Study of Numerical Approximations of Gas Flow into Vacuum

C.-D. Munz, R. Schneider, M. Göz, A. Herschel
Institut für Neutronenphysik und Reaktortechnik

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Institut für Neutronenphysik und Reaktortechnik

KfK 5237

Study of Numerical Approximations of Gas Flow into Vacuum

C.-D. Munz, R. Schneider, M. Göz, A. Herschel

Study of Numerical Approximations of Gas Flow into Vacuum

Abstract

In this paper, we study numerical approximations of the expansion of gas flow into vacuum. A new tracking method of the gas vacuum interface is proposed which may be combined with numerical methods for the equations of gas dynamics, the Euler equations. This tracking prevents the difficulties of the numerical approximation introduced by the vacuum as a region where the Euler equations are not valid due to the failure of the continuum assumption. The tracking algorithm is based on the exact or an approximate solution of the vacuum Riemann problem. This is the initial value problem with two constant states, one being the gas and the other the vacuum state, and a limit case of the usual Riemann problem. In this approach, the gas vacuum boundary is sharply resolved within one mesh interval. For a test problem, the numerical results of the tracking algorithm for gas flow into vacuum are presented which indicate that the gas vacuum boundary is captured very well. These results are compared with the quasi vacuum approach where the vacuum is replaced by a gas of low density and pressure.

Untersuchung von numerischen Approximationsmethoden zur Berechnung der Gasausbreitung ins Vakuum

Zusammenfassung

In diesem Bericht untersuchen wir die numerische Approximation eines expandierenden Gases ins Vakuum. Dabei wird eine neue Tracking-Methode zur Verfolgung des Gas-Vakuum Randes vorgestellt. Dieser neue Ansatz wird mit numerischen Methoden zur Lösung der gasdynamischen Gleichungen (Eulergleichungen) zusammengeführt, was zu einer eleganten Berechnung der numerischen Flüsse in der Gas-Vakuum-Zone führt. Durch Verwendung der Tracking-Methode lassen sich insbesondere die auftretenden Schwierigkeiten bei der numerischen Approximation der Eulergleichungen umgehen, wo die Grundlage dieser Gleichungen, nämlich die Kontinuumsannahme, keine Gültigkeit mehr besitzt. Zur Konstruktion des Tracking-Algorithmus wird wahlweise die exakte oder genäherte Lösung des Vakuum-Riemann-Problems (VRP) herangezogen. Das VRP stellt einen Grenzfall des gewöhnlichen Riemann-Problems dar, wo die im allgemeinen vier Zustände zu nur einem Gas- und Vakuumszustand verschmelzen. Dieser auf dem VRP basierender Zugang führt dazu, daß die Gas-Vakuum-Grenzschicht erstaunlich scharf, nämlich innerhalb einer Maschenzelle, aufgelöst wird. Numerische Resultate, die wir anhand einer Vielzahl von Testrechnungen erhielten, führen uns die Stärke und Effizienz der vorgeschlagenen Tracking-Methode vor Augen. Diese Resultate werden verglichen mit denen der Quasivakuum-Methode, bei der das Vakuum durch ein Gas kleiner Dichte und niedrigen Druckes ersetzt wird.

Contents

1.   Introduction

The approximation of gas flow with a gas vacuum boundary may give rise to severe difficulties in the numerical schemes. If the conservation laws of gas dynamics are formulated in an Eulerian frame of reference, which is fixed in space, the computational domain contains the vacuum region and the gas vacuum boundary. As the equations of gas dynamics, usually called Euler equations, are based on the continuum assumption, they are no longer valid in the vacuum and a numerical approximation based on these equations will fail.

A common procedure is to replace the vacuum region by a gas of low density and pressure and it is assumed that the gas flow into this rarefied gas is quite similar to a flow into real vacuum. But if this auxiliary state contains very low densities, still severe difficulties are faced when using the numerical method. This is due to the fact that the dominant energy mode becomes kinetic near the vacuum and vacuum gas boundary. Usually, the gas flow is computed using a conservative numerical approximation of the conservation laws of mass, momentum, and energy. To obtain the internal energy and, thus, the pressure and temperature, the kinetic energy calculated from the conserved variables of mass and momentum are subtracted from the conservative approximation of total energy. In regions of low densities, the resulting internal energy may then be negative which, of course, leads to a failing of the numerical scheme. For a special class of numerical schemes, the Godunov-type schemes, this effect has been observed and studied in [5]. If the negative value of the internal energy is simply replaced by a small positive one, the conservation laws are violated and nonlinear instabilities may be generated.

These difficulties in the approximation of a gas vacuum boundary may be avoided using Lagrangean coordinates which move with the flow. In this case the vacuum boundary is situated at the boundary of the computational domain for all times. Furthermore, Lagrangean numerical methods are much more robust in regions of low densities (see [13]). However, the equations in the Eulerian frame of reference have the widest field of applications, because they are ideal for fluid flow with large deformations and Lagrangean numerical methods are much more difficult to extend to the multidimensional case.

In this paper we propose a tracking method for a gas vacuum interface of gas flow in the Eulerian frame of reference by means of which the difficulties mentioned above are avoided. The propagation of the gas vacuum boundary is followed and the information about its actual location is used to determine the numerical flux between the grid zones at the vacuum interface. The tracking method is based on the exact solution of a free boundary value problem, which is referred to as the vacuum Riemann problem and represents a limit of the usual Riemann problem for gas dynamics. The basic ideas of this method are described in detail in one space dimension. Due to the invariance of the Euler equations according to a Galileian transformation, these considerations are also valid in the multidimensional case in the normal direction of the gas vacuum boundary. In two or three space dimensions however, the tracking of the gas vacuum boundary becomes more complicated and exceeds the scope of the present paper. We shall only make a few remarks about multidimensional extensions.

This paper is organized as follows. In section 2, the equations of compressible fluid flow are listed and the Riemann problem and its solution are reviewed briefly. The limit with one of the initial states of the Riemann problem tending to vacuum is called the vacuum Riemann problem (VRP). Its solution is used in section 3 to propose a tracking method of the gas vacuum boundary which may be easily incorporated into Eulerian numerical methods. The information about the location of the vacuum is used to determine the numerical fluxes in the grid zone adjacent to the vacuum. In section 4, the exact solution of the Riemann problem is replaced by an approximate one and the tracking algorithm and appropriate numerical flux calculations for this approximation are formulated. By means of this approximation the computational effort of the tracking method and the flux calculation is reduced. The paper is concluded by presenting results for a test problem in section 5. Here, we compare the numerical results of the tracking method with those obtained by replacing the vacuum by a gas of low density and pressure and solving numerically the Euler equations in this region, too.

2.   The Euler Equations and the Vacuum Riemann Problem

The Euler equations in one space dimension may be written in the conservation form

(1)
$$u_t + f(u)_x = 0$$

with

(2)
$$u = \begin{pmatrix} \rho \\ m \\ e \end{pmatrix} \quad , \quad f(u) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(e+p) \end{pmatrix} \quad .$$

Here ρ is the density, $m = \rho v$ is the momentum per unit volume, v is the fluid velocity, and e the total energy per unit volume. The pressure p is given as a function of the density and specific internal energy : $p = p(\rho, \varepsilon)$ with $\varepsilon = e/\rho - v^2/2$; we now consider the case of a perfect gas where the corresponding equation of state has the form

(3)
$$p = (\gamma - 1)\rho \varepsilon$$

with the adiabatic exponent γ. Here, the formulation of the Euler equations in the conservation form (1) is preferred, because this formulation is favorable for the approximation of smooth as well as weak solutions, see e.g. [2]. There are three families of waves with the wave speeds being associated with the eigenvalues of the Jacobian $df(u)/du$:

(4)
$$a_1 = v - c \quad , \quad a_2 = v \quad , \quad a_3 = v + c$$

where c denotes the sound velocity given by the relation $c^2 = \gamma p/\rho$.

An important initial-value problem of the Euler equations is the Riemann problem with the piecewise constant initial values: A left constant state u_l for $x < 0$ and right state u_r for $x > 0$. A detailed description of the Riemann problem and its solution in the perfect gas case is reviewed, e.g., by Chang and Hsiao [2] or Halter [8]. The general solution is given by a fixed point problem and a fast iterative method for its solution is proposed by Halter in [8]. The solution consists of four constant states u_l, u_1, u_2, and u_r, which are separated by elementary waves. The right and left wave is either a shock wave or a rarefaction wave. The intermediate states u_1 and u_2 are separated by a contact discontinuity, because the charac-

teristic field associated with a_2 is linear degenerate in the sense of Lax, while the other characteristic fields are strictly nonlinear (see [12]). At a contact discontinuity, the density is discontinuous, while the pressure and velocity remain constant: $p_1 = p_2$, $v_1 = v_2$. In a shock wave, all variables jump according to the Rankine-Hugoniot conditions. A rarefaction wave is a continuous transition from the left to the right values. Figure 1 shows a typical situation, where a shock wave moves into the undisturbed state u_r on the right, followed by a contact discontinuity. A rarefaction wave travels to the left. For more details see, e.g., [2] or [8].

We will now discuss the situation with one state of the initial values of the Riemann problem being a vacuum, e.g., the right state. In vacuum, the conserved variables have the values of $\rho_r = m_r = e_r = 0$ and both the pressure p_r and the sound velocity c_r also vanish: $p_r = 0$, $c_r = 0$. The specification of a vacuum fluid velocity has, of course, no physical meaning. We call this problem for the Euler equations (1) with the initial values

$$
(4) \qquad u(x, 0) = \begin{cases} (\rho_l, m_l, e_l) & \text{for } x < 0 \\ (0, 0, 0) & \text{for } x > 0 \end{cases}
$$

the "vacuum Riemann problem (VRP)". It must be notetd that this problem is no real initial value problem. It is a free boundary value problem, because vacuum is no solution of the Euler equations. But the solution of the vacuum Riemann problem may be obtained as a limit of the solution of the usual Riemann problem. This has been observed and discussed by Halter in [8] and we will study this solution briefly (see also [2]).

The contact discontinuity wave travels with the local fluid velocity and establishes the interface between the right and left material. Hence, the right wave in Figure 1 disappears in the vacuum. Furthermore, the pressure p^* has to be constant across the contact and is identical with the pressure in the vacuum: $p_r = 0$. This means that the contact discontinuity also disappears in the sense that it coincides with the right boundary of the left wave. Hence, the left state is connected with the vacuum by one elementary wave only (see Figure 2). This wave cannot be a shock wave because the Rankine Hugoniot conditions cannot be satisfied in this case. Consequently, it has to be a rarefaction wave. The solution of (1), (4) is given explicitly by

$$
(5) \qquad u(x, t) = \begin{cases} (\rho_l, m_l, e_l) & \text{for} & x/t < v_l - c_l \\ (\rho_0, m_0, e_0) & \text{for} & v_l - c_l < x/t < v_l + 2\,c_l/(\gamma-1) \\ (0, 0, 0) & & \text{otherwise} \end{cases}
$$

with

$$m_0 = \rho_0 v_0 \quad , \quad e_0 = p_0 / (\gamma - 1) + \frac{1}{2} \rho_0 v_0^2 \quad ,$$

$$v_0 = [(\gamma - 1) v_l + 2 (x/t + c_l)] / (\gamma + 1) \quad ,$$

(6)

$$\rho_0 = [(v_0 - x/t)^2 \rho_l^\gamma / (\gamma \, p_l)]^{(1/(\gamma - 1))} \quad ,$$

$$p_0 = (\rho_0 / \rho_l)^\gamma p_l \quad ,$$

(see [8]). For completeness we also write down the formula for the vacuum being situated on the left hand side. It is connected to the right state by the simple rarefaction wave

$$v_0 = [(\gamma - 1) v_r + 2 (x/t + c_r)] / (\gamma + 1)$$

(7)

$$\rho_0 = [(x/t - v_0)^2 \rho_r^\gamma / (\gamma \, p_r)]^{(1/(\gamma - 1))}$$

$$p_0 = (\rho_0 / \rho_r)^\gamma p_r$$

within the region $v_r - 2 c_r / (\gamma - 1) \leq x/t \leq u_r + c_r$. We remark that these solutions are selfsimilar, because u is a function of x/t only.

3. The Tracking Algorithm

We assume that the numerical method for the Euler equations (1) is a scheme in the conservation form

$$(8) \qquad u_i^{n+1} = u_i^n - \frac{\Delta t}{x_{i+1/2} - x_{i-1/2}} \left(g_{i+1/2}^n - g_{i-1/2}^n \right) .$$

Here, Δt denotes the time step $t_{n+1} - t_n$ and u_i^n is the approximation of the mean value of the conservative variables in the grid zone $[x_{i-1/2}, x_{i+1/2}]$, $g_{i+1/2}^n$ is the approximation of the flux $f(u)$ at the right grid zone boundary $x_{i+1/2}$ during $[t_n, t_{n+1}]$, and $g_{i-1/2}^n$ is the approximation of the flux at the left grid zone boundary $x_{i-1/2}$. This assumption is not really necessary, but simplifies the notation. It is straightforward to extend the tracking idea to other schemes than those in conservation form.

As discussed in the previous section the Euler equations are no longer valid within the vacuum and, hence, neither is (8) at consistent approximation. Furthermore, near the gas vacuum interface very small values of density and pressure may occur inside the gas region. Any approximation or rounding error, which introduces a small undershoot in the numerical approximation, may lead to negative values of the density and/or pressure and causes a breakdown of the calculation. Especially, because the internal energy is nearly zero and the dominant energy mode is kinetic, small errors in the subtraction of kinetic energy from the total energy may result in negative internal energy and a failing of the numerical approximation. The very small values of the density and pressure are often accentuated by a numerical smearing effect. If a small amount of gas is in the i-th grid zone and the (i + 1)-th zone contains vacuum, then the pressure difference leads to the flow of a small amount of gas into the (i + 1)-th grid zone. In each time step, this numerical smearing of the gas vacuum boundary will propagate at a rate of one grid zone per time step, introducing smaller and smaller values of density and pressure. A similar numerical smearing may occur in other parts of a flow, but these small errors do not result in a failure of the approximation due to the nonvalidity of the equations.

The idea in this paper is to track the gas vacuum boundary in a first step. This will give an estimation of the real movement of the gas vacuum boundary. According

to this, the numerical flux is determined in a second step in such a way that the gas vacuum interface remains sharp.

