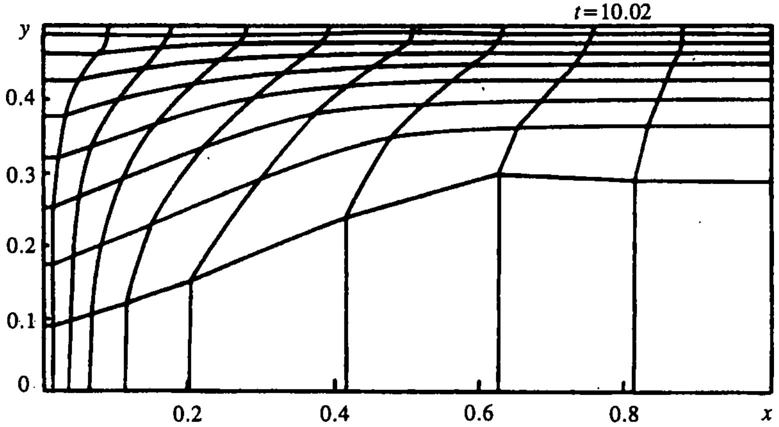

Figure 8.

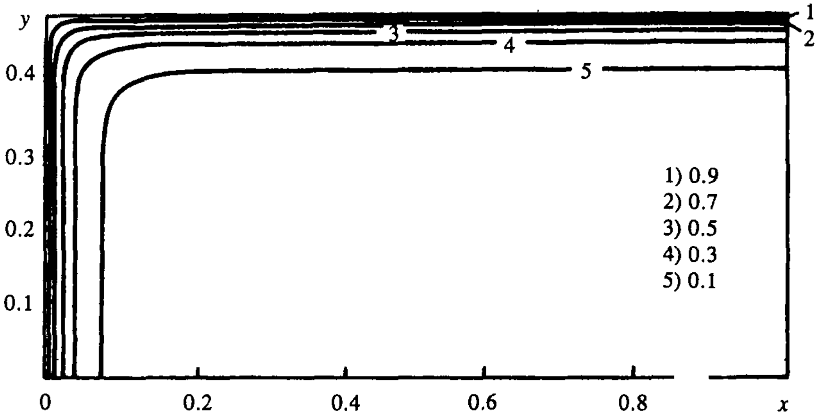

Figure 9.

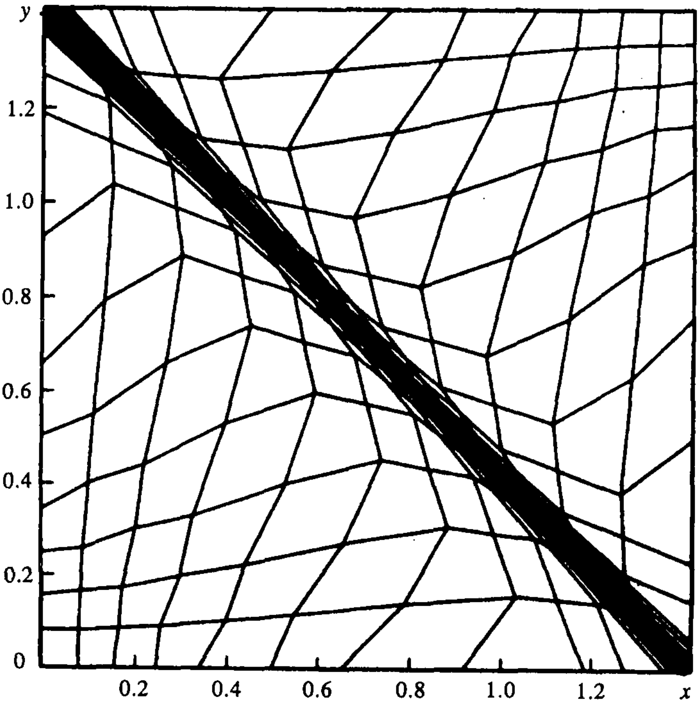

Figure 10.

