KfK 4353
Mai 1988

Numerical Simulations of Compressible Hydrodynamic Instabilities

C.-D. Munz, L. Schmidt
Institut für Neutronenphysik und Reaktortechnik

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Institut für Neutronenphysik und Reaktortechnik

KfK 4353

Numerical Simulations of Compressible Hydrodynamic Instabilities

C.-D. Munz
L. Schmidt[1)]

[1)] Institut für Angewandte Mathematik, Universität Karlsruhe

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

Abstract

This paper presents results for the numerical simulation of compressible hydrodynamic interface instabilities. The numerical results are produced by the direct simulation via numerical solution of the two-dimensional Euler equations. The numerical method considered here is a MUSCL-type scheme and belongs to the class of high resolution schemes. We study the instabilities of interfaces separating two domains of a fluid which move at different velocities, namely the Kelvin-Helmholtz instability, the Rayleigh-Taylor instability and the instability of jets. The interfaces are treated in a Lagrangean fashion according to the calculated Eulerian flow field.

Numerische Simulation kompressibler hydrodynamischer Instabilitäten

Kurzfassung

Diese Arbeit zeigt Ergebnisse der numerischen Simulation kompressibler hydrodynamischer Instabilitäten. Die numerischen Ergebnisse werden durch direkte Simulation mittels numerischer Lösung der zweidimensionalen Eulergleichungen erzielt. Das numerische Verfahren, welches hier angewendet wird, ist ein MUSCL-Typ Verfahren und gehört zur Klasse der "High Resolution Schemes". Instabilitäten von Grenzflächen zwischen Strömungen, welche mit verschiedenen Geschwindigkeiten strömen, werden untersucht, insbesondere die Kelvin-Helmholtz-, die Rayleigh-Taylor Instabilität und die Instabilität eines Überschallstrahls. Die Grenzflächen werden in Lagrangescher Weise entsprechend dem berechneten Eulerschen Strömungsfeld behandelt.

Table of Contents

1. Introduction

Interface instabilities arise in a wide variety of physical contexts: e.g., inertial laser fusion, instabilities of layers in stars, astrophysical jets, or foils accelerated by energy beams. In this paper, we examine the numerical simulation of the large-scale motion of these two-dimensional interfacial instabilities. We study the instability of interfaces separating two domains of the same compressible fluid which move at different velocities, namely the Kelvin-Helmholtz instability, the Rayleigh-Taylor instability, and the instability of supersonic jets. Our calculations are based on the two-dimensional equations of compressible fluid flow which can be written in the Lagrangean or in the Eulerian form. Numerical methods based on the Lagrangian formulation use a computational mesh traveling with the fluid. Hence, the Lagrangean methods are ideal for solving problems which involve interfaces between two fluids. However, two-dimensional Lagrangean calculations can typically be carried out for only small time spans. Then severe mesh distortion or mesh tangling will occur and will destroy the calculations. To continue the calculations rezoning must be performed in which all computational quantities are transferred to a new Lagrangean mesh. As this rezoning calls for much computational effort, a Lagrangean method does not seem to be favorable for large-scale computations. Eulerian methods, in which the mesh is fixed, are ideal for flows with large deformations. But interfaces are smeared out over some grid zones and the development of the interfaces can hardly be seen.

In this paper we use a combined method. The flow field is calculated by a Eulerian method, while the interfaces are moved in a Lagrangean fashion according to the Eulerian flow field. This means that we discretize the interface. In each time step we calculate at first the Eulerian flow field and then we calculate the new position of the discretized interface. To solve the Euler equations we use a shock-capturing finite difference scheme which is a so called high resolution scheme. The high resolution schemes are defined to have the following properties: They are at least second-order accurate on smooth parts of the flow, they produce sharp monotone discrete shock profiles without generating spurious oscillations, and they do not require artificial viscosity. The directions of the finite differences are upwind biased which means that they are locally defined according to the direction of the wave propagation. Here, we consider a high resolution scheme which is based on the MUSCL scheme of van Leer /18/. The treatment of the interfaces may be considered as a marker particle

algorithm. But this algorithm is used only to visualize the movement of the interface in the fluid flow. It may be also considered as the simplest method of front tracking, but we do not decompose our computational domain into two components separated by the interface as Glimm et al. /3/ do. Hence, our method is limited to the numerical simulation of instabilities in a homogeneous fluid.

The format of this paper is as follows. In Section 2 we shall briefly describe the mathematical model - the equations of compressible gas dynamics. In Section 3 we shall give the design principles of the numerical method we used. Section 4 is a description of the treatment and visualization of the interfaces. The numerical code is fully vectorized on the Cyber 205 vector computer. We shall add some remarks on the vectorization of the algorithm in Section 5. Section 6 contains a description of the interfacial instabilities considered here and the numerical results.

2. Euler Equations

We consider the two-dimensional equations of compressible fluid mechanics without thermal conduction and viscosity, written in the conservation form

$$U_t + f(U)_x + g(U)_y = 0 \qquad (2.1)$$

where U denotes the vector of the conserved variables and f, g denote the physical fluxes:

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ e \end{pmatrix}, \quad f(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e+p) \end{pmatrix}, \quad g(U) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e+p) \end{pmatrix} \qquad (2.2)$$

Here ρ denotes the density, u and v denote the velocity components in x and y direction, respectively, p denotes the pressure and e denotes the total energy per unit volume. Equations (2.1), (2.2) are derived from the integral conservation laws for density, momentum and energy. They are usually called Euler equations. The pressure p is functionally related to the other variables via the equation of state. In this paper we only consider that of an ideal gas

$$p = (\gamma - 1)(e - \frac{1}{2}\rho(u^2 + v^2)) \tag{2.3}$$

where γ denotes the adiabatic exponent.

3. Numerical Method

For the Euler equations (2.1), (2.2) it is favorable to use a scheme in conservation form which reproduces the integral conservation properties. This provides the proper propagation rates of shock waves. The numerical method considered here is based on dimensional splitting, also termed method of fractional steps (see e.g., /26/, /27/). According to this method the two-dimensional Euler equations (2.1), (2.2) are split into two one-dimensional problems

$$U_t + f(U)_x = 0 \tag{3.1}$$

$$U_t + g(U)_y = 0 \tag{3.2}$$

each of which containing either the x or y derivatives. These problems are then solved successively in each time step. In our calculations we use the two-cycle splitting method of Strang /26/ in which after each xy step the order is reversed for the following time interval: xy - yx. This version is of second-order accuracy as regards the time t. The systems (3.1), (3.2) resemble in structure the one-dimensional Euler equations and the numerical methods for these equations can be conveniently transferred to the systems (3.1), (3.2). Dimension splitting is often used to extend a one-dimensional method to two dimensions and seems to work very well (see /27/, /10/).

We will restrict ourselves now to a description of the numerical method used to treat (3.1); equation (3.2) can be treated in an analogous fashion. A one-dimensional explicit scheme in conservation form reads

$$U_i^{n+1} = U_i^n - \lambda(h_{i+1/2}^n - h_{i-1/2}^n), \qquad \lambda = \frac{\Delta t}{\Delta x} \tag{3.3}$$

where U_i stands for an approximation of the mean value of the solution U in the ith lattice interval at time t_n, Δx and Δt denote the increments. The function h is

called numerical flux: $h_{i+1/2}$ is an approximation of the physical flux f (U) between the ith and the (i + 1)th grid zone.

The one-dimensional schemes in conservation form, considered here, are based on van Leer's MUSCL-scheme /18/ which is a second-order version of Godunov's first order upwind scheme. We use the generalization of this scheme given in /16/. With this concept it is easy to convert every first order upwind scheme to second order accuracy in space and time. It is formulated in a two-step format:

In the first step, by means of interpolation, a piecewise linear representation of the approximate solution is calculated from the integral approximate values

$$U^n(x) = U_i^n + (x - x_i) S_i^n, \qquad x \in [x_{i-1/2}, x_{i+1/2}]. \tag{3.4}$$

The value S_i stands for the slope in the ith lattice interval. The boundary values of the ith lattice interval - U_{i+} on the right and U_{i-} on the left - are given by

$$U_{i\pm}^n = U_i^n \pm \frac{\Delta x}{2} S_i^n \tag{3.5}$$

(see Fig. 3.1).

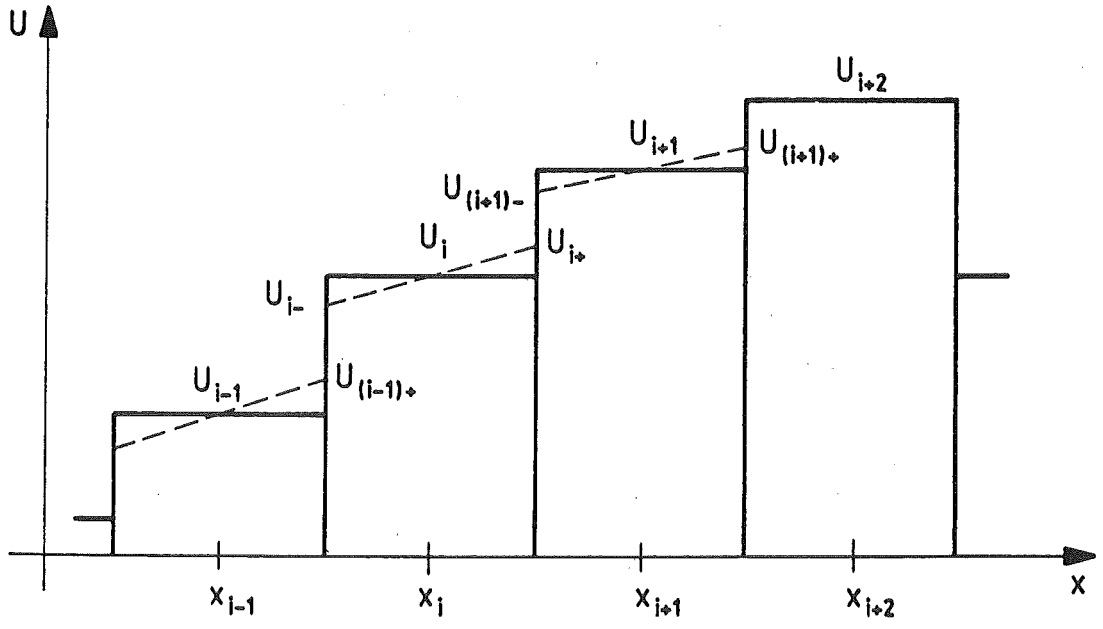


Fig. 3.1 Piecewise linear distribution of the approximate solution at time t_n

In order to obtain second-order accuracy with respect to time a midpoint rule is used: the boundary values (3.5) are advanced to $t_{n+1/2}$

$$U_{i\pm}^{n+1/2} = U_{i\pm}^{n} - \frac{\lambda}{2}\,(f(U_{i+}^{n}) - f(U_{i-}^{n})). \tag{3.6}$$

In the second step, an upwind scheme is applied to these data

$$U_{i}^{n+1} = U_{i}^{n} - \lambda\,(h\,(U_{i+}^{n+1/2},\ U_{(i+1)-}^{n+1/2}) - h\,(U_{(i-1)+}^{n+1/2},\ U_{i-}^{n+1/2})) \tag{3.7}$$

where h denotes the numerical flux of the upwind scheme.

The upwind scheme in the second step picks out the proper direction of the differences. Any first-order upwind method can be used for this purpose. A survey of various upwind methods is contained in /13/. The slope S must satisfy a number of conditions. A first necessary condition for second-order accuracy in space says that S_i is a first-order approximation of U_x at (x_i, t_n). In order to avoid oscillations at strong gradients the piecewise linear representation must satisfy some monotonicity constraints. There are several ways of calculating the slopes. One method, used in the MUSCL-scheme of van Leer /18/ or in the PPM- or PLM-scheme of Colella and Woodward /6/ or Colella and Glaz /5/, is to compute slopes in terms of the primitive variables ρ, u, v, p. Another possibility is to use the conservative variables. We compute the slopes in terms of characteristic variables and use the scalar theory of slope calculation. In the case of a scalar equation of conservation laws or in the case of a system with constant coefficients new extrema should not be introduced and the total variation should not increase. For a scalar conservation law various suitable calculations of slopes have been indicated and analyzed (see /20/). We are using this scalar theory in our two-step methods by transferring the scalar slopes to the systems (3.1), (3.2) with an extension method based on Roe's /22/ and Huang's /14/ method.

This method relies on local linearization of the nonlinear system which defines a local system of characteristic fields. The linearized system can be converted to the characteristic form where the individual equations are decoupled. Then the scalar slope calculation is applied scalarly to each of the characteristic equations. In each grid zone an average value $\overline{U}_i$ is determined; for instance, by $\overline{U}_i = (U_{i+1} + 2U_i + U_{i-1})$ or also $\overline{U}_i = U_i$. Here and in the following studies the time index n is omitted as long as no misunderstandings can arise. The vector r_i^k denotes the k-

th right eigenvector of the Jakobi matrix evaluated at U_i. The left hand side and the right hand side difference quotients are then expanded in terms of this system of right eigenvectors

$$\frac{1}{\Delta x}(U_{i+1} - U_i) = \Sigma_{k=1}^{4} \alpha_i^k r_i^k \ ,$$

$$\frac{1}{\Delta x}(U_i - U_{i-1}) = \Sigma_{k=1}^{4} \beta_i^k r_i^k \ . \qquad (3.8)$$

The coefficients α^k, β^k measure the change of the difference quotients in direction of the kth eigenvector. A suitable vector of slopes S_i is obtained by applying a scalar slope calculation to these coefficients. By using Roe's minmod-function we obtain, e.g.

$$S_i = \Sigma_{k=1}^{4} \text{minmod}(\alpha_i^k, \beta_i^k) r_i^k \qquad (3.9)$$

where the minmod-function is given by

$$\text{minmod}(a,b) = \begin{cases} a \ \text{for} \ |a| \leq |b|, \ ab > 0 \\ b \ \text{for} \ |a| > |b|, \ ab > 0 \\ 0 \ \text{for} \ ab < 0 \ . \end{cases} \qquad (3.10)$$

A survey of other suitable choices of the scalar slope calculation is given in /20/, /21/. The main advantage of the slope calculation in terms of characteristic variables is that different slope calculations may be applied to the genuinely nonlinear characteristic fields and to the linearly degenerate fields. Especially in large scale computations a severe problem is that numerical damping of a contact discontinuity increases in time in contrast to shock waves (see e.g. /12/). This will imply - for a classical second order accurate scheme like Lax-Wendroff scheme /15/ - those contact discontinuities vanish for large time spans and the asymptotic numerical solution for $t \to \infty$, if existent, does not possess any contact discontinuity. Within a high resolution scheme this numerical damping may be reduced or prevented by using a very compressive slope on the linearly degenerate field. Slope calculations can be used which give the best results for linear advection problems. A very compressive slope has been proposed by Roe:

$$\text{maxmod}(a,b) = \text{sign}(a) \max\{|\text{minmod}(2a,b)|, |\text{minmod}(a,2b)|\} \qquad (3.11)$$

which is called superbee-function. This slope calculation yields a very narrow transition zone for a linear or contact discontinuity. However, the superbee-function exhibits a slightly over-compressive behavior and tends to compress all monotone transitions into discontinuities. In our calculations we use a class of slopes, proposed by Sweby, in terms of schemes using flux limiters which in terms of a parameter l provides a continuous transition between the minmod and superbee-function

$$s_l(a,b) = \mathbf{sign}(a)\,\mathbf{max}\{|\mathbf{minmod}(la,b)|, |\mathbf{minmod}(a,lb)|\} \qquad (3.12)$$

where l is in the range $1 \leqq l \leqq 2$. The slope s_1 equals the minmod-function, the most compressive s_2 equals the superbee-function. Another possibility to reduce the numerical dissipation of the contact discontinuities is to additionally apply Harten's /12/ artificial compression method in a second step on the linear degenerate field. This method deepens the contact discontinuity. Very compressive slopes or the artificial compression method should not be applied to the genuinely nonlinear fields. Because they may be over-compressive and may compress each monotone profile into a discontinuity, they may introduce at centered rarefaction waves non-physical discontinuities which are usually called rarefaction or expansion shocks and which violate the entropy condition. If the slopes are calculated in terms of the primitive or conservative variables, the different waves cannot be treated in a different fashion. Hence, less compressive slopes must be used or near sonic points a correction mechanism must be added which switches to a less compressive slope or which increases the entropy (see /21/). Another advantage of the characteristic slope calculation is that new ideas for the scalar conservation law can be easily extended to systems.