We assume that the location of the vacuum material boundary at the time level t_n is known and situated in the i-th grid zone. Its location is named x_V^n. On the left hand side of x_V^n we have a gas, while on the right a vacuum exists (see Figure 3a, b). To determine the location of the vacuum gas boundary at the next time level t_{n+1}, we solve the vacuum Riemann problem (1), (5) where the left state u_l is given by the formula

$$(9) \qquad u_l = \alpha \, \bar{u}_i^n + (1 - \alpha) u_{i-1}^n$$

with

$$(10) \qquad \alpha = \frac{x_V^n - x_{i-1/2}}{\Delta x} \quad \text{and} \quad \bar{u}_i^n = \frac{x_{i+1/2} - x_{i-1/2}}{x_V^n - x_{i-1/2}} u_i^n$$

where Δx is the typical length of a grid zone; a good choice may be $\Delta x = x_{i+1/2} - x_{i-1/2}$. According to (5) the location of the vacuum boundary at t_{n+1} is given by

$$(11) \qquad x_V^{n+1} = x_V^n + (t_{n+1} - t_n) v_V^n \quad \text{with} \quad v_V^n = v_l + \frac{2}{\gamma - 1} c_l \quad .$$

The motivation for the mean value (9) is as follows. The time step $t_{n+1} - t_n$ for the explicit scheme in the conservation form (8) is restricted by the fact that the waves do not move through more than one grid zone within one time step. If we use u_i^n as the left value, then waves generated at $x_{i-1/2}$ will interact with the vacuum Riemann problem and change the solution (6). Hence, to guarantee that these waves do not reach the material vacuum boundary, we introduce a left state in such a way that it is constant in the interval of length Δx. In other words, the stencil for the vacuum Riemann problem must be enlarged beyond the adjacent cell. Replacing u_i^n in (9) by $\bar{u}_i^n$ we use the knowledge of the location of the vacuum material boundary at the time t_n. The value u_i^n is an approximation of the integral value of the solution in the whole i-th grid interval. As $[x_V^n, x_{i+1/2}]$ contains a vacuum, we redistribute the integral value over the interval $[x_{i-1/2}, x_V^n]$.

We remark that it may be favorable to calculate the average (9) not in the conservative variables but in the primitive variables ρ, v, and p. This ensures that p_l is a convex average of $\bar{p}_i^n$ and p_{i-1}^n and remains positive. Otherwise, the pressure p_l

has to be recalculated from the average of the conservative variables in a region of low density and internal energy, which introduce the difficulties mentioned above.

By the algorithm (9) - (11) we obtain an approximation of the gas vacuum boundary at time t_{n+1} in a first step. In the second step, we use this information to calculate the numerical flux $g_{i+1/2}$ near the gas vacuum boundary. There are two cases to distinguish:

Case 1: If $x_V^{n+1} \leq x_{i+1/2}$, then the grid zone interface at $x_{i+1/2}$ lies in the vacuum during the whole time step (see Figure 3a) and, hence, the numerical flux must be zero:

$$(12) \qquad g_{i+1/2}^n = 0 \ .$$

The information on the real movement of the gas vacuum boundary via (9) - (11) and the conclusion (12) keeps the approximation of gas vacuum interface sharp.

Case 2: If $x_V^{n+1} > x_{i+1/2}$, then the gas vacuum boundary moves across the grid zone interface. In this case, the flux vector at $x_{i+1/2}$ becomes nonzero. A good numerical flux calculation should be given by using the flux of the VRP of the tracking step evaluated at $x = x_{i+1/2}$ and averaged over the time step

$$(13) \qquad g_{i+1/2}^n = \frac{1}{t_{n+1} - t_n} \int_{t_n}^{t_{n+1}} f(u\,(x_{i+1/2} - x_V^n, t))\,dt$$

where u is the solution (5) of the VRP. According to (5) the integral (13) can be reformulated in the following way. If t'_n denotes the intersection of the gas vacuum boundary with the line $x = x_{i+1/2}$:

$$(14) \qquad t'_n = t_n + \frac{1}{v_V}\,(x_{i+1/2} - x_V^n)$$

and t'_{n+1} is the intersection of the left rarefaction fan boundary curve with this line in the case of $u_l - c_l > 0$:

$$(15) \qquad t'_{n+1} = \min\!\left(t_{n+1}, t_n + \frac{1}{u_l - c_l}\,(x_{i+1/2} - x_V^n)\right)$$

and $t'_{n+1} = t_{n+1}$ in the case of $u_l - c_l < 0$ (see Figure 3.b), then the formula of the numerical flux becomes

$$(16) \qquad g^n_{i+1/2} = \frac{1}{t_{n+1} - t_n} \left[\int_{t'_n}^{t'_{n+1}} f(u_0(x_{i+1/2} - x^n_v, t)) \, dt + (t_{n+1} - t'_{n+1}) f(u_l) \right]$$

where u_0 is the vector of the physical variables (p_0, m_0, e_0) in the rarefaction fan (6). The integrals in (16) can be determined analytically. The evaluation of the formulae for the integrals needs much computational effort and should be avoided, even in the one-dimensional case. If the numerical fluxes of the method used within the gas flow is, e.g., second-order accurate in time, then it should be sufficient to approximate the integral at the same order of accuracy.

The idea of using the exact solution of the Riemann problem to determine the numerical flux was proposed for the first time by Godunov [6]. In Godunov's method the numerical flux $g_{i+1/2}$ is defined to be the physical flux $f(u(0, t))$ where u is the exact solution of the Riemann problem with the left and right state $u_l = u_i^n$ and $u_r = u^n_{i+1}$, respectively. In our tracking approach for the gas vacuum boundary the center of the VRP is in general not located at a grid zone interphase. This avoids the smearing of that boundary and keeps it sharp, but results in the complicated flux evaluation (16).

A simpler way is to use the tracking step only to obtain information about the progress of the gas vacuum boundary. In Case 1 the flux is given by (12) and in Case 2 the flux is calculated by using the solution of the VRP centered at the grid zone interface $x_{i+1/2}$ with the data:

$$(17) \qquad u(x, 0) = \begin{cases} u_i^n & \text{for } x < 0 \\ \\ \text{vacuum} & \text{for } x > 0 \end{cases}$$

where the approximate integral value in the i-th grid zone is the left state. This flux calculation only extends the idea of Godunov and his construction of a numerical method to the gas vacuum boundary. The general solution of the Riemann problem for the Euler equations is replaced by the solution of the vacuum Riemann problem at the grid zone interface $x_{i+1/2}$. In this case, the physical flux becomes

$$(18) \qquad g^n_{i+1/2} = \frac{1}{t_{n+1} - t_n} \int\limits_{t_n}^{t_{n+1}} f(u(0,t)) \, dt$$

where u is the VRP solution to the initial data (17). Since u is constant along the lines of x/t = constant, the numerical flux becomes very simple:

$$(19) \qquad g^n_{i+1/2} = \begin{cases} f(u_i^n) & \text{if } u_i^n - c_i^n > 0 \\[2mm] f(u_0(0,t)) & \text{otherwise} \end{cases}$$

with u_0 given by (6).

We remark that when $x_v^{n+1} - x_{i+1/2}$ is positive, but very small, then the flux at $x_{i+1/2}$ into the (i + 1)-th grid zone may be very small which leads to small approximate integral values of the physical quantities. This may again give rise to the difficulties outlined above. A more robust algorithm is obtained by the switching of Case 1 to Case 2 not at zero but at a small positive value $\varepsilon \, \Delta x$. For ε the value 0.01 is used in the calculations. The results obtained are shown in section 5.

4. Approximate Solutions of the Vacuum Riemann Problem

To obtain an appropriate numerical flux in the case of the gas vacuum boundary moving from one grid zone to the other, the exact solution of the vacuum Riemann problem is averaged in (13) or (18) over the time interval. The local structure of the solution is lost in this averaging process. The idea is now to replace the exact solution by a much simpler averaged one to obtain a very simple flux calculation. The following approximation extends in some sense the idea of Harten, Lax and van Leer, who constructed in [9] a simple approximation of the general Riemann problem solution by replacing the intermediate states by one average state. They used this approximation to define the numerical flux of a so called Godunov-type scheme.

In the following section, we approximate the exact solution within the rarefaction fan of the VRP by a constant value. This approximate VRP solution is constructed such that the consistency with the integral conservation laws and with the entropy condition is established. It has the form

$$(20) \qquad w(x, t) = \begin{cases} u_l & \text{for} & x/t < v_l - c_l \quad, \\ u_{lr} & \text{for} & v_l - c_l < x/t < v_l + 2\,c_l/\gamma - 1 \quad, \\ \text{vacuum} & & \text{otherwise} \quad. \end{cases}$$

The constant state u_{lr} is calculated from the consistency condition with the integral conservation:

$$(21) \qquad \int_{-\Delta x/2}^{v_V \Delta t} w(x, \Delta t)\, dx = \frac{\Delta x}{2}\, u_l + \Delta t\, f(u_l)$$

which is obtained by integrating the Euler equations (1) over the region bound by $t = 0, t = \Delta t, x = -\Delta x/2$ and $x = v_V\, t$ under the assumption

$$(22) \qquad \Delta t \max \{|v_l - c_l|, |v_V|\} \le \Delta x/2$$

(see Figure 4). Inserting (20) into the left hand side of (21) and eliminating u_{lr} from this equation, we obtain

(23)
$$u_{lr} = \frac{f(u_l)}{v_V - v_l + c_l} - \frac{(v_l - c_l)u_l}{v_V - v_l + c_l} \quad .$$

Using this approximate Riemann solution (23), we can simplify the flux calculations at the gas vacuum boundary. The tracking step remains unchanged, because the propagation velocity of the gas into the vacuum is the same in the exact and the approximate Riemann solutions.

We first consider the modification of the flux calculation (16) where the vacuum Riemann solution of the tracking step was used. We assume that our flux calculation is consistent with the integral conservation principle. If the jump of the initial data is located at $x = 0$, then we look for the flux at $x_{i+1/2} - x_V^n$ given by the approximate Riemann solver (20). The integral conservation is satisfied, if the relation

(24)
$$\int_{-\Delta x/2}^{x_{i+1/2} - x_V^n} w(x, \Delta t)\,dx = \frac{\Delta x}{2} u_l - \Delta t\, g_{i+1/2} + \Delta t\, f(u_l)$$

holds. The evaluation of the integral on the left hand side by inserting (20) and some manipulation leads to the numerical flux:

(25)
$$g_{i+1/2}^n = f(u_l) - \frac{1}{\Delta t}\min\left\{(v_l - c_l)\Delta t, x_{i+1/2} - x_V^n\right\} u_l - \frac{1}{\Delta t}\max\left\{0, x_{i+1/2} - x_V^n - (v_l - c_l)\Delta t\right\} u_{lr} \quad .$$

If we use the information on the transition of the gas vacuum boundary to the neighboring grid zone only, the numerical flux is then determined by the approximate Riemann problem with the integral approximate value u_i^n as left state and the center $x = 0$. The analogy to (24), the ansatz is

(26)
$$\int_{-\Delta x/2}^{0} w(x, \Delta t)\,dx = \frac{\Delta x}{2} u_i^n - \Delta t\, g_{i+1/2}^n + \Delta t\, f(u_i^n) \quad .$$

This relation leads to the simple flux calculation

(27)
$$g_{i+1/2}^n = \frac{v_V}{v_V - \min(0, v_l - c_l)}\left(f(u_i^n) - \min(0, v_l - c_l)u_i^n\right) \quad .$$

The exact solution of the vacuum Riemann problem is continuous and satisfies another conservation law: The conservation of the entropy. If S denotes the entropy, then the differential form of this conservation law has the form

(28) $$(\rho S)_t + (\rho v S)_x = 0$$

(see e.g.[1]). In the case of a perfect gas, the entropy is given by

(29) $$S = \left(\frac{\partial \varepsilon}{\partial T}\right)_V \left(C_0 + \ln \frac{p}{\rho^\gamma}\right)$$

where $(\partial \varepsilon / \partial T)_V$ is the derivative of the internal energy with respect to temperature T at a constant volume and C_0 an arbitrary constant. The entropy is a Riemann invariant within the rarefaction and, hence, S_0 is constant: $S_0 = S_l$. We remark that the entropy equation is no longer valid, if discontinuous solutions occur (see, e.g. [1], [11]). In this case, the second law of thermodynamics states that any physically relevant solution satisfies the entropy inequality

(30) $$(\rho S)_t + (\rho v S)_x \geq 0$$

in an integral sense. This indicates that the total entropy does not decrease.

The numerical approximation in general does not satisfy such an additional conservation law. But, if the entropy decreases within a numerical approximation, it is possible that the numerical solution becomes unphysical. Then, instead of the continuous expansion into the vacuum, an unphysical rarefaction shock wave may be approximated. To prevent this the numerical approximation should be consistent with the entropy inequality (30) in any case. We will show in the following section that our approximation of the VRP is physically reasonable and does not lead to a decrease in entropy.

At first, we consider the exact vacuum Riemann problem solution u of (5). This solution satisfies the local entropy equation (28) at all points where u can be differentiated (U (u), F (u)) with

(31) $$U(u) = -\rho S \quad \text{and} \quad F(u) = -\rho v S$$

is referred to as the entropy pair and it is noted that U (u) is a convex function (positive definite Hesse matrix, see [11]). Integrating (28) over the region of the

vacuum Riemann problem bound by $t = 0, t = \Delta t, x = -\Delta x/2$ and $x = v_V t$ under the assumption (22), the integral formulation of the entropy conservation is obtained in the form

$$(32) \qquad \int_{-\Delta x/2}^{v_V \Delta t} U(u(x, \Delta t)) \, dx = \frac{\Delta x}{2} U(u_l) + \Delta t \, F(u_l)$$

(cf (21) and see Figure 4).

Lemma 4.1: The approximate vacuum Riemann problem solution $w = w(x, t)$ given by (20), (23) satisfies the integral entropy inequality

$$(33) \qquad \int_{-\Delta x/2}^{v_V \Delta t} U(w(x, \Delta t)) \, dx \leq \frac{\Delta x}{2} U(u_l) + \Delta t \, F(u_l) \quad .$$

Proof: According to (20), we obtain the relation

$$(34) \qquad \int_{-\Delta x/2}^{v_V \Delta t} U(w(x, \Delta t)) \, dx = \left(\frac{\Delta x}{2} + (v_l - c_l) \Delta t \right) U(u_l) + \left(v_V \Delta t - (v_l - c_l) \Delta t \right) U(u_{lr}) \quad .$$

As the boundaries of the constant state u_{lr} are identical with the boundaries of the rarefaction fan of the exact solution u, we conclude that

$$(35) \qquad u_{lr} = \frac{1}{(v_V - v_l + c_l) \Delta t} \int_{(v_l - c_l) \Delta t}^{v_V \Delta t} u(x, \Delta t) \, dx$$

due to the consistency with the integral conservation (21). As $U(u)$ is convex, Jensen's inequality may be applied (see e.g. [9], [14]) and furnishes

$$(36) \qquad U(u_{lr}) \leq \frac{1}{(v_V - v_l + c_l) \Delta t} \int_{(v_l - c_l) \Delta t}^{v_V \Delta t} U(u(x, \Delta t)) \, dx \quad .$$

If (36) is inserted into (34), the two terms on the right hand side may be combined and the inequality

(37)
$$\int_{-\Delta x/2}^{v_V \Delta t} U(w(x, \Delta t)) \, dx \le \int_{-\Delta x/2}^{v_V \Delta t} U(u(x, \Delta t)) \, dx$$

is obtained. Replacing the integral on right hand side by the right expression in (32) leads to the inequality (33) and finishes the proof.

