NUCLEAR SCIENCE AND ENGINEERING: 123, 169–189 (1996)

Numerical Simulations of Nonstationary Fronts and Interfaces by the Godunov Method in Moving Grids

V. E. Fortov

High Energy Density Research Center, IVTAN, Moscow 127412, Russia

B. Goel and C.-D. Munz

Forschungszentrum Karlsruhe, Technik und Umwelt
Institut für Neutronenphysik und Reaktortechnik
D-76021 Karlsruhe, Germany

and

A. L. Ni, A. V. Shutov, and O. Yu. Vorobiev

Institute of Chemical Physics, Moscow Region, Chernogolovka 142432, Russia

Received May 22, 1995
Accepted November 17, 1995

Abstract—*A Godunov scheme is proposed for the simulation of impact problems and detonations where nonstationary fronts and interfaces are tracked as boundaries of subregions that move in time. In each subregion and at each time step, a new grid is created by the use of boundary-fitted coordinates. The numerical method is based on a finite-volume approach in the space-time domain, and the fluxes are calculated using the solution of Riemann problems. Numerical results are shown for several impact and detonation problems, showing the efficiency of this approach.*

I. INTRODUCTION

Safety-related problems are very important in nuclear energy production because of the extremely high concentration of energy and the strong radioactivity of the working media in nuclear reactors. Most modern accident scenarios are associated either with an explosion of air-hydrogen mixtures and condensed explosives or with hypervelocity impacts of condensed fragments. Direct experimental study of these complicated phenomena is cumbersome and expensive. Theoretical methods of investigation are based on mathematical modeling of nonsteady nonuniform flows of compressible media that may be chemically active and may undergo phase transitions. The state of matter may change over a wide range: from hot gas to highly compressed solid states at high pressures. The only way to solve the multidimensional nonlinear equations of motion together with wide-range equations of state (EOSs) and reaction kinetics equations is by numerical simulation. This paper concentrates on numerical aspects of the problems of hypervelocity impact and detonations.

One of the fundamental problems in these computations is the numerical treatment of fronts and interfaces. Practical problems of this type are hypervelocity impact problems. In a typical case, an interface separates two nonmixing materials. The numerical calculation introduces some numerical smearing and artificial mixing of the components, but no physical meaningful EOS for the mixed material is available. In modeling these phenomena by way of the conservation laws of mass, momentum, and energy for a continuous medium, the fluid dynamics description with sharp resolution of the discontinuities is essential for the validity and quality of the numerical simulation.

Usually, the conservation equations are written for a coordinate system fixed in space, the so-called Eulerian coordinates. Numerical approximations then use a

mesh, also fixed in space, which makes them ideal for use in flows with large deformations. These Eulerian methods are widely used in fluid dynamics as the multidimensional numerical treatment is straightforward. However, within a fixed grid, moving interfaces are bound to be smeared out over a couple of grid zones. There are two approaches to reduce drastically this numerical dissipation of moving fronts: the high-resolution schemes, which have been developed by a number of authors in the past 15 yr (for a review, see Refs. 1 and 2), and adaptive grid techniques (see, e.g., Refs. 3 and 4). In the case of a material interface, these methods do not succeed, as there is at least one intermediate state within a material layer that requires an artificial EOS. Hence, high-velocity impact problems are usually based on Lagrangian grids, moving with the flow, or on the so-called particle methods. In a Lagrangian method, material interfaces can be kept sharp without introducing artificial intermediate states because they become stationary in this frame of reference (see, e.g., Ref. 5). Unfortunately, Lagrangian grids break down when fluid distortions become large because the grid cells deform and get tangled. To continue the calculation, one must start a mesh rezoning procedure, which consumes much computer time and, in addition, may introduce the unacceptable smearing of the interfaces in this step. A review of these methods as well as particle methods and their applications to the hypervelocity impact problems is given in Refs. 6 and 7.

In this paper, a strategy is proposed that combines both the Eulerian and the Lagrangian approaches. It is based on the Arbitrarily Lagrangian-Eulerian (ALE) method of Hirt, Amsden, and Cook[8] and the ideas of Godunov et al.[9] The new philosophy in our approach is to divide the computational domain into several subregions in such a way that each subregion contains only one material and material interfaces or fronts coincide with boundaries of subregions. The conservation equations are formulated and approximated on a time-dependent grid called a moving grid. Boundaries of the subregions associated with material interfaces are moved at the local fluid velocity, which corresponds to a Lagrangian treatment of the interfaces. The grid inside each subregion is generated at each time step using boundary-fitted coordinates. Hence, fluid distortions inside each subregion are well captured. But, strong deformations of the material interfaces may cause difficulties. To avoid the breakdown of the calculation in this case, we introduce interactive and adaptive generations of new subregions during the calculations.

An accurate treatment of interfaces is important not only for nonmixing materials but also for nonstationary fronts at which chemical reactions or phase transitions take place. Diffusion errors at the front can significantly change the hydrodynamic behavior of the entire flow. Material vacuum boundaries that are a difficult problem for a Eulerian method (see Ref. 10) can also be treated in an efficient way by a tracking method. They are favorable in general for all problems in which a few moving fronts determine the dynamic behavior of the system and become difficult to apply when different fronts interact or the front becomes highly distorted. Material mixing by instabilities and turbulence at the interface cannot be treated.

There exist several approaches to track fronts and interfaces. A review of these concepts is given by Hyman.[11] Besides moving mesh methods, volume- and surface-tracking methods are applied usually on Cartesian grids. In a volume-tracking method, the regions with different materials are identified by fraction variables that, e.g., change from one to zero along the interface. The fractional volume is calculated and advanced during the computation by the numerical solution of a scalar transport equation. At each time level, this variable is used to reconstruct the interface. For surface-tracking methods, the location of the interface is either specified by interpolation between marker particles or given as a certain level set of a scalar function. The first approach is used, e.g., by Chern et al.[12]; the level set approach is described in detail by Mulder, Osher, and Sethian.[13] In all these tracking methods on fixed grids, the information of the interface location is used in some sophisticated intracell considerations to obtain appropriate values for the numerical fluxes at the cell face, which do not smear out the interface (see also Ref. 14). In our moving grid technique, the interface is always defined to be identical to a subregion boundary, and hence, the fluxes can be calculated in the usual manner.

The outline of this paper is as follows. Section II contains a survey of the proposed overall strategy. Section III describes the mathematical modeling. The discretized equations and the basics of the numerical scheme in moving grids are treated in Sec. IV. The building block in the flux calculation between different grid zones, which is the solution procedure for the Riemann problem, is given in Sec. V. This completes the description of the numerical approximation within each subregion. The determinations of the time evolution of the subregion boundaries and the grid generation in the subregions are described in Secs. VI and VII, respectively. Dividing the computational region and the adaptive generation of subregions is proposed in Sec. VIII. In Sec. IX, this method is applied to several problems of hypervelocity impact and the time evolution of detonation fronts. A summary and conclusions in Sec. X close this paper.

II. THE CALCULATION STRATEGY

The basic idea of interface tracking, as described in this paper, is dividing the computational region into different subregions in such a way that boundaries of subregions are identified with the interfaces. In each

subregion, the grid is defined independent of the neighboring ones. Within each time cycle, the overall procedure consists of the following three steps:

1. calculation of the movement of the boundaries for all subregions
2. generation of a new grid in each subregion
3. calculation of the approximate values at the new time level.

The first step serves to move any subregion boundary with an appropriate front velocity, e.g., with the local fluid velocity, if it is associated with a material interface. Within a conservative scheme, this allows the interface to be resolved without any intermediate points being introduced; it corresponds to a local use of Lagrangian mass coordinates. Within this frame of reference, the interface becomes stationary, and the numerical method can now combine conservation with an exact resolution of an interface (see, e.g., Ref. 5).

The information about the movement of the interface is obtained by solving Riemann problems, which are initial-value problems for the governing equations with piecewise constant initial values, one constant state on either side of the interface. The solution provides information about the breakup of the jump between the right and the left states into different waves and, hence, an estimate of the wave velocities. These local wave velocities are used to move the grid points. The idea of using the solution of the Riemann problem within the numerical framework is due to Godunov.[15] He assumed the approximation to be constant within each grid zone. The fluxes between adjacent grid zones are then calculated by solving the Riemann problem with the corresponding constant data. We will adopt this procedure also for the numerical method inside each subregion. In the first step, we solve only the Riemann problem at the subregion boundaries to obtain an estimate of their movement. As the location and the number of grid points at the interfaces from different sides may not coincide because of different discretizations in the subregions, appropriate averages have to be calculated as initial data for the Riemann problem data. Noninterface boundary conditions for the subregion boundaries are implemented in a similar way by defining appropriate initial data for the Riemann problems.

When the location of the subregion boundaries has been defined at the new time level within this first step, a new grid is generated in each subregion. This is done in a second step using boundary-fitted coordinates, which requires the numerical solution of an elliptic equation (see, e.g., Ref. 16). A good approximation of the new grid is available from the previous time level as long as the displacement of the interfaces during one time step is not excessive, and the elliptic equation can be solved quickly by a simple iterative method. In steps 1 and 2, an appropriate grid has been defined in each subregion at the old and the new time levels, and the conservation equations are solved within this time-dependent grid. This is done in an extension of the usual finite-volume approach. The conservation equations are formulated in an integral form, where the domain of integration is a grid zone considered as part of the space-time domain. This formulation of the conservation equations and an appropriate approximation has been proposed by Hirt, Amsden, and Cook[8] in their so-called ALE methods and by Godunov et al.[9]

The structure of the program is sketched in Fig. 1. The different steps are described in more detail later. We remark that within this numerical framework, it is possible to move the subregion boundaries with any velocity, e.g., that of material interfaces, shock waves, or flame fronts.

III. THE EQUATIONS

The motion of a continuous medium may be described by the integral laws of the conservation of mass, momentum, and energy:

$$\frac{d}{dt}\int_V \rho\, dV = -\int_{\partial V} \rho \boldsymbol{v}\cdot\boldsymbol{n}_r\, dS\ , \tag{1a}$$

$$\frac{d}{dt}\int_V \rho \boldsymbol{v}\, dV = -\int_{\partial V} [(\rho \boldsymbol{v}\circ\boldsymbol{v}) + pI]\boldsymbol{n}_r\, dS\ , \tag{1b}$$

and

$$\frac{d}{dt}\int_V e\, dV = -\int_{\partial V} [(e+p)\boldsymbol{v}]\cdot\boldsymbol{n}_r\, dS\ , \tag{1c}$$

where

ρ = density

$\boldsymbol{v} = (u,v,w)^T$ = velocity vector

p = pressure

e = total energy per unit volume, which may be split into an internal and a kinetic part by

$$e = \rho\epsilon + \tfrac{1}{2}\rho\boldsymbol{v}^2\ , \tag{2}$$

where ϵ is the specific internal energy and $\boldsymbol{v}^2 = \boldsymbol{v}\cdot\boldsymbol{v}$. The set of conservation equations is completed by an EOS

$$p = p(\rho,\epsilon)\ . \tag{3}$$

Equations (1) are the common formulation of the physical integral conservation laws: The change in time of mass, momentum, and energy within an arbitrary constant volume V is due to the fluxes through the surface ∂V of V. The vector $\boldsymbol{n}_r$ in Eqs. (1) denotes the outward directed unit normal of ∂V. In this formulation, V is assumed to be fixed and not to change in time.