As an explicit method the one-dimensional schemes have to satisfy a stability condition, usually called the CFL - condition according to Courant, Friedrichs and Lewy,

$$\frac{\Delta t}{\Delta x}\,\mathbf{max}(|u| + c) \leq 1 \qquad (3.13)$$

which gives a restriction on the time increments. In the calculations presented here we used within the two-step algorithm the first-order upwind schemes of van Leer /17/ and of Einfeldt /9/ which is based on the work of Harten, Lax and van Leer /13/. In terms of the computational effort these schemes seem to be the best. In Chapter 5 we will give a comparison of the computer times we needed.

4. Visualization of Interfaces

At the beginning of the calculations the surfaces between the fluids are discretized and the surfaces are replaced by a number of discrete points. In the following we will term these points marker particles. In each time step, at first the new flow field is calculated by the Eulerian two-step method and then the massless marker particles are advected in a Lagrangean fashion according to the local flow field. The movement of the interface can then be visualized by graphic display of these marker particles. The particles are overlaid on the fixed computational grid and are advected without any collisional effect between them. This algorithm will be described in detail below.

By discretization of an interface we obtain a set of marker particles. At time t_n the kth particle is locatd at a point (x_k, y_k) of the computational domain and possesses the velocity (u_k, v_k). After calculation of the flow field at time t_{n+1} by our Eulerian method the new locations of the marker particles are calculated by using

$$x_k^{n+1} = x_k^n + \frac{\Delta t}{2}(u_k^{n+1} + u_k^n)$$

$$y_k^{n+1} = y_k^n + \frac{\Delta t}{2}(v_k^{n+1} + v_k^n) \qquad (4.1)\,.$$

The velocities u_k , v_k in (4.1) are determined by bilinear area weighting interpolation. At first we need the location of the particle according to the computational grid. Here we only need the location of the lower left grid zone near the particle. We will call this grid zone $\Gamma_{i,j}$; i,j is given by the formula

$$i: = \mathrm{int}\left(\frac{x_k^n - x_{le}}{\Delta x}\right) \qquad j: = \mathrm{int}\left(\frac{y_k^n - y_{lo}}{\Delta y}\right) \qquad (4.2)$$

where x_{le}, y_{lo} denote the position of the left and lower boundary of the computational grid, respectively. Next we calculate the areas A_1, A_2, A_3, A_4 in reference to Fig. 4.1 which are given as

$$A_1 = (x_{i+1} - x_k^n)(y_{j+1} - y_k^n)$$

$$A_2 = (x_k^n - x_i)(y_{j+1} - y_k^n) \qquad (4.3)$$

$$A_3 = (x_k^n - x_i)(y_k^n - y_j)$$

$$A_4 = (x_{i+1} - x_k^n)(y_k^n - y_j).$$

The values without superscripts refer to the fixed grid quantities. The velocity of the k-th marker particle at time t_{n+1} is then determined from the quantities of the Eulerian flow field by the interpolation formula

$$u_k^{n+1} = B_1 u_{i,j}^{n+1} + B_2 u_{i+1,j}^{n+1} + B_3 u_{i+1,j+1}^{n+1} + B_4 u_{i,j+1}^{n+1}$$

$$v_k^{n+1} = B_1 v_{i,j}^{n+1} + B_2 v_{i+1,j}^{n+1} + B_3 v_{i+1,j+1}^{n+1} + B_4 v_{i,j+1}^{n+1} \qquad (4.4)$$

where B_l denotes the area ratios $B_l = A_l / (\Delta x \, \Delta y)$, $l = 1, \ldots, 4$ and the values i,j are given by (4.2).

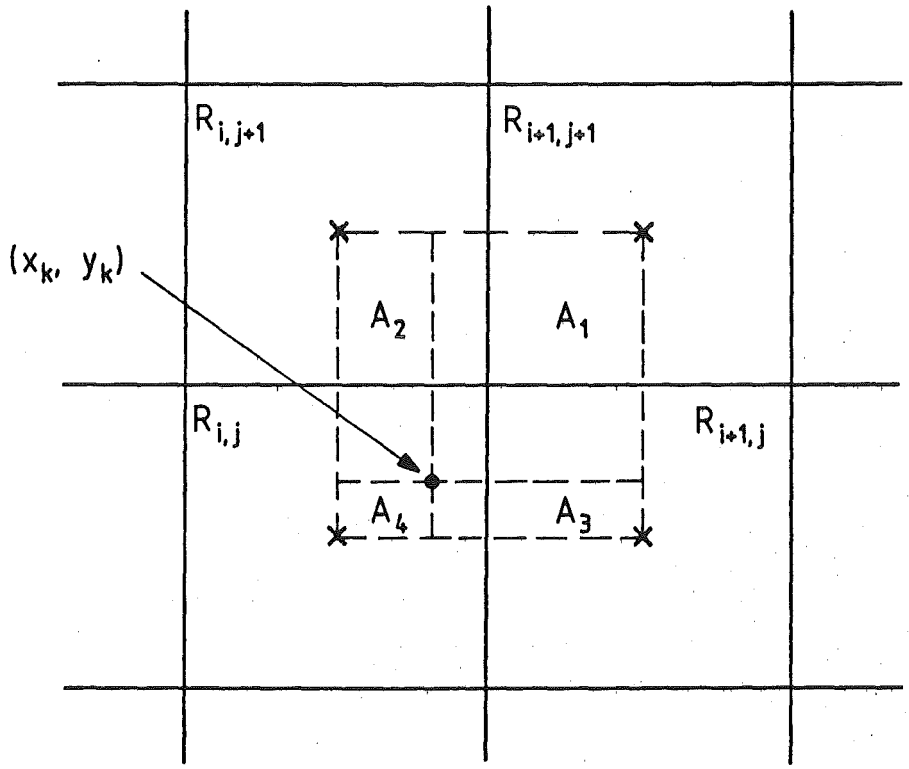


Fig. 4.1: Area weighting interpolation

At the boundary some modifications are needed according to the physical boundary conditions. For instance, in the case of periodic boundary conditions it is favorable that a particle which leaves the computational domain reappears at the opposite boundary. In this paper we only consider a rectangular computational domain and it is very easy to determine the location of a particle as given in (4.2). For general domains and grids this is more difficult and necessitates much more computational effort.

5. Vectorization

The one-dimensional two-step algorithm (3.5) - (3.7) is an explicit algorithm and can be efficiently implemented on a vector computer. But, if the upwind scheme of Godunov is used in the second step to determine the exact solution of the Riemann problem, a fixed point problem has to be solved at each grid point by an iteration scheme. This iteration scheme introduces some difficulties for an efficient vectorization, because it is a recursive process. The approximate Riemann solvers or the flux-vector splitting schemes do not contain fixed point iterations and they can be vectorized in a straightforward manner. For the two-dimensional algorithms some other difficulties arise. To vector computers and especially to the Cyber 205 for which our numerical code is optimized, it is very important that the data used in the calculations are contiguously stored vectors. For the x-step of the splitting algorithm the first index of the two-dimensional variables can be chosen to be the index of the inner loops. Then the two-dimensional calculations can be replaced by calculations with contiguously stored long one-dimensional arrays. In the y-step this situation changes. To obtain contiguously stored vectors the physical variables have to be transposed. After the transposition, the index of the inner loops can be chosen to be again the first index of the two-dimensional arrays, and the two-dimensional calculation can be replaced by calculations with contiguously stored vectors. For Strang-type splitting one transposition of the physical variables per time step has to be performed. For this reason it turned out that Strang-type splitting is also favourable for the vectorization of two-dimensional algorithms.

Within the two-step algorithm we used several upwind schemes in the second step. The fastest schemes turned out to be the Godunov-type scheme of Harten, Lax and van Leer /13/. The scheme of Harten, Lax and van Leer has been implemented as proposed in /9/. The different computer times we needed in the

case of 100 x 100 grid zones are listed in Table 5.1. All calculations have been done on the Cyber 205 vector computer. The values in Table 5.1 do not give a true comparison, but show only the trend. In an early stage of our optimization for vector computers we recognized that the flux-vector splitting scheme of van Leer seemed to be the fastest. Hence, we made more effort to optimize this scheme. Then, because of its simplicity the upwind scheme of Harten, Lax and van Leer seemed to be able to compete with this scheme. As the schemes of Steger-Warming and Roe call for more computational effort we did not invest the same time for their optimization. We think that by further optimization of our codes the computer times may be reduced for these schemes by about 10 - 20 percent. The two-step algorithm consumes additionally about 0.008 seconds per time step, if the slope calculation is performed in terms of primitive variables and using (3.10). If the slope calculation is performed by (3.12) in terms of characteristic values the algorithm add to the computer times of the upwind schemes 0.12 per time tep. In Table 5.2 we listed the computer times for a grid with 200 grid points in each space direction.

Upwind scheme	Computer times
Harten, Lax and van Leer /13/	0.073 s per time step
van Leer /17/	0.073 s per time step
Steger-Warming /24/	0.09 s per time step
Roe /22/	0.12 s per time step

Table 5.1: Computer times for different upwind schemes on Cyber 205

Upwind scheme	Slope calculation in terms of Primitive variables	Characteristic variables
Harten, Lax, van Leer	0.265 s	0.393 s
van Leer	0.248 s	0.376 s

Table 5.2: Computer times on Cyber 205 for two-step schemes using 200 x 200 grid zones per time step

In the table above which shows the computer times we did not take into consideration the marker particle algorithm to visualize the movement of the interfaces. Since we only consider rectangular grids and uniform step sizes, it is very easy to determine the location of a particle. The movement of 2000 marker particles in each time step according to the Eulerian flow field only needs 0.003 s of computer time. For general domains and grids the calculation of the position of the particles becomes more difficult and requires more computational effort. A fast vectorizable algorithm has been proposed recently by Seldner and Westermann /23/.

In the case of one-dimensional calculations the numerical results of two-step schemes based on different upwind schemes turned out to be quite similar (see /21/). This is also valid for our two-dimensional calculations. We will present in this paper only numerical results of our fastest two-step scheme. This is based on van Leer's flux vector splitting scheme and on the slope calculation in terms of the primitive variables with the minmod-function of Roe. A comparison of numerical results using different upwind schemes and slope calculations will be contained in a further paper.

6. Interfacial Instabilities and Numerical Results

In this chapter we will apply a two-step scheme to simulate two-dimensional compressible interfacial instabilities. We will consider instabilities of interfaces which separate two domains of the same fluid which move at different velocities. These instabilities were recognized and formulated notably by Helmholtz, Kelvin, Rayleigh and Taylor. A review of two-dimensional hydrodynamic instabilities and their mathematical description have been given by Birkhoff /1/. The mathematical description can be given only approximatively. An important method is the linear theory based on the concept of a normal mode (see /1/). But the linear theory is limited to short time spans and small initial perturbations. For longer time spans nonlinear effects become important and a linear theory cannot give satisfactory results. Furthermore the linear theory is only favorable for an incompressible fluid, it becomes very complicated in the case of a compressible fluid. Another important approximation - also limited to an incompressible fluid - is the vortex method (see /1/, /19/). This numerical technique is based on the representation of an interface by a vortex sheet, i.e. a surface across which the fluid velocity has a continuous normal component, but a

discontinuous tangential component. The jump measures the strength of the sheet. The vortex sheet is replaced by a suitable distribution of discrete vortices. This leads to an autonomous system of ordinary differential equations which have to be solved numerically. By vortex methods the nonlinear development of the interface can be approximated, but the methods run into difficulty for large scales (see /1/, /19/). Especially, vortex methods are unreliable to study e.g., vortex sheet rollup and the development of a Karman vortex street.

The technique adopted here is the direct numerical solution of the equations of the fluid flow. By that it is possible to perform large scale computations. We will show that if fully vectorized fast numerical methods are used the direct simulation may not be very costly in terms of computer time. We will study interfacial instabilities which are purely inertial phenomena. We adopted the three basic examples of hydrodynamic instabilities: the Kelvin-Helmholtz instability, the Rayleigh-Taylor instability and the instability of a jet as described by Birkhoff (/1/, Figure 1 - 3). We will present a series of computations with a two-step scheme based on the slope calculation (3.9) with the minmod-function. Applied to a sinusoidally perturbed vortex sheet for a homogeneous fluid the numerical results show the roll-up of the sheet into spirals. If another - more compressive - slope calculation is used, the amount of numerical dissipation and viscosity which is inherent in the scheme is smaller. Besides the movement of the vortex sheet towards the rollup small amplitude waves will additionally arise. These higher frequency perturbations disturb the rollup of the vortex sheets into spirals. The small perturbations might be introduced by the errors of the approximation of the initial values on the Cartesian grid. We may also conclude that the rollup of a perturbed vortex sheet corresponding to a local concentration of vorticity will only arise, if the fluid possesses a small amount of viscosity. This result agrees with the statements of Birkhoff /1/, /2/ for the case of incompressibility. But there are several uncertainties which have to be carefully studied. In this paper we will restrict ourselves to show the numerical results of our calculations. They demonstrate that the high resolution schemes combined with the Lagrangean tracking of the interfaces give a very efficient method to study hydrodynamic instabilities. Further investigations will be performed to clarify the role of viscosity and to obtain better insight in the development of the instabilities. Numerical results with viscosity terms and finer grids have to be examined.

At first we apply the two-step schemes to study the time evolution of the compressible Kelvin-Helmholtz instability. The initial values for our problem are shown in Fig. 6.1. The density and pressure are equal to 1.0 in the whole computational domain. The velocity in x-direction in the upper part is equal to -0.5, while in the lower part it is equal to 0.5, the velocity in y-direction v is zero everywhere. The two fluids are separated by a surface S represented in the form

$$S: \quad y = a \sin(2\pi k x) \tag{6.1}$$

where a denotes the amplitude and k the mode number corresponding to a sinusoidal perturbation of the shear layer y = 0. At the boundaries of our computational domain R = [-0.5, 0.5] x [-0.5, 0.5] we impose the following boundary conditions: At the flow entrance and exit, corresponding to the right side and left side in Fig. 6.1, we impose periodic conditions, at the top and the