This lemma shows that the approximate vacuum Riemann problem is consistent with the entropy inequality. Hence the numerical methods based on the flux calculations (25) or (27) at the gas vacuum boundary are not supposed to produce unphysical rarefaction shocks.

5.   Test Problems and Numerical Results

In this section we consider two test problems for a gas vacuum expansion wave. In the first case the real vacuum region is replaced by a gas of low density and pressure, assuming that the flow into the quasi vacuum is quite similar to the flow into the real vacuum. With the general RP having the form

$$(38) \qquad (\rho, v, p) = \begin{cases} \left(1.0, \ 0.0, \ \dfrac{1}{\gamma - 1}\right) & \text{for } x < 0 \\[2ex] (\delta\rho_r, 0.0, \delta p_r) & \text{for } x > 0 \end{cases} \qquad ,$$

we study systematically the numerical solution for different small $\delta\rho_r$ and δp_r. The exact reference solution for the sequence of calculations is given by a fixed point problem which could be solved by fast iterative method [8].
Considering in the second case a computational domain containing real vacuum, we have to apply the tracking method proposed in section 3, based on the solution of the special VRP which reads as

$$(39) \qquad (\rho, v, p) = \begin{cases} \left(1.0, \ 0.0, \ \dfrac{1}{\gamma - 1}\right) & \text{for } x < 0 \\[2ex] \text{vacuum} & \text{for } x > 0 \end{cases} \qquad .$$

Numerical results for different flux calculation approximations as described in section 3 and 4 will be presented. To compare these results with the exact solution we use the equations (5) and (6).
The numerical calculations are performed on a grid with 100 grid zones covering the computational region form - 0.3 to 0.7. The numerical results are shown at the time t = 0.1; the time step size Δt is controlled in an adaptive manner according to the CFL condition

$$\Delta t = \sigma \frac{\Delta x}{|a_{max}|} \quad ,$$

where σ is hold fix to 0.8 and a_{max} is the maximum velocity of the propagating waves. The scheme in conservation form for the approximation of the flow within the gas region is the HLL method. The a priori estimates of the signal velocities as given by Einfeldt [4] are used. We call this scheme the HLLE scheme. In [5] it is shown that this scheme is quite robust near the vacuum. Hence, it should be a good candidate method for the testing of the quasi vacuum approach as well as of the tracking algorithm. For more details about the HLL scheme see [4].

5.1 The Quasi Vacuum Approach

As a first test problem, we consider the numerical solution of the general RP (38). There is an open question how to define the values of the low pressure and density to get a good approximation of the vacuum and the movement of the gas vacuum boundary. The values should be as small as possible, of course. But, what means possible? This depends very strongly on the numerical method used. The approximation of low density states is a very serious problem of the numerical approximation. We outlined this fact in our introduction and will describe this difficulty more precisely in the following. If we use a numerical scheme approximating the conservation equations (1), then we obtain within one cycle approximate values at the new time levels for the conservative variables

$$\rho, \rho v, e \quad .$$

From these values, the other variables are calculated, e.g. the internal energy according to

$$\varepsilon = \frac{e}{\rho} - \frac{1}{2} \frac{(\rho v)^2}{\rho}$$

and from this pressure and temperature. Near vacuum density ρ, momentum ρv as well as ε tend to zero. The main energy mode is kinetic. This kinetic energy is calculated as a quotient $(\rho v)^2/\rho$ of the small values of momentum and density. Any truncation of rounding error may strongly be accentuated. This quotient next is subtracted from e/ρ which value is very close to this kinetic term. This means that a small error may easily lead to a negative value of the internal energy and the numerical scheme breaks down.

The "correction" by fixing the internal energy to be positive may introduce instabilities and will destroy the integral conservation properties.

In [5] this effect has been studied and it was shown that even Godunov-type schemes which incorporates the wave propagation into the numerical approximation may fail and no appropriate linearization will exist. A scheme which is quite robust in the vicinity of vacuum has been named positively conservative in [5]. The HLLE scheme as mentioned above is a positively conservative scheme and guarantees that the numerical approximation does not produce unphysical values

of the physical variables. Hence, this method is appropriate to approximate the solution of the problem (38) for very small values δp_r and $\delta \rho_r$.

We performed calculations varying the values of $\delta \rho_r$ and δp_r between 10^{-2} up to 10^{-6}. We assumed that, beside the absolute values, the quotient $\delta p_r / \delta \rho_r$ will also influence the quality of quasi vacuum approximation, because the sound velocity and, hence, acoustic wave propagation depend on this ratio. The first series of calculations have been performed with data where the value of the pressure of the right state is set equal to the value of right state density. For different small but finite values of the density $\delta \rho_r$ ($= 10^{-2} \ldots 10^{-6}$), the numerical solution of the RP (38) (small open circles), which is in good agreement with the exact one obtained from a fixed point problem (not shown here), is compared with the exact solution of the VRP (solid lines) in figure 5. Replacing the vacuum by a relatively dense quasi vacuum ($\delta \rho_r = 10^{-2}$), the resulting numerical solution of the density, pressure, total energy and velocity are a rather crude approximation of the exact VRP as it is seen in figure 5a. Decreasing successively the density of the quasi vacuum, the agreement of the approximation of the conserved quantities (namely density and total energy) with the exact VRP solution is quite good. However, the pressure slightly exceeds the line of the exact values of the VRP. Obviously, as it is depicted in figure 5, the velocity for the different dense quasi vacuum clearly underestimate the exact result of the VRP. It is remarkable that the internal energy and the density remains positive throughout the computation for all considered right state densities $\delta \rho_r$. This reflects the fact that the HLLE-scheme applied is positively conservative [5].

A second series of calculations are performed where the pressure of the right state of the general RP (38) is related to the density via the relation

$$\delta p_r = 0.1 \, \delta \rho_r \quad .$$

The numerical result of the density, pressure, total energy and velocity is shown in figure 6 for two different right state densities $\delta \rho_r$ ($= 10^{-4}, 10^{-5}$). Additionally, the solid lines in figure 6 indicate the exact solution of the VRP. We can ascertain from figure 6 that the two conserved variables as well as the pressure approximate the exact VRP solution in an acceptable manner, while the exact velocity of the VRP is also underestimated. A comparison between figure 5 and 6 convince us from the remarkable fact, that the sound velocity of the right state of the RP (38) do not influence the quality of the approximation of the exact solution of the VRP.

5.2   The Tracking Method Approach

Figure 7 shows the numerical results of a gas expanding into vacuum using the tracking algorithm combined with different kinds of flux calculations near the gas vacuum boundary. By turns, we apply the numerical flux calculations given by the equation (19), (16) [15], (27), and (25) depicted in figure 7a till 7d. The numerical values for the density, pressure, total energy and velocity are indicated by open circles, while the exact solution is plotted by a straight line. As it is presented by figure 7, the agreement of the approximation with the exact values is quite good: The numerical results for the same gas vacuum expansion wave (39), are very similar for all flux estimations. The numerical results for the conservative variables as well as for the pressure were nearly identical. Some differences become visible at the tracked velocity of the gas vacuum boundary.

Since the tracking method proposed in section 3 is only accurate to first order up to now, we restrict ourselves to the original first order upwind scheme to test the gas vacuum boundary tracking algorithm. Obviously, the HLLE scheme applied introduce large numerical dissipation, which is clearly visible at the left boundary of the rarefaction fan. This numerical damping is considerably reduced using a second order extension of the HLLE scheme (see [4] and references cited there).

The approximation of the location of the gas vacuum boundary is much more sensitive than the approximation of the conservative variables, the pressure or the velocity. This is due to the fact that the sound velocity of the left state is used for the evaluation of the propagation velocity v_v of the gas vacuum boundary. If the sound velocity is calculated according to the usual relation

(40a)
$$c = \sqrt{\frac{\gamma p}{\rho}} \, ,$$

we found that it is much larger than the sound velocity of the exact solution. Consequently, the calculated gas vacuum location should enormous overestimate (up to a factor more than two) the exact one, as it becomes obvious from figure 8. We note that this overestimation of the velocity of gas vacuum boundary do not tremendously influence the approximation of the conserved variables within the real gas region. In the vacuum limit, the exact pressure and density tend to zero such that the quotient p/ρ tends to zero, too. In the numerical approximation this is no longer valid. The quotient p/ρ near the vacuum is strongly influenced by approximation errors, when p and ρ tend to zero. It is obvious from figure 7 that

the approximate values of the pressure slightly exceeds the line of the exact values near the vacuum transition, while those of the density are found to be at the line. Hence, the sound velocity does not tend to zero but to a finite value; in our test problems approximately to 0.8. Consequently, the approximate value of v_v is larger than the exact one.

The numerical evaluation of the limit p/ρ may be circumvented by approximating the Euler equations near the gas vacuum interface by the isentropic equations of compressible fluid flow. This is of course a consistent approximation, because the expansion wave into vacuum is isentropic as mentioned above. In the isentropic approximation, the Euler equations are reduced to two equations and the relation for the sound velocity becomes

(40b)
$$c = \sqrt{\gamma \rho^{\gamma - 1}}$$

(see [2]). The use of this relation for the determination of the sound velocity c_i and the calculation of the velocity [11] of the gas vacuum boundary results in a better approximation of the exact solution. The results are depicted in figure 9 in comparison with the exact solution. Using the isentropic sound velocity (40b), the velocity of the gas vacuum boundary is slightly underestimated. As we have already seen in figure 7, the kind of numerical flux calculation near the gas vacuum interface have a petty influence to the approximated gas vacuum location.

We have to emphasize that the mean value v_i in (9) is obtained directly in terms of the velocities $\bar{v}_i^n$ and v_{i-1}^n. Averaging the conserved variables and calculating v as m/ρ results in a slight overestimation of the velocity of the gas vacuum boundary by the tracking step. This again shows the sensitivity of this variable.

6.   Conclusion and Outlook

The HLLE scheme combined with the proposed tracking algorithm is an accurate and effective method to solve the Euler equations numerically for a computational domain containing a gas region, a gas vacuum boundary and a vacuum region. Additionally, with the proposed numerical flux calculation near the gas vacuum layer, encouraging results are obtained for a simple test configuration. However, to take the full advantage of the high resolution upwind schemes into account, it seems necessary to extend the tracking algorithm to second order accuracy.

In the two-dimensional case, the situation becomes more complicated. The gas vacuum boundary now is a curve in the two-dimensional computational region, which is defined implicitly by the solution and evolves dynamically with it. A proper method to extend one-dimensional tracking ideas to this situation is the following method which resembles the ideas of Chern et al. [3] in their tracking scheme for contact discontinuities and shock waves. The curve is discretized by a number of points. The propagation of the gas vacuum boundary now involves the motion of these points. The Euler equations (1) may be locally transformed to a system with the derivatives occuring only in the normal and tangential direction of the gas vacuum boundary. Using operator splitting in the normal and tangential direction, the system is approximated by solving the tangential equations and normal equations one after the other within one time step. Via our one-dimensional tracking algorithm, the movement of the points of the gas vacuum curve into normal direction is obtained. The left state is obtained by integrating the approximate solution in the gas state over a rectangle with the length Δx and Δy. From the movement of the gas vacuum boundary, only the information on the transition from one grid zone to the other should be extracted. According to this, the flux should be zero or calculated in analogy to (19) or (27). The other flux calculations are too complicated in two space dimensions.

Other numerical methods to track interfaces in two dimensions may also be extended to track the gas vacuum boundary using our one-dimensional considerations. An overview of tracking algorithms has been given by Hyman in [10]. In [7], Goel et al. use the VRP to determine the movement of the boundary of a two-dimensional computational region. This boundary of the computational region is

then moved according to this propagation rate. In each time step a new Eulerian grid is generated within the computational region by using boundary fitted coordinates. The conservation equations are approximated by the Godunov scheme formulated on such a moving grid. Especially, the generation of a new grid at each time level needs much computational effort, but this method seems to be quite accurate.

We restricted ourselves to the case of the equation of state of perfect gas. But the tracking algorithm may be easily extended to more general equations of state. The general assumption is that an expansion into vacuum is a continuous solution of the Euler equations. The solution of the VRP consists of one rarefaction wave connecting the left state with the vacuum. The velocity of the gas vacuum boundary is then obtained from the conditions that within the rarefaction the conservative values tend to zero while the Riemann invariant associated with the first characteristic field remains constant (for further details see [2]).

Appendix A: The VRP for a general Equation of State

Smooth similarity solutions of (1) are determined by the equations

$$- \xi \frac{du}{d\xi} + \frac{d}{d\xi} f(u) = 0 \quad , \quad \text{with} \quad \xi = x/t \quad .$$

Explicitly,

$$(v - \xi) \rho_\xi + \rho v_\xi = 0 \quad ,$$

$$\rho (v - \xi) v_\xi + p_\xi = 0 \quad , \qquad \qquad \text{(A.1)}$$

$$\rho (v - \xi) \varepsilon_\xi + p v_\xi = 0 \quad .$$

This system has a solution if and only if

$$(v - \xi) \cdot [(v - \xi)^2 - C^2] = 0 \quad , \quad C^2 = p_\rho + p p_\varepsilon / \rho^2 \quad . \qquad \text{(A.2)}$$

The first alternative, $v = \xi$, is no valid solution, because it contradicts the conclusion from (A.1) that $v_\xi = 0$. This could only be avoided, if $|\rho_\xi| = \infty$ in contradiction to the assumption of smoothness. In fact, it describes a contact discontinuity, across which ρ can jump while v and p remain constant.

For the second possibility we obtain from (A.1)

$$\pm C \rho_\xi + \rho v_\xi = 0 \quad ,$$

$$p_\xi \pm C_\rho v_\xi = 0 \quad ,$$

$$p v_\xi \pm C_\rho \varepsilon_\xi = 0 \quad .$$

Combining the first two equations leads to

$$\frac{dp}{d\rho} = C^2 \quad , \qquad \qquad \text{(A.3)}$$

combining the first and third equation gives

$$\frac{d\varepsilon}{d\rho} = \frac{p}{\rho^2} \quad ; \qquad \qquad \text{(A.4)}$$

the first equation is accordingly rewritten as

$$\frac{dv}{d\rho} = \mp \frac{C}{\rho} \quad . \qquad\qquad (A.5)$$

Defining the vacuum state as being one of vanishing density, i.e. $\rho_{vac} = 0$, the general VRP can be solved exactly by integrating the equations (A.3) - (A.5) from ρ_{vac} to the state of matter, say ρ_0. Usually this has to be done numerically. Note that the integrals must converge inspite of the singularity of the integrands at $\rho = 0$.