Viscous terms, thermal conductivity, diffusion, and radiation are neglected in Eqs. (1). It is assumed that the stress tensor is diagonal: $\Pi_{ik} = p\delta_{ik}$, and that each element of the medium is in the state of thermodynamic

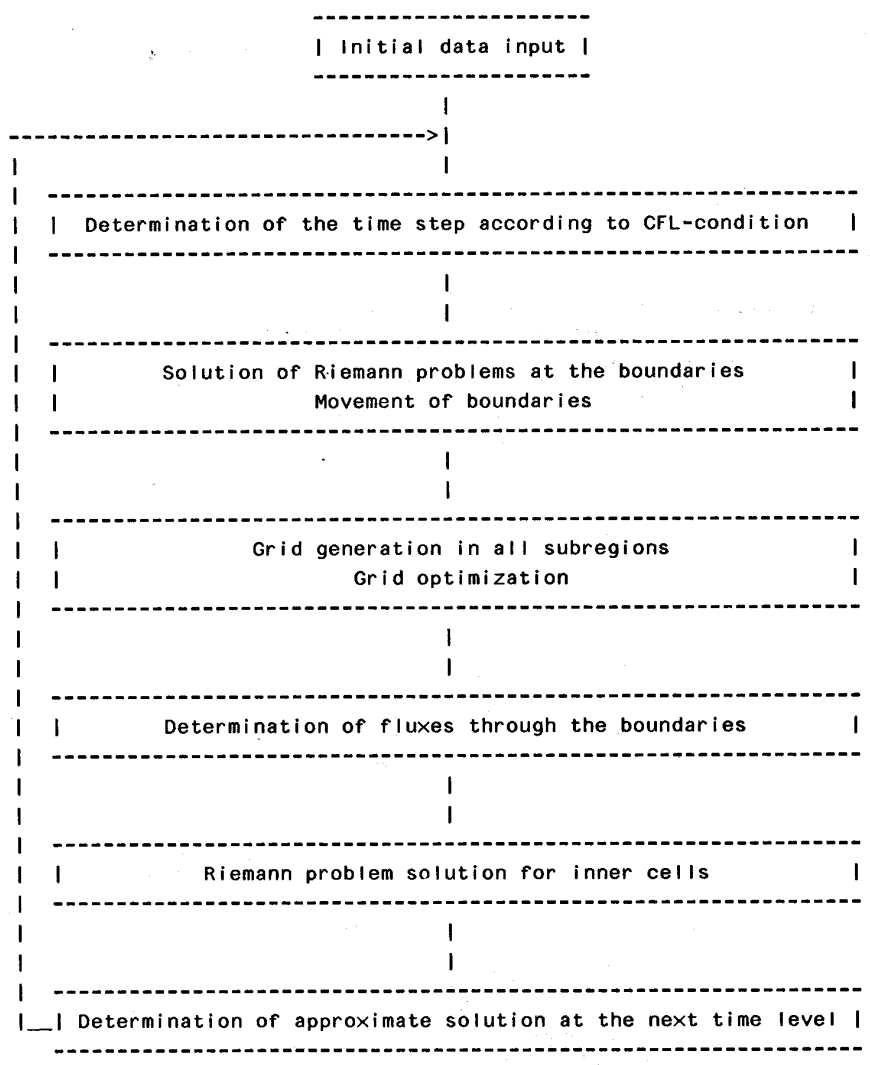


Fig. 1. Block diagram of the algorithm in moving grids.

equilibrium, so that the equations of motion (1) can be closed. In this case, Eqs. (1) mathematically model inviscid compressible fluid flow and are usually called the Euler equation in integral form. Beside problems of gas dynamics, Eqs. (1) also establish the hydrodynamic approximation of shock waves in condensed media. If the pressure developed in a solid body is much higher than at its yield point, this model provides an accurate description of the wave propagation. Typical applications include high-velocity impact problems or interaction of laser and charged-particle beams, and strong detonations with solid targets. As the load pulse at the material decreases, the elastoplastic properties of the medium become more significant for the dynamics, and deviatoric stresses and relaxation processes have to be taken into account. A comprehensive treatment of the physical phenomena and their mathematical modeling are given in the classical books of Zel'dovich and Raizer[17] and Courant and Friedrichs.[18] Elastic-plastic effects lead to equations that are more complex, but they belong to the same class of hyperbolic quasilinear equations[19,20] and may be numerically solved by similar methods. The simulation of elastic-plastic phenomena is not a subject of this paper.

The integral conservation equations (1) are the starting point for the so-called finite-volume schemes. The physical space is divided into a number of volumes, the grid cells. A numerical scheme is then obtained by direct approximation of the integral equations (1) for each volume. To develop schemes in which the boundary of the volumes may move, we reformulate the integral conservation equations (1). The volume Ω, which

will be considered, is now part of the four-dimensional space $\mathbb{R}^3 \times \mathbb{R}_0^+$. The conservation equations then have the form

$$\int_{\partial\Omega} [\rho \nu_t + \rho \boldsymbol{v} \cdot \boldsymbol{\nu}_r] \, dV = 0 \ , \tag{4a}$$

$$\int_{\partial\Omega} [\rho \boldsymbol{v} \nu_t + (\rho \boldsymbol{v} \circ \boldsymbol{v} + pI)\boldsymbol{\nu}_r] \, dV = 0 \ , \tag{4b}$$

and

$$\int_{\partial\Omega} [e\nu_t + (e+p)\boldsymbol{v} \cdot \boldsymbol{\nu}_r] \, dV = 0 \ , \tag{4c}$$

where $\partial\Omega$ is the hypersurface bounding the volume Ω. The outer unit normal of $\partial\Omega$ is given by $\boldsymbol{\nu} = (\boldsymbol{\nu}_r, \nu_t)^T$. Equations (4) again express the integral conservation, now formulated in the space-time domain: The change of mass, momentum, and energy within an arbitrary volume Ω is due to the fluxes through the surfaces $\partial\Omega$. If Ω is fixed in time, then Eqs. (4) reduce to Eqs. (1).

We denote by $W = W(t)$ the projection of Ω onto a hyperplane $t =$ constant being a volume in $\mathbb{R}^3$ and assume that the hypersurface $\partial W(t)$ for any time t is given by the set of points, which satisfy

$$S(\boldsymbol{r}, t) = 0 \tag{5}$$

with $r = (x, y, z)^T$. The function S is called the evolving function of the hypersurface. Let $\boldsymbol{r} = \boldsymbol{r}(t)$ be the path of a point on the moving surface. If this point propagates in its normal direction n_r at the velocity $D = D[\boldsymbol{r}(t)]$, this results in

$$\frac{\partial \boldsymbol{r}(t)}{\partial t} = D[\boldsymbol{r}(t)] n_r \ . \tag{6}$$

Next, the evolving function $S = S[\boldsymbol{r}(t), t]$ is differentiated with respect to t, and the chain rule is used to obtain

$$\frac{\partial S}{\partial t} + \nabla S \cdot \boldsymbol{n}_r D = 0 \tag{7}$$

with $\nabla S = (\partial S/\partial x, \partial S/\partial y, \partial S/\partial z)^T$. On the other hand, the unit normal $\boldsymbol{\nu}$ of $\partial\Omega$ is given by

$$\boldsymbol{\nu} = \frac{1}{L}\left(\nabla S, \frac{\partial S}{\partial t}\right)^T$$

with

$$L = \left[\nabla S^2 + \left(\frac{\partial S}{\partial t}\right)^2\right]^{1/2} \ .$$

Because of $n_r = \nabla S/|\nabla S|$, and replacing $\partial S/\partial t$ via relation (7), we rewrite $\boldsymbol{\nu}$ as

$$\boldsymbol{\nu} = \frac{|\nabla S|}{L}\begin{pmatrix} n_r \\ -D \end{pmatrix} = \frac{1}{(1+D^2)^{1/2}}\begin{pmatrix} n_r \\ -D \end{pmatrix} . \tag{8}$$

For the construction of a numerical method, it is convenient to take Eqs. (4) in a special case, where Ω is a region of the four-dimensional space bounded by hyperplanes $t = t_1$ and $t = t_2$ with $t_2 > t_1$. Using Eq. (8), the conservation equations (4) may then be simplified to

$$\int_{W(t_2)} \rho \, dV = \int_{W(t_1)} \rho \, dV - \int_{\partial\Omega'} \rho(\boldsymbol{v} \cdot \boldsymbol{n}_r - D) \, d\sigma \ , \tag{9a}$$

$$\int_{W(t_2)} \rho \boldsymbol{v} \, dV = \int_{W(t_1)} \rho \boldsymbol{v} \, dV - \int_{\partial\Omega'} [\rho(\boldsymbol{v} \cdot \boldsymbol{n}_r - D)\boldsymbol{v} + p\boldsymbol{n}_r] \, d\sigma \ , \tag{9b}$$

and

$$\int_{W(t_2)} e \, dV = \int_{W(t_1)} e \, dV - \int_{\partial\Omega'} [e(\boldsymbol{v} \cdot \boldsymbol{n}_r - D) + p\boldsymbol{v} \cdot \boldsymbol{n}_r] \, d\sigma \ , \tag{9c}$$

where

$W(t_1)$ = projection of Ω onto hyperplane $t = t_1$

$W(t_2)$ = projection of Ω onto hyperplane $t = t_2$

$\partial\Omega' = \partial\Omega \backslash [W(t_1) \cup W(t_2)]$

$d\sigma = |\nabla S|/L dV$ = component of the volume element dV along the normal $\boldsymbol{n}_r$.

The additional terms in the fluxes are generated by the movement of the grid and are convection terms involving velocity D. If D vanishes, Eqs. (1) are reestablished. The numerical scheme in a moving grid proposed in Sec. IV is designed as an approximation of these integral equations (9).

IV. GODUNOV SCHEME IN MOVING GRIDS

In the following, we restrict ourselves to two space dimensions. For simplicity, it is assumed that the computational region $W(t)$ at a fixed time $t = t_n$ is a curvilinear quadrangle with the boundary Γ_r^n, Γ_l^n, Γ_f^n, and Γ_b^n (see Fig. 2) and that a structured grid for $W(t_n)$ exists. A grid zone (i,j) at the time t_n is a quadrangle denoted by $R_{i,j}^n$ and is given via the coordinates of the four corners

$$P_{i-1/2,j-1/2}^n = (x_{i-1/2,j-1/2}^n, y_{i-1/2,j-1/2}^n) \ ,$$

$$P_{i+1/2,j-1/2}^n = (X_{i+1/2,j-1/2}^n, y_{i+1/2,j-1/2}^n) \ ,$$

$$P_{i-1/2,j+1/2}^n = (x_{i-1/2,j+1/2}^n, y_{i-1/2,j+1/2}^n) \ ,$$

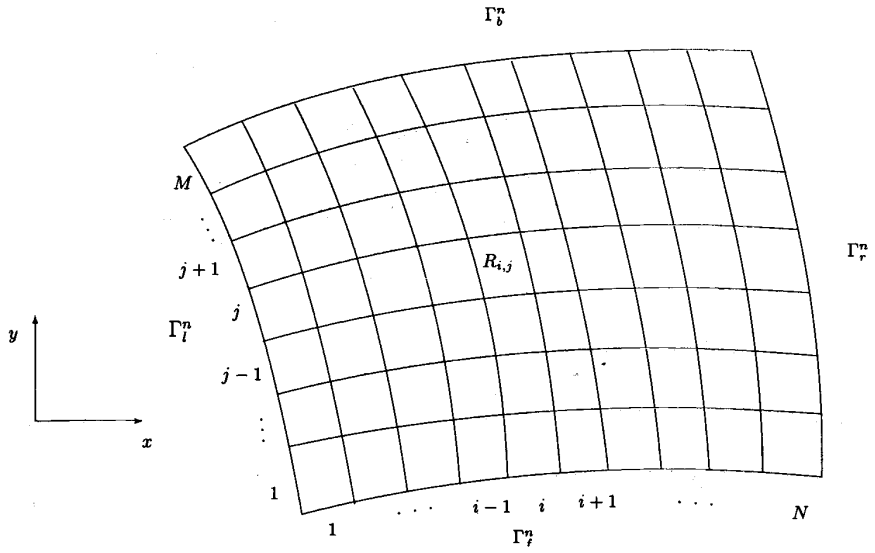


Fig. 2. Computational region and corresponding grid at the fixed time $t = t_n$.

and

$$P^n_{i+1/2,j+1/2} = (x^n_{i+1/2,j+1/2}, y^n_{i+1/2,j+1/2}) \quad (10)$$

with $i = 1,\ldots,N$ and $j = 1,\ldots,M$ (see Fig. 2). If these corners move in time, the grid zone $R^{n+1}_{i,j}$ is defined at the next time level $t_{n+1} = t_n + \Delta t$ by the new set of points

$$P^{n+1}_{i-1/2,j-1/2}, P^{n+1}_{i+1/2,j-1/2}, P^{n+1}_{i-1/2,j+1/2}, P^{n+1}_{i+1/2,j+1/2} \quad (11)$$

with coordinates defined analogously to Eq. (10). The control volume $\Omega^n_{i,j}$, which is a region in the three-dimensional space $(x,y,t) \subseteq \mathbb{R}^2 \times \mathbb{R}^+_0$, is then defined by the two quadruples of points (10) and (11) (see Fig. 3).