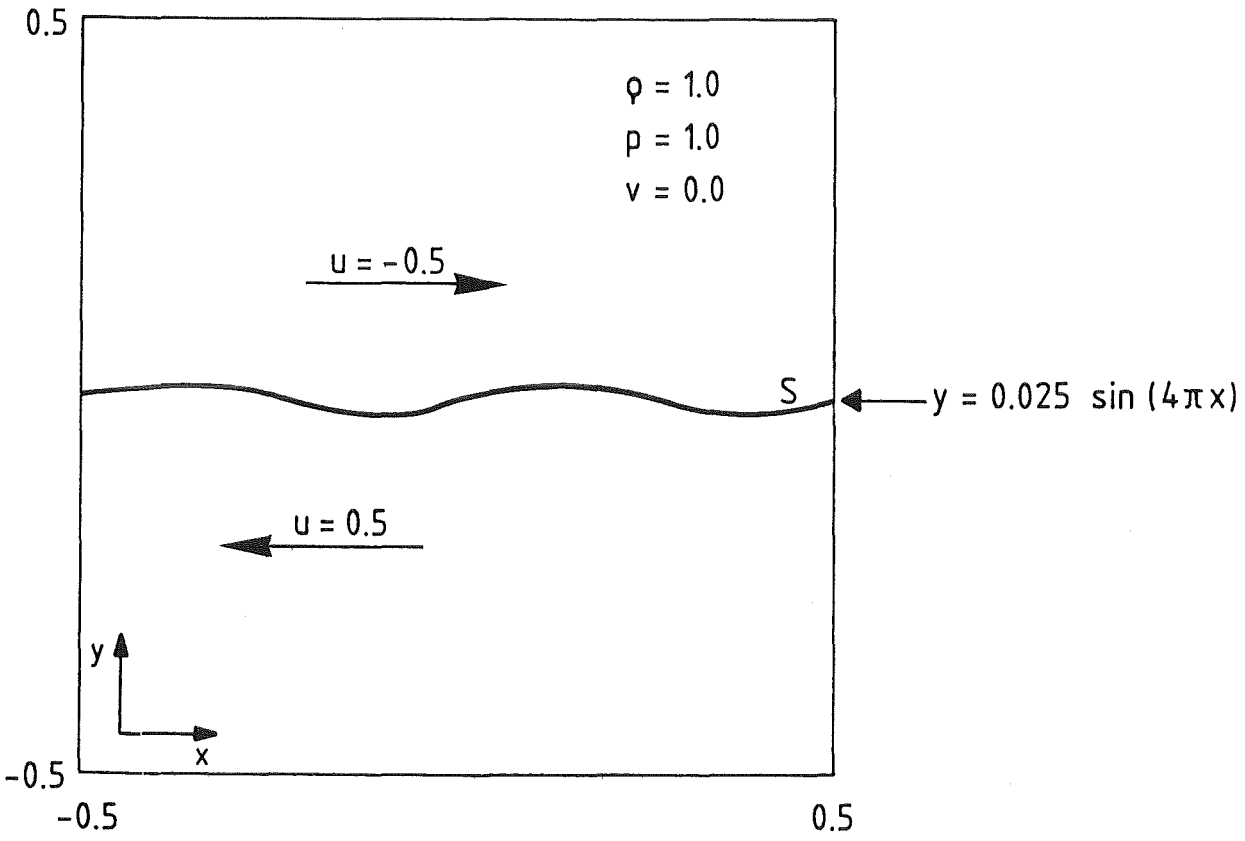
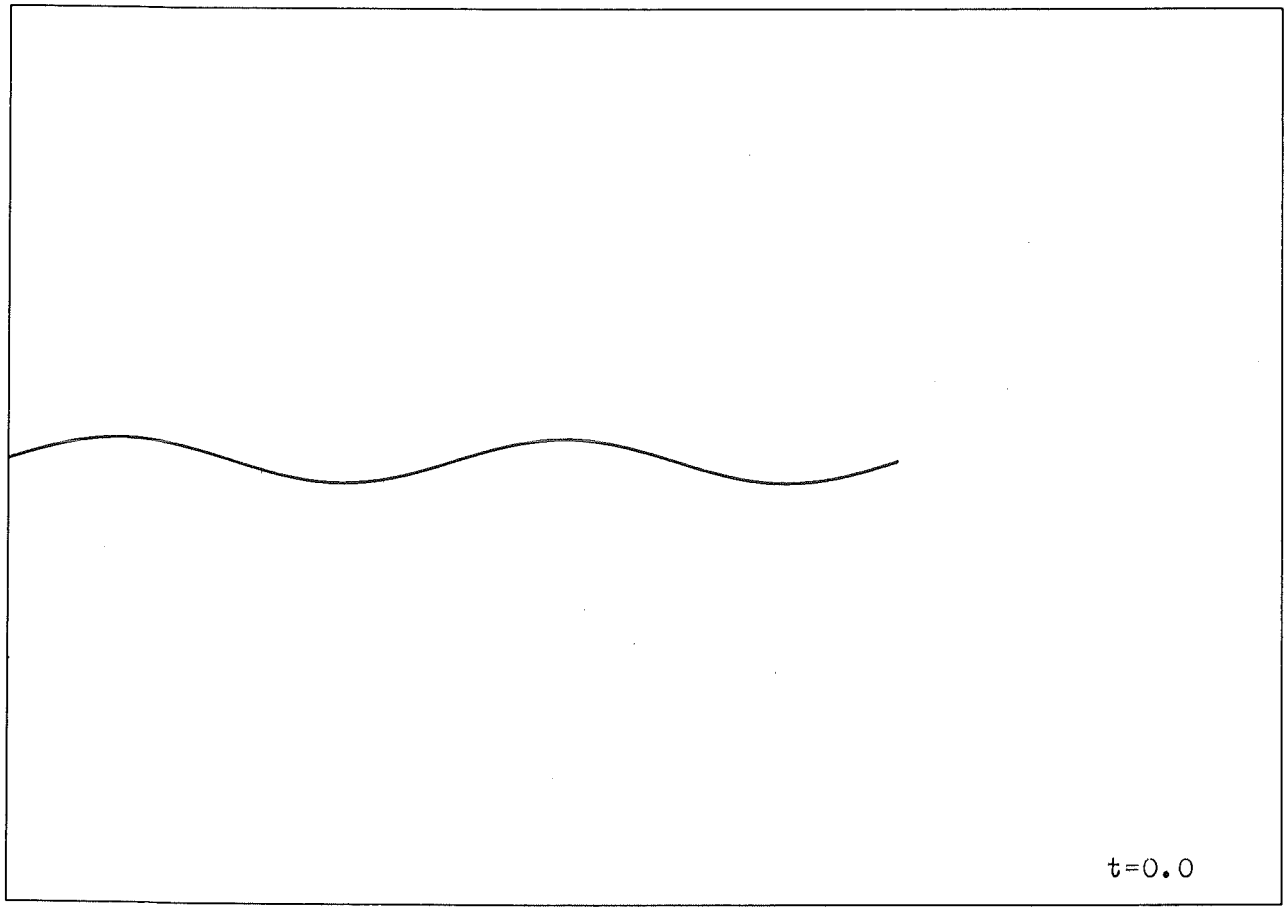


Fig. 6.1: Initial values of a Kelvin-Helmholtz instability

bottom we impose conditions of a reflecting wall. For the numerical calculations we used a uniform grid with 100 x 100 grid zones corresponding to step sizes $\Delta x = 0.01$, $\Delta y = 0.01$. The time increment is at each time step adaptively chosen according to the CFL-condition (3.1).

The marker particles are placed on the interface S. The development of this interface can be seen by displaying the marker particle field. We used 2000 marker particles. The initial perturbation (6.1) is of mode 2 and the amplitude is $a = 0.025$. As initial data for the numerical scheme we prescribed in each grid zone approximative mean values. The profile of the vortex sheet at 10 points in time are shown in Figures 6.2 - 6.6. In Fig. 6.2 the initial shape and the slope at time $t = 0.4$ of the sheet are plotted. The sheet differs from the sinusoidal profile as predicted by the linear theory. At $x = -0.25, 0.25$ the sheet becomes vertical and starts to roll up. This rollup into a pair of spirals becomes very obvious from results at time $t = 0.8$ (Fig. 6.3). The vortex sheet tends to roll-up into local concentrations of vorticity. The next figures indicate that this roll-up will continue. The increasing rate seems to be constant; during a time step $\Delta t = 0.4$ the roll-up precedes half a rotation. Experimental result for the incompressible case can be found in /7/ (page 85). They are in good agreement with our numerical results for the compressible case.

If the amplitude of the sinusoidal perturbation (6.1) becomes larger, the movement of the vortex sheet will change. A diagram of initial data with large amplitude perturbations is plotted in Fig. 6.7. The computational region agrees with that of our last problem, the amplitude of the sinusoidal perturbation (6.1) is now $a = 0.1$. The self-induced motion of this vortex sheet differs strongly from the motion of the vortex sheet with a small amplitude. The numerical results are plotted in Figures 6.8 - 6.11. In contrast to the results for the Kelvin-Helmholtz instability where the rollup process starts, the results at the times $t = 0.2, 0.4$ are quite similar to those obtained by Birkhoff (Figure 8.6 in /1/) for an incompressible fluid. He used a point vortex method. Our next example is the numerical simulation of the Rayleigh-Taylor instability. This instability may occur when two superposed fluids of different densities are accelerated in a direction perpendicular to their interface. If the acceleration is directed from the lighter to the heavier fluid, the interface is unstable. A mathematical theory can also be given for the case of an incompressible fluid only, with small amplitude perturbations of simple types of motion. In the present investigation we consider initial values as given in Fig. 6.12. A fluid with the density $\rho = 1.0$ is superposed



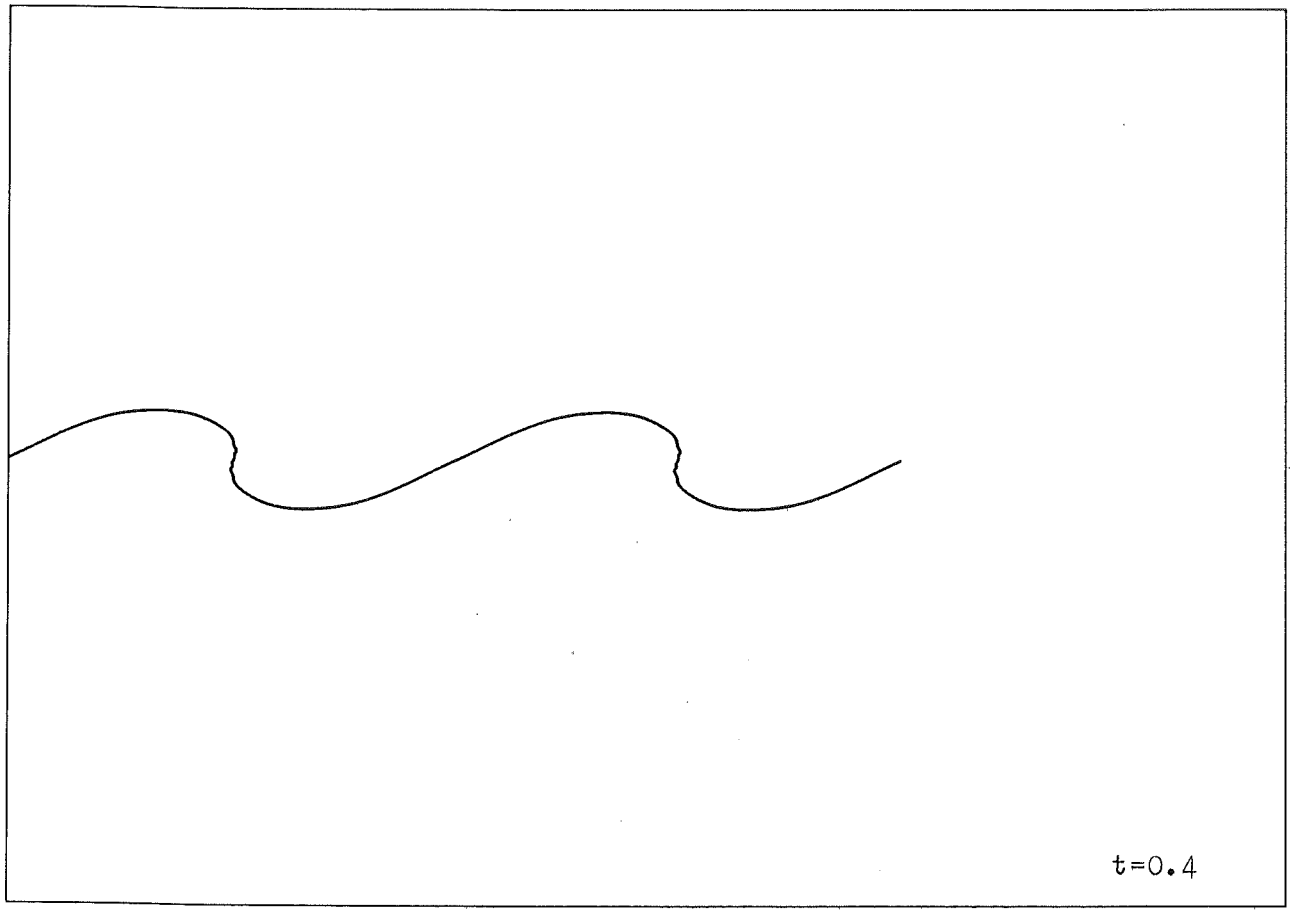
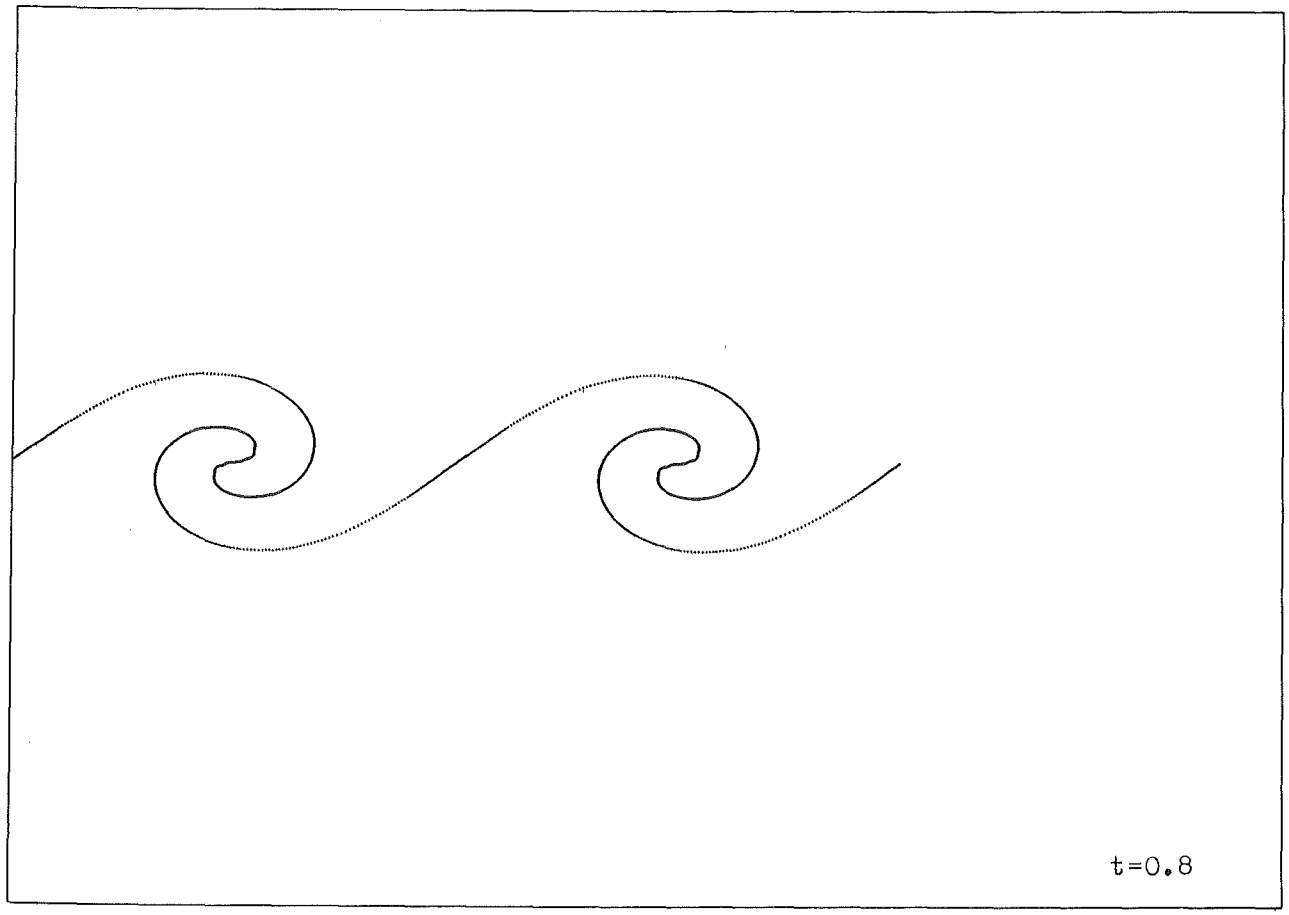


Fig. 6.2: Numerical simulation of a Kelvin-Helmholtz instability at time t = 0.0, 0.4