We also need the other state variables of the vacuum as described in (4). These are found most easily by considering the limit of vanishing density of a dilute perfect gas, and by ascribing the vacuum a temperature of zero.

But we may also consider the above expansion problem for perfect gases. Then (A.3) and (A.4) can be integrated explicitly giving

$$p = (\gamma - 1)\rho\varepsilon \ , \ C^2 = \gamma p/\rho \Rightarrow p = p_0 (\rho/\rho_0)^\gamma \ , \ \varepsilon = \varepsilon_0 (\rho/\rho_0)^{\gamma-1} \quad . \qquad (A.6)$$

Hence, knowing that the transition from a state of matter to vacuum must be smooth, we can define the latter by letting $\rho \to 0$. Then $p \to 0$, and also $\varepsilon \to 0$ which is of course reasonable (the inner energy corresponds to temperature, and quantum effects are neglected). In addition, we can define $C_{vac} = 0$, since $C^2 \sim p/\rho \sim \rho^{\gamma-1} \to 0$ due to $\gamma > 1$. These formulae indicate how one should choose the values of the state variables when one tries to solve the VRP numerically by approximating the vacuum as a limit of a dilute perfect gas.

Thus, we can find the solution to the expansion of a perfect gas into vacuum by simple integration. In particular, from (A.5) we get

$$\frac{dv}{d\rho} = \mp \frac{C}{\rho} = \mp \left(\frac{\gamma \, p_0}{\rho_0^\gamma} \rho^{\gamma-3} \right)^{1/2} \quad ,$$

hence

$$v_{vac} - v_0 = \pm \frac{2}{\gamma - 1} C_0 \quad , \quad C_0 = (\gamma p_0 / \rho_0)^{1/2} \qquad (A.7)$$

(the sign depends on the relative location of gas and vacuum). This relation defines the "vacuum velocity"; more precisely, it yields the propagation speed of the boundary between gas and vacuum, i.e. the expansion velocity. The explicit solution has been given at the end of section 2.

Main program

Read input
100 continue

(CALL TRACK)

No calculation of slope necessary

(CALL HLL)

goto 100
Print or plot all data

time loop

Subroutine TRACK

Localiation of the gas boundary
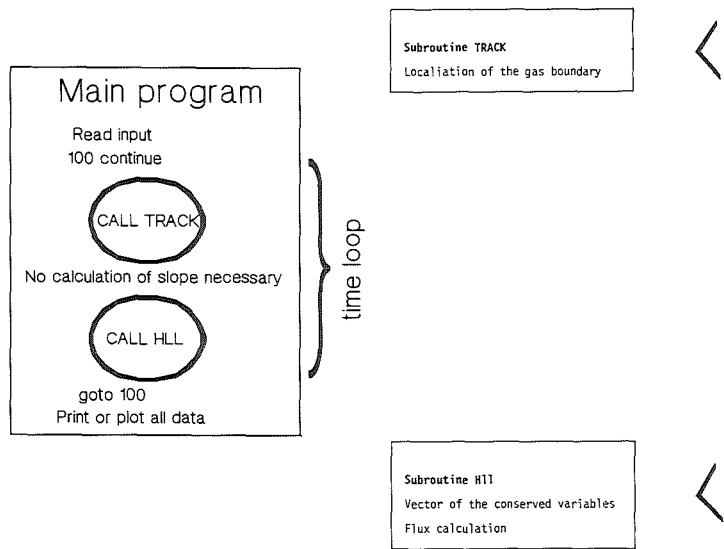
```
IF ( (XVAC(NPVAC) - XIR) .GT. (1.0E-2*DX(IVAC+1)) ) THEN
   LCASE2 = .TRUE.
   ULMCL  = UVACL - CVACL
ELSE
   LCASE2      = .FALSE.
   XVAC(NPVAC) = MIN(XVAC(NPVAC),XIR)
ENDIF
```

Subroutine Hll

Vector of the conserved variables

Flux calculation

```
IF (LCASE2) THEN
   .
   .
   IF    (CFLUX .EQ. 'VRGOD') THEN
      CALL VRP   (...)
   ELSEIF (CFLUX .EQ. 'VRHLE') THEN
      CALL VRPHLE (...)
   ELSEIF (CFLUX .EQ. 'VRHLS') THEN
      CALL VRPHLS (...)
   ELSEIF (CFLUX .EQ. 'VRIFX') THEN
      CALL VRPIFX (...)
   ENDIF
   .
   .
ENDIF
```

CFLUX IS INPUT PARAMETER

LCASE2=.TRUE.
Flux calculation

FLUX=.FALSE.
Fluxes become zero Flux calculation

t
t_{n+1}
t_n
x_v^{n+1}
vacuum
x_v^n
$x_{i-1/2}$ $x_{i+1/2}$ x

t
t_{n+1}
t_n
x_v^{n+1}
vacuum
x_v^n
$x_{i-1/2}$ $x_{i+1/2}$ x
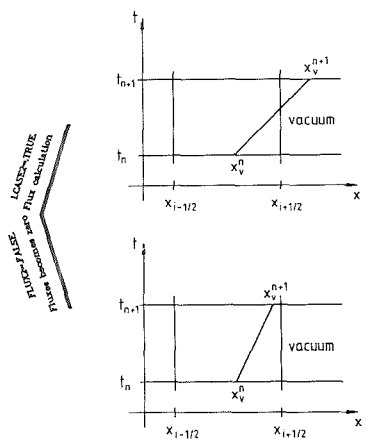
```fortran
      SUBROUTINE VRP ( X,T,RHOL,VL,PL,RO1,RO2,
     &                 RHO,V,P,GG1,GG2,GG3,GG4,GG5 )
*
***********************************************************************
*                 S U B R O U T I N E     V R P                      *
*                                                                    *
* APPROXIMATION OF THE VACUUM RIEMANN PROBLEM                        *
* CONTAINING IDEAS OF HARTEN, LAX AND VAN LEER                       *
* INPUT: X, T, RHOL, VL, PL                                          *
* AOUTPUT: RHO, V, P, GG1, GG2, GG3, GG4, GG5                        *
* REMARK: IN THIS ROUTINE THE FLUXES ARE CALCULATE                   *
*         GP1=1/(GAMMA-1) IS DEFINED IN COMMON/EOS                   *
*                                                                    *
*AH                                                    FEBRUAR'93*
***********************************************************************
*
      INCLUDE(EOS1M)
      REAL RHOL,VL,PL,RO1,RO2
      REAL RHO,V,P
      REAL GG1,GG2,GG3,GG4,GG5
      CL  = SQRT(GAMMA*PL/RHOL)
      XSI = X/T
      IF (XSI .LE. VL-CL) THEN
         RHO = RHOL
         V   = VL
         P   = PL
      ELSEIF (XSI .LE. VL+2.0*GP1*CL) THEN
         V   = ( (GAMMA-1.0)*VL + 2.0*(XSI+CL) )/(GAMMA + 1.0)
         RHO = ( (V-XSI)**2 * RHOL**GAMMA/(GAMMA*PL) )**GP1
         P   = (RHO/RHOL)**GAMMA*PL
      ELSE
         RHO = 0.0
         V   = 0.0
         P   = 0.0
      ENDIF
      EVRP = GP1*P + 0.5*RHO*V*V
C
C---> F(U_L) FLUESSE
C
      GG1  = RHO*V
      GG2  = RHO*V*V + P
      GG3  = V*(EVRP + P)
      GG4  = RO1*V
      GG5  = RO2*V
      END
```

```fortran
      SUBROUTINE VRPHLE( DYVAC,DTCFL,
     &                   RVL,UVL,PVL,UVAC,ULMCL,RO1,RO2,
     &                   GG1,GG2,GG3,GG4,GG5 )
*
*****************************************************************************
*              S U B R O U T I N E    V R P H L E                           *
*                                                                           *
* CALCULATION OF THE NUMERICAL FLUXES                                       *
* INPUT: DYCAV, DTCFL, RVL, UVL, PVL, UVAC, ULMCL,RO1,RO2                    *
* OUTPUT: GG1, GG2, GG3, GG4, GG5                                           *
* REMARKG: THE EVALUATION OF THE INTEGRAL EXPANTED WITH THE                 *
*          HLL-APPROXIMATION CONDUCTS TO THE NUMERICAL FLUXES               *
*          GP1=1/(GAMMA-1) IS DEFINED IN COMMON/EOS                         *
*                                                                           *
*AH                                                         FEBRUAR'93 *
*****************************************************************************
*
      INCLUDE(EOS1M)
      REAL DYVAC,DTCFL,RVL,UVL,PVL,UVAC,ULMCL,RO1,RO2
      REAL GG1,GG2,GG3,GG4,GG5
      REAL MVL,EVL
      REAL UL1,UL2,UL3,UL4,UL5
      REAL HELP1,HELP2
      REAL ULR1,ULR2,ULR3,ULR4,ULR5,
     &     HMAX,HMIN
      WRITE(6,*) ' DYVAC=',DYVAC,'  UVAC=',UVAC,'  ULMCL=',ULMCL
      WRITE(6,*) ' DTCFL=',DTCFL
C
C---> FESTLEGUNG DER KONSTANTEN
C
      MVL      = RVL * UVL
      EVL      = GP1*PVL + 0.5*RVL*UVL*UVL
      HELP1    = UVAC - ULMCL
      HELP2    = 1.0 / HELP1
C
C---> U_L ZUSTANDSVEKTOR
C
      UL1      = RVL
      UL2      = MVL
      UL3      = EVL
      UL4      = RO1
      UL5      = RO2
C
C---> F(U_L) FLUESSE
C
      GG1      = MVL
      GG2      = RVL*UVL*UVL + PVL
      GG3      = UVL*(EVL + PVL)
      GG4      = RO1*UVL
      GG5      = RO2*UVL
C
C---> MITTLERER ZUSTAND
C
      ULR1     = ( GG1 - ULMCL*UL1 ) * HELP2
      ULR2     = ( GG2 - ULMCL*UL2 ) * HELP2
      ULR3     = ( GG3 - ULMCL*UL3 ) * HELP2
      ULR4     = ( GG4 - ULMCL*UL4 ) * HELP2
      ULR5     = ( GG5 - ULMCL*UL5 ) * HELP2
C
C---> FLUSSBERECHNUNG
C
      HMIN     =   MIN( ULMCL*DTCFL, DYVAC ) / DTCFL
      HMAX     =   MAX( 0., DYVAC - ULMCL*DTCFL ) / DTCFL
      GG1      = GG1 - HMIN*UL1 - HMAX*ULR1
      GG2      = GG2 - HMIN*UL2 - HMAX*ULR2
      GG3      = GG3 - HMIN*UL3 - HMAX*ULR3
      GG4      = GG4 - HMIN*UL4 - HMAX*ULR4
      GG5      = GG5 - HMIN*UL5 - HMAX*ULR5
C
      END
```