For the integral Euler equations (9), we introduce for this situation the shorthand notation

$$\int_{R^{n+1}_{i,j}} \boldsymbol{u}\, dV = \int_{R^n_{i,j}} \boldsymbol{u}\, dV - \sum_{m=1}^{4} Q^n_{m,i,j} , \quad (12)$$

where $\boldsymbol{u}$ denotes the vector of the conserved variables $\boldsymbol{u} = (\rho, \rho u, \rho v, e)^T$ with the velocity $\boldsymbol{v} = (u,v)^T$ and $Q^n_{m,i,j}, m = 1,\ldots,4$, denotes the integral of the fluxes into the normal direction through the four sides of Ω^n_{ij}.

The side faces of Ω^n_{ij} are the surfaces connecting the corresponding corners of $R^n_{i,j}$ and $R^{n+1}_{i,j}$; e.g., the side face S_l on the left is defined by

$$P^n_{i-1/2,j-1/2}, P^n_{i-1/2,j+1/2}, P^{n+1}_{i-1/2,j-1/2}, P^{n+1}_{i-1/2,j+1/2} \quad (13)$$

(see Fig. 4). The other sides on the right, front, and back, with reference to Fig. 4, are called S_r, S_f, and S_b, respectively. As side faces in general are not plane, there is some arbitrariness in their definition within the numerical framework. According to Eqs. (9), for the flux evaluation, the velocity D of the sides into the normal direction is needed, and an appropriate calculation of these values is proposed as follows.

For this, we consider the left side face S_l, which separates $\Omega^n_{i,j}$ and $\Omega^n_{i-1,j}$; the velocity of the side into the normal direction is a function of (x,y,t). For construction of the numerical scheme, we define an appropriate average by

$$D = |\Pi(S_l)| / (d_l \Delta t) , \quad (14)$$

where $\Pi(S_l)$ is the projection of S_l onto the plane $t = t_n$. This projection is used as a measure of the change of the area of grid zone $R^n_{i,j}$ caused by the movement of the edges $P^n_{i-1/2,j-1/2}, P^n_{i-1/2,j+1/2}$. The change into the normal direction is obtained by dividing an average length of S_l in the tangential direction denoted d_l. In our calculations, this average length is defined by the distance of the midpoints

$$P^{n+1/2}_{i-1/2,j-1/2} = \tfrac{1}{2}(P^{n+1}_{i-1/2,j-1/2} + P^n_{i-1/2,j-1/2})$$

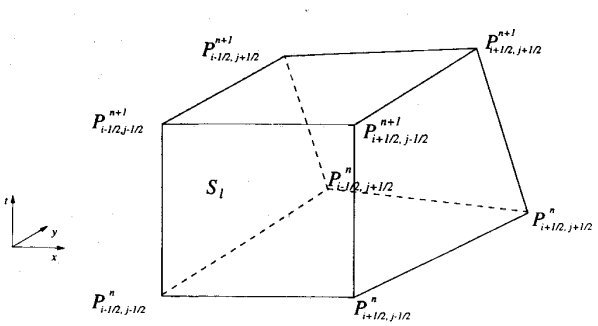


Fig. 3. Grid zone $\Omega^n_{i,j}$ in the space-time domain with corner points and the side face S_l.

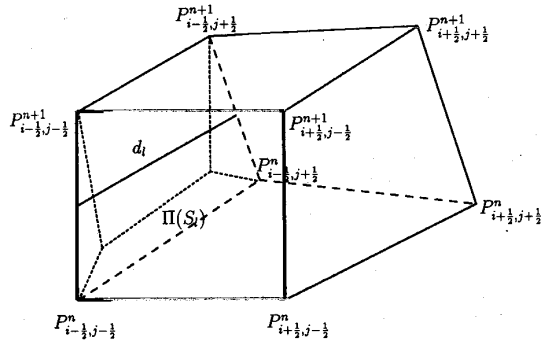


Fig. 4. Grid zone $\Omega^n_{i,j}$ with corner points and the projection $\Pi(S_l)$ of the side face S_l onto the plane $t = t_n$.

and

$$P^{n+1/2}_{i-1/2,j+1/2} = \tfrac{1}{2}\,(P^{n+1}_{i-1/2,j+1/2} + P^{n}_{i-1/2,j+1/2}) \ . \quad (15)$$

If a side face is a plane rectangle, the velocity D as defined in Eq. (14) coincides with the local velocity of the side at each point.

The numerical method considered in this paper is a direct approximation of the integral equation (12) and may be called a finite-volume method in a moving grid. It has the form

$$|R^{n+1}_{i,j}|\,\boldsymbol{u}^{n+1}_{i,j} = |R^{n}_{i,j}|\,\boldsymbol{u}^{n}_{i,j} - \boldsymbol{F}^{n}_{l} - \boldsymbol{F}^{n}_{r} - \boldsymbol{F}^{n}_{f} - \boldsymbol{F}^{n}_{b} \ , \quad (16)$$

where $\boldsymbol{u}^n_{i,j}$ denotes the approximate value of the average of the solution $\boldsymbol{u}$ within $R^n_{i,j}$ and $\boldsymbol{F}^n_l$, $\boldsymbol{F}^n_r$, $\boldsymbol{F}^n_f$ and $\boldsymbol{F}^n_b$ denote the approximation of the integrated fluxes between the grid zones, called numerical fluxes. To calculate these fluxes, we need an appropriate approximation of the normal directed outward from the surfaces. A very obvious choice, e.g., for the surface S_l, is the normal of the segment defined by points (15), which have already been used to calculate the velocity D of the surface S_l. This normal $\boldsymbol{n}_l$ and the tangent $\boldsymbol{\tau}_l$ are given by

$$\boldsymbol{n}_l = -\frac{1}{d_l}\begin{pmatrix} \Delta y_l \\ -\Delta x_l \end{pmatrix}$$

and

$$\boldsymbol{\tau}_l = \frac{1}{d_l}\begin{pmatrix} \Delta x_l \\ \Delta y_l \end{pmatrix} \ , \quad (17)$$

respectively, where Δx_l and Δy_l are defined to be

$$\Delta x_l = x^{n+1/2}_{i-1/2,j+1/2} - x^{n+1/2}_{i-1/2,j-1/2}$$

and

$$\Delta y_l = y^{n+1/2}_{i-1/2,j+1/2} - y^{n+1/2}_{i-1/2,j-1/2} \ , \quad (18)$$

and d_l denotes the length of the segment.

Based on the integral approximate values, Godunov,[15] in his construction of a numerical scheme, defined a piecewise constant approximation by

$$\boldsymbol{u}^n_\Delta(x,y) = \boldsymbol{u}^n_{i,j} \quad \text{for } (x,y) \in R^n_{i,j} \ . \quad (19)$$

To obtain the fluxes between neighboring grid zones, he then solved the problem of the breakup of the jumps into waves. The numerical fluxes between the grid zones are then calculated from the fluxes generated by their wave propagation. The approximation of the integral of the fluxes is performed by the midpoint rule. In this concept, the initial value problem must be solved for the conservation laws with constant left and right states. Introducing the velocity components into the normal and tangential directions v_n and v_τ, respectively, makes this problem one-dimensional into the normal direction and reads as

$$\boldsymbol{u}_t + f(\boldsymbol{u})_\zeta = 0 \ , \quad \boldsymbol{u}(\xi,0) = \begin{cases} \boldsymbol{u}_l & \text{for } \zeta < 0 \\ \boldsymbol{u}_r & \text{for } \zeta > 0 \end{cases} \ , \quad (20)$$

where

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho v_n \\ \rho v_\tau \\ e \end{bmatrix} \ , \quad \boldsymbol{f}(\boldsymbol{u}) = \begin{bmatrix} \rho v_n - \rho D \\ \rho v_n^2 - \rho v_n D + p \\ \rho v_\tau v_n - \rho v_\tau D \\ v_n(e+p) - eD \end{bmatrix} \ . \quad (21)$$

This may be deduced from Eqs. (9) in a straightforward way in the case of the continuity and the energy equations (9a) and (9c), respectively. In the case of the momentum equations, the flux vector is multiplied by the matrix

$$\mathbf{T} = \begin{pmatrix} a & b \\ -b & a \end{pmatrix} \ , \quad (22)$$

where a and b are the components of the unit normal, e.g., in the case of the flux F_l through the left side face,

$$a = -\frac{1}{d_l}\,\Delta y_l \ , \quad b = \frac{1}{d_l}\,\Delta x_l \ . \quad (23)$$

Multiplication by Eq. (22) corresponds to a local change of the coordinate system with new coordinate axes into the normal and the tangential directions. The change of the transformed fluxes locally determines the time development of the normal component of the momentum as well as the tangential component due to convection into the normal direction. This Riemann problem then must be solved. The solution procedure and the structure of the solution are described in Sec. V.

The flux of the Riemann problem at $\zeta = 0$ is a good approximation to the local flux introduced by the nonlinear wave propagation into the normal direction. The momentum flux must be rotated back by multiplication by

$$\mathbf{T}^{-1} = \begin{pmatrix} a & -b \\ b & a \end{pmatrix} \tag{24}$$

for the components in the x and y directions to be obtained. According to the midpoint rule, the flux is multiplied by the area of the corresponding side face to obtain the finite-volume formulation (16). For a structured grid fixed in time, i.e., $D = 0$, the foregoing procedure is well known and described in detail, e.g., in Refs. 1 and 9.

The scheme in moving grids is now fully defined, when the positions of the grid zone vertices are known at the old and the new time levels. Equation (14) allows the average velocity D of each side face to be approximated and the average unit normal (17) to be defined. Then the fluxes between the neighboring grid cells are calculated from the solution of the Riemann problem. This introduces nonlinear wave propagation into the numerical solution and is the main reason for the overall robustness of the Godunov scheme. The time step in the procedure is limited because of the assumption that waves from neighboring grid zone interfaces do not interact. This leads to the usual time-step restriction of explicit schemes for the Euler equations: the CFL condition. It expresses the fact that waves should run at most through one grid cell in one time step.

In smooth regions of the flow, the accuracy of the Godunov scheme is of the first order in time and space only. Although higher order modifications have been proposed for fixed grids (see, e.g., Refs. 21 and 22), which may be extended to moving grids, we restrict ourselves in this paper to the foregoing basic algorithm and focus our attention on the accurate resolution of interfaces.

V. THE RIEMANN PROBLEM

The solution of the Riemann problem, Eqs. (20) and (21), is the basic building block in the Godunov method. In this section, we discuss the solution procedure used in our calculations. If the grid velocity D vanishes, the flux of the Riemann problem is identical to that of the one-dimensional Euler equations. The additional equation for the tangential momentum component is a pure transport equation and can be solved within the one-dimensional solution procedure. The terms involving the grid velocity D are convection terms only and introduce no additional difficulties. Hence, D will be set equal to zero first, and then the modification for the nonzero case will be described.