one-dimensional variation, the solution is

$$u(l) = (e^{-1/\lambda^{1/2}} - e^{-(l-2)/\lambda^{1/2}})/(1 - e^{-2/\lambda^{1/2}}). \tag{10}$$

The calculations were performed on a fixed and on an adaptive grid with the same total number of nodes (10 × 10), one continued until the u profile stabilized completely. The

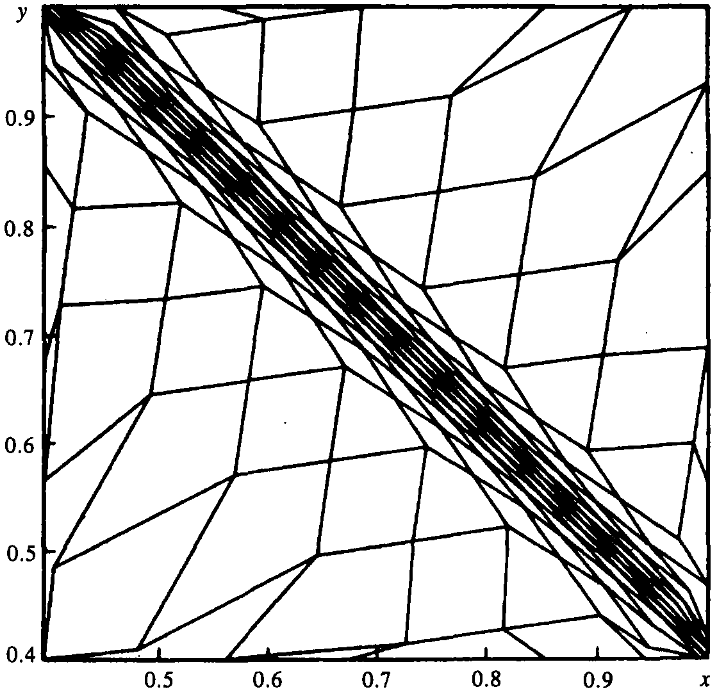

Figure 11.

results of the calculations were compared with each other and with the analytical solution in the regions where the latter exists. The comparison showed that when using a fixed grid, only one coordinate line extends to within the boundary layer region where the u_{ij} values fall by approximately one order of magnitude. With the adaptive grid with the same number of nodes, one is able to concentrate five grid coordinate lines in the same region (Figure 8). The node-by-node comparison with the analytical solution in the boundary layer region showed that the maximum error occurred in the nodes of the coordinate line nearest to the boundary, and comprised 1 to 3%.

The numerical solution obtained on the adaptive grid is presented as isolines in Figure 9. The node concentration in regions in which the solution underwent significant change resulted in an appreciable two-dimensional deformation of the grid in physical space (see Figure 8). The adaptive grid's deformation can be measured in terms of the Jacobian ψ_{ij}, which describes the areas of the cells with indices (ij). For uniform fixed grids, the cell areas are the same and remain constant over time, $\psi_{ij} = \text{const}$. The nodes of the adaptive grids coincide with the nodes of the fixed grids at $t = 0$. By the time the solution stabilizes, the values of ψ_{ij} fall by a factor of 4–5 in the boundary layer, while in the zero-solution region they increase by 3–6-fold. Thereby the computational grid in physical space becomes orthogonal (Figure 8).

An even greater deformation of the grid was achieved in the second version of the problem, in which the region was a square measuring 1.4 on a side, and the boundary condition $u = 1$ was fixed along one of the principal diagonals. Thus, the boundary layer was located along the principal diagonal. An analytical solution as in (10) exists in the direction perpendicular to this diagonal.