- 28 -

```
      SUBROUTINE VRPHLS( DYVAC,DTCFL,
     &                   RVL,UVL,PVL,UVAC,ULMCL,RO1,RO2,
     &                   GG1,GG2,GG3,GG4,GG5 )
*
*******************************************************************************
*                 S U B R O U T I E    V R P H L S                          *
*                                                                           *
* SIMPLE CALCULATION OF THE FLUXES                                          *
* INPUTR: DYVAC, DTCFL ( WIRD NICHT BENOETIGT! )                            *
*         RVL, UVL, PVL, UVAC, ULMCL, RO1, RO2                              *
* OUTPUT: GG1, GG2, GG3, GG4, GG5                                           *
* REMARK: RESAMBLE ROUTINE VRPHLE, BUT THE CALCULATION OF THE FLUXES        *
*         IS DIFFERENT                                                      *
*         GP1=1/(GAMMA-1) IS DEFINED IN COMMON/EOS                          *
*                                                                           *
*AH                                                       FEBRUAR'93        *
*******************************************************************************
*
      INCLUDE(EOS1M)
      REAL DYVAC,DTCFL,RVL,UVL,PVL,UVAC,ULMCL,RO1,RO2
      REAL GG1,GG2,GG3,GG4,GG5
      REAL MVL,EVL
      REAL UL1,UL2,UL3,UL4,UL5
      REAL HMIN,HELP1,HELP2
C
      MVL     = RVL * UVL
      EVL     = GP1*PVL + 0.5*RVL*UVL*UVL
C
C---> U_L ZUSTANDSVEKTOR
C
      UL1     = RVL
      UL2     = MVL
      UL3     = EVL
      UL4     = RO1
      UL5     = RO2
C
C---> F(U_L) FLUESSE
C
      GG1     = MVL
      GG2     = RVL*UVL*UVL + PVL
      GG3     = UVL*(EVL + PVL)
      GG4     = RO1*UVL
      GG5     = RO2*UVL
C
C---> FLUSSBERECHNUNG
C
      HMIN    = MIN( 0.0, ULMCL )
      HELP1   = UVAC - HMIN
      HELP2   = 1.0 / HELP1
      GG1     = ( GG1 - HMIN*UL1 ) * UVAC * HELP2
      GG2     = ( GG2 - HMIN*UL2 ) * UVAC * HELP2
      GG3     = ( GG3 - HMIN*UL3 ) * UVAC * HELP2
      GG4     = ( GG4 - HMIN*UL4 ) * UVAC * HELP2
      GG5     = ( GG5 - HMIN*UL5 ) * UVAC * HELP2
C
      END
```

```
        SUBROUTINE VRPIFX ( DYVAC,DTCFL,
     &                      RVL,UVL,PVL,UVAC,ULMCL,CVL,RO1,RO2,
     &                      GG1,GG2,GG3,GG4,GG5 )
*
*******************************************************************
*                 S U B R O U T I E    V R P F I X                *
*                                                                 *
* CALCULATION OF THE FLUXES WITH INTEGRALS                        *
* INPUT: DYVAC, DTCFL                                             *
*        RVL, UVL, PVL, UVAC, ULMCL, RO1,RO2                      *
* OUTPUT: GG1, GG2, GG3, GG4, GG5                                 *
* REMARK: THE ROUTINE REQUIRES THE FUNCTIONS RPINTE, BIKO, FAC    *
*         GP1=1/(GAMMA-1) IS DEFINED IN COMMON/EOS                *
*                                                                 *
*AH                                                  17.3.93      *
*******************************************************************
*
        INCLUDE(EOS1M)
        REAL DYVAC,DTCFL,RVL,UVL,PVL,UVAC,ULMCL,CVL,RO1,RO2
        REAL GG1,GG2,GG3,GG4,GG5
        REAL SOLINT
        REAL DTST,DTST1,XI1,XI2,XIMAX,ALFA
        REAL HELP1,HELP2,HELP3,HELP4
        INTEGER NU,MU,SIG

        IF ( DYVAC.GT.0.0 ) THEN
C
C-------> VORBEREITUNG
C
        DTST  = DYVAC/UVAC
        IF ( ULMCL .GT. 0.0 ) THEN
          DTST1 = MIN(DTCFL , DYVAC/ULMCL)
        ELSE
          DTST1 = DTCFL
        ENDIF
        XI1   = DYVAC / CVL / DTST1
        XI2   = DYVAC / CVL / DTST
        HELP1 = DYVAC / CVL / DTCFL
        HELP2 = 1.0 - DTST1/DTCFL
        XIMAX = UVAC/CVL
        NU    = INT( 2./(GAMMA-1.0) )
        ALFA  = (GAMMA - 1.0)/(GAMMA + 1.0)
C
        WRITE(6,*) ' DYVAC=',DYVAC,'  UVAC=',UVAC,'   ULMCL=',ULMCL
        WRITE(6,*) '  DTST=',DTST,' DTST1=',DTST1,'   CVL=',CVL
        WRITE(6,*) '   XI1=',XI1,'  XI2=',XI2
C
C-------> BERECHNUNG DES NUMERISCHEN FLUSSES
C
**********ERSTER FLUSS*********
        HELP3 = SOLINT( XI1,XI2,XIMAX,NU,1,NU )
        N1    = NU + 1
        GG1   = HELP1 * ALFA**N1 * XIMAX**N1 *
     &          RVL * CVL * HELP3
**********ZWEITER FLUSS*********
        HELP3 = SOLINT( XI1,XI2,XIMAX,NU,2,NU )
        HELP4 = SOLINT( XI1,XI2,XIMAX,NU,0,NU+2 )
        N1    = NU + 2
        GG2   = HELP1 * ALFA**N1 * XIMAX**N1 *
     &          ( RVL * CVL*CVL * HELP3 +
     &            PVL *            HELP4   )
**********DRITTER FLUSS*********
        HELP3 = SOLINT( XI1,XI2,XIMAX,NU,3,NU )
        HELP4 = SOLINT( XI1,XI2,XIMAX,NU,1,NU+2 )
        N1    = NU + 3
        GG3   = HELP1 * ALFA**N1 * XIMAX**N1 *
     &          ( 0.5 * RVL * CVL*CVL*CVL * HELP3 +
     &            GAMMA*GP1 * PVL * CVL *    HELP4   )
**********VIERTER FLUSS*********
        HELP3 = SOLINT( XI1,XI2,XIMAX,NU,1,NU )
        N1    = NU + 1
        GG4   = HELP1 * ALFA**N1 * XIMAX**N1 *
     &          RO1 * CVL * HELP3
```

```
**********FUENFTER FLUSS*********
        HELP3 = SOLINT( XI1,XI2,XIMAX,NU,1,NU )
        N1    = NU + 1
        GG5   = HELP1 * ALFA**N1 * XIMAX**N1 *
     &            RO2 * CVL * HELP3
C
        GG1   = GG1 + HELP2*  RVL*UVL
        GG2   = GG2 + HELP2*( RVL*UVL*UVL + PVL)
        GG3   = GG3 + HELP2*(0.5*RVL*UVL*UVL*UVL + GAMMA*GP1*PVL*UVL)
        GG4   = GG4 + HELP2*RO1*UVL
        GG5   = GG5 + HELP2*RO2*UVL
C
     ELSEIF ( DYVAC.EQ.0.0 ) THEN
        WRITE(6,*) '    DYVAC .EQ. 0       '
        GG1   = RVL * UVL
        GG2   = RVL*UVL*UVL + PVL
        GG3   = 0.5*RVL*UVL*UVL*UVL + GAMMA*GP1*PVL*UVL
        GG4   = RO1*UVL
        GG5   = RO2*UVL
     ELSE
        WRITE(6,*) ' >>>>>> ERROR IN VRPIFX ERROR <<<<<<'
        WRITE(6,*) ' REASON: DYVAC=',DYVAC,'  <0'
     ENDIF
     END
```

```
      REAL FUNCTION SOLINT(XI1,XI2,XIMAX,NU,MU,SIG)
C
C     ******************************************************************
C     ***                                                          ***
C     ***                    FUNCTION RPINTE                       ***
C     ***                                                          ***
C     ***   CALCULATE THE INTEGRAL OF THE EXACT RIEMANNFLUESSE     ***
C     ***                                                          ***
C     ***   INPUT  :  > INTEGRATIONSINTERVALL {XI1, XI2}           ***
C     ***   ------                                                 ***
C     ***             > EXPONENTS MU, SIG                          ***
C     ***                                                          ***
C     ***             > KONSTANTE  NU & XIMAX                      ***
C     ***                                                          ***
C     ***   OUTPUT :  > VALUE OF THE INTEGRAL SOLINT               ***
C     ***   ------                                                 ***
C     ***                                                          ***
C     ***                                                          ***
C     ******************************************************************
C
      INTEGER  BIKO
      INTEGER  NU,MU,SIG
      INTEGER  M,K,NNN,NNNN
      REAL     XI1,XI2,XIMAX
      REAL     A1,A2,A3,A4,A5
      REAL     RNU,RMU,RSIG,RBIKO1,RBIKO2
C
      IF( NU .LT. 0  .OR.  MU .LT. 0  .OR.  SIG .LT. 0 ) THEN
        WRITE(6,*) ' >>>>> ERROR IN RPINTE GERUFEN VON VRPIFX        '
        WRITE(6,*) '                            GRUND:               '
        WRITE(6,*) '                              NU<0    NU=',NU
        WRITE(6,*) '                              MU<0    MU=',NU
        WRITE(6,*) '                              SIG<0   SIG=',SIG
        WRITE(6,*) '                                            <<<<< '
        STOP
      ENDIF
C
      RNU    = REAL(NU)
      RMU    = REAL(MU)
      RSIG   = REAL(SIG)
C
C...  ERSTER TERM DES INTEGRAL
      A1     = 1.0/XI1 - 1.0/XI2
C
C...  ZWEITER TERM DES INTEGRAL
      A2     = (RNU*RMU-RSIG) * LOG(XI2/XI1) / XIMAX
C
C...  DRITER TERM DES INTEGRAL
      IF( MU .EQ. 0 ) THEN
        A3     = 0.0
      ELSE
        A3     = RNU**MU * RSIG / XIMAX**(MU+1) *
     &           ( XI2**MU - XI1**MU ) / RMU
      ENDIF
C
C...  VIERTER TERM DES INTEGRAL
      A4     = 0.0
      DO 100 M=2,MU
C--------------------------------> HELP-GROESSEN
                     NNN     = BIKO(MU,M-1)
                     RBIKO1 = REAL(NNN)
      A4       = A4 + RNU**(M-1) / XIMAX**M * RBIKO1 *
     &           (RNU/REAL(M)*(RMU-REAL(M)+1.0)-RSIG) *
     &           ( XI2**(M-1) - XI1**(M-1) ) / REAL(M-1)
100   CONTINUE
C
```

```
C...   FUENFTER TERM DES INTEGRAL
       A5       = 0.0
       DO 300 K=2,SIG
          DO 200 M=K,MU+K
C--------------------------------> HELP-GROESSEN
                                   NNN    = BIKO(SIG,K)
                                   NNNN   = BIKO(MU,M-K)
                                   RBIKO1 = REAL(NNN)
                                   RBIKO2 = REAL(NNNN)
          A5       = A5 + (-1.0)**K * RNU**(M-K) / XIMAX**M *
     &                    RBIKO1 * RBIKO2 *
     &                    ( XI2**(M-1) - XI1**(M-1) ) / DBLE(M-1)
 200      CONTINUE
 300   CONTINUE
C
C...   INTEGRALWERT DURCH ADDITION
       SOLINT = A1 + A2 - A3 + A4 + A5
C
       END
C
C======================================================================
C
       INTEGER FUNCTION BIKO(N,M)
C
C      ***************************************************************
C      ***                                                        ***
C      ***                   FUNCTION BIKO                        ***
C      ***                                                        ***
C      ***   CALCULATE THE BINOMINALKOEFFIZIENTS      | N |       ***
C      ***                                            | M |       ***
C      ***                                                        ,***
C      ***************************************************************
C
       INTEGER N,M,FAK
       INTEGER N1,M1,NM
C
       IF ( N .LT. M ) THEN
         WRITE(6,*) ' >>>>> ERROR IN BIKO GERUFEN VON SOLINT      '
         WRITE(6,*) '          GRUND:  N=',N,'   <  M=',M
         STOP
       ENDIF
       N1    = FAK(N)
       NM    = FAK(N-M)
       M1    = FAK(M)
       BIKO  = N1/NM/M1
C
       END
C
C======================================================================
C
       INTEGER FUNCTION FAK(N)
C
C      ***************************************************************
C      ***                                                        ***
C      ***                   FUNCTION FAK                         ***
C      ***                                                        ***
C      ***   BERECHNET DIE FAKULTAET N!                           ***
C      ***                                                        ***
C      ***************************************************************
C
       INTEGER N,I
C
       IF ( N .LT. 0 ) THEN
         WRITE(6,*) ' >>>>> ERROR IN FAK GERUFEN VON BIKO        '
         WRITE(6,*) '          GRUND:  N=',N,'   <  0 '
         STOP
       ENDIF
       IF ( N .EQ. 0 ) THEN
         FAK       = 1
         GOTO 999
       ENDIF
       FAK     = 1
       DO 100 I=2,N
         FAK       = FAK*I
 100   CONTINUE
 999   CONTINUE
C
       END
```

References

[1]   Bruhn, G.; 'Erhaltungssätze und schwache Lösungen in der Gasdynamik',
        Math. Meth. in the Appl. Sci. 7, 470 - 479 (1985)

[2]   Chang, T. and Hsiao, L.; 'The Riemann Problem and Interaction of Waves in
        Gas Dynamics', Longman Group UK Limited, Essex 1989

[3]   Chern, I.L.; Glimm, J.; McBryan, O.; Plohr, B. and Yaniv, S.; 'Front tracking
        for gas dynamics', J. of Comput. Phys. 62, 83 - 110 (1986)

[4]   Einfeldt, B.; 'On Godunov-type methods for gas dynamics', SIAM J. Numer.
        Anal. 25 (1988), 294 - 318

[5]   Einfeldt, B.; Munz, C.D.; Roe, P.L. and Sjögreen, B., 'On Godunov-type
        methods near low densities', J. Comput. Phys. 92, 273 - 295 (1991)

[6]   Godunov, S.K.; 'Finite difference method for numerical computation of
        discontinuous solutions of the equations of fluid dynamics', Mat Sbornik 47
        (1959), 271 - 306 (in Russian)

[7]   Goel, B.; Munz, C.D.; Shutov, A.V.; Vorobiev, O.Y. and Ni, A.L.; 'Application
        of Godunov's scheme on a moving grid to the numerical simulation of ion
        beam target interaction', in preparation

[8]   Halter, E., 'A fast solver for Riemann problems', Math. Meth. in the Appl.
        Sci. 7, 101 - 107 (1985)

[9]   Harten, A.; Lax, P.D. and Leer, B. van, 'On upstream differencing and
        Godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25,
        35 - 62 (1983)

[10]   Hyman, J.M.; 'Numerical methods for tracking interfaces', Physica 12D,
        396 - 407 (1984)

[11]   Lax, P.D.; 'Shock waves and entropy', Proc. Symposium at the University of
        Wisconsin, E.H. Zarantonello (Ed.), 603 - 634 (1971)

[12]   Lax, P.D.; 'Hyperbolic systems of conservation laws II', Comm. Pure and Appl. Math. 10 (1957), 537 - 566

[13]   Munz, C.D.; 'On Godunov-type schemes for Lagrangean gas dynamics', to appear in SIAM J. Numer. Anal. 1993

[14]   Roberts A.W.; Varberg, D.E.; Convex Functions, Academic Press, New York, 1973

[15]   Schneider, R., Munz, C.-D.; KfK 5213, Karlsruhe, 1993

Legends

Figure 1:   (x, t)-diagram of a solution of the Riemann problem

Figure 2:   (x, t)-diagram of a solution of the vacuum Riemann problem

Figure 3:   Tracking of the gas vacuum boundary within the Eulerian grid
             a.) Case 1   b.) Case 2

Figure 4:   Approximation (20) of a solution of the vacuum Riemann problem

Figure 5:   Comparison between the exact solution of the VRP (solid lines) and the numerical solution of the RP (38) with finite $\delta p_r = \delta p_r$ (open circles). (a) $\delta p_r = 10^{-2}$, (b) $\delta p_r = 10^{-4}$, (c) $\delta p_r = 10^{-5}$ and (d) $\delta p_r = 10^{-6}$

Figure 6:   Numerical solution of the RP (38) with $\delta p_r = 0.1 \, \delta p_r$ (open circles) in comparison with the exact VRP (solid lines). δp_r is choosen equal to 10^{-4} (a) and equal to 10^{-5} (b)

Figure 7:   Numerical (open circles) and exact (solid lines) of the VRP (39). The numerical flux is calculated according equation (19) (a), (16) (b), (27) (c), and 25 (d), respectively

Figure 8:   Location of the gas vacuum boundary as a function of time;
             —— exact solution, ooo numerical results using the relation (40a) for the calculation of the sound velocity