We will consider a general EOS, but we will assume that it satisfies the following conditions: If the EOS is given as $p = p(V,S)$, where V is the specific volume $1/\rho$ and S is the entropy, then the inequalities

$$p_V < 0 \ , \quad p_{VV} > 0 \ , \quad p_S > 0 \tag{25}$$

are valid. These relations are usually called convexity conditions and guarantee that the characteristic fields of the Euler equations are either genuinely nonlinear or linearly degenerated in the sense of Lax.[23] This ensures that the solution of the Riemann problem for the Euler equations exists and has a structure as given in Fig. 5 (see, e.g., Refs. 24 and 25). The solution is a function of ξ/t and consists of four constant states separated by three elementary waves. The right and the left elementary waves are shock or rarefaction waves, while the wave in between is a contact discontinuity, associated with a linearly degenerated characteristic field. Across a contact discontinuity, the pressure as well as the velocity is continuous. The values of this intermediate state are denoted by p_* and v_* for pressure and velocity, respectively. Solving the Riemann problem requires finding out in phase-space how to connect the right and the left states by a set of appropriate elementary waves. As velocity and pressure are constant in the intermediate state, the projection of the phase-space is best considered onto the (v,p) plane. The structure of the solution procedure is discussed later, and some remarks will be added about possible approximations to save computational effort.

If the left wave is assumed to be a shock wave, the Rankine-Hugoniot relations must apply across that discontinuity and provide the connection between the shock velocity s and the values of the physical quantities behind the shock wave identified by the subscript 1. If the shock velocity is eliminated in these expressions, the remaining two relations are

$$v_* - v_l = [(p_* - p_l)(V_1 - V_l)]^{1/2} \tag{26}$$

and

$$\epsilon_1 - \epsilon_l = \tfrac{1}{2}(p_* + p_l)(V_1 - V_l) \ . \tag{27}$$

Together with the EOS, these form a nonlinear system of three equations with four unknown quantities. If the EOS is given in an inverse form

$$\epsilon = \epsilon(V,p) \ , \tag{28}$$

the internal energy may be replaced in Eq. (27). Elimination of V_1 in Eq. (27) and inserting it into Eq. (26) results in a relation of the form

$$v_* = F_l(p_*;\rho_l,v_l,p_l) \ . \tag{29}$$

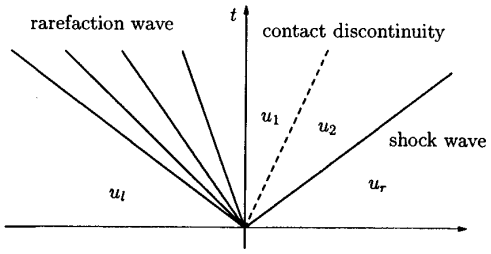


Fig. 5. The (ξ,t) diagram for a solution of the Riemann problem, Eqs. (20) and (21), with $D = 0$.

This relation corresponds to a curve in the (v,p) plane (see Fig. 6, solid line). It is defined for values of the pressure $p_* > p_l$ only. Otherwise, no shock wave will occur, and a rarefaction wave will connect the (v_l,p_l) state with the (v_*,p_*) intermediate state.

Within such a rarefaction wave, the two Riemann invariants associated with the second and third characteristic fields must be constant. From this, it follows that

$$S_1 - S_l = 0 \tag{30}$$

and

$$v_* - v_l = \int_{p_l}^{p_*} \frac{dp}{\rho} c \ , \tag{31}$$

where the sound velocity is evaluated at constant entropy. If the EOS is available in the form of $p = p(\rho,S)$, the velocity of sound is given by

$$c^2 = \left.\frac{\partial p}{\partial \rho}\right|_S \ . \tag{32}$$

Equations (30) and (31) together with the EOS can also be reduced to an equation of the form of relation (29). If the EOS is reformulated as $S = S(\rho,p)$, the density ρ_1 may be eliminated in Eq. (30) and inserted into Eq. (31). The resulting equation of the form of relation (29) defines the isentropic curve within the (v,p) plane (see Fig. 6, dotted line). This curve consists of all points that can be connected to (v_l,p_l) by a rarefaction wave.

Analogous considerations for the right waves establish a relationship

$$v_* = F_r(p_*,\rho_r,v_r,p_r) \ , \tag{33}$$

which defines another curve in the (v,p) plane through the point (v_r,ρ_r).

The point of intersection of these curves is the desired solution (v_*,p_*). To calculate the solution p_*, one must solve the fixed-point problem $F_l = F_r$. All other quantities then can be calculated from the relations (26) through (33). Until now, grid velocity D was assumed to be zero. In this case, the flux is calculated at $\xi/t = 0$: $f[u(0;u_l,u_r)]$ from the Riemann solution $u = u(\xi/t;u_l,u_r)$. After rotation back of the momentum components through multiplication by T^{-1}, it is defined as the numerical flux of the Godunov scheme in the fixed grid. As the terms involving D in a time-dependent grid describe pure convection of the conservative variables, the appropriate flux of the Riemann problem, Eqs. (20) and (21), is given by the flux of the problem for zero grid velocity but now evaluated at $\xi/t = D$. Hence, any solution procedure for the Riemann problem as developed for a Godunov scheme on a fixed grid may be used in this context. For more details about the solution of the Riemann problem, see Refs. 9, 24, 25, and 26.

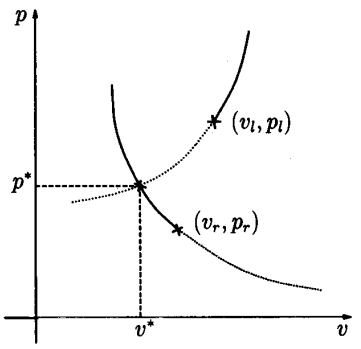


Fig. 6. The (v,p) plane with the right and left states u_r and u_l and the corresponding Hugoniot curves (dashed lines) and isentropes (dotted lines) with intersection point (v^*,p^*).

As the solution procedure outlined earlier requires much computational effort and must be calculated at every grid zone interface, several approximations have been proposed in the literature. Roe's[27] idea was to replace the exact solution by the exact solution of a specifically linearized Riemann problem, which can be solved in an explicit way using the theory of characteristic. For an extension to a general EOS and a review of other approaches, see Ref. 28. Several authors proposed a direct approximation within the fixed-point solution procedure.[29-31]

For the calculations described in this paper, we also introduce an acceleration technique for the fixed-point iteration. In a first step, an approximation of the unknown quantities v_* and p_* is calculated using the isentropic assumption, which means that the shock adiabates in the v-p diagram are replaced by isentropes calculated explicitly. In the case of small jump parameters, it corresponds to the acoustic approximation. The intersection point of these curves furnishes the isentropic values of the contact pressure and the velocity p_* and v_*. For small jumps, this approximation is quite good and is used to determine the flux, while for a strong jump, this solution is a good initial estimate of fixed-point iterations.

While for simple EOSs the overall number of calculations is to be reduced to accelerate the solution procedure, complicated EOSs with, e.g., table look-up, require the number of EOS evaluations to be minimized, as this becomes the most expensive part with respect to the computer time. Optimizing this solution procedure for vector computers is difficult, as there is no a priori knowledge of the number of iterations to be performed locally for the desired accuracy to be achieved. Any local criterion for the breakup of the iteration will destroy the vectorization efficiency of the algorithm. The strategy proposed in Ref. 32 is to perform a fixed number of iterations, which usually guarantees that ~90% of the Riemann problems are solved within the desired accuracy, and the rest are solved

using a local criterion. The parallelization of this procedure is straightforward and very efficient because a fine granularity exists and data transfer is minimal as long as all data for the EOS evaluation are stored in the local memories of each processor.

VI. THE MOVEMENT OF SUBREGION BOUNDARIES

The first step of the algorithm is the movement of the subregion boundaries, as sketched in the block diagram in Fig. 1. The boundary of each subregion is first represented in the array of the corresponding grid coordinates. Furthermore, for any boundary between adjacent subregions, the coordinates of all nodes belonging to one or the other subregion are stored subsequently in a one-dimensional array. Nodes corresponding to the beginning and the end of a segment of boundaries are marked.

The Riemann problem with flow parameters from adjacent zones in contact is solved along every boundary link to determine the velocity of a zone boundary. At material interfaces, this is a Riemann problem where a discontinuity in the flux function occurs because the EOS in the right state is different from that in the left one. But, the different materials are separated by the contact discontinuity, and the solution procedure given in Sec. V may be applied in such a way that relation (29) concerning the left moving wave uses the left EOS and relation (33) uses the right EOS.

The velocities of boundary nodes can be calculated after the Riemann problems at the ribs connecting the nodes have been solved. In Fig. 7, a typical distribution of the nodes at an interface between two subregions is shown. To determine the movement of point P in Fig. 7 first one calculates velocities v_1 and v_2 into the normal directions by solving the Riemann problems between grid zones B, A and C, A, respectively. To minimize the mean-square deviation between the boundary displacement obtained from the Riemann problem solution, when every rib moves at an appropriate velocity, and the boundary displacement determined by the node movement, one should use the formula

$$v^* = \frac{v_1 l_1 + v_2 l_2}{l_1 + l_2} \tag{34}$$

given by Ni et al.[33]

In the present algorithm, v^* is defined as the simple arithmetic average $(v_1 + v_2)/2$. This calculation procedure is used for an interface between different materials as well as the same one material. With a moving grid, tracking a boundary condition becomes easier than in a Eulerian grid. Five boundary conditions are implemented: a boundary with vacuum, a boundary with another material, axes of symmetry, a reflecting wall, and a so-called "special boundary" through which material can flow. The type of boundary can be changed

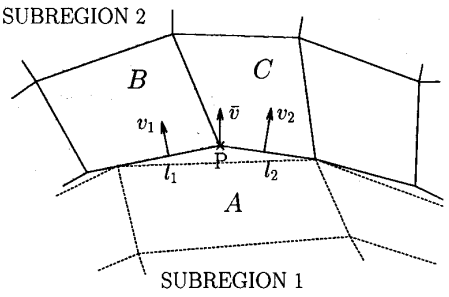


Fig. 7. Boundary segment between different subregions and calculation of the velocity $\bar{v}$ for the boundary node P of subregion 2.

from one time step to another. The implementation of the boundary conditions follows the flux calculation inside the computational region. For the proper behavior at the boundary, appropriate initial values for the Riemann problem are prescribed. The reflecting wall, e.g., is generated by prescribing an artificial initial state with the same values for density and pressure and a change of sign in the velocity. At a boundary with a vacuum, the so-called vacuum Riemann problem is solved, which is a limit case of the usual problem (see Refs. 25 and 34).

VII. GRID GENERATION

When the new positions of the boundaries of each subregion have been determined and the nodes on them arranged, the grid zones must be constructed inside each region. For this purpose, boundary-fitted coordinates are used as reviewed in Ref. 16. This algorithm is based on univalent mapping of a unit square in a so-called logic or parametric (ξ, η) plane onto the domain of the physical (x, y) plane. The problem can be formulated mathematically in the following way: Bijective mapping of the physical domain onto a square is determined by the solution of the system of elliptic equations

$$\xi_{xx} + \xi_{yy} = 0 \ , \quad \eta_{xx} + \eta_{yy} = 0 \tag{35}$$

with the boundary conditions

$$\xi(s) = \xi_1(s) \ , \quad \xi(s) = \xi_2(s) \ ,$$
$$\xi(s) = \xi_3(s) \ , \quad \xi(s) = \xi_4(s)$$

and

$$\eta(s) = \eta_1(s) \ , \quad \eta(s) = \eta_2(s) \ ,$$
$$\eta(s) = \eta_3(s) \ , \quad \eta(s) = \eta_4(s) \ , \tag{36}$$

where s parameterizes the sides of the physical domain.

This mapping can easily be proved to be bijective by the maximum principle for harmonic functions. It

is more convenient to solve the system of equations for the inverse maps $x = x(\xi,\eta)$ and $y = y(\xi,\eta)$:

$$\alpha x_{\xi\xi} + 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = 0$$

and

$$\alpha y_{\xi\xi} + 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = 0 \qquad (37)$$

with

$$\alpha = x_\xi^2 + x_\eta^2 \ , \quad \beta = x_\xi y_\eta + x_\eta y_\xi \ , \quad \gamma = y_\xi^2 + y_\eta^2$$

and the Dirichlet boundary conditions corresponding to conditions (36).