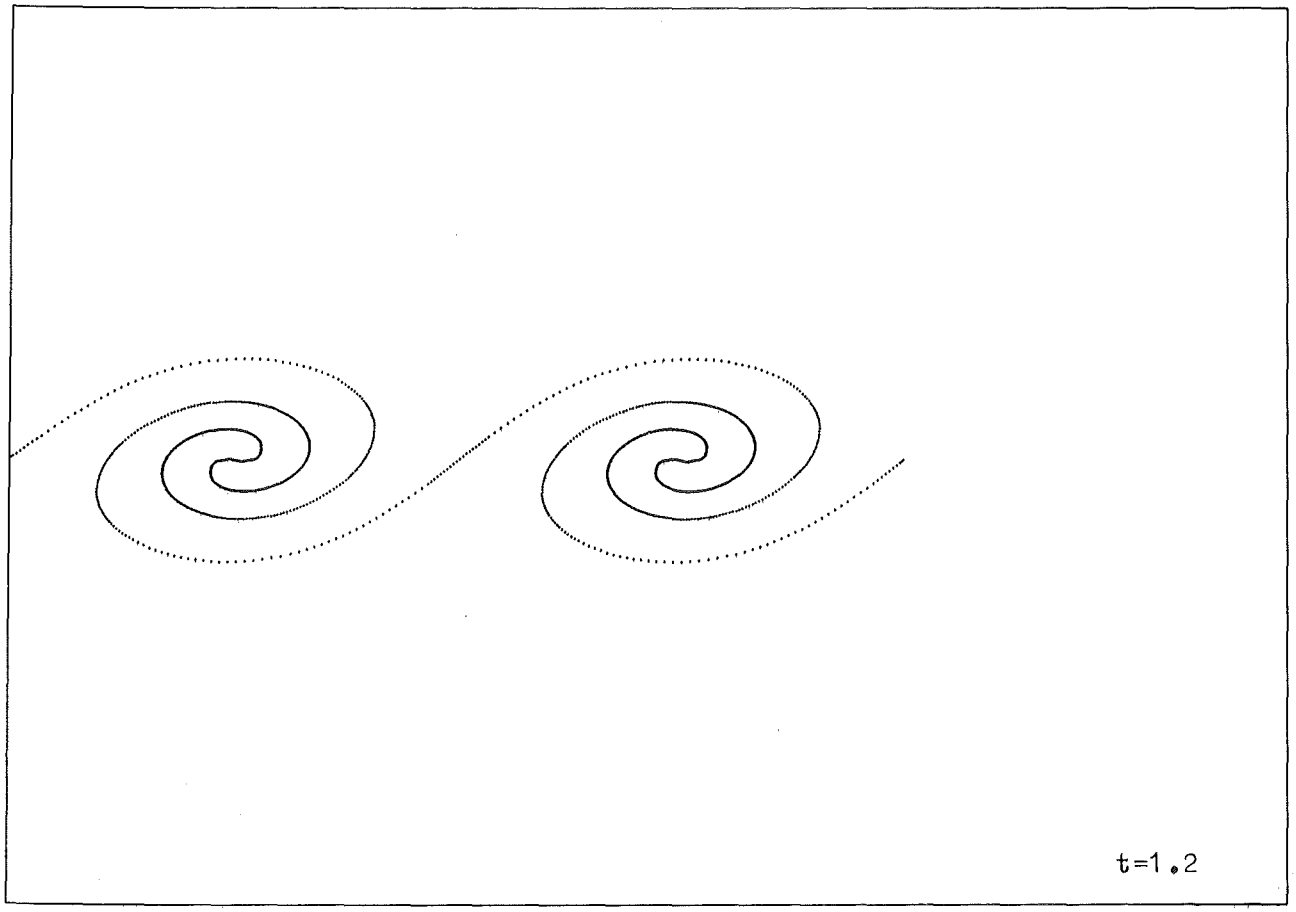
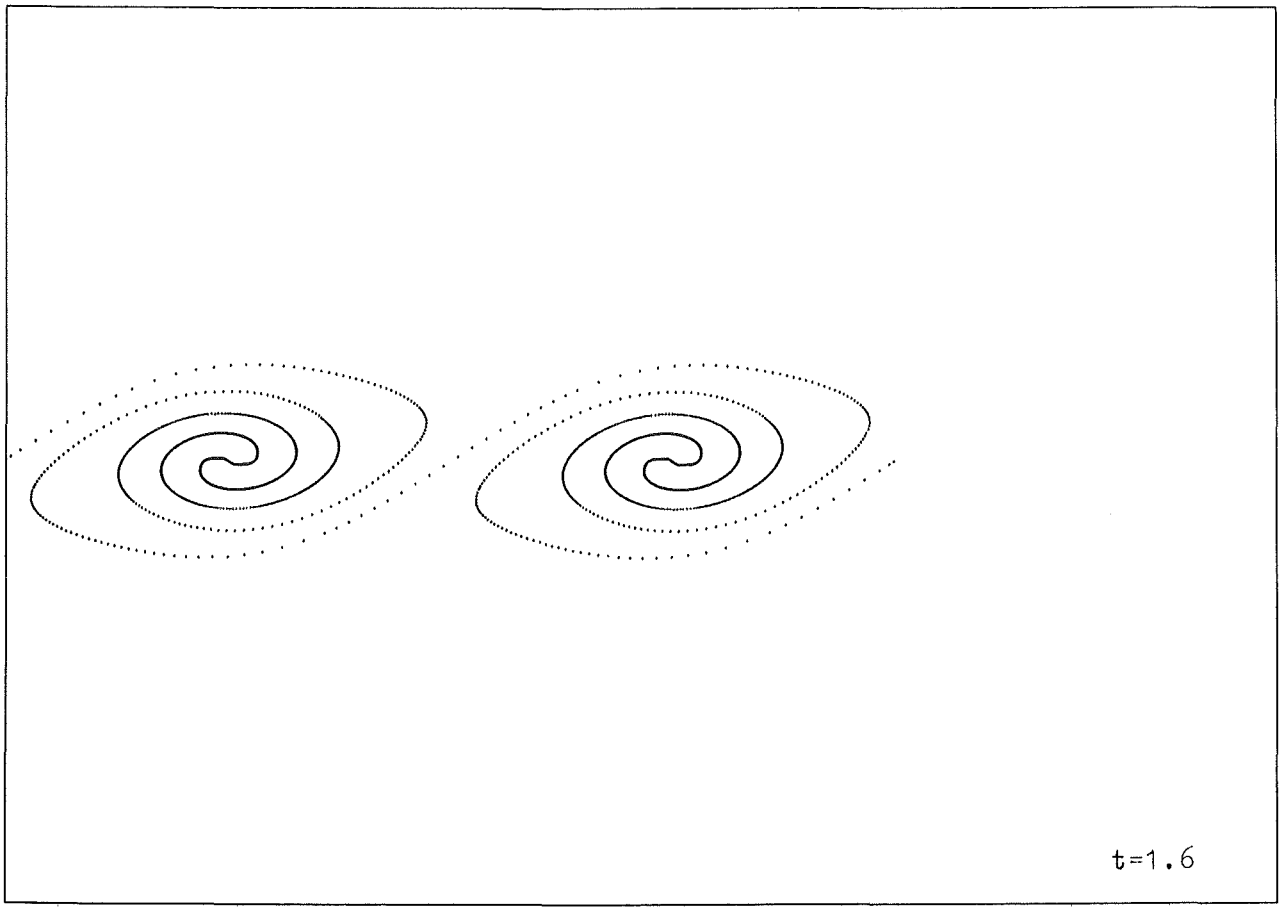


Fig. 6.3: Numerical simulation of a Kelvin-Helmholtz instability at time t = 0.8, 1.2



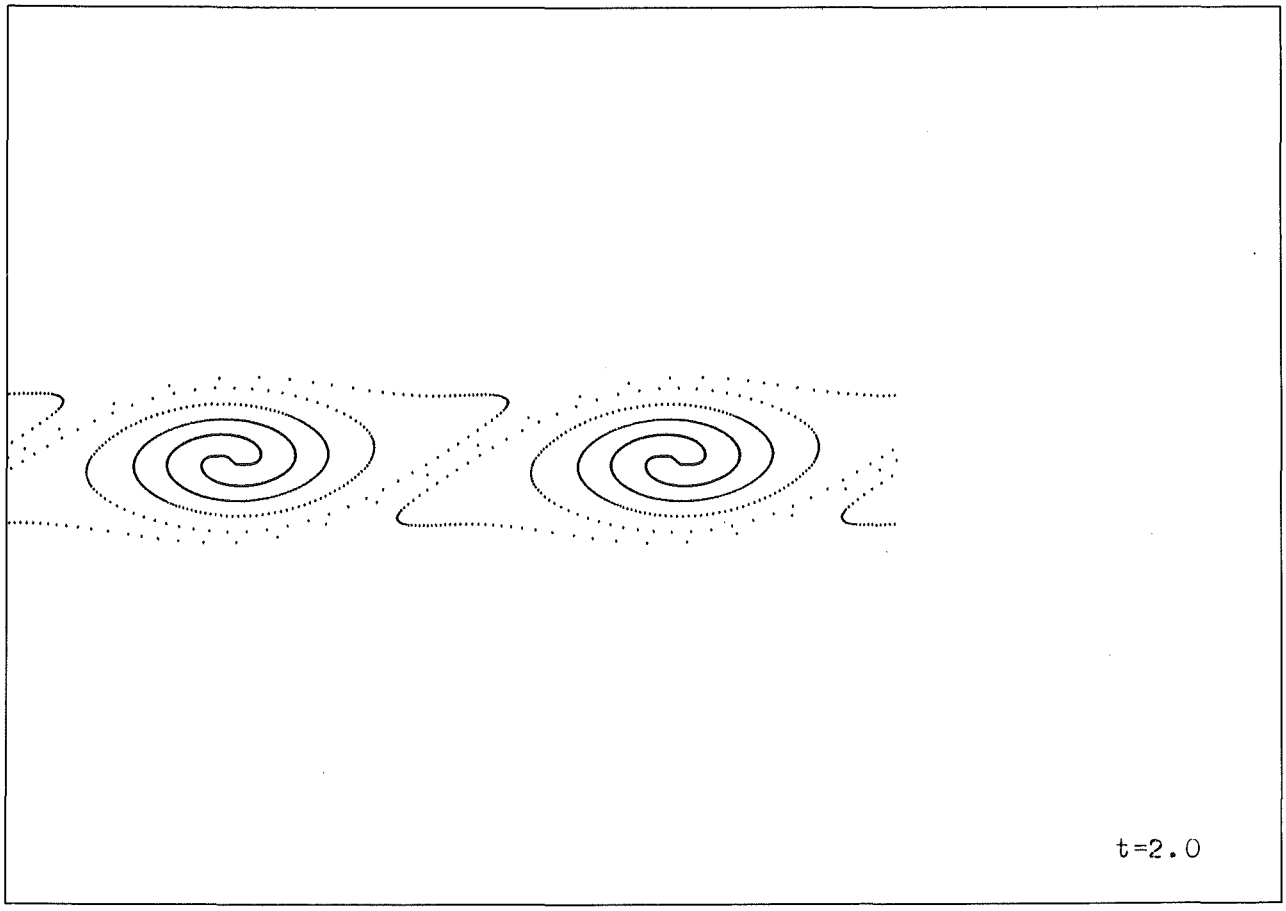
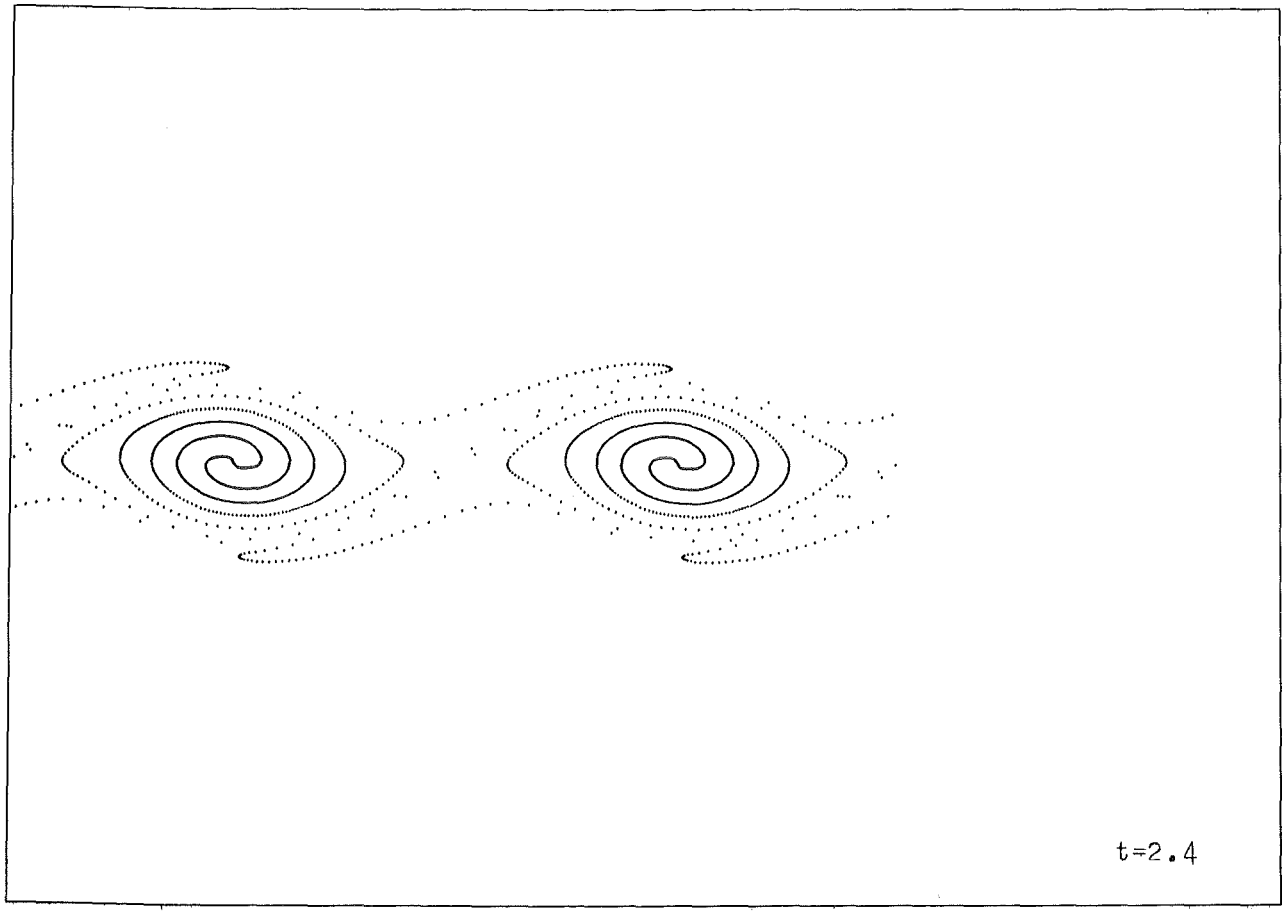


Fig. 6.4: Numerical simulation of a Kelvin-Helmholtz instability at time t = 1.6, 2.0