A uniform orthogonal grid was introduced in this region, the total number of nodes being 20 × 20. The grid steps exceeded the characteristic boundary layer thickness $l \approx 3.3 \cdot 10^{-2}$ by a factor of more than 2. As before, the calculations were continued until complete stabilization and then the results were compared with the exact ones. The configuration of the adaptive grid when the solution had stabilized is shown in Figure 10. Figure 11 shows a magnified section of the solution. Up to seven cells of the grid were concentrated in the boundary layer region within the characteristic thickness l, which led to extreme deformations of the grid. Note that the maximal difference between the cell areas in the boundary layer region and the cell area in the rest of the domain was two orders of magnitude.

It is widely believed that the accuracy of the calculations is low on very nonuniform grids, especially for multidimensional problems. This may occur because of the large error in approximating mixed derivatives and fluxes on nonorthogonal grids. However, these effects are smoothed on adaptive grids dynamically coupled with the solution. The calculations were carried out in a computational space with a fixed rectangular grid. Information about the deformation of the grid in the physical space is transmitted via metric coefficients, and since the grid deformation is determined by the solution's dynamics, the changes in the solution and in the node arrangement occur simultaneously. This implies that in regions where the solution changes significantly, the transition Jacobian and the metric coefficients decrease in proportion to the gradients, which should reduce the truncation error. In the regions where the solution changes only slightly, the cell areas increase, as do the values of ψ and the metric coefficients. However, the values of the derivatives in this region are small, and hence the truncation error is also small. For example, the error of the numerical solution in the boundary layer region was under 2%.

In conclusion, we should note that the mathematical modeling we carried out indicates that the method suggested here for generating adaptive grids is quite efficient for multidimensional boundary value problems of mathematical physics.

REFERENCES

1. Samarskii, A. A., Theory of Difference Schemes [In Russian], Nauka Press, Moscow, 1977.
2. Thompson, J.F., Grid generation techniques in computational fluid dynamics, *AIAA J.*, vol. 22, No. **11**, pp. 1505–1523, 1984.
3. Thompson, J.F., Numerical Grid Generation, North-Holland, 1982, 909 pp.
4. Yanenko, N. M., Danayev, N. T., and Liseikin, V. D., On methods variational grid generation, *Chislennye metody mekhaniki sploshnoi sredy*, vol. 8, No. **4**, pp. 157–163, Novosibirsk, 1977.
5. Brackbill, J. and Saltzmann, J., 'Adaptive zoning for singular problems in two dimensions', *J. Comput. Phys.*, No. **46**, pp. 342–368, 1982.
6. Degtyarev, L. M. and Drozdov, V. V., Solution-adaptive grids in elliptic problems on a plane, *Differentsialnyye uravneniya*, vol. 20, No. **7**, pp. 1194–1203.
7. Nakahashi, K. and Dewert, Three-dimensional adaptive grid method, *AIAA J.*, vol. 24, No. **3**, pp. 945–954, 1986.
8. Kreis, R. I., Thames, F. C., and Hassan, H., Application of a variational method for generating adaptive grids, Ibid., vol. 24, No. **3**, pp. 404–410, 1986.
9. Anderson, D. A., 'Equidistribution schemes, Poisson generators and adaptive grids', *Appl. Math. Computers*, No. **24**, pp. 211–227, 1987.
10. Rai, M. M. and Anderson, D. A., 'Application of an adaptive grid to fluid-flow problem with asymptotic solutions', *AIAA J.*, vol. 20, No. **4**, pp. 469–502, 1982.