The quasilinear system of elliptic equations (37) is approximated numerically by a finite difference method. The derivatives are replaced by central difference quotients. The discrete equations then must be solved by an iteration procedure. The approach in this paper is based on a time-marching algorithm. Equations (37) are modified in such a way that a time dependency is introduced by replacing the zero right-hand side with the time derivatives of x and y. Hence, the elliptic boundary value problems are shifted to parabolic initial-boundary value problems, with the grid of the previous time step as initial data. The solution of the elliptic problem is obtained as the steady state of the time-dependent problem. The whole procedure may be considered to be a special iteration technique as the time here has no physical meaning and may be interpreted as a relaxation parameter. Within each iteration step, the method of alternating directions is used (see, e.g., Ref. 35). In a first step, terms containing derivatives over η are kept fixed. In a second step, derivatives over ξ are taken from calculations in the first stage. Hence, the problems in each stage are similar to one-dimensional heat transfer problems.

The accuracy of the results as well as the robustness of the numerical methods are known to depend on the grid. Especially, orthogonal grid lines are very well suited for numerical calculations, while strongly distorted grids may introduce errors and instability. When boundary-fitted coordinates are used, the node distribution at the boundary strongly influences the grid generated inside. Hence, after movement of the boundary, the nodes are shifted so that the grid lines tend to become orthogonal (see Ref. 33). Vorobiev et al.[36] noted that physical sharp angles of the boundary are smoothed out from one time step to another when the nodes are connected by straight lines and moved along these. To avoid this effect, one must use other interpolation formulas in the redistribution of the boundary nodes (see Ref. 36 for more details). Usually three iterations are enough to obtain an appropriate grid. Another possibility is to introduce source terms in the elliptic equations[37] with which orthogonality can be achieved in the grid generation algorithm, as proposed in Ref. 12. Our experience is that several iterations are needed additionally to achieve boundary orthogonality by this procedure. Because the grid generation has to be performed in every time step, this results in much more computational effort.

VIII. ADAPTIVE GENERATION OF SUBREGIONS

Other severe problems arise when strong deformations of the computational region occur because of the fluid flow. Redistribution of the nodes to generate orthogonality in the grid may help again. For strong distortions, however, it should be possible to divide the computational regions into new subregions during computation. Examples are shown in Sec. IX. We implemented in our code a method where this can be performed interactively by the users. The main strategy is obvious: Numerical regions with complex form are split into a number of subregions with a simpler form, possibly similar to a rectangle. We also developed an algorithm that automatically proposes or introduces a subdivision by analyzing the distortion of the boundary of the subregion during the computation. This approach is described in this section.

There are two restrictions. In each time step, every subregion can be split once, and the splitting can be performed at a grid line only. According to Sec. VII, the grid is formed by lines ξ = constant and lines η = constant. The set of these lines is called the ξ family and the η family, respectively. For example, we consider the situation when the region is to be split along a grid line of the η family.

To introduce an automatic splitting procedure, we need some indicators for the distortion of subregions. Critical situations occur when a part of the subregion becomes strongly curved. To get an appropriate indicator of this, we first define the function Θ, which gives for any η the length of the corresponding η line:

$$\Theta(\eta) = \int_{\xi_0(\eta)}^{\xi_N(\eta)} \left\{ \left[\frac{\partial x(\xi,\eta)}{\partial \xi} \right]^2 + \left[\frac{\partial y(\xi,\eta)}{\partial \xi} \right]^2 \right\}^{1/2} d\xi \ , \qquad (38)$$

where ξ_0 and ξ_N denote the starting and the end points, respectively, of the η line.

The values of length function (38) and their changes contain the information of the grid distortion along the η lines. We obtained the best practical results by the indicator function

$$F(\eta) = \frac{1}{\Theta(\eta)} \frac{\partial^2 \Theta(\eta)}{\partial \eta^2} \ , \qquad (39)$$

which uses the information about the curvature of the length function normalized by the length. The region is split along an η line when the function F exceeds a given value F^* for some η. The splitting line $\eta = \eta_k$ is chosen according to the criterion

$$F(\eta_k) = \max_\eta F(\eta) > F^* \ . \qquad (40)$$

For the numerical grid with the grid points $(x_{i,j}, y_{i,j})$, the discrete analogs of relations (38) and (39) are then given by

$$\Theta_j = \sum_{i=1}^{N} [(x_{i,j} - x_{i-1,j})^2 + (y_{i,j} - y_{i-1,j})^2]^{1/2} \quad (41)$$

and

$$F_j = \frac{1}{\Theta_j} (\Theta_{j+1} - 2\Theta_j + \Theta_{j-1}) \; . \quad (42)$$

If any F_j is larger than the given value F^*, the region is split at η_k with $F_k = \max_j F_j$. To see in which subregion grid refinement has to be introduced, we use the gradients of density and pressure to identify shock waves and contact discontinuities.

IX. APPLICATIONS

In this section, we demonstrate the usefulness of the developed method to calculate some practical problems arising in nuclear reactor safety and in meteorite impact of space vehicles. Internal explosions, recriticalities of melted cores, oxygen-fuel reactions, hydrogen detonations, or terrorism attacks can generate high loads at the containment walls and accelerate solid fragments to high velocity. Their impact on other components can cause severe damage and lead to emergency conditions. These conditions have to be understood and analyzed thoroughly so that damage to the installation and surroundings is reduced to a manageable level. Numerical simulations of shock wave dynamics in condensed targets under intense impulsive loads are required to predict possible consequences of high-velocity impacts. Results of these simulations depend on both the physical model used in simulations and the accuracy of the numerical solution. In this section, some test results for hypervelocity impact problems as well as the tracking of an explosion and detonation front are shown, demonstrating the ability and the efficiency of the Godunov scheme in moving grids to solve such problems.

IX.A. Impact of a Sphere on a Plane Plate

In the first example, we numerically simulate the impact of a sphere on a plane target. This example is important in the foregoing context as well as in the context of meteorite impact on space vehicles or such phenomena as the impact of the Levy-Shumaker asteroid on Jupiter, which has happened recently. In our simulation, this 1-cm-diam aluminum sphere penetrates a 2-mm-thick plane plate. The velocity of the impacting aluminum sphere is 10 km/s. We assumed a symmetry with respect to the azimuthal angle of the sphere that allows the problem to be calculated in two space dimensions in the (r,z) plane. Figures 8a, 8b, and 8c show numerical results for different discretizations at the same time $t = 710$ ns to indicate the influence of grid refinement. The moved grid is shown on the upper half of Figs. 8, and the isolines for the pressure are shown on the lower part. The different pressure levels indicate the increase of the pressure by 100 kbar, starting at the first level with 100 kbar.

Numerical results using three subregions are shown in Fig. 8a. The aluminum sphere is discretized by a grid with 33 × 17 cells. The plate is divided into two subregions: One is in front of the impacting aluminum sphere, and the other establishes the connecting region with the other part of the plate. The subregions consist of 23 × 4 and 7 × 7 grid zones, respectively. The impact deforms the plate and the sphere. The plate has become cambered at this time, while the sphere has been squeezed together. At the boundary of the sphere, low-density vaporized matter is blown off into the backward direction as a debris cloud. The third subregion has been introduced to prevent strong grid distortions and to get a better resolution.

The impact generates a shock wave moving through the aluminum sphere. It is followed by a rarefaction wave that is formed at the free surface of the aluminum plates. Figure 8b shows the results for the same physical problem at the same time $t = 710$ ns but with a better numerical resolution due to grid refinement. The grid of the sphere is refined into the r direction by a factor of 2; the subregion of the plate in front of the sphere is refined in both directions by the same factor. The neighboring subregion is split into two new subregions with fine grids. The different subregion boundaries are plotted in Figs. 8 as straight lines without marks. The number of grid zones are increased by a factor of ~2, from 700 to 1400. Figure 8c shows the numerical results of a calculation where additionally the grid has been refined. The number of grid zones is here ~2500. Figures 8 indicate that the physical process in the sphere is well captured. The position of the shock wave as well as the shape of the sphere at $t = 710$ ns are quite the same in all three calculations. On the fine grid, of course, the width of the shock wave becomes very small compared with the coarse grids. Obvious differences occur in the dynamic behavior of the debris cloud.

Figures 9a and 9b again show the numerical result at a later time $t = 1900$ ns for different discretizations again. The case of Fig. 9b exactly corresponds to that in Fig. 8b, while in Fig. 9a, a coarser grid than in Fig. 8a has been used. But, now the velocity into the z direction is plotted here. The shock wave has passed the sphere at this time, and the pressure distribution becomes smooth. These results confirm the statement that the overall dynamic behavior is well captured; the velocity distribution inside the sphere is quite similar in both cases. The simulation on the fine grid predicts a smaller thickness of the missile. Because of negative pressures, a rupture of the material was necessary to be introduced into the calculation on the coarse grid. For the coarse grid calculation in Fig. 8a, 290 time steps have to be calculated to reach the time $t = 710$ ns.

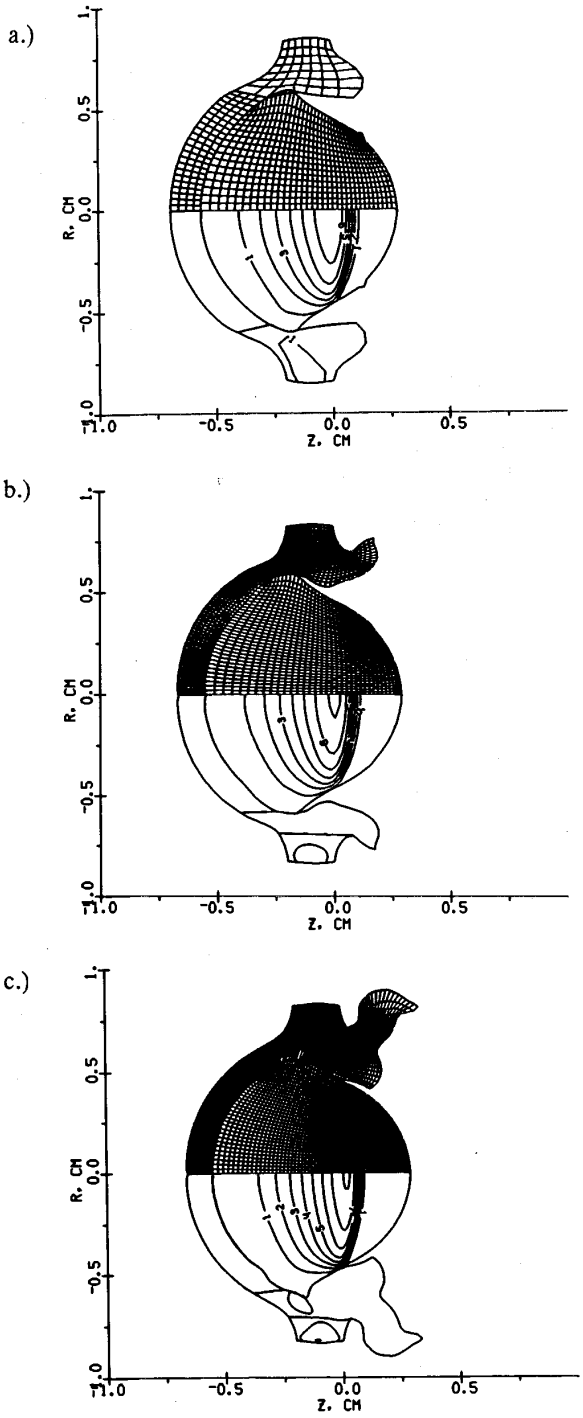


Fig. 8. Moved numerical grid and isolines of the pressure for the impact of an aluminum sphere on a plane plate at time $t = 710$ ns for different discretizations with about (a) 700 grid zones, (b) 1400 grid zones, and (c) 2500 grid zones.