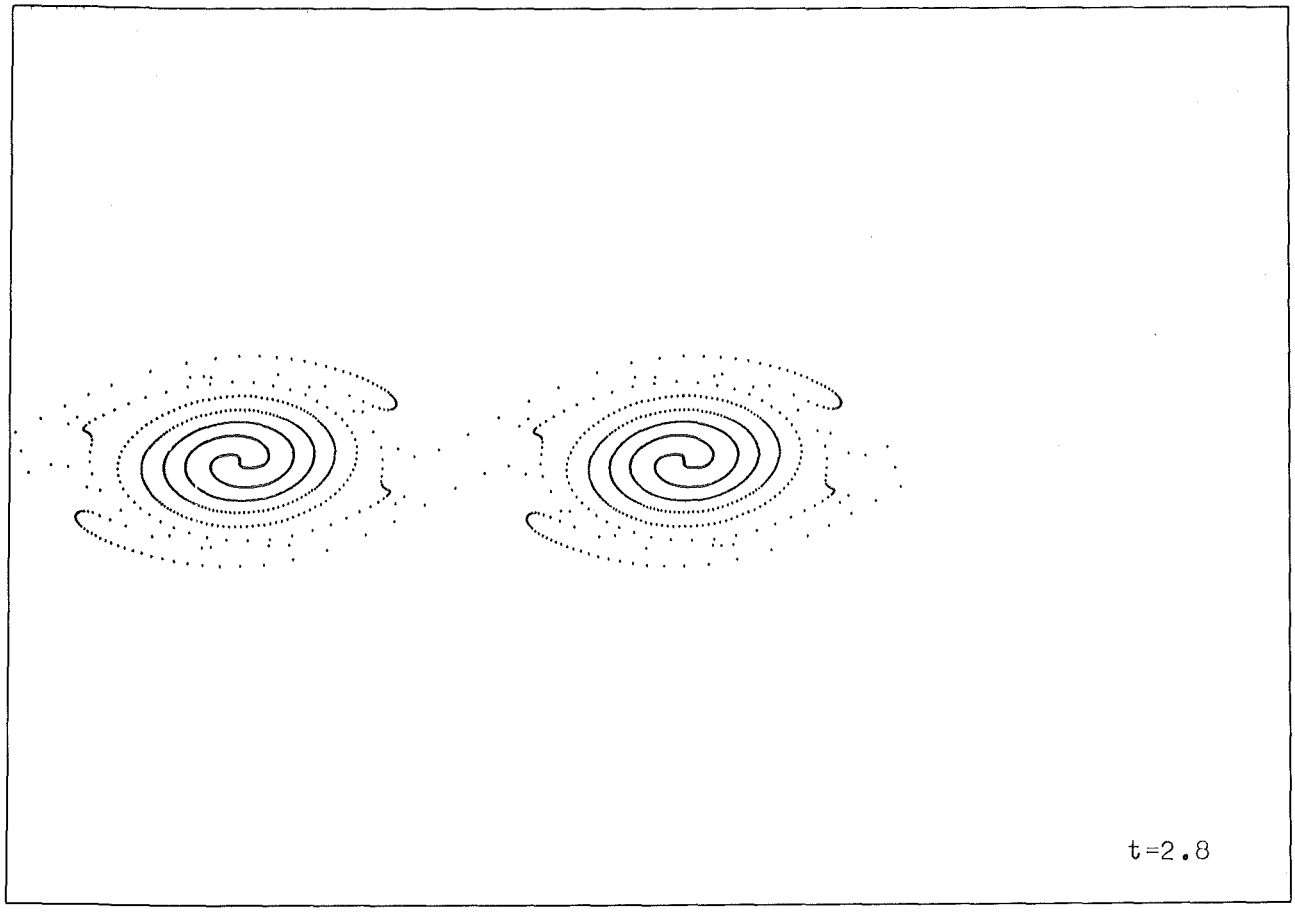
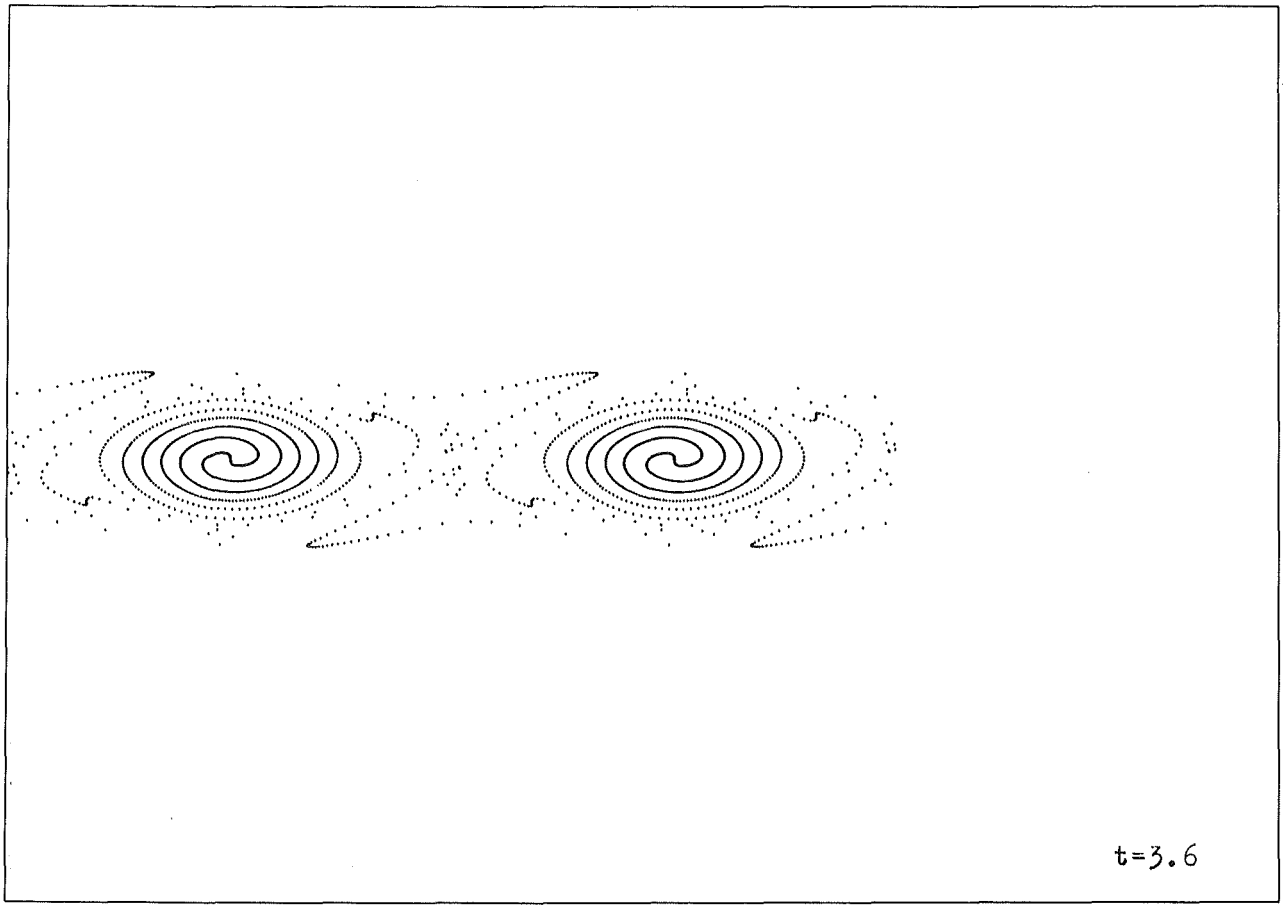


Fig. 6.5: Numerical simulation of a Kelvin-Helmholtz instability at time t = 2.4, 2.8



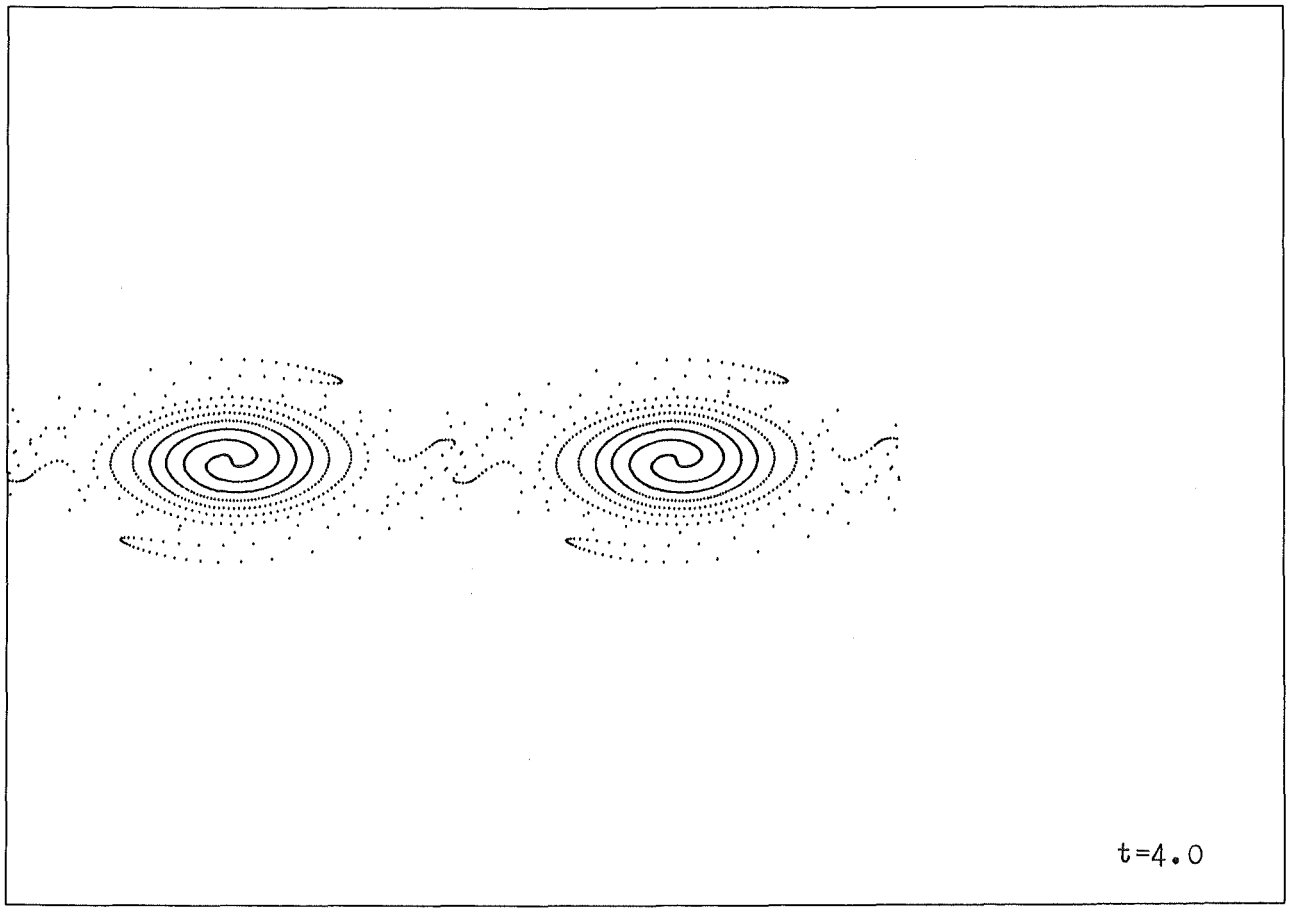


Fig. 6.6: Numerical simulation of a Kelvin-Helmholtz instability at time t = 3.6, 4.0

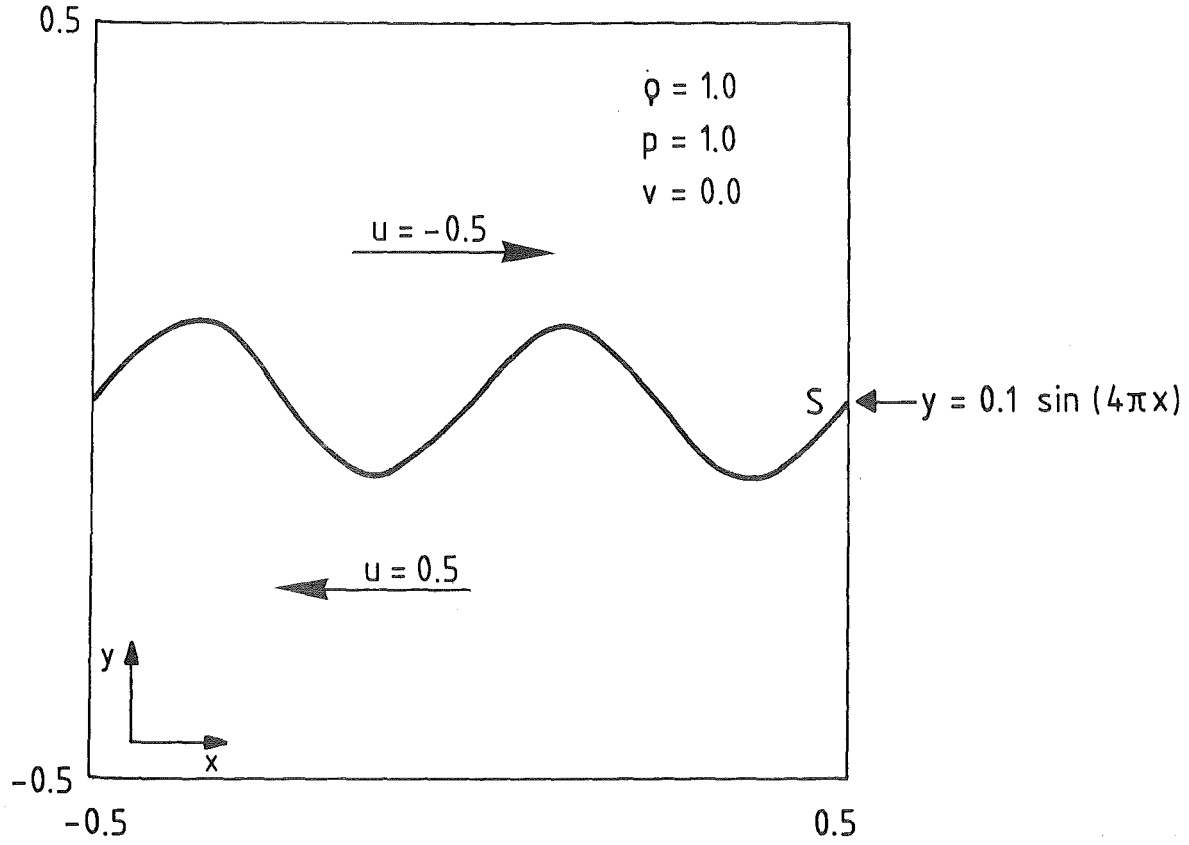


Fig. 6.7: Initial values for a Kelvin-Helmholtz instability with a large amplitude

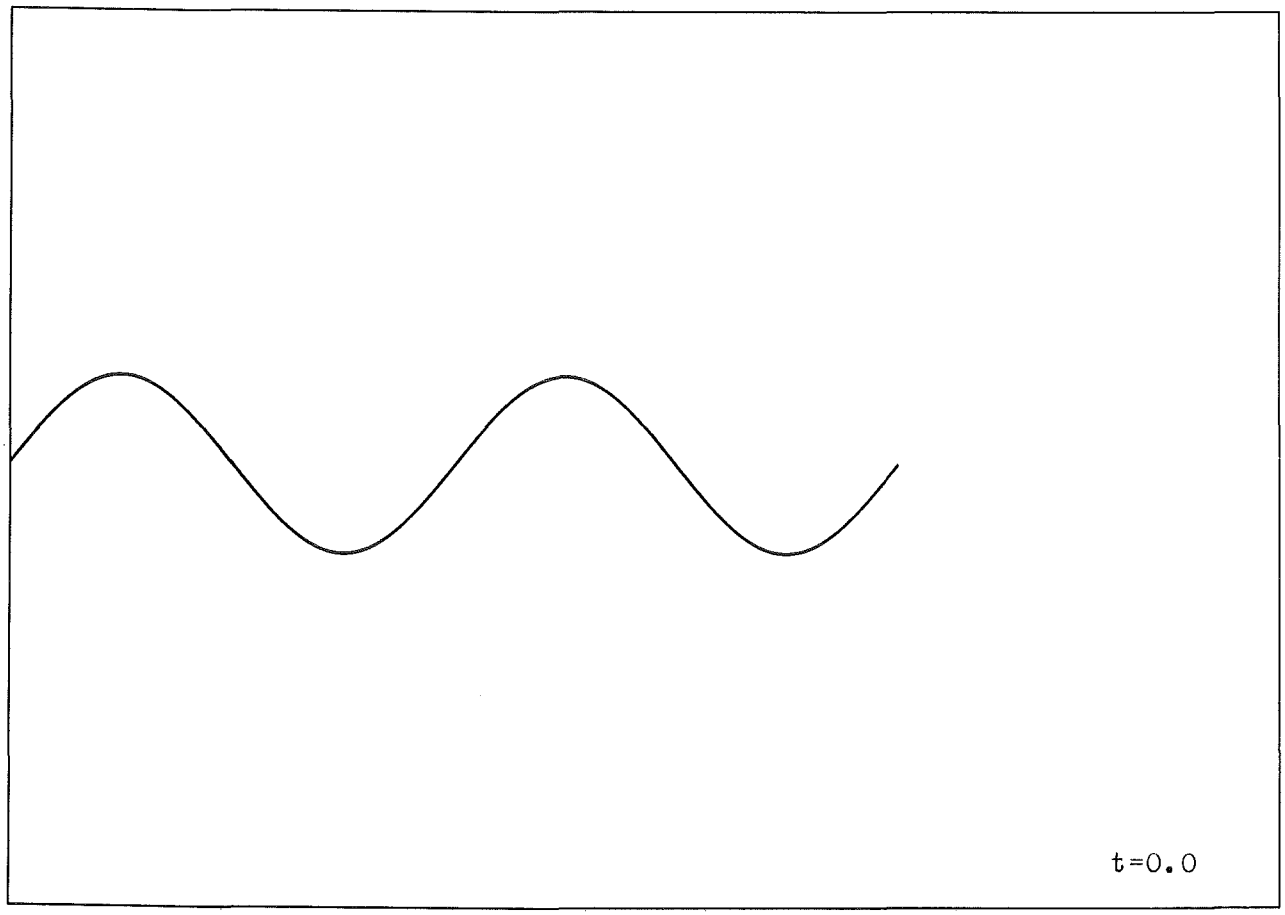
by a heavier fluid with the density $\rho = 10.0$. The lighter fluid is accelerated against the heavier fluid due to the gravitational force. Instead of the homogeneous Euler equations (2.1) the equations

$$\mathbf{U}_t + \mathbf{f}(\mathbf{U})_x + \mathbf{g}(\mathbf{U})_y = \mathbf{h}(\mathbf{U}) \qquad (6.2)$$

are numerically solved where due to the gravitational forces the source term

$$\mathbf{h}(\mathbf{U}) = (0, 0, -\rho, -\rho v)^T \qquad (6.3)$$

occurs in addition on the right hand side. The source term is treated by a splitting technique. Equation (6.2) is decomposed into the two one-dimensional problems (3.1), (3.2) and the equation