Figure 9:   Exact (——) and numerical (ooo) solution of the location of the gas vacuum interface as a function of time. The sound velocity is calculated according (40b), while the numerical flux is determined from equation (19) (left upper), (27) (right upper), (16) (left lower) and (25), respectively
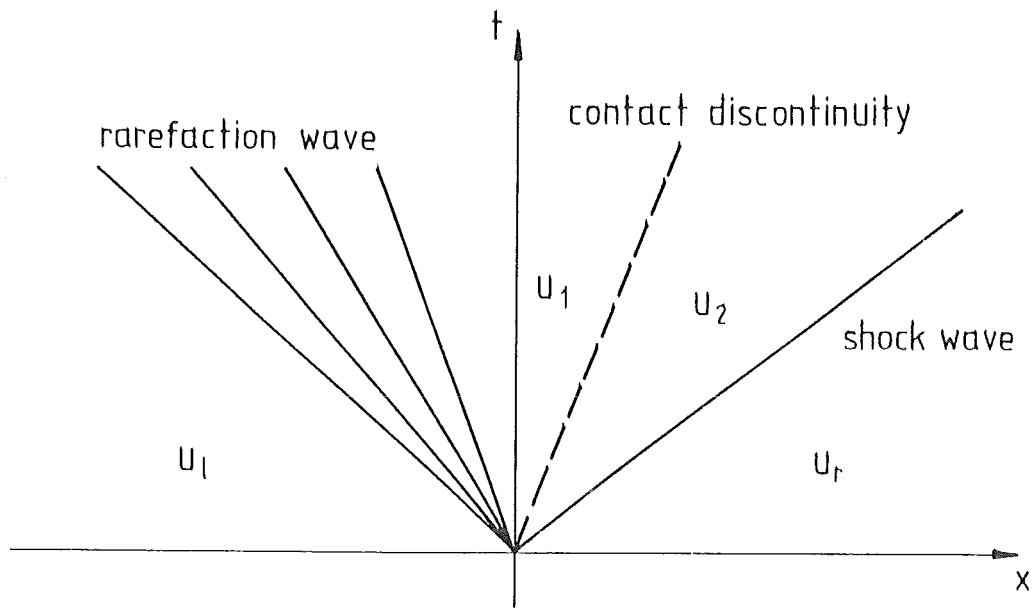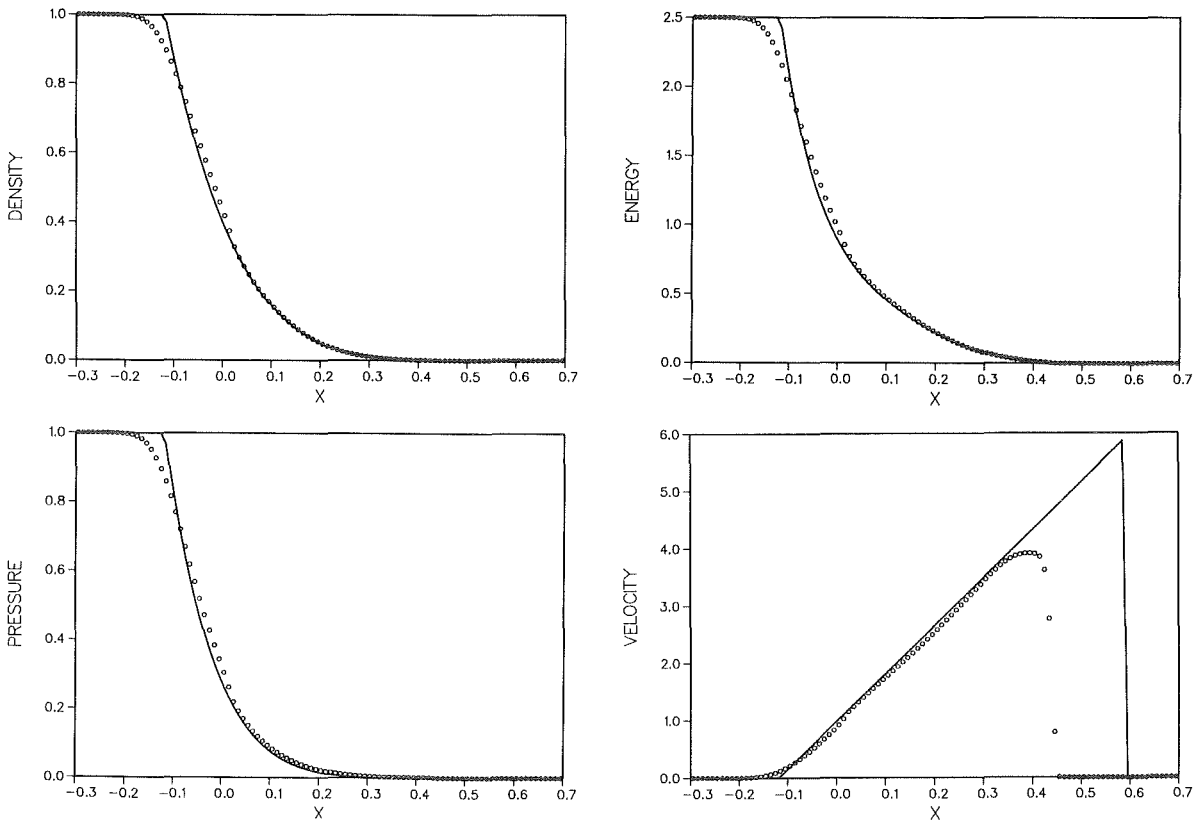
Figure 1
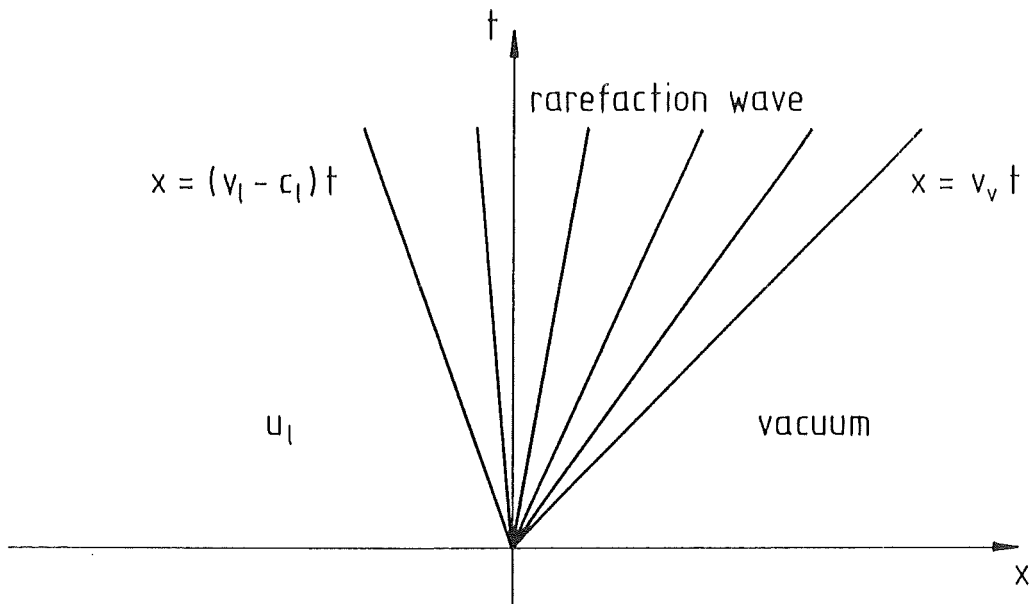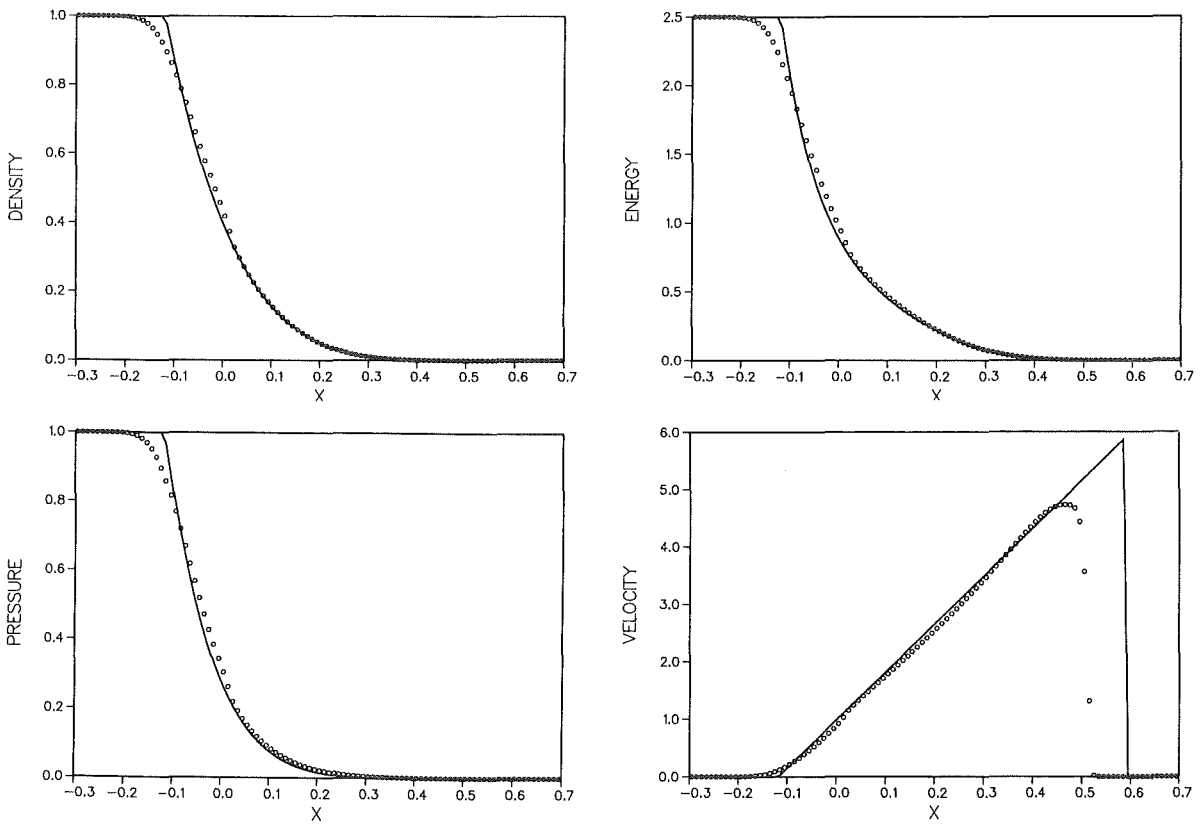