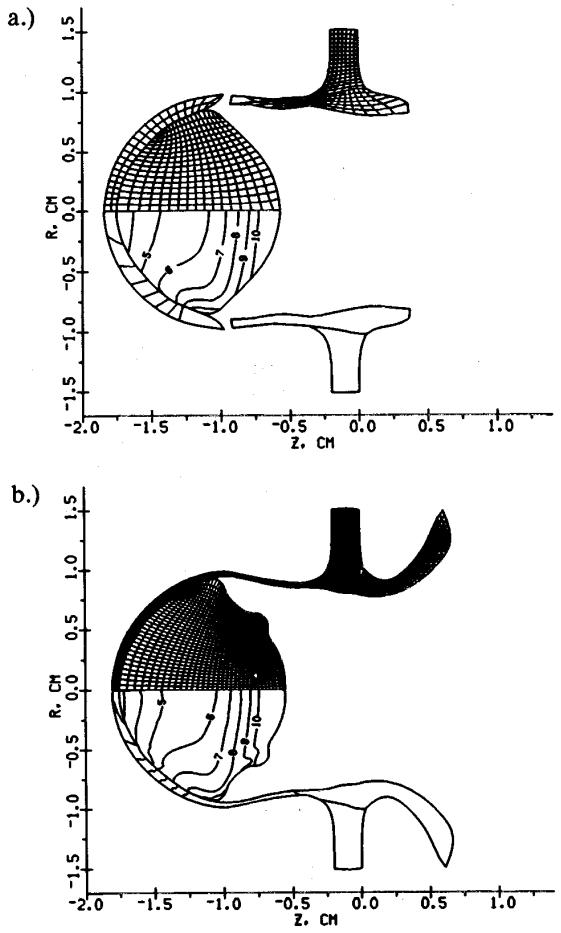


Fig. 9. Moved numerical grid and isolines of the velocity into z direction for the impact of an aluminum sphere on a plane plate at time $t = 1900$ ns for different discretizations with about (a) 550 grid zones and (b) 1400 grid zones.

For our explicit scheme, the time step has to be reduced in accordance with the refinement of the spatial grid. In all subregions, we used in these calculations the same time step. For the grid in Fig. 8b, we needed 887 time steps, while for the finest grid, 1536 time steps become necessary to reach the time $t = 710$ ns. All calculations have been performed on a workstation dec 3000/300 lx. The computer times for one time step were 0.7, 1.3, and 1.8 for the different space discretizations, as given in Figs. 8a, 8b, and 8c, respectively.

IX.B. Detonations in High-Explosive Charges of Finite Diameter

The failure detonation problem is the problem of the minimum explosive charge diameter, required for

a self-sustained detonation to exist. This example is quoted to demonstrate the flexibility and adaptability of the numerical algorithm with respect to geometry and the physics of flow. The flow is described by the Euler equations with a source term in the energy equation due to chemical reaction. To elucidate whether the detonation becomes steady or not, one must calculate the evolution of the detonation process along the distance of several diameters. As the flow downstream of the shock is determined by the chemical reaction kinetics and, in particular, by the amount of energy release, the shock wave must be calculated accurately.

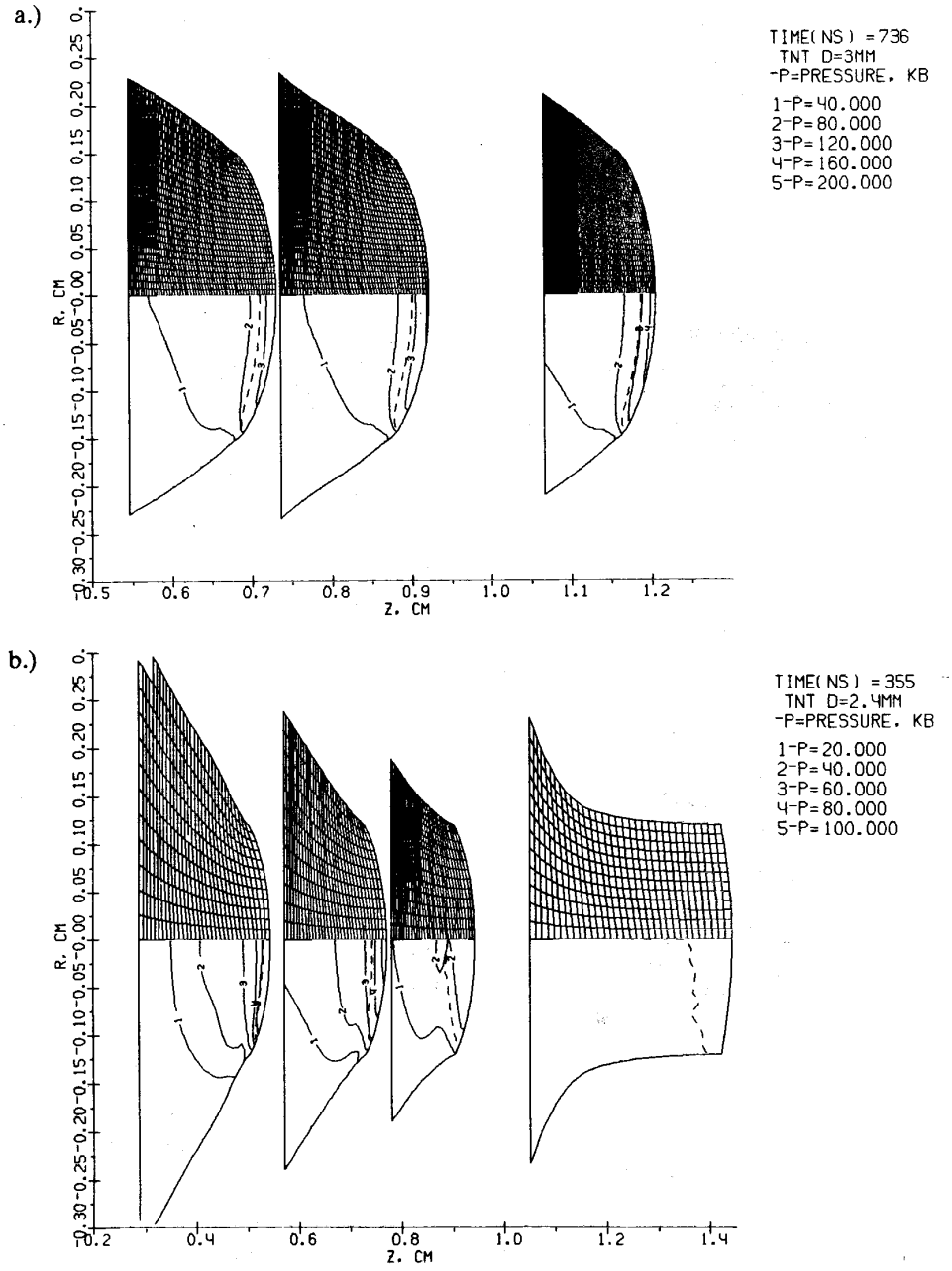


Fig. 10. Computational grids and pressure isolines for the failure detonation problem at different times and with (a) a charge diameter of 3 mm and (b) a charge diameter of 2.4 mm.

The following outline of the computational strategy is chosen in accordance with the problems described earlier. The computational region represents a curvilinear quadrangle. The sides of the quadrangle are the leading shock wave, the contact between the detonation products and the vacuum, the axis of symmetry, and straight line perpendicular to the axis of symmetry. The numerical results are given in Figs. 10a and 10b at different times and for different diameters. The curved shock wave is the right boundary of the computational domain, the axis of symmetry is the line $r = 0.0$, the free contact surface is the aforementioned boundary, and the left boundary is the artificial one introduced due to computational reasons. The velocity of this left side is assumed to be directed along the axis and to be equal to max $(v + c, s)$, where s is the velocity of the shock wave to the right at the axis of symmetry, while $v + c$ is the characteristic velocity. It follows that this boundary moves ahead of matter at least at sonic velocity. Therefore, the values of the arguments for the fluxes of mass, momentum, and energy through this side are taken from the adjacent internal cells. A pressure of 200 kbar, normal density, and zero velocity are given as initial data for these calculations. According to our simulations, the critical diameter was found to be between 2.4 and 3 mm. Numerical results of simulations corresponding to these two cases are shown in Figs. 10a and 10b.

In the 2.4-mm-diam case,the detonation decays, as can be seen in Fig. 10b. Calculations for this diameter were performed with different initial pressures. The moving grid again is shown in the upper part of Figs. 10 at different times, while the isolines of the pressure are given in the lower part. The isolines indicate that the initial high pressure of ~100 kbar decreases in time. This corresponds to the fact that the detonation is going out. No steady detonation was obtained throughout. In Fig. 10a, the 3-mm-diam case is shown. Here the detonation wave becomes steady in the frame of reference moving with detonation velocity. This fact allows us to conclude that the failure diameter of TNT lies in the interval between 2.4 and 3 mm, which agrees very well with experimental findings.[38] In these calculations, we used only one subregion for our calculations and took advantage only of the possibility of moving its boundaries at any velocity. By doing this, we were able to simulate this problem on a small grid and with low computational effort. These studies can also be done on a fixed grid, but with a much larger computational region. The moving grid gives the possibility of cutting down the calculation to the domain of influence in an adaptive way.

IX.C. Propagation of an Explosion Wave

Figure 11 shows the initial configuration for the numerical simulation of an explosion. As in the other examples, the upper part of the plot shows the grid, while the lower part shows the isolines of the pressure. The computational domain is divided into two parts. Subregion 1 is a channel filled with the hot products of an explosion. Ahead of the channel is a cylindrical containment filled with air closed by a spherical dome. In the initial stage, subregion 2 contains the air and is situated at the end of the channel. It is only a thin strip located on the right side in Fig. 11. There is no need to introduce a grid for the whole containment in the early stages of the fluid flow, as the flow in the containment is at rest initially. The explosion wave develops at the end of the channel and propagates into the containment. The boundary conditions for the high-pressure subregion 1 are rigid walls on the upper, lower, and left sides with respect to the diagram in Fig. 11. At the channel front, it is connected to subregion 2 by a contact interface between two subregions. The whole situation is symmetric to the axis. This fact is used in the calculations, which are performed in the upper part of the computational region only. Symmetry boundary conditions are prescribed at the axis. The boundaries of subregion 2 are defined as "characteristic velocity boundaries," except for that connected to subregion 1. The boundary velocity thus is given by the value of fluid velocity plus the sound velocity in the undisturbed air.