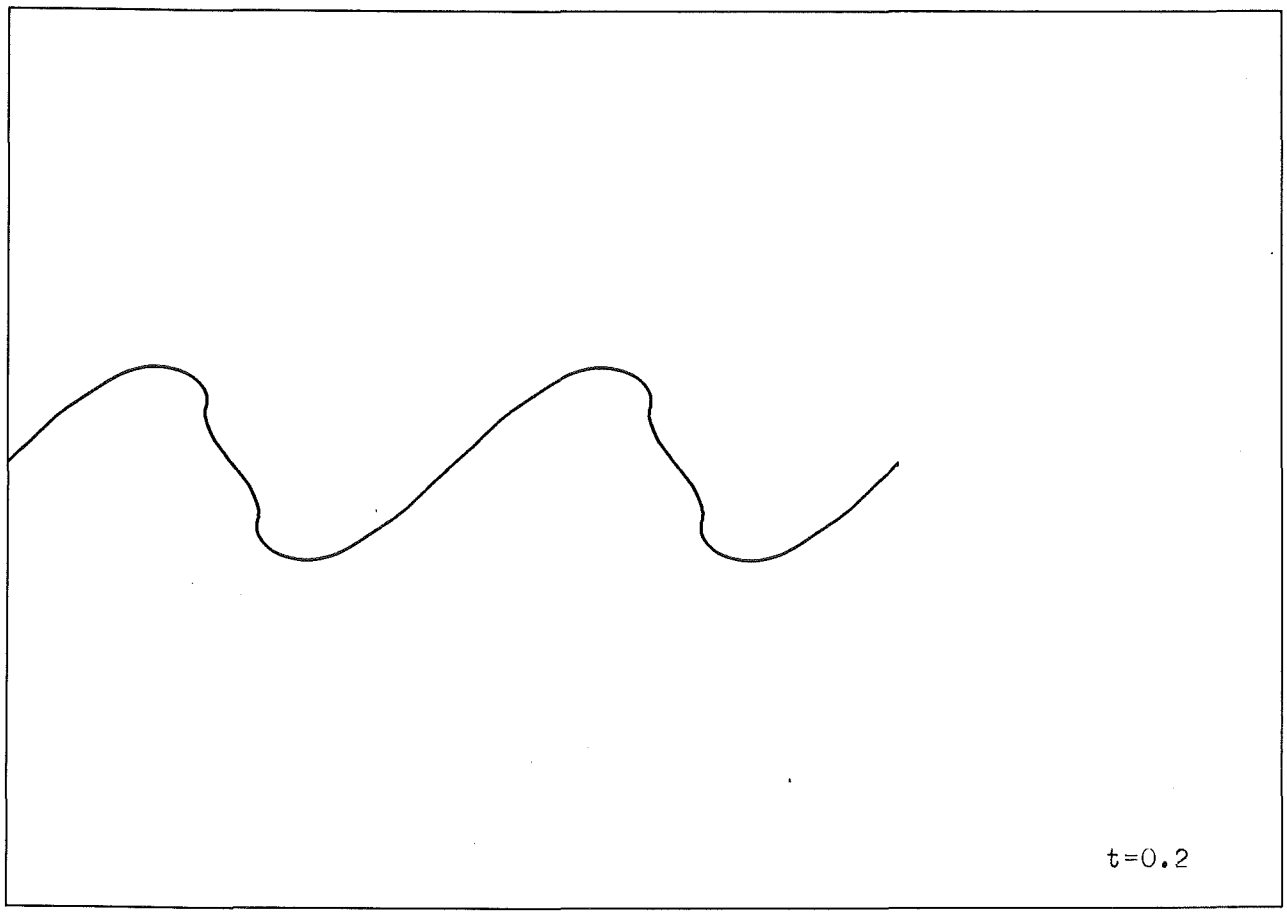
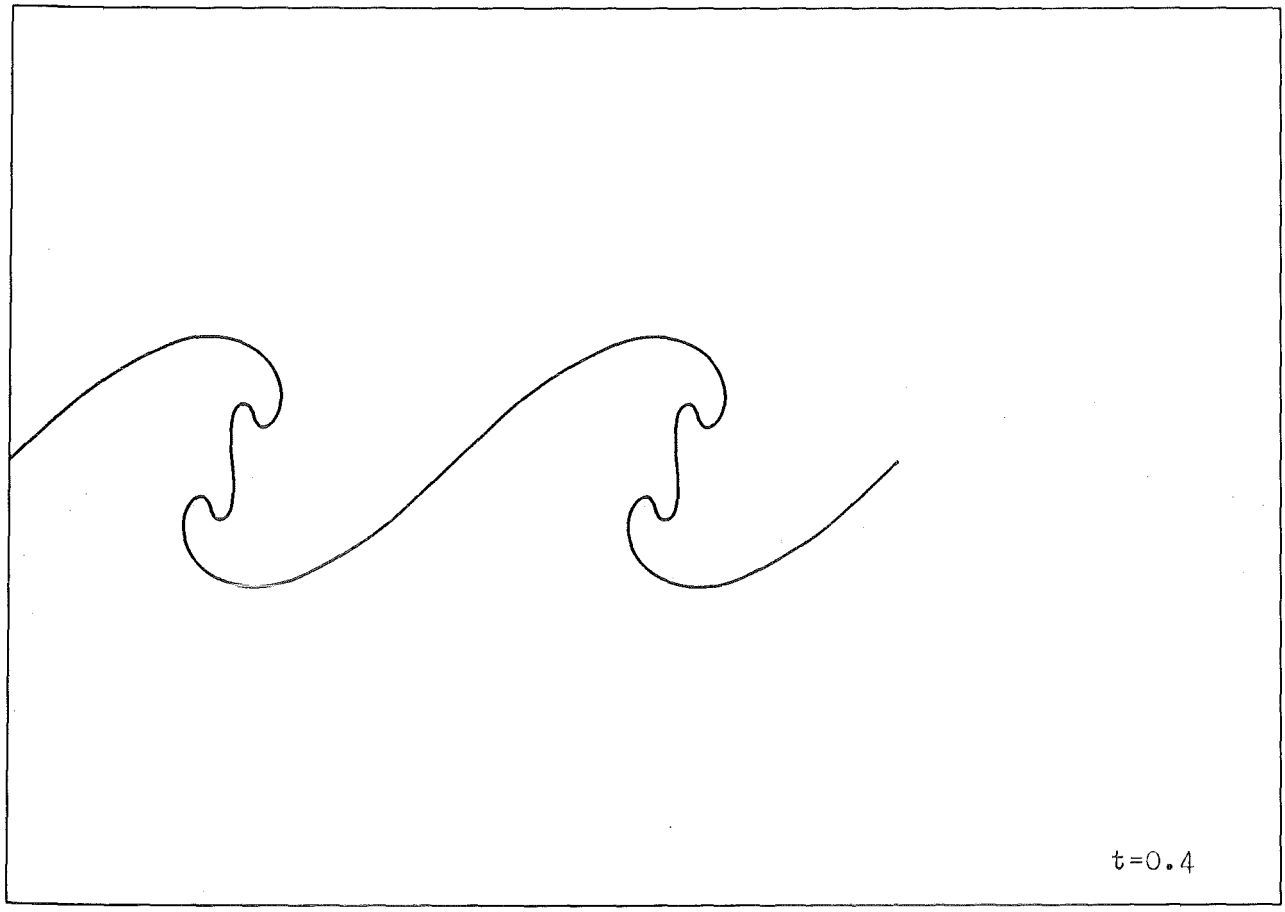


Fig. 6.8: Numerical simulation of a Kelvin-Helmholtz instability with a large amplitude at time t = 0.0, 0.4



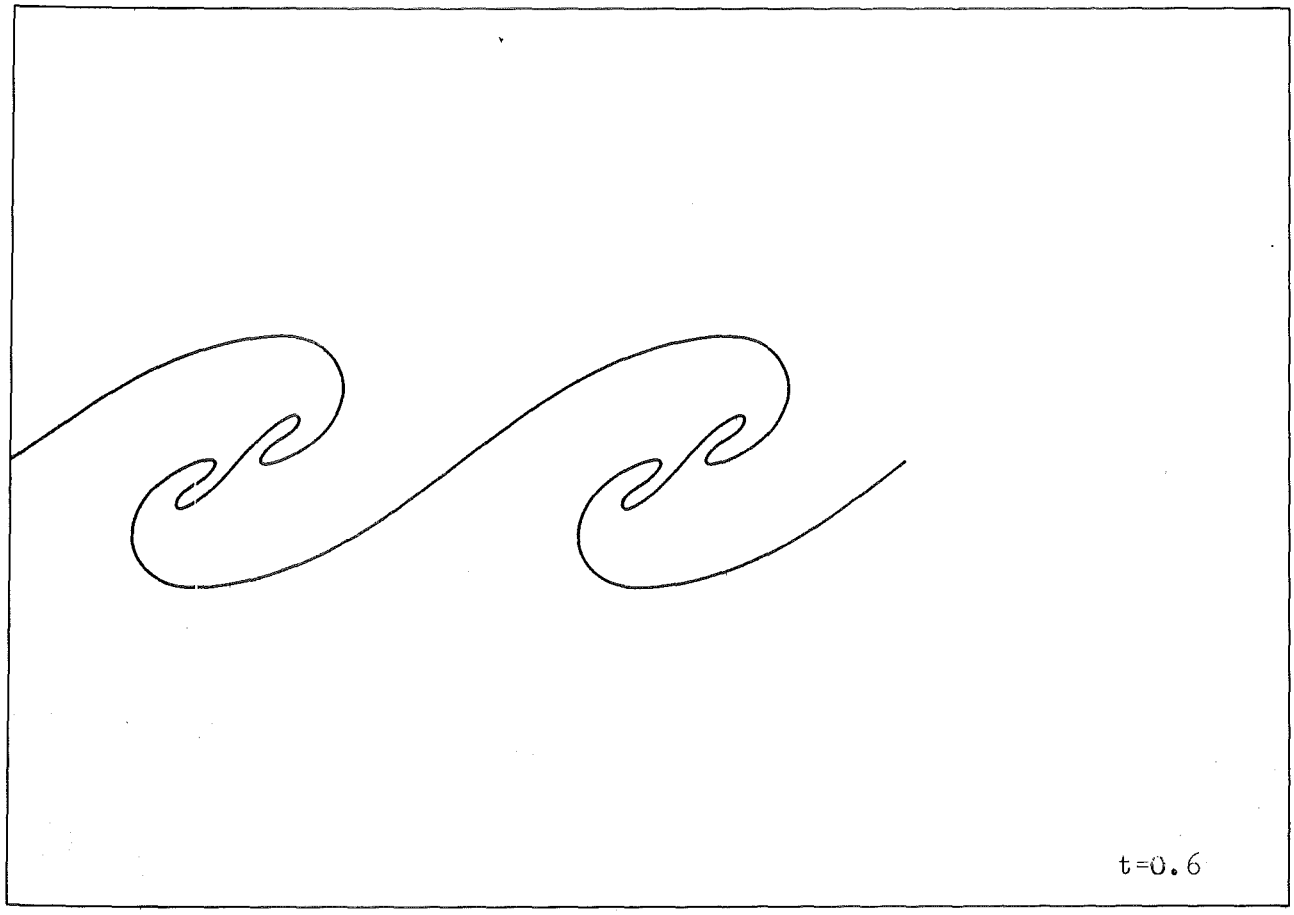
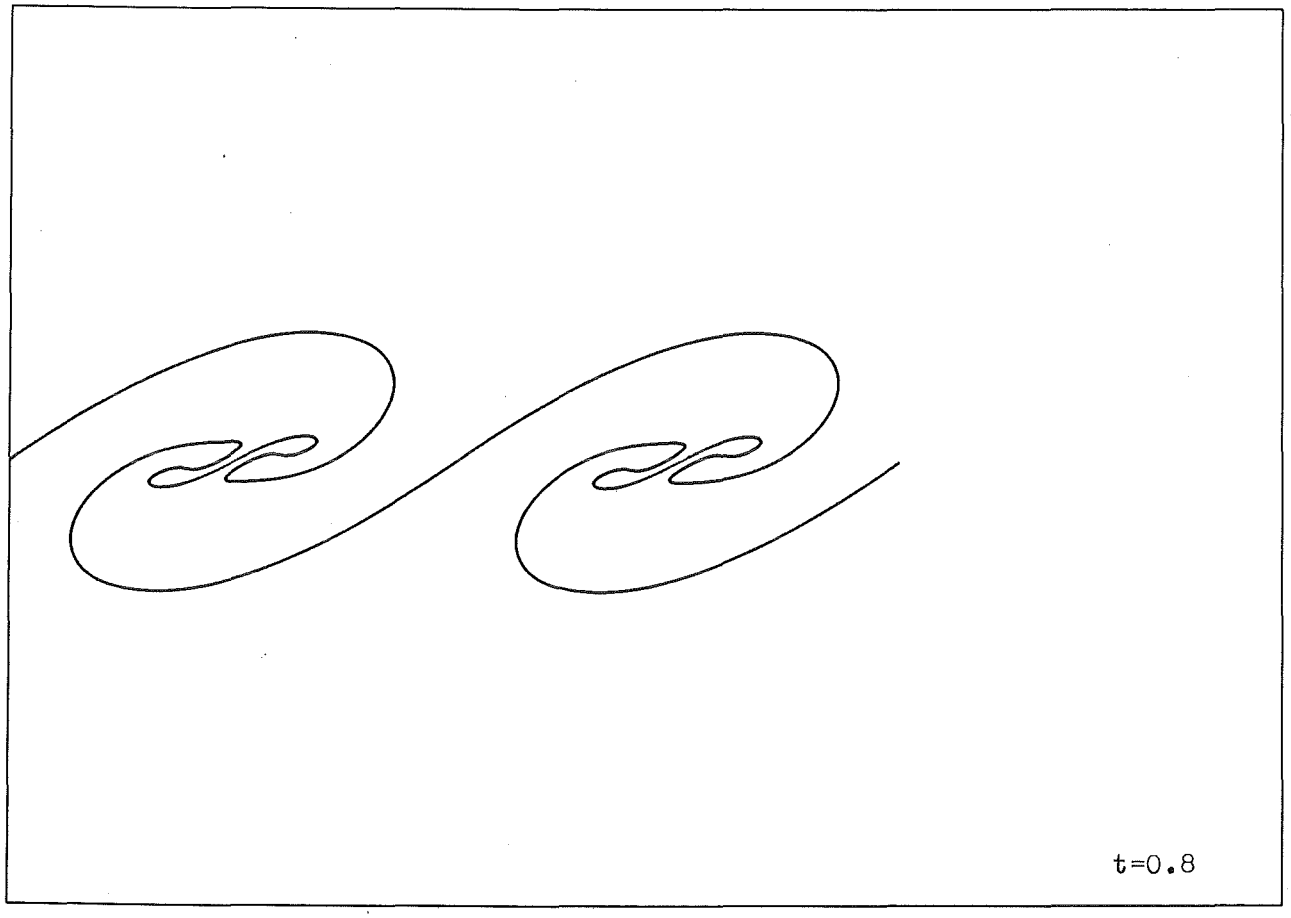


Fig. 6.9: Numerical simulation of a Kelvin-Helmholtz instability with a large amplitude at time t = 0.4, 0.6