Figure 2
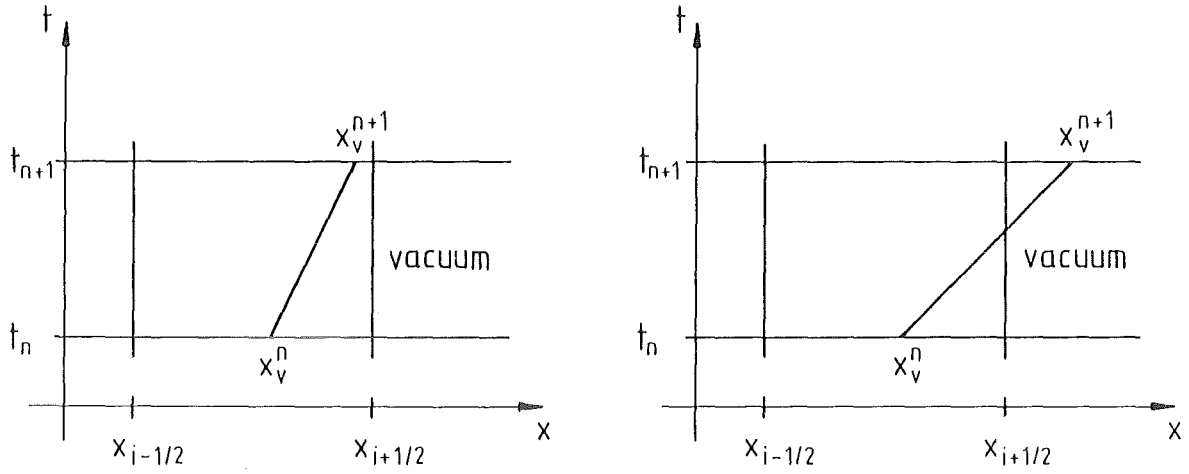
Figure 3
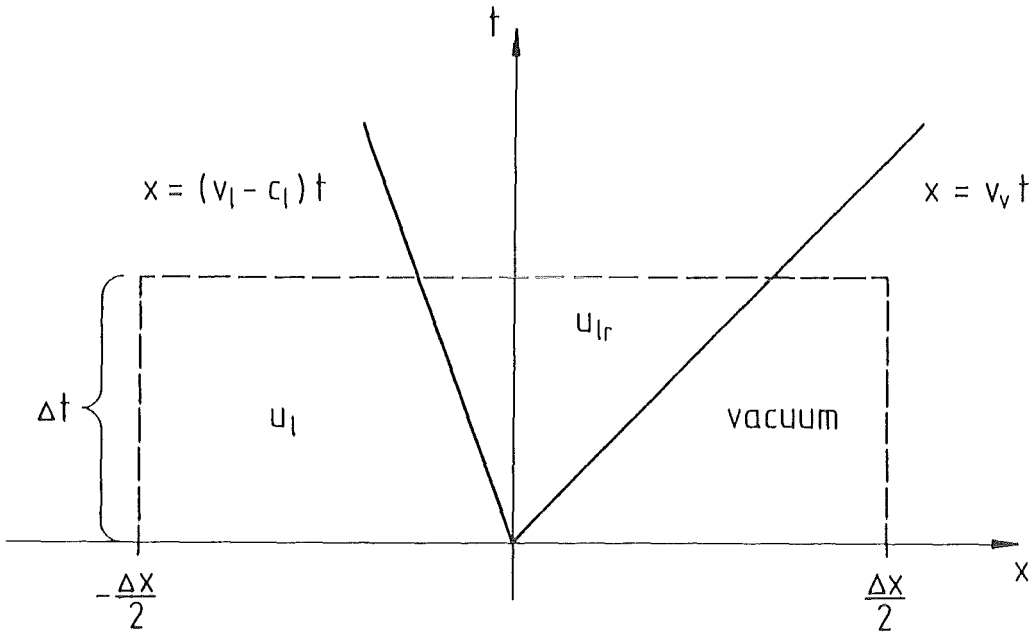
Figure 4
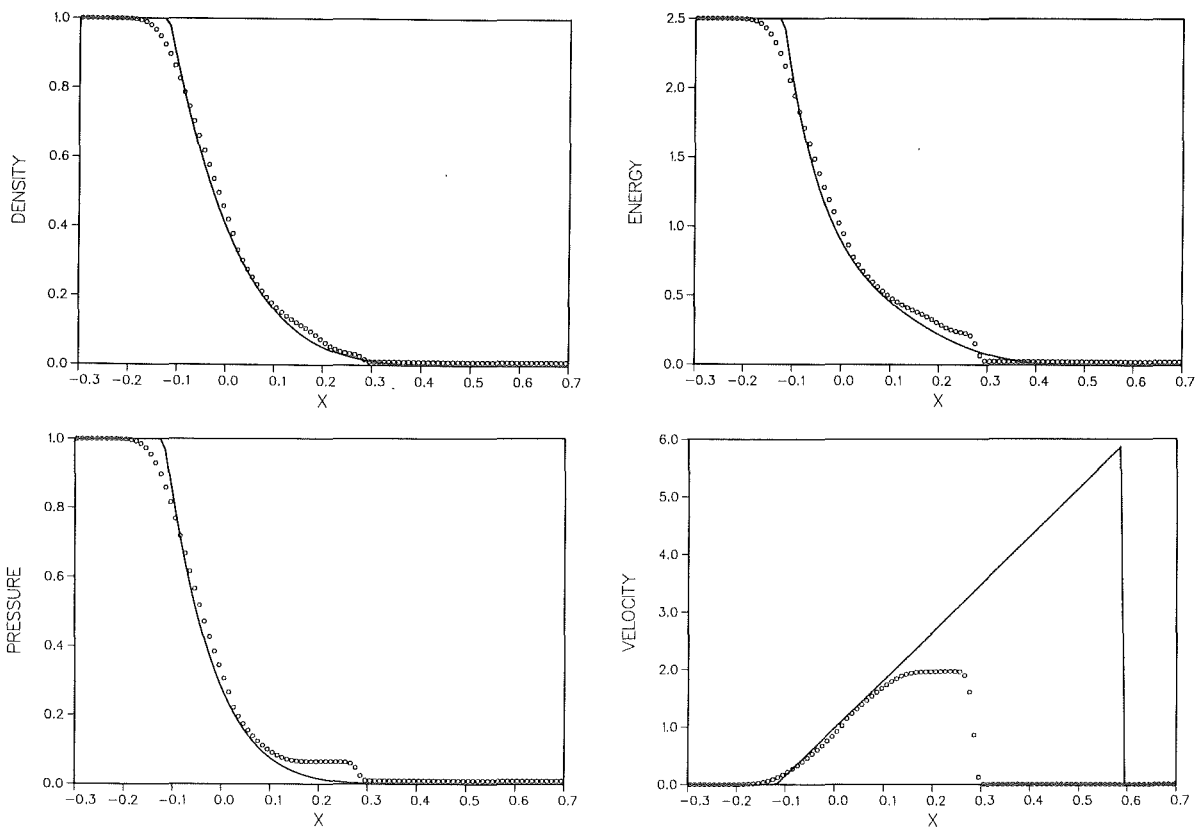
Figure 5a
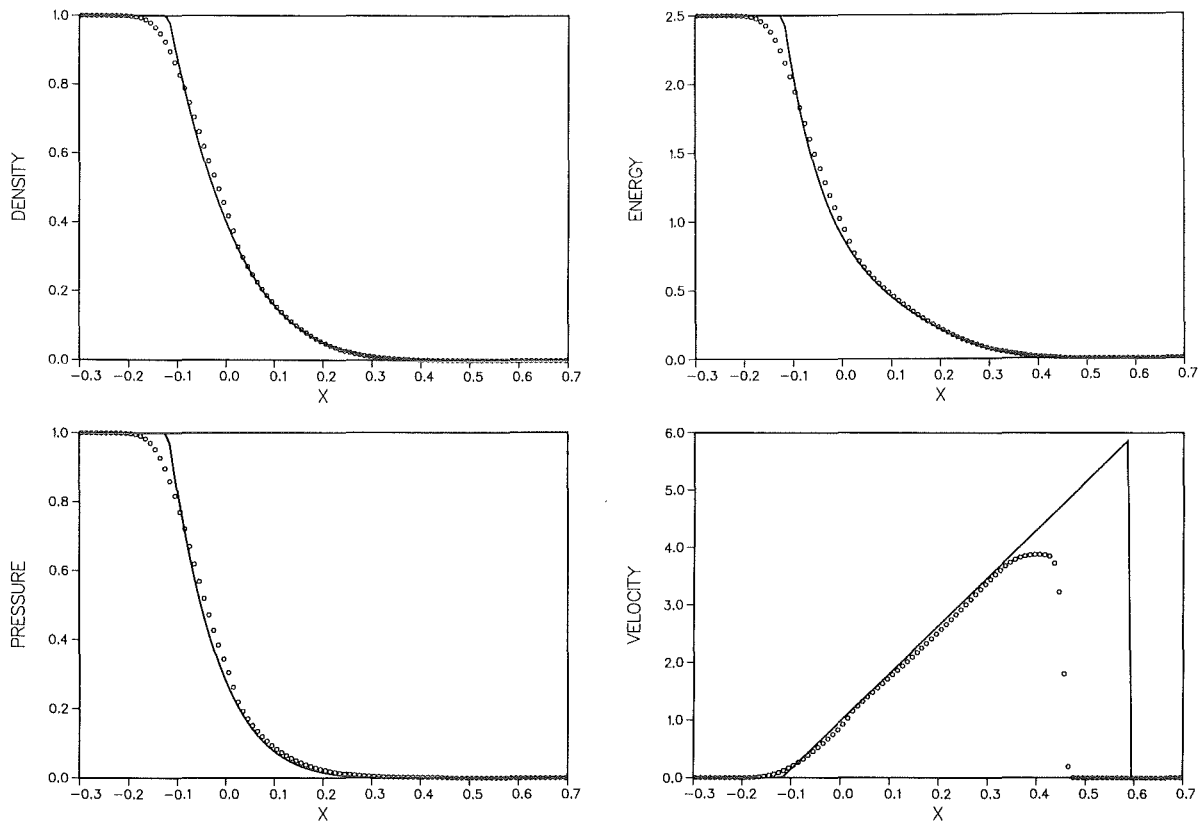
Figure 5b
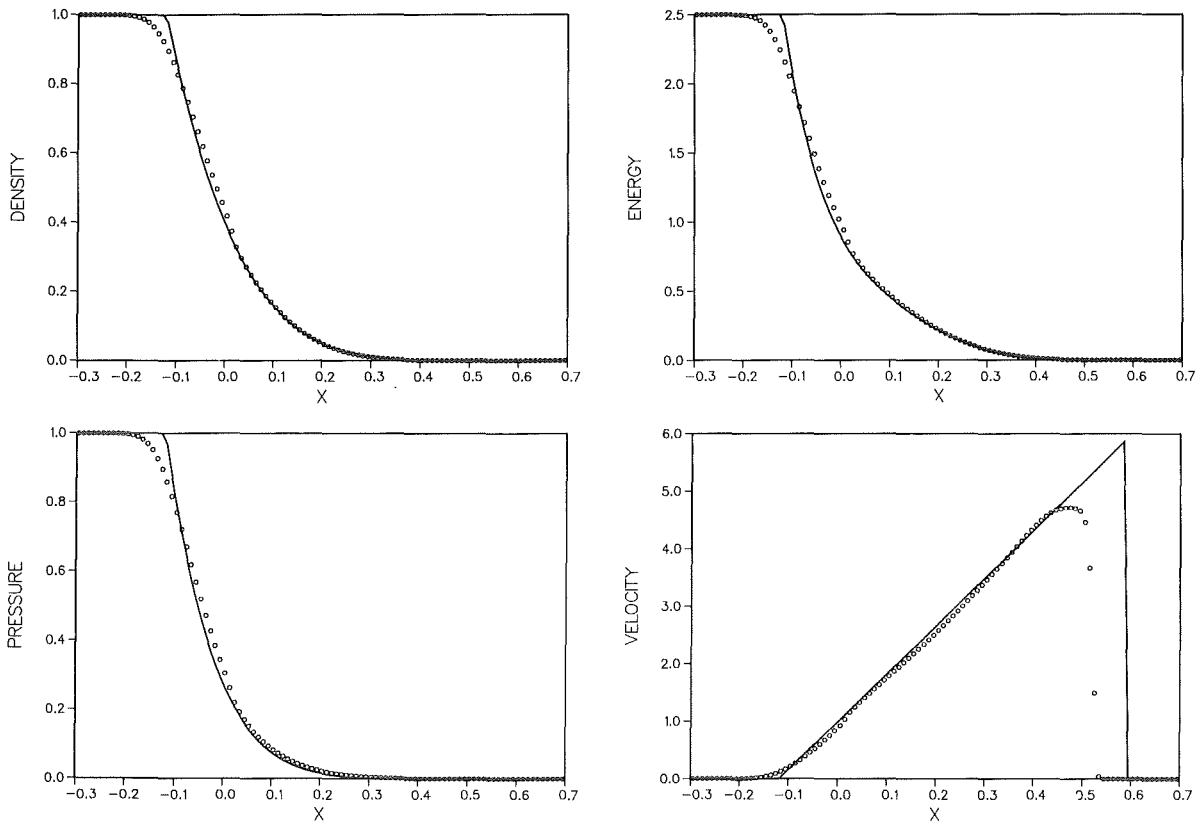
Figure 5c
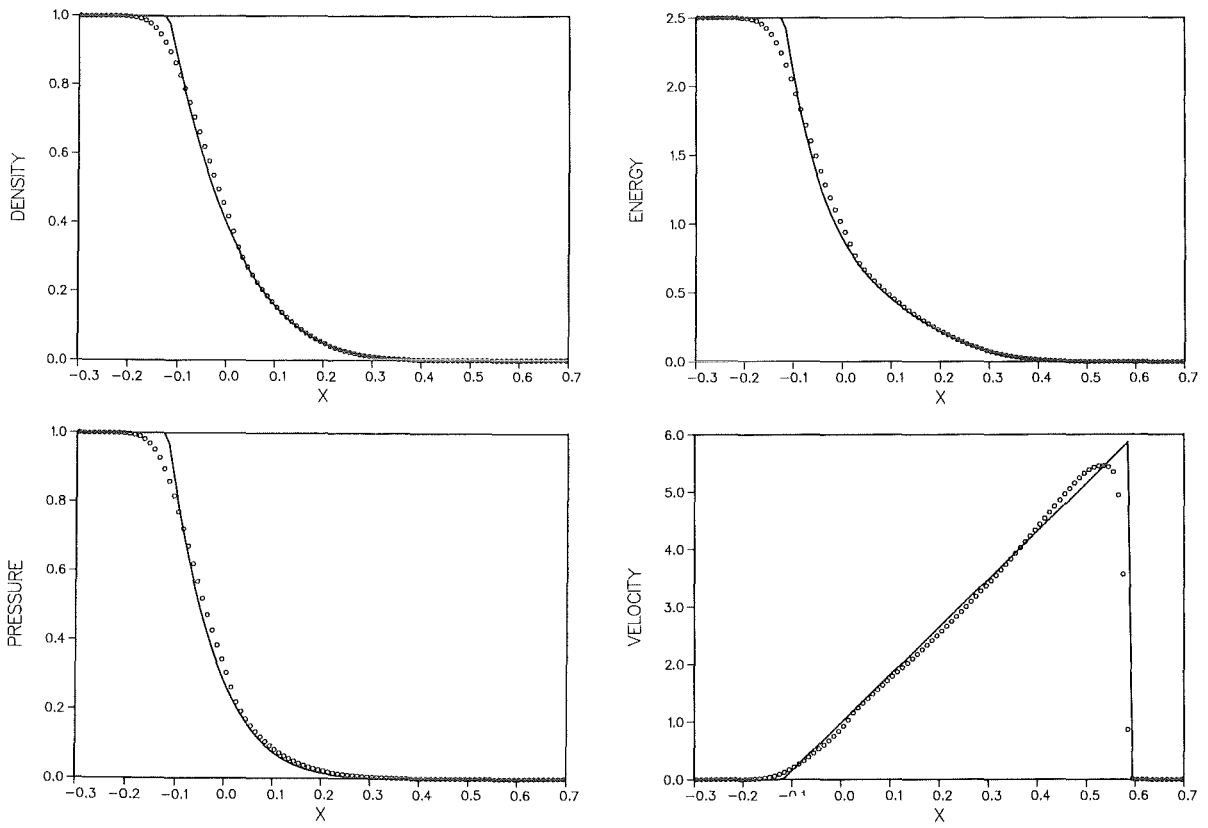
Figure 5d