11. Dwyer, H. A., Kee, R. J., and Sanders, B. R., 'Adaptive grid method for problems in fluid mechanics and heat transfer', Ibid., vol. 18, No. **10**, pp. 1205–1212, 1980.
12. Dwyer, H. A., Grid adaptation for problems in fluid dynamics, Ibid., vol. 22, No. **12**, pp. 1705–1712, 1984.
13. Bell, J. B. and Shubin, C. R., An adaptive grid finite difference method for conservation laws, *J. Comput. Phys.*, vol. 52, pp. 569–591, 1983.
14. Samarskii, A. A. and Popov, Yu. P., Difference Methods for Fluid Dynamics Problems [In Russian], Nauka Press, Moscow, 1980, 352 pp.
15. Godunov, S. K., Zabrodin, A. V., Ivanov, M. Ya., *et al.*, Numerical Solution of Multidimensional Gas Dynamics Problems [In Russian], Nauka Press, Moscow, 1976, 400 pp.
16. Belotserkovskiy, O. M., Numerical Modeling in Mechanics of Continuous Media, Nauka Press, Moscow, 1984, 520 pp.
17. Dar'in, N. A. and Mazhukin, V. I., Adaptive Grid Generation Method for One-Dimensional Boundary-Value Problems [In Russian], *Preprint* No. **33**, 1987, Inst. Problems of Mechanics, Moscow, 26 pp.
18. Dar'in, N. A. and Mazhukin, V. I., On an approach to adaptive difference grid generation, *Dokl. Akad. nauk SSSR*, vol. 298, No. **1**, pp. 64–68, 1988.
19. Dar'in, N. A., and Mazhukin, V. I., On an approach to adaptive grid generation for unsteady-state problems [In Russian], *Zh. vychisl. mat. mat. fiz.*, vol. 28, No. **3**, pp. 454–460, 1988.
20. Dar'in, N. A. and Mazhukin, V. I., On numerical solution of Stefan problem on an adaptive grid [In Russian], *Preprint* No. **51**, 1987, Inst. Appl. Math., Moscow, 17 pp.
21. Dar'in, N. A. and Mazhukin, V. I., Mathematical modeling of the Stefan problem on an adaptive grid, *Differentsialnyye uravneniya*, vol. 23, No. **7**, pp. 1154–1160, 1987.
22. Breslavskiy, P. V. and Mazhukin, V. I., 'Mathematical modeling of pulsed melting and evaporation of metals with explicitly separated phase boundaries', *Inzh.-fiz. zh.*, vol. 56, No. **6**, 1987.
23. Dar'in, N. A. Mazhukin, V. I., and Samarskii, A. A., A finite-difference method for gas dynamics equations using adaptive grids [In Russian], *Preprint* No. **115**, 1987, Inst. Problems of Mechanics, Moscow, 38 pp.
24. Dar'in, N. A., Mazhukin, V. I., and Samarskii, A. A., A finite-difference method for solving one-dimensional gas dynamics problems on adaptive grids, *Dokl. Akad. nauk SSSR*, vol. 302, No. **5**, pp. 1078–1081, 1988.
25. Dar'in, N. A., Mazhukin, V. I., and Samarskii, A. A., A finite-difference method for solving one-dimensional gas dynamics problems on adaptive grids dynamically coupled with the solution, *Zh. vychisl. mat. mat. fiz.*, vol. 28, No. 8, pp. 1210–1225, 1988.
26. Dar'in, N. A., Mazhukin, V. I., and Samarskii, A. A., A finite-difference method for solving unsteady-state two-dimensional boundary value problems on an adaptive grid dynamically coupled with the solution [In Russian], *Preprint* No. **117**, Inst. Appl. Math., Moscow, 27 pp.
27. Bakirova, O. I., On some methods of solving the Stefan problem, *Differentsialnyye uravneniya*, vol. 19, No. **3**, pp. 491–500, 1983.
28. Davis, M., Kapadia, P., and Dowden, J., 'Solution of a Stefan problem in the theory of laser welding by the method of lines', *J. Comput. Phys.*, vol. 60, pp. 534–548, 1985.
29. Carslaw, H. S. and Jaeger, J. C., Conduction of Heat in Solids. Oxford. Univ. Press, 1959, Russian translation (1964), pp. 534–548.
30. Dar'in, N. A. and Mazhukin, V. I., 'Mathematical modeling of an unsteady-state two-dimensional Stefan problem on an adaptive grid' [In Russian], *Preprint* No. **52**, Inst. Appl. Math., Moscow, 1987, 25 pp.