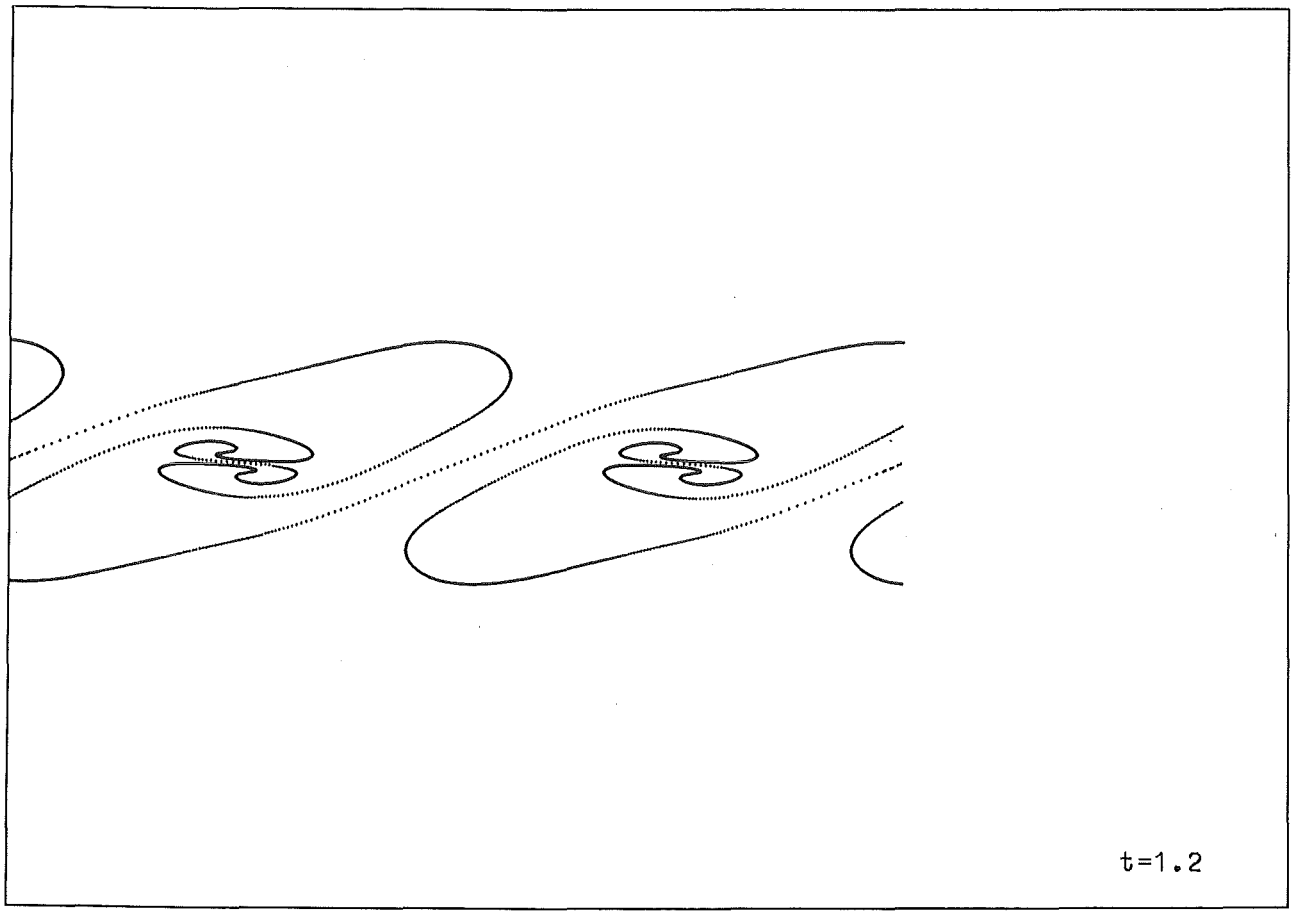
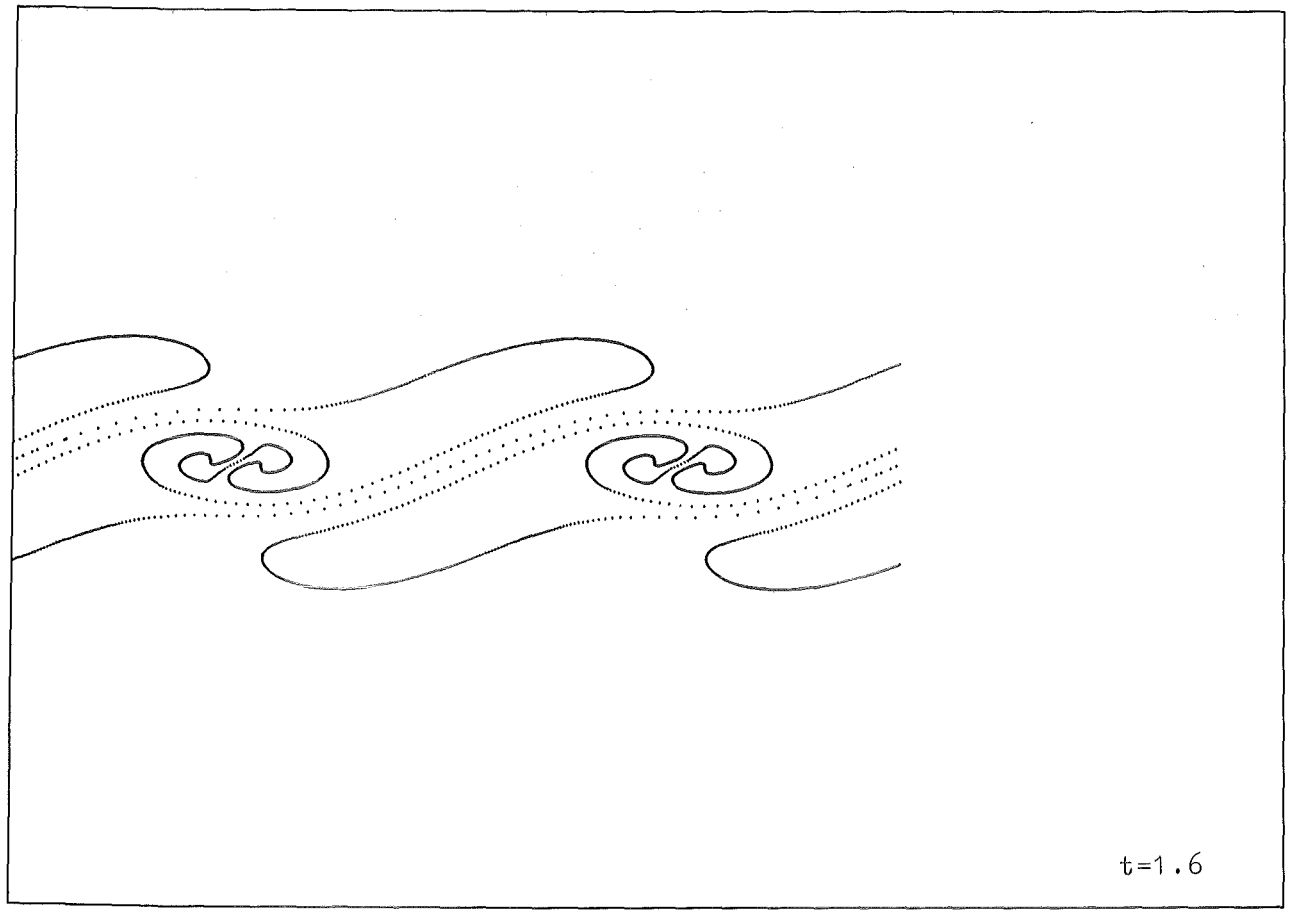


Fig. 6.10: Numerical simulation of a Kelvin-Helmholtz instability with a large amplitude at time t = 0.8, 1.2



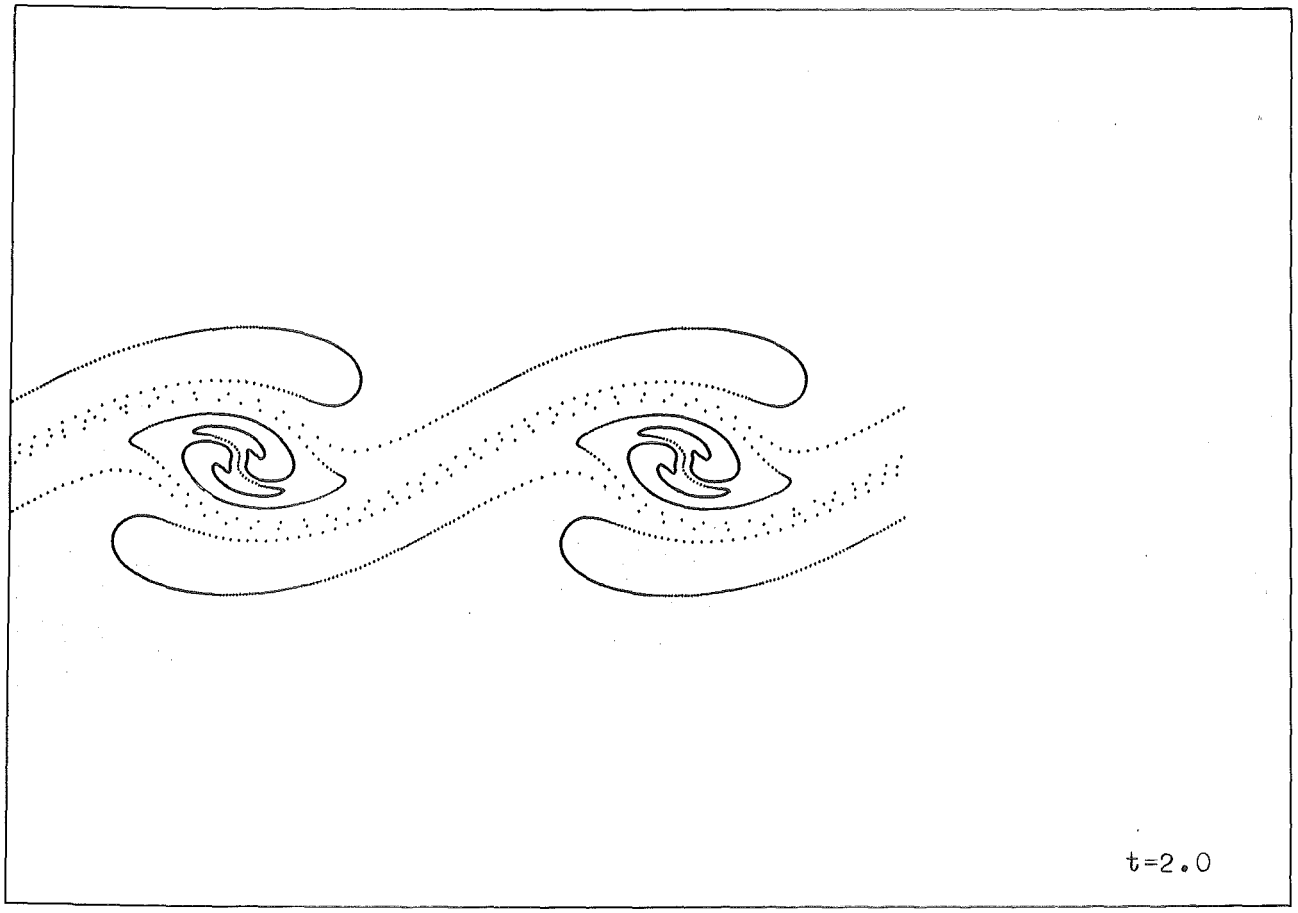


Fig. 6.11: Numerical simulation of a Kelvin-Helmholtz instability with a large amplitude at time t = 1.6, 2.0

$$U_t = h(U). \tag{6.4}$$

According to the splitting technique these problems are then solved successively in each time step. Hence, within this splitting algorithm the treatment of the source term is reduced to the solution of ordinary differential equations. To maintain the second-order accuracy of the algorithm we solved the ordinary differential equations (6.4) by the second-order accurate method of Heun (/25/).

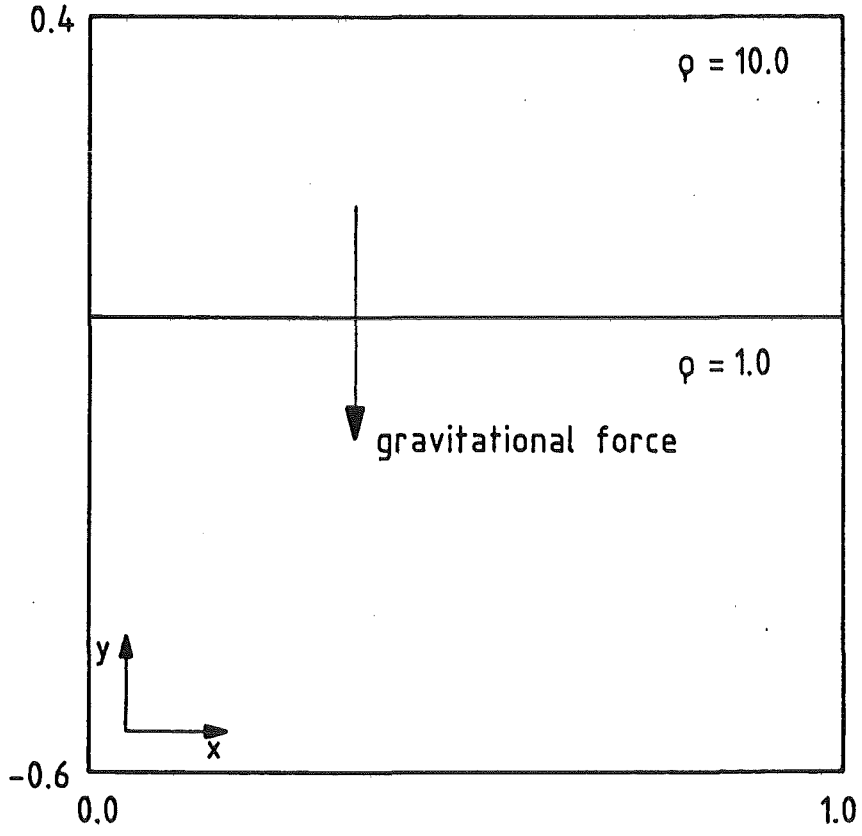
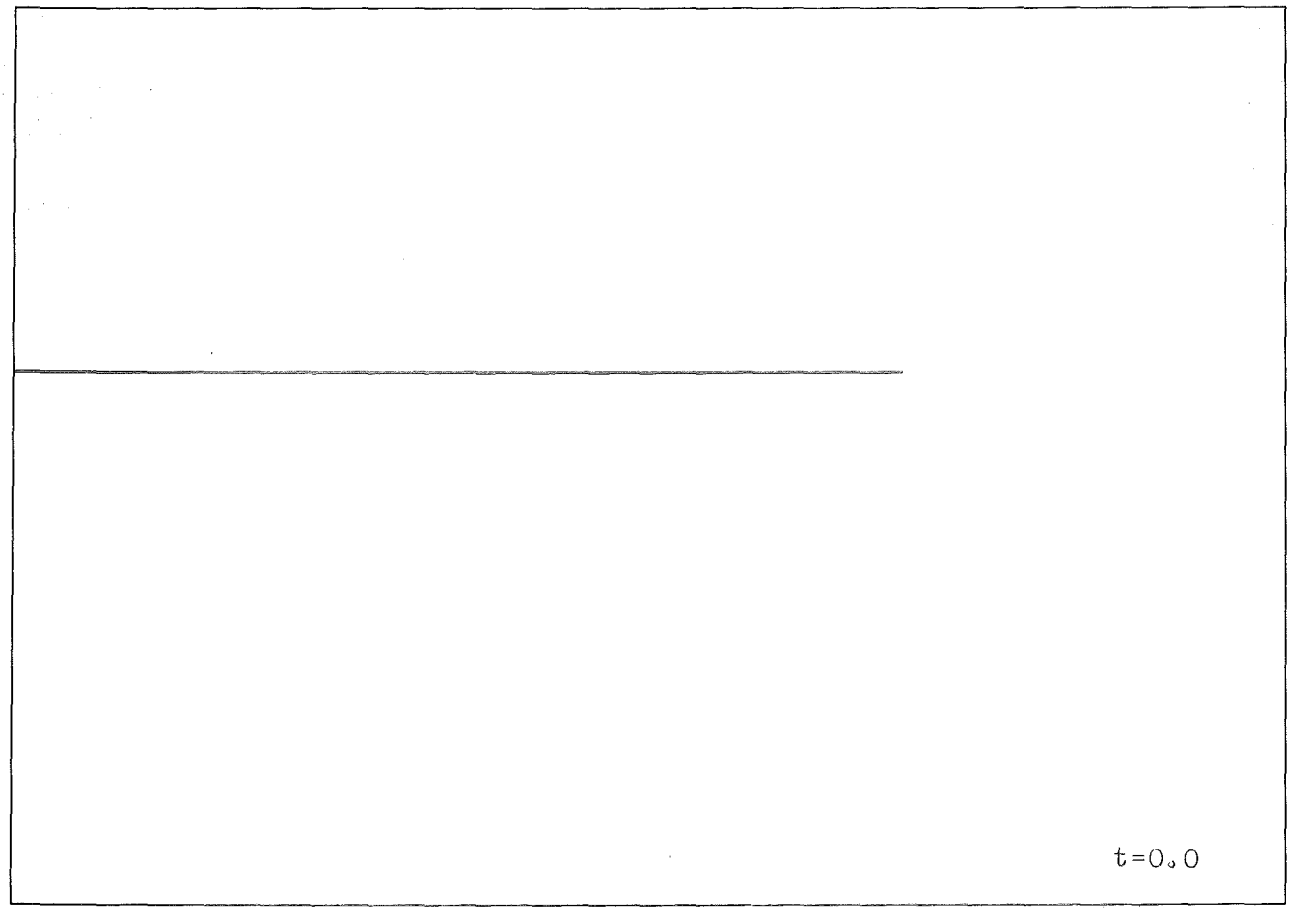