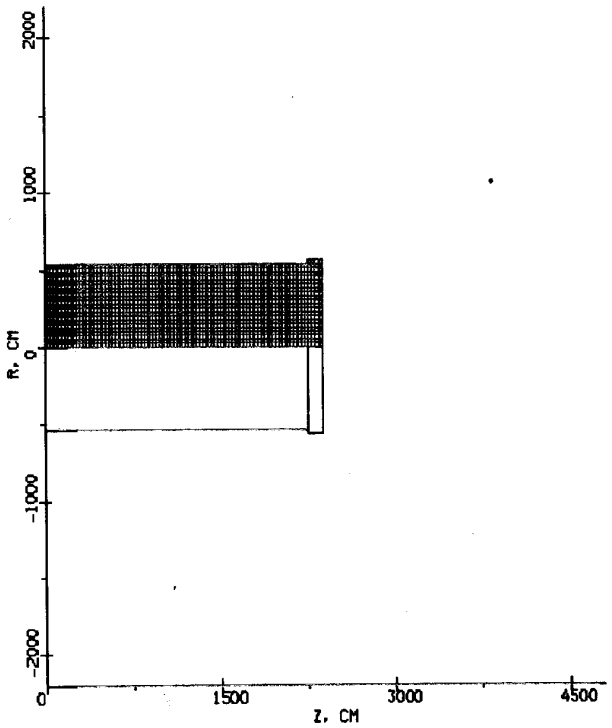
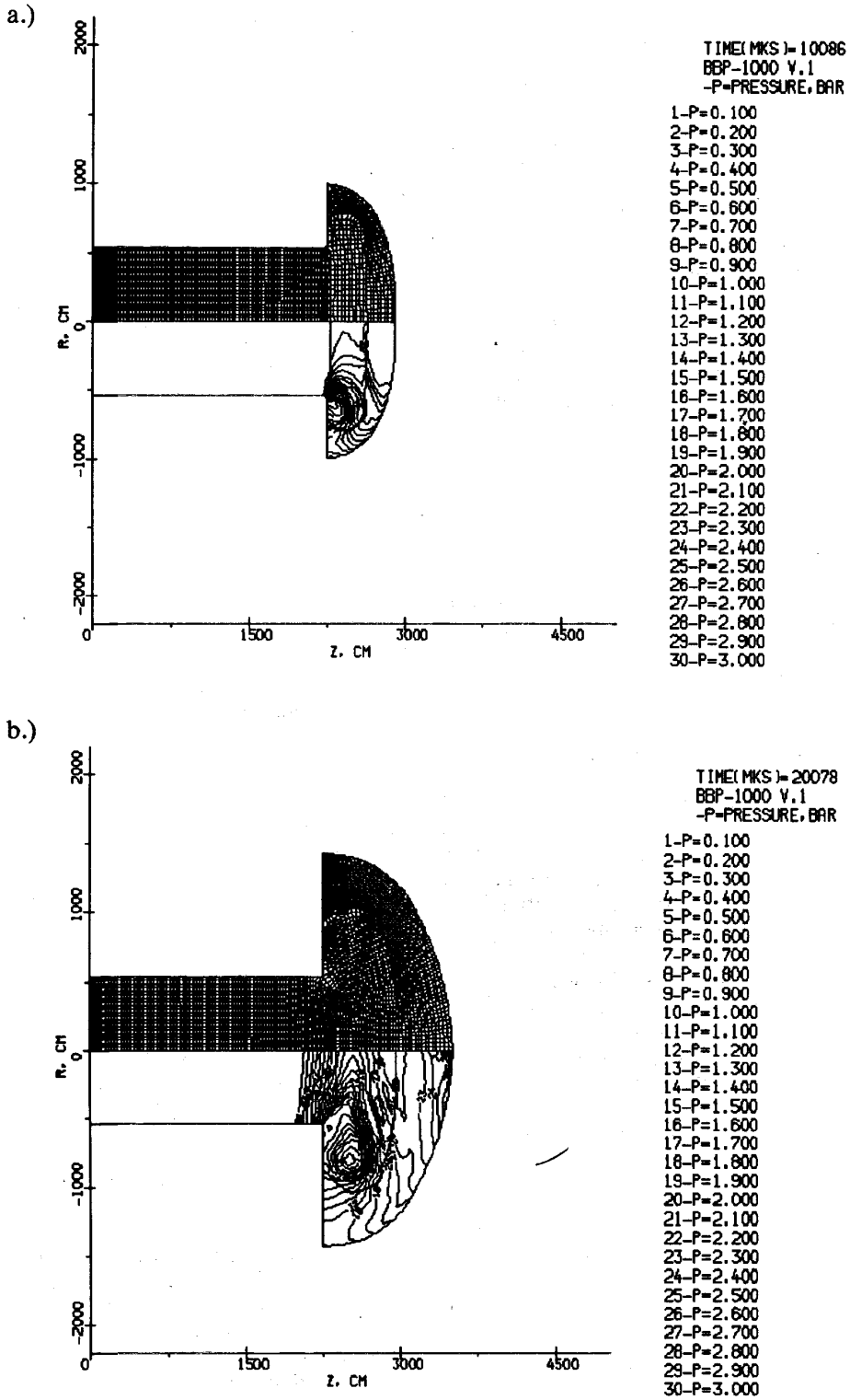
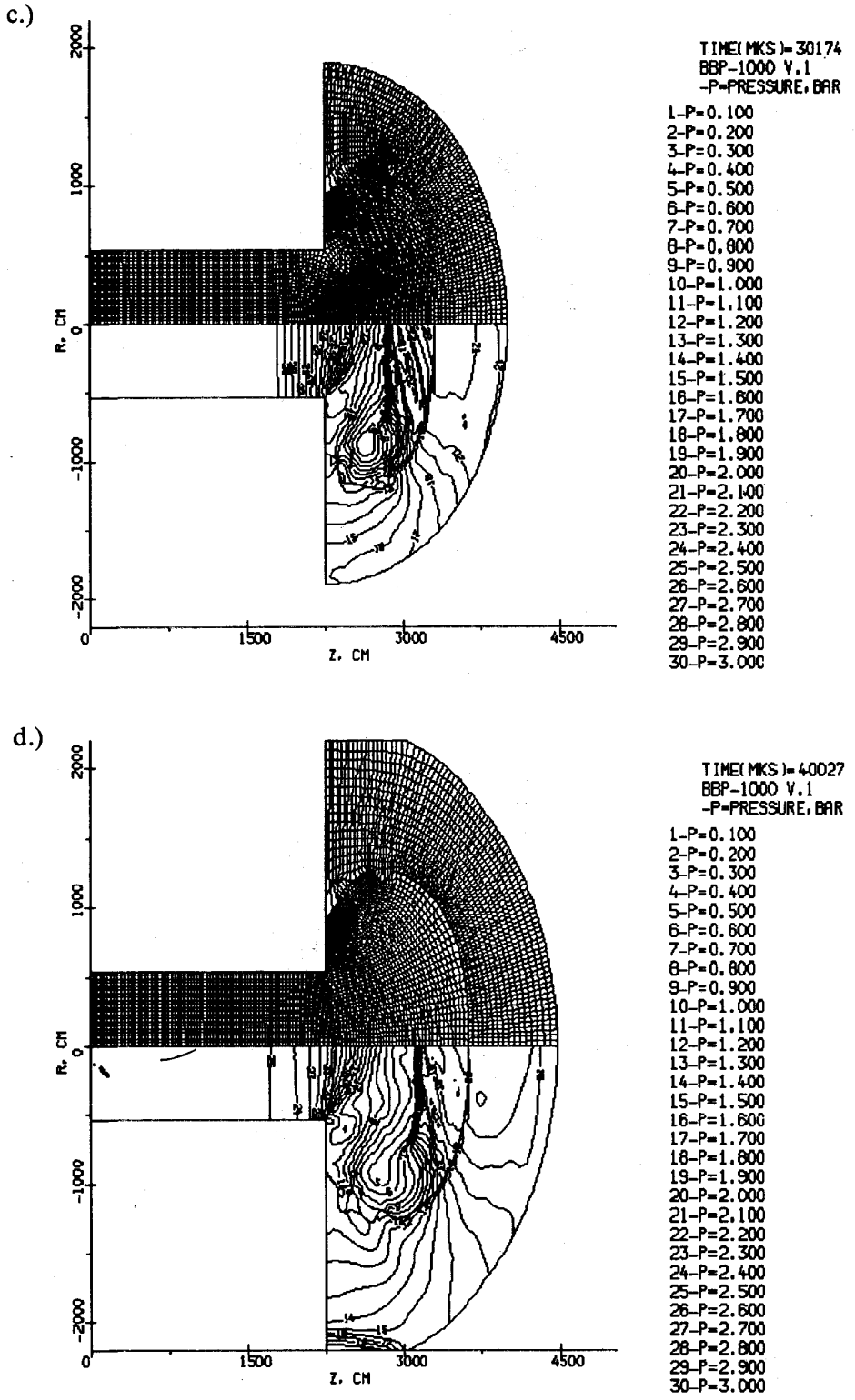


Fig. 11. Numerical simulations of an explosion in a containment: computational region and initial grid.

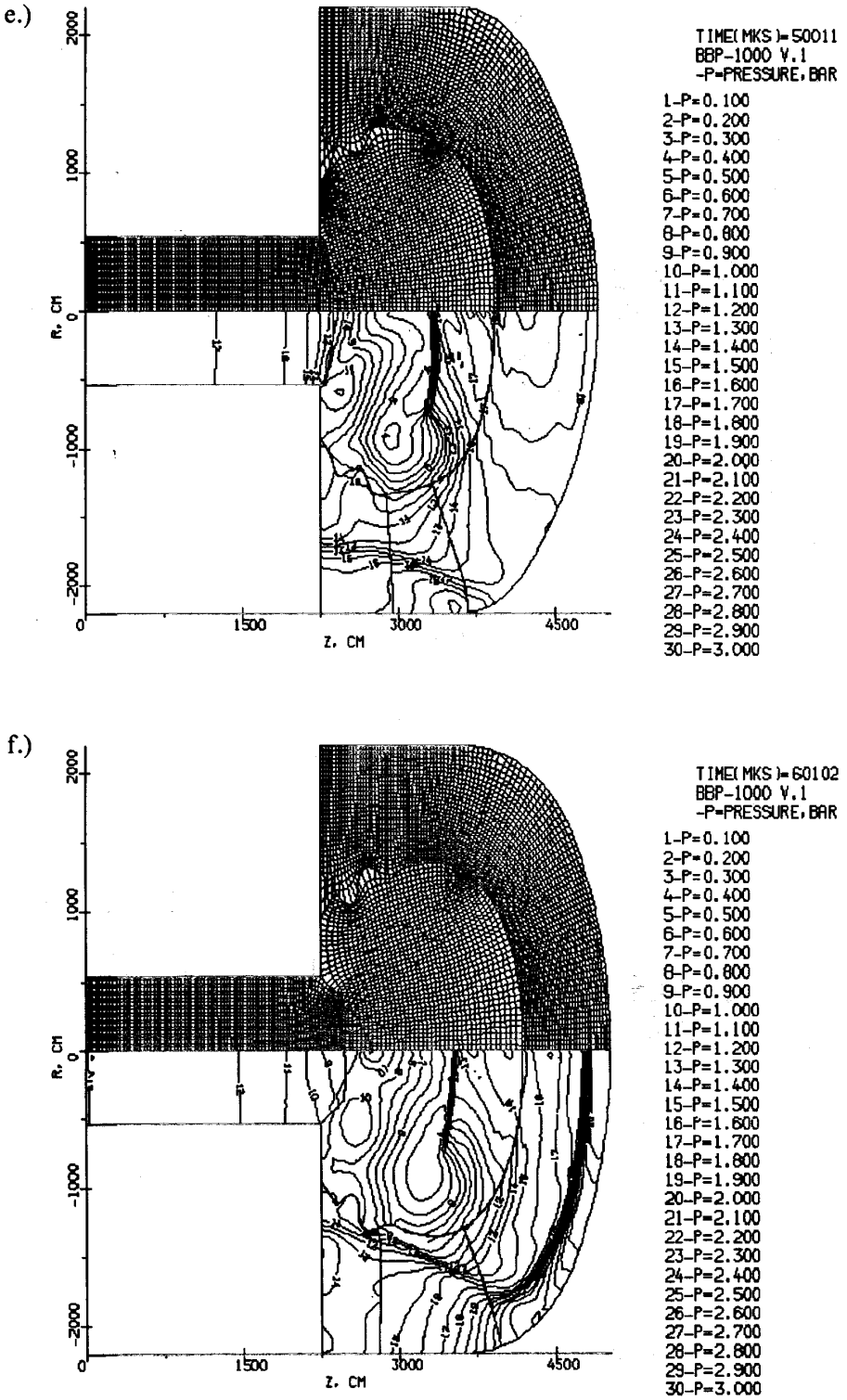
Figures 12a through 12f show the time evolution of the explosion propagating into the containment.



Figs. 12a and 12b. Numerical simulations of an explosion in a containment: computational grids in the upper half and pressure isolines in the lower half at different times of computation at (a) 10.0 ms and (b) 20.0 ms



Figs. 12c and 12d. Numerical simulations of an explosion in a containment: computational grids in the upper half and pressure isolines in the lower half at different times of computation at (c) 30.2 ms and (d) 40.0 ms.