Figure 6a

Figure 6b
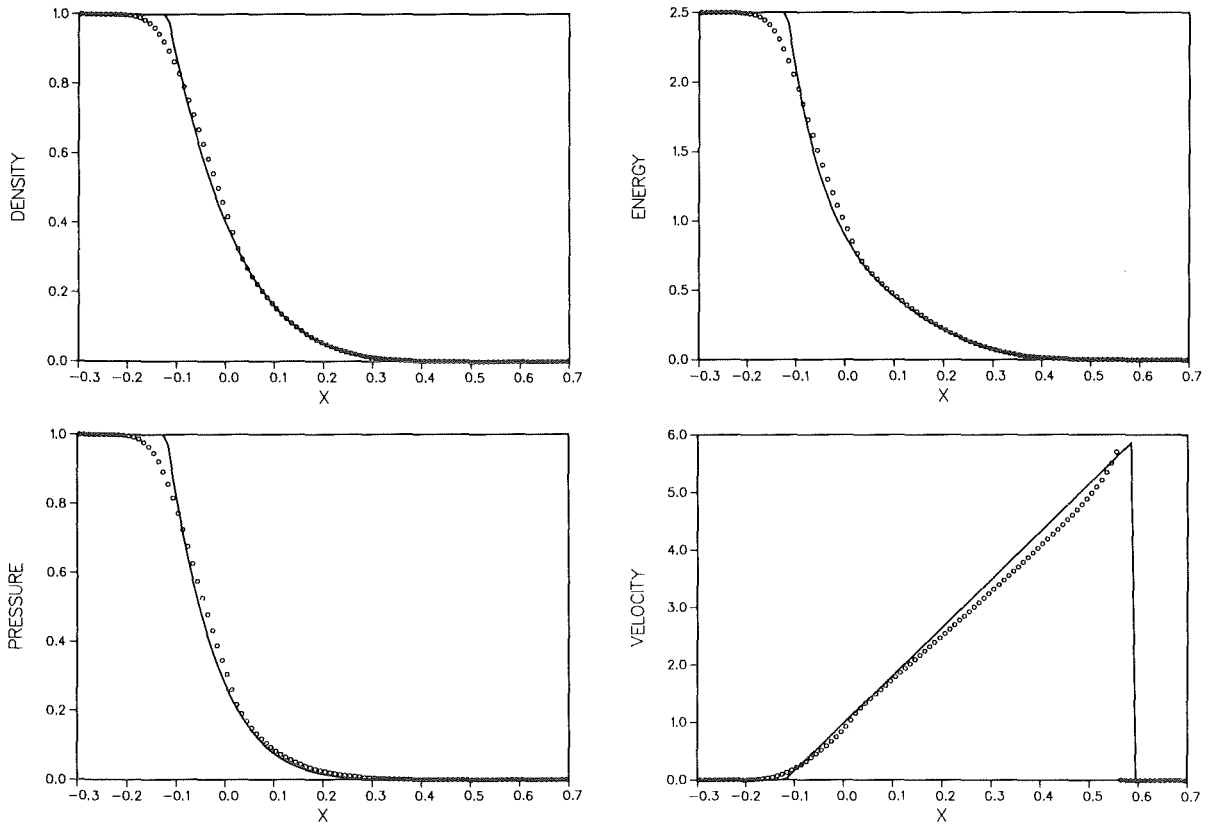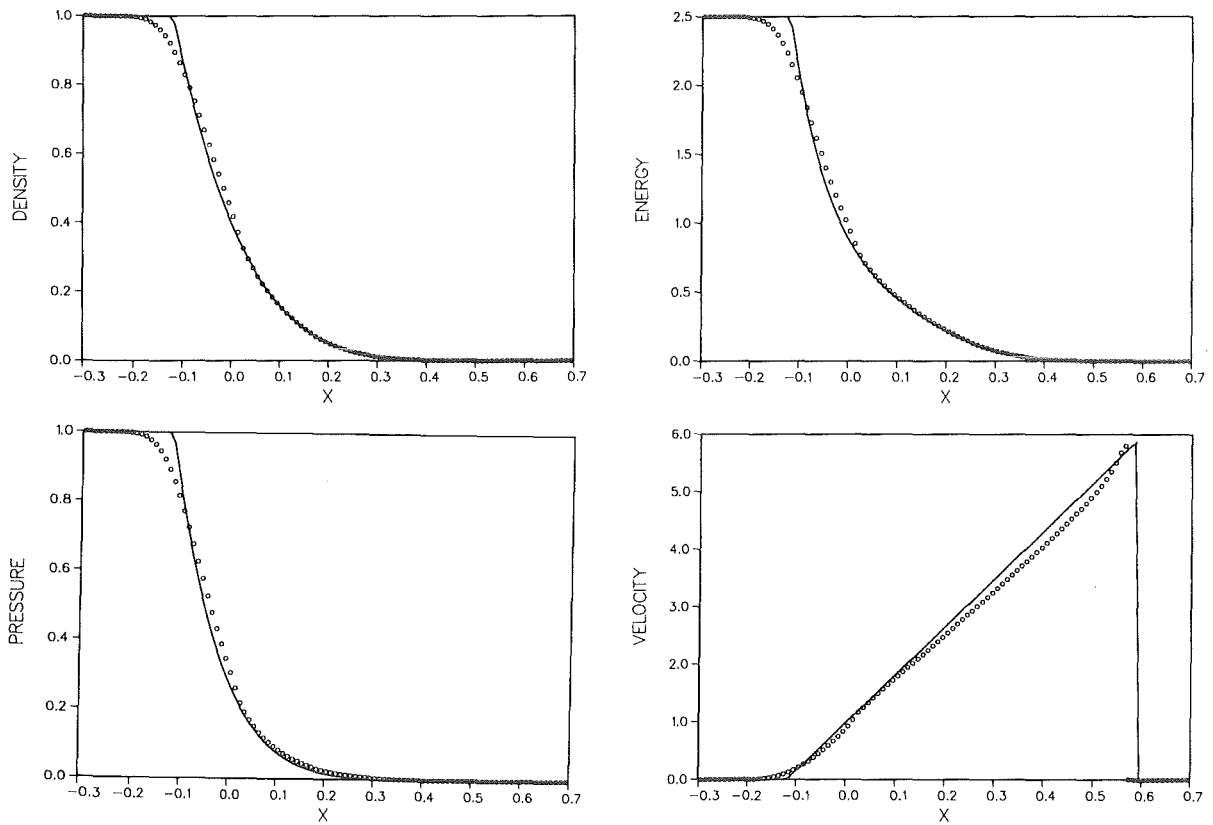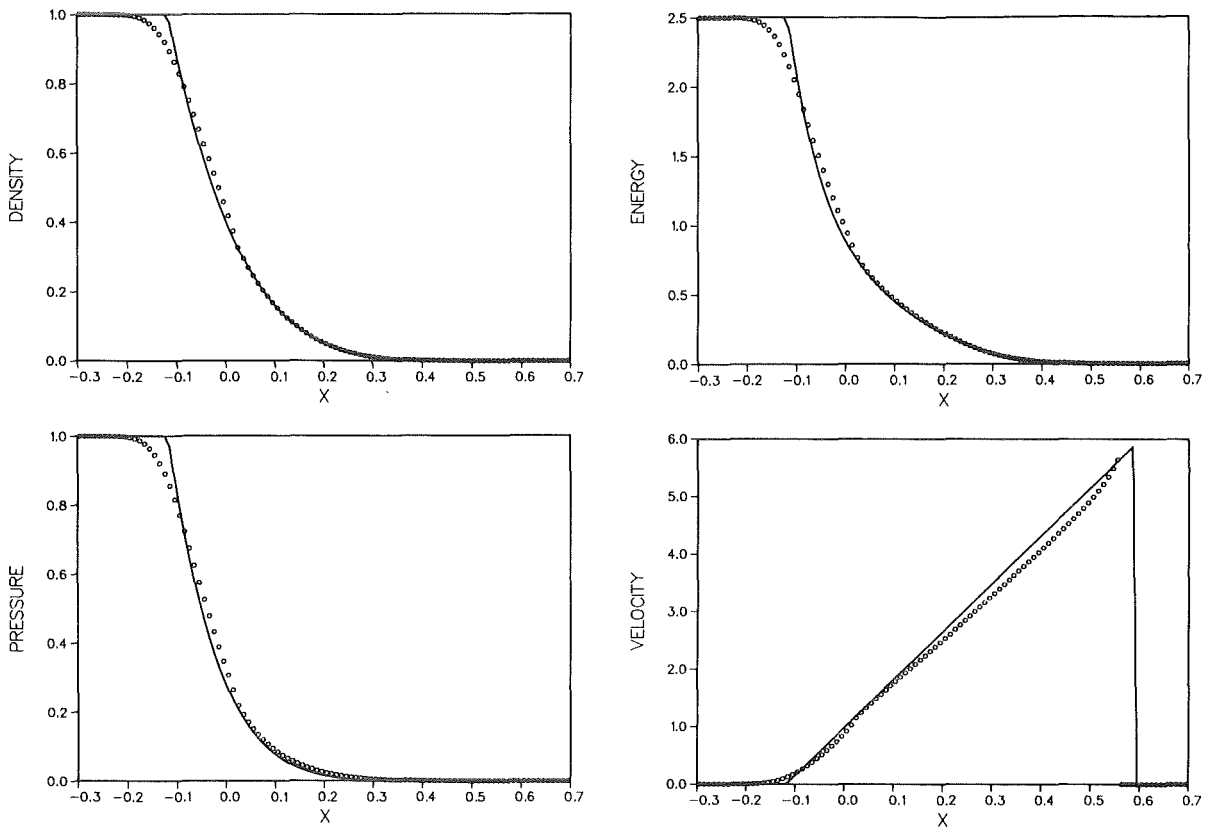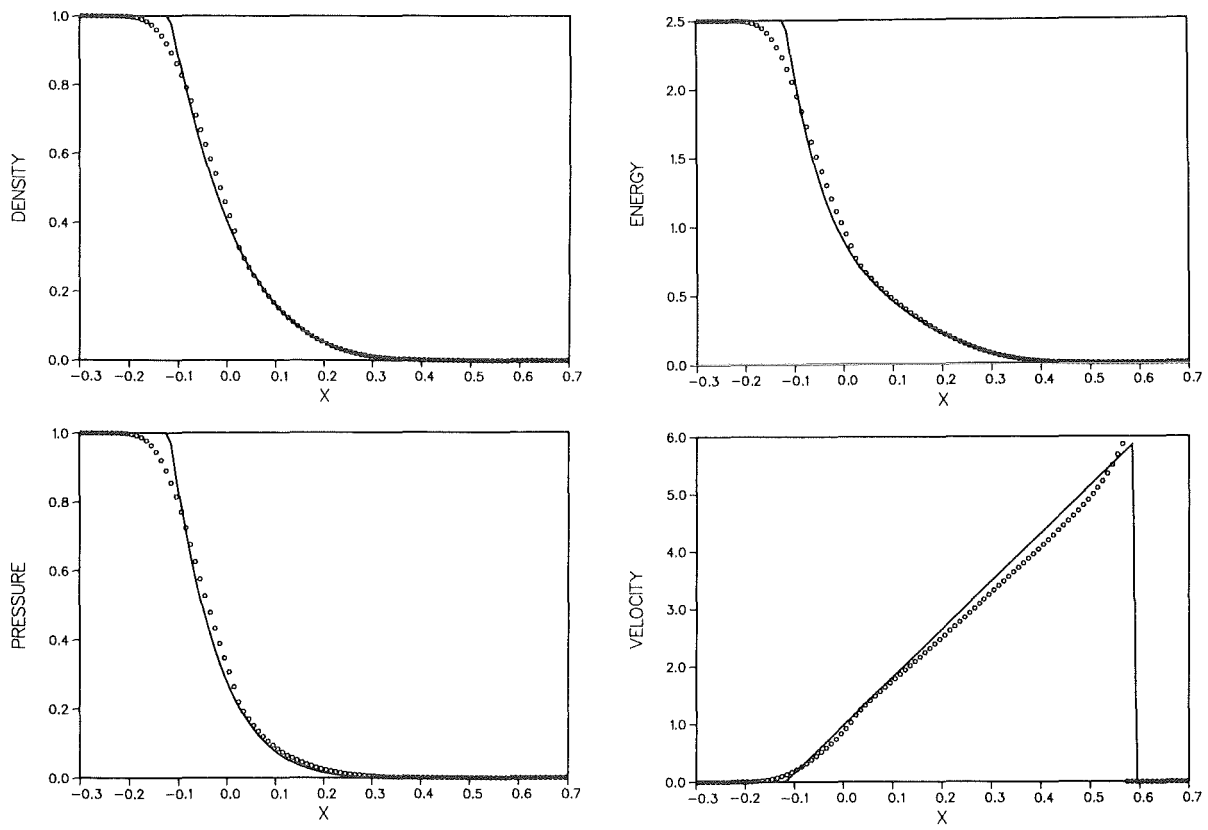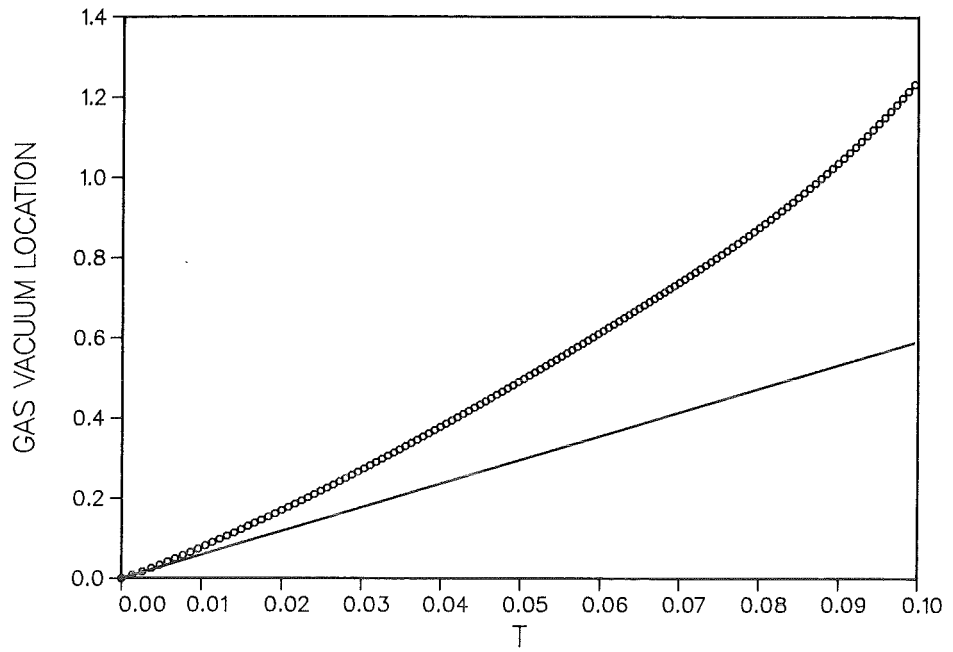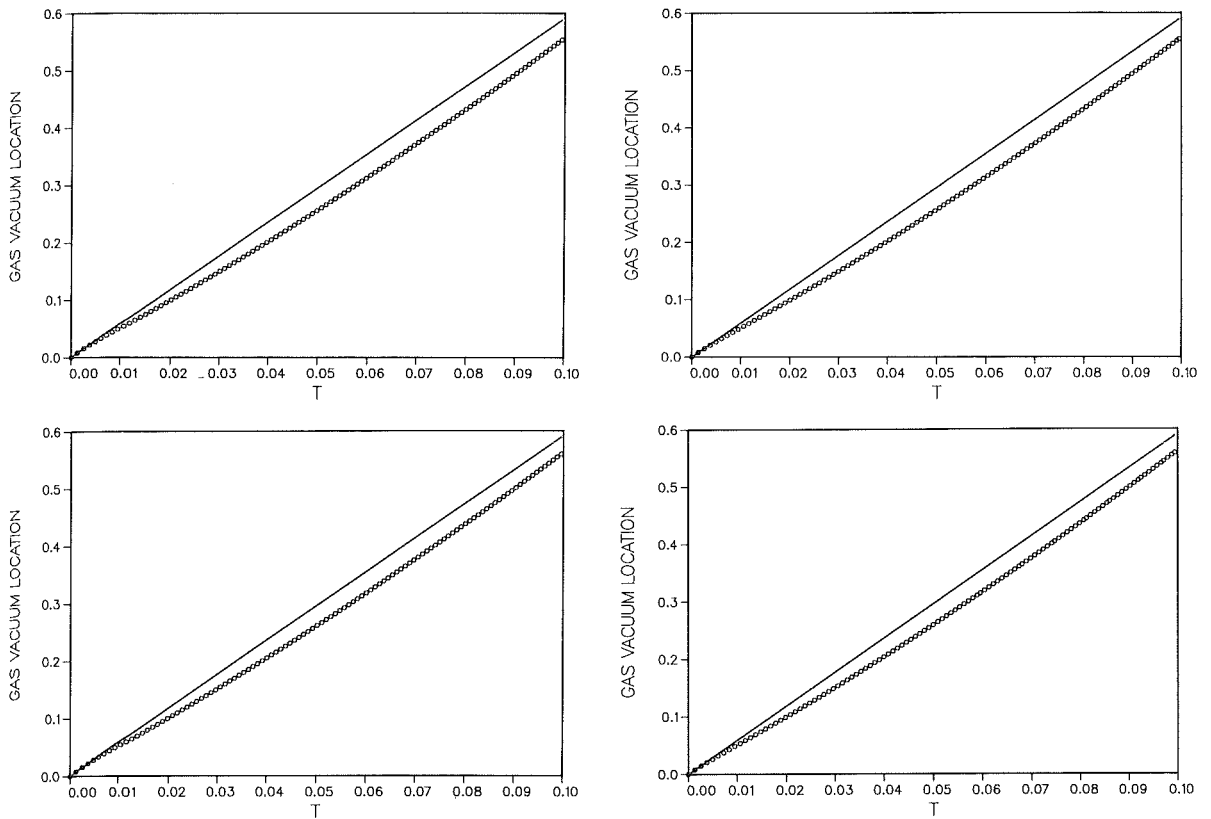
Figure 7a

Figure 7b

Figure 7c

Figure 7d

Figure 8

Figure 9