Figs. 12e and 12f. Numerical simulations of an explosion in a containment: computational grids in the upper half and pressure isolines in the lower half at different times of computation at (e) 50.0 ms and (f) 60.1 ms.

The first plot shows the results after 10 ms. Many activities had taken place by this time. Subregion 2 was decomposed into two subregions. One of these subregions contains the moving front. The boundary of this subregion, which we called the characteristic velocity boundary, now moves at the velocity of the shock wave generated by the explosion. The program switched to the shock speed being higher than the local characteristic velocity when the shock reached the boundary. The other new subregion is introduced to decouple the flow around the corner of the channel from the front propagation because there the fluid distortions become pronounced and irregular. The boundaries of this subregion are contact boundaries.

The next plots in Fig. 12 show the situation at a later point in time. The shock wave in front propagates toward the rigid walls of the containment boundary. The plot at the time $t = 40.0$ ms shows the reflection of the propagating front by the upper and lower walls of the containment with respect to Figs. 12. When the front contacts the containment walls, the boundary conditions are changed to rigid wall conditions. At that time, it also becomes clearly visible near the channel corners that the boundary between the subregions becomes highly distorted by the fluid flow. To stabilize the calculations of the front tracking, we again divide the subregion into three subregions, two in the containment corner and one in the dome. The subregion boundaries are visualized in the plot of pressure isolines. Between $t = 50$ ms and $t = 60.1$ ms, where the numerical results are shown in Figs. 12g and 12f, the propagating front reaches the dome boundary and is reflected. The reflected shock wave can clearly be seen in Fig. 12f.

The main advantage in using the moving grid technique in this problem lies in the front being sharply resolved and in the possibility of extracting the loads at the containment walls from the numerical values in a reliable way. The adaptivity of the calculations allows the direct application to similar problems. For these calculations, ~10 000 time steps have to be performed on a Dec 3000/300 workstation with a total computational effort of ~3 h.

X. SUMMARY AND CONCLUSIONS

A Godunov method on moving grids is proposed as a method of tracking fronts and contact surfaces. The computational domain is split into several subregions, and their boundaries are defined to be identical to the fronts to be tracked. The subregion boundaries are then moved according to the local velocity associated with the front. Within the subregions, at each time step, a new grid is generated by the boundary-fitted coordinate technique. This takes extra computational effort, but the iteration procedure converges very fast because the old grid constitutes a good starting vector. The moving grid method allows interfaces to be resolved numerically without numerical smearing.

The high resolution of fronts or interphases is very important in problems where artificial smearing destroys the physical behavior. Typical examples are material interphases where for an artificial intermediate state between the materials, no EOS makes any physical sense or thin reaction zones where the reaction kinetics become inaccurate. A gas-vacuum interphase can also be treated in a straightforward manner. The main advantage of this approach is that the interphases are treated in a Lagrangian fashion as subregion boundaries to produce a sharp resolution, while for the interior flow, a shock-capturing Godunov method is used that is neither Eulerian nor Lagrangian. In this way, the difficulties in the Lagrangian methods associated with strong grid distortions can be avoided. The method proposed is a logical extension of the ideas of Godunov et al.[9] The numerical results of the simulation of the impact of a sphere on a plate, the failure detonation problem, and the propagation of an explosion wave clearly indicate the efficiency and the possibilities of this approach.

In this paper, we did not consider the higher order extensions of the Godunov method. These modifications of the Godunov method can be included in the presented approach to improve the accuracy of the calculations in subregions, especially for smooth parts of the flow. Another aspect of our approach that was not mentioned is the decomposition of the computational domain into different subregions that can be used for parallelization. The interior flow problems in each subregion may be calculated independently, and only the values at the boundary must be exchanged between neighboring regions, i.e., between different processors.

ACKNOWLEDGMENTS

This work could not have been performed without the gracious hospitality offered by G. Kessler. The authors, A. L. Ni and O. Yu. Vorobiev, are grateful to him for the opportunity to stay at the Institute for Neutron Physics and Reactor Technology of Forschungszentrum Karlsruhe. We thank him and our colleagues A. V. Bushman, A. V. Utkin, I. V. Lomonosov, and A. N. Kraiko for numerous stimulating discussions.

REFERENCES

1. H.C. YEE, G. H. KLOPFER, and J.-L. MONTANGE, "High Resolution Shock-Capturing Schemes for Inviscid and Viscous Hypersonic Flow," *J. Comput. Phys.*, **88**, 31 (1990).

2. W. J. RIDER, "Methods for Extending High Resolution Schemes to Non-Linear Systems of Hyperbolic Conservations Laws," *Int. J. Numer. Methods Fluids*, **17**, 861 (1993).

3. Th. SONAR, V. HANNEMANN, and D. HEMPEL, "Dynamic Adaptivity and Residual Control in Unsteady

Compressible Flow Computation," *Math. Comput. Mod.*, **20**, 201 (1994).

4. M. BERGER and P. COLELLA, "Local Adaptive Mesh Refinement for Shock Hydrodynamics," *J. Comput. Phys.*, **82**, 64 (1985).

5. C.-D. MUNZ, "On Godunov-Type Schemes for Lagrangean Gas Dynamics," *SIAM J. Numer. Anal.*, **31**, 17 (1994).

6. W. E. JOHNSON, "History and Applications of Hydrocodes in Hypervelocity Impact," *Int. J. Impact Eng.*, **5**, 423 (1987).

7. A. V. BUSHMAN, G. I. KANEL, A. L. NI, and V. E. FORTOV, "Intense Dynamic Loading of Condensed Matter," Taylor & Francis, Washington, D.C. (1993).

8. C. W. HIRT, A. A. AMSDEN, and J. L. COOK, "An Arbitrary Lagrangian-Eulerian Computing Method for All Speeds," *J. Comput. Phys.*, **14**, 227 (1974).

9. S. K. GODUNOV, V. A. ZABRODIN, M. L. IVANOV, A. N. KRAIKO, and G. P. PROKOPOV, "Numerical Solution of Multidimensional Problems of Gasdynamics," *Nauka*, Moscow (1976).

10. B. EINFELDT, C.-D. MUNZ, P. L. ROE, and B. SJÖGREEN, "On Godunov-Type Methods Near Low Densities," *J. Comput. Phys.*, **92**, 273 (1991).

11. J. M. HYMAN, "Numerical Methods for Tracking Interfaces," *Physica*, **12D**, 396 (1984).

12. I.-L. CHERN, J. GLIMM, O. McBRYAN, B. PLOHR, and S. YANIV, "Front Tracking for Gasdynamics," *J. Comput. Phys.*, **62**, 83 (1986).

13. W. MULDER, S. OSHER, and J. A. SETHIAN, "Computing Interface Motion in Compressible Gas Dynamics," *J. Comput. Phys.*, **100**, 209 (1992).

14. R. LeVEQUE, "Shock-Tracking with the Large Time Step Method," *Computing Methods in Applied Science and Engineering*, p. 511, R. GLOWINSKI and J.-L. LIOUS, Eds., Elsevier Science Publishers, North-Holland, Amsterdam (1986).

15. S. K. GODUNOV, "Finite Difference Method for Numerical Computation of Discontinuous Solutions to the Equations of Fluid Dynamics," *Mat. Sb.*, **47**, 271 (1959).

16. J. F. THOMPSON, Z. WARSI, and C. W. MARTIN, "Boundary Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations—A Review," *J. Comput. Phys.*, **47**, 1 (1982).

17. Y. B. ZEL'DOVICH and Y. P. RAIZER, *Physics of Shock Waves and High-Temperature Hydrodynamic Phenomena I, II*, Academic Press, New York (1967).

18. R. COURANT and K. FRIEDRICHS, *Supersonic Flow and Shock Waves*, Interscience, New York (1956).

19. M. L. WILKINS, "Calculation of Elastic-Plastic Flow," *Methods in Computational Physics*, Vol. 3, p. 211, B. ADLER, S. FERNBACH, and M. ROTENBERG, Eds., Academic Press, New York (1964).

20. J. A. TRANGENSTEIN and R. B. PEMBER, "Numerical Algorithms for Strong Discontinuities in Elastic-Plastic Solids," *J. Comput. Phys.*, **103**, 63 (1992).

21. B. VAN LEER, "Towards the Ultimate Conservative Difference Scheme V. A Second Order Sequel to Godunov's Method," *J. Comput. Phys.*, **32**, 101 (1979).

22. P. COLELLA, "A Direct Eulerian MUSCL Scheme for Gas Dynamics," *SIAM J. Sci. Stat. Comput.*, **6**, 104 (1985).

23. P. D. LAX, "Hyperbolic Systems of Conservation Laws II," *Comm. Pure Appl. Math.*, **10**, 537 (1957).

24. R. MENIKOFF and B. J. PLOHR, "The Riemann Problem for Fluid Flow of Real Material," *Rev. Mod. Phys.*, **61**, 75 (1989).

25. R. G. SMITH, "The Riemann Problem in Gas Dynamic," *Trans. Am. Math. Soc.*, **249**, 1 (1979).

26. T. CHANG and L. HSIAO, "The Riemann Problem and Interaction of Waves in Gas Dynamics," Longman Group, Essex (1989).

27. P. L. ROE, "Approximate Riemann Solvers, Parameter Vectors and Difference Schemes," *J. Comput. Phys.*, **43**, 357 (1981).

28. M. S. LIOU, B. VAN LEER, and J.-S. SHUEN, "Splitting of Inviscid Fluxes for Real Gases," *J. Comput. Phys.*, **87**, 1 (1990).

29. J. K. DUKOWICZ, "A General, Non-Iterative Riemann Solver for Godunov's Method," *J. Comput. Phys.*, **61**, 119 (1985).

30. P. COLELLA And H. GLAZ, "Efficient Solution Algorithm for the Riemann Problem for Real Gases," *J. Comput. Phys.*, **59**, 264 (1985).

31. Ja. HOLODOV and O. Yu. VOROBIEV, "Effective Methods of Integrating 1D Gas Dynamics Problems," *Zh. Vychisl. Mat. Mat. Fiz.* (in Russian) (to be published).

32. C.-D. MUNZ and L. SCHMIDT, "High Resolution Hydrodynamic Schemes and Their Implementation on Vector Computers," *Proc. Joint Int. Conf. Mathematical Methods and Supercomputing in Nuclear Applications*, Karlsruhe, Germany, April 19-23, 1993, p. 135, Kernforschungszentrum Karlsruhe (1993).

33. A. L. NI, T. N. FORTOVA, K. G. SHKADINSKI, A. V. SHUTOV, and A. V. UTKIN, "Numerical Simulation of Interaction of High Velocity Impactors with Condensed Targets," Preprint Institute of Chemical Physics (1989) (in Russian).

34. C.-D. MUNZ, "A Tracking Method for Gas Flow into Vacuum Based on the Vacuum Riemann Problem," *Math. Methods Appl. Sci.*, **17**, 597 (1994).

35. G. I. MARCHUK, *Methods of Numerical Mathematics*, Springer-Verlag, New York/Heidelberg/Berlin (1975).

36. O. Yu. VOROBIEV, A. L. NI, A. V. SHUTOV, and V. E. FORTOV, "Application of Godunov's Method with Movable Grids for Calculations of Dynamics of Condensed Matter Under Intense Impulsive Actions," *Model. Mech.*, **6**, *4*, 3 (1992) (in Russian).

37. A. V. BUSHMAN, V. E. FORTOV, and I. V. LOMONOSOV, in *High-Pressure Equation of State: Theory and Applications*, p. 249, S. ELIEZER, Ed., Elsevier Science Publishers, North-Holland, Amsterdam (1991).

38. A. V. UTKIN, T. N. FORTOVA, G. I. KANEL, and K. G. SHKADINSKI, "Calculation of Nonideal Detonation in TNT on the Base of Macrokinetics Empirical Equation," *Chem. Phys.*, **7**, 1257 (1988).