Fig. 6.12: Initial values for a Rayleigh-Taylor instability

The computational domain is R = [0.0, 1.0] x [-0.6, 0.4]. The interface between the two fluids is situated at y = 0. The initial pressure is given by the hydrostatic pressure

$$p(x, y) = \int_y^{0.4} \rho(x, y)\, d\, y. \tag{6.5}$$



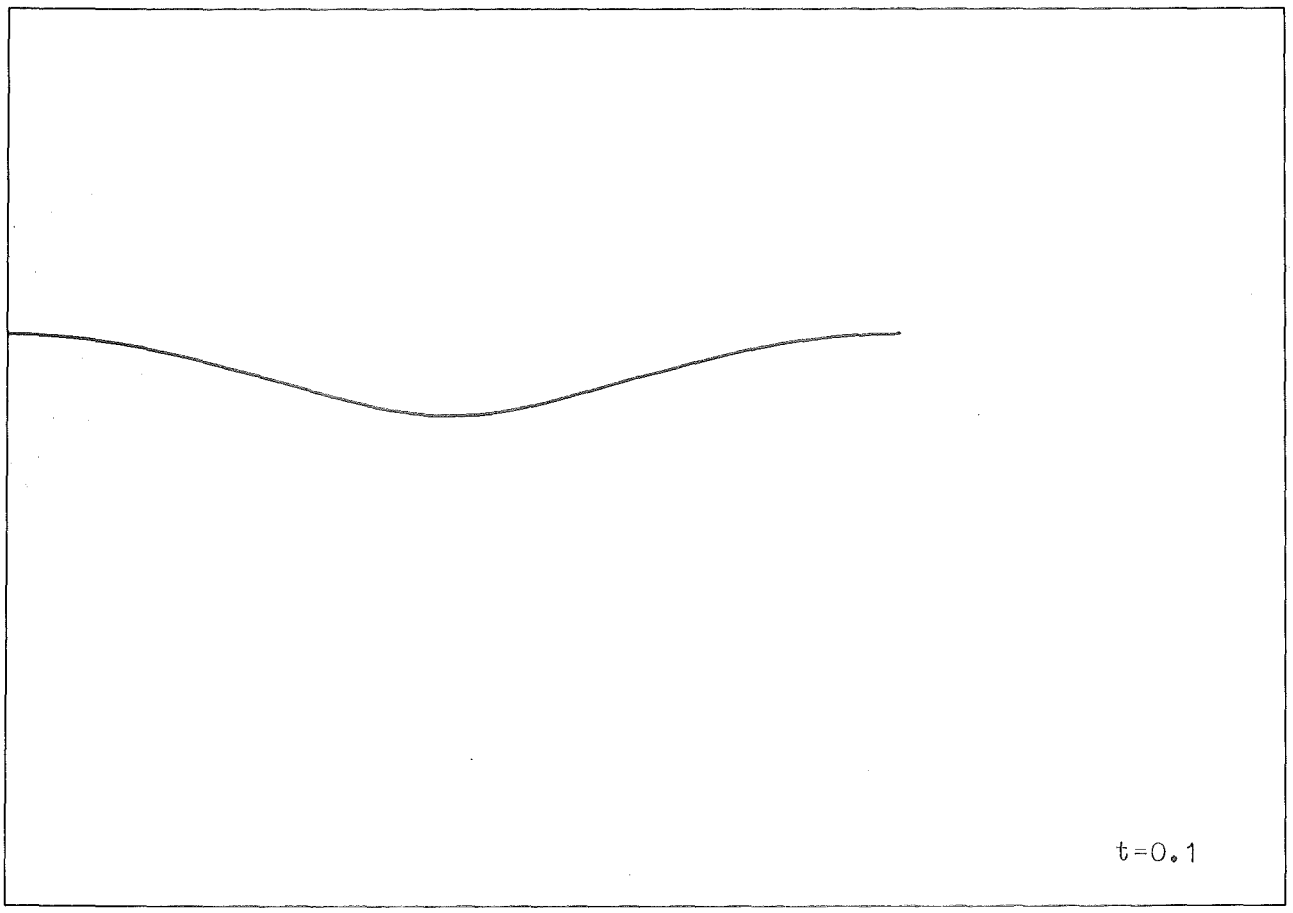
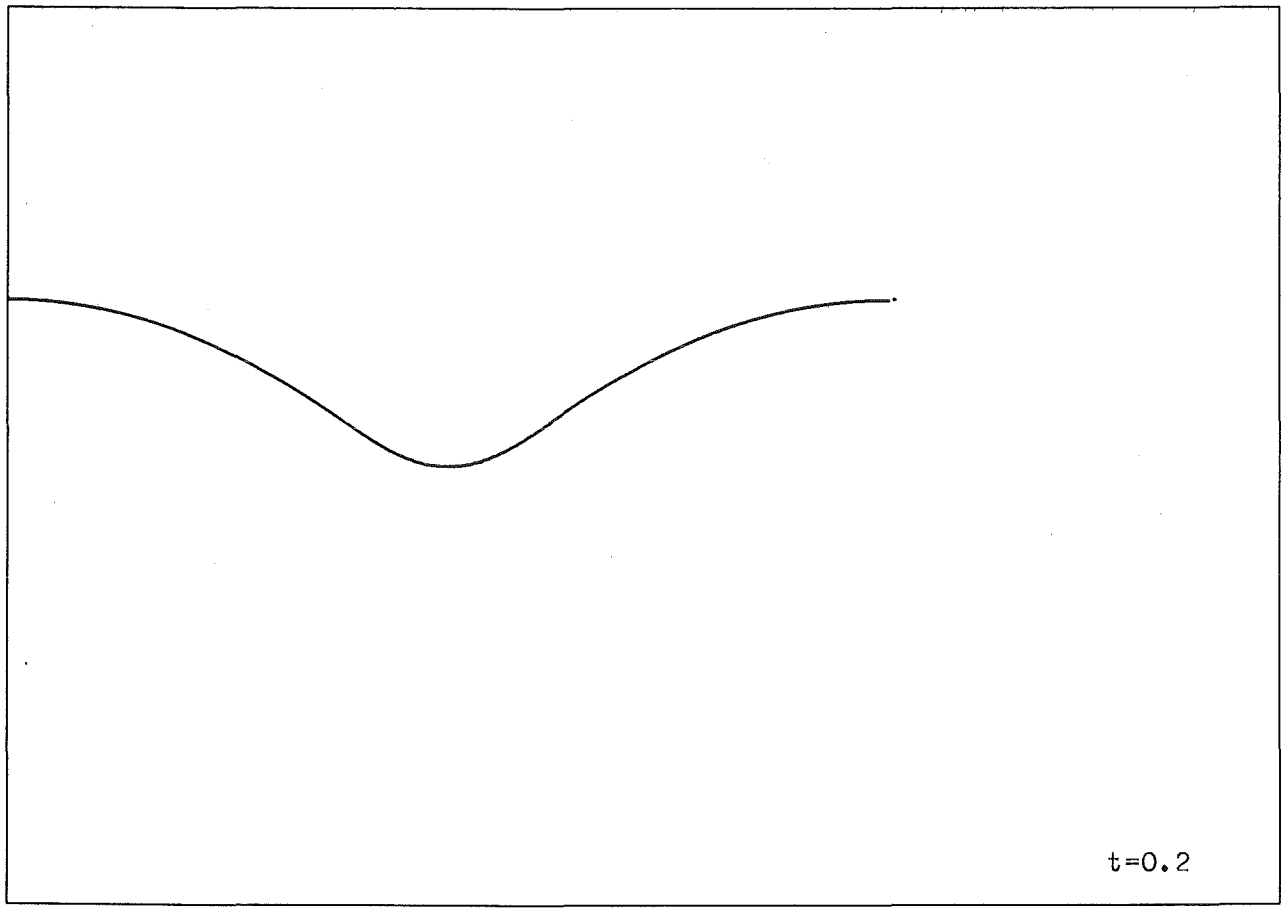


Fig. 6.13: Numerical simulation of a Rayleigh-Taylor instability at time t = 0.0, 0.1



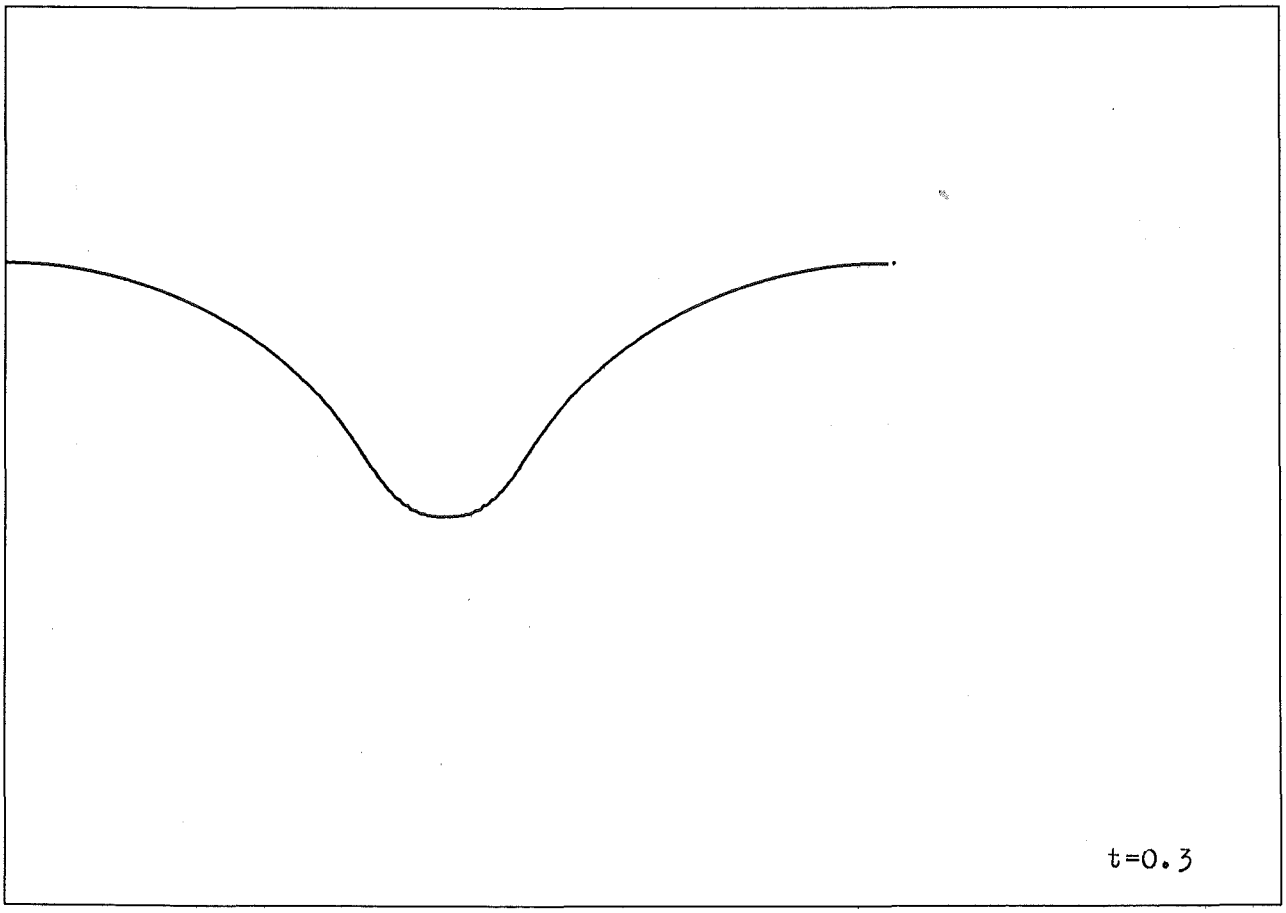
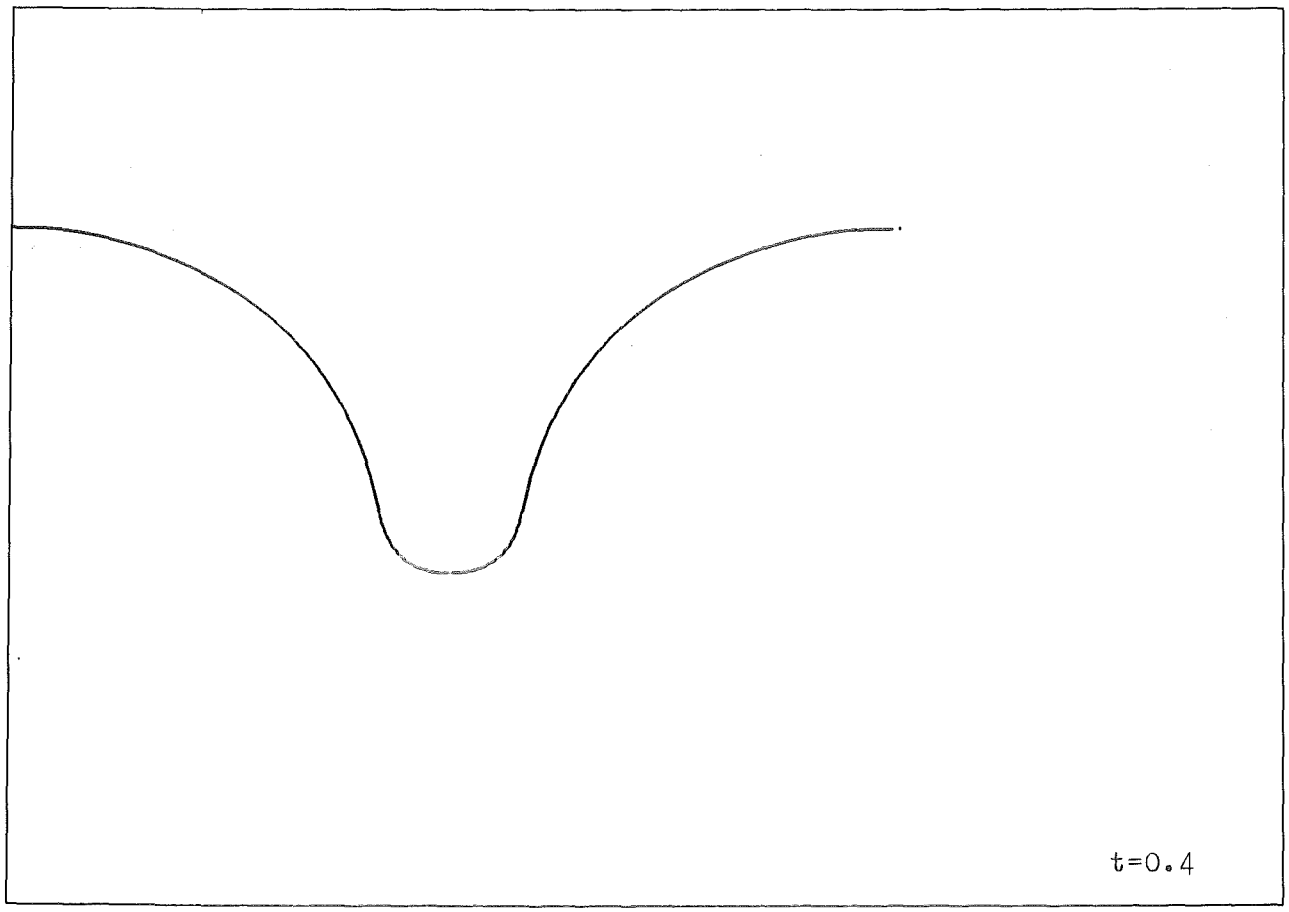


Fig. 6.14: Numerical simulation of a Rayleigh-Taylor instability at time t = 0.2, 0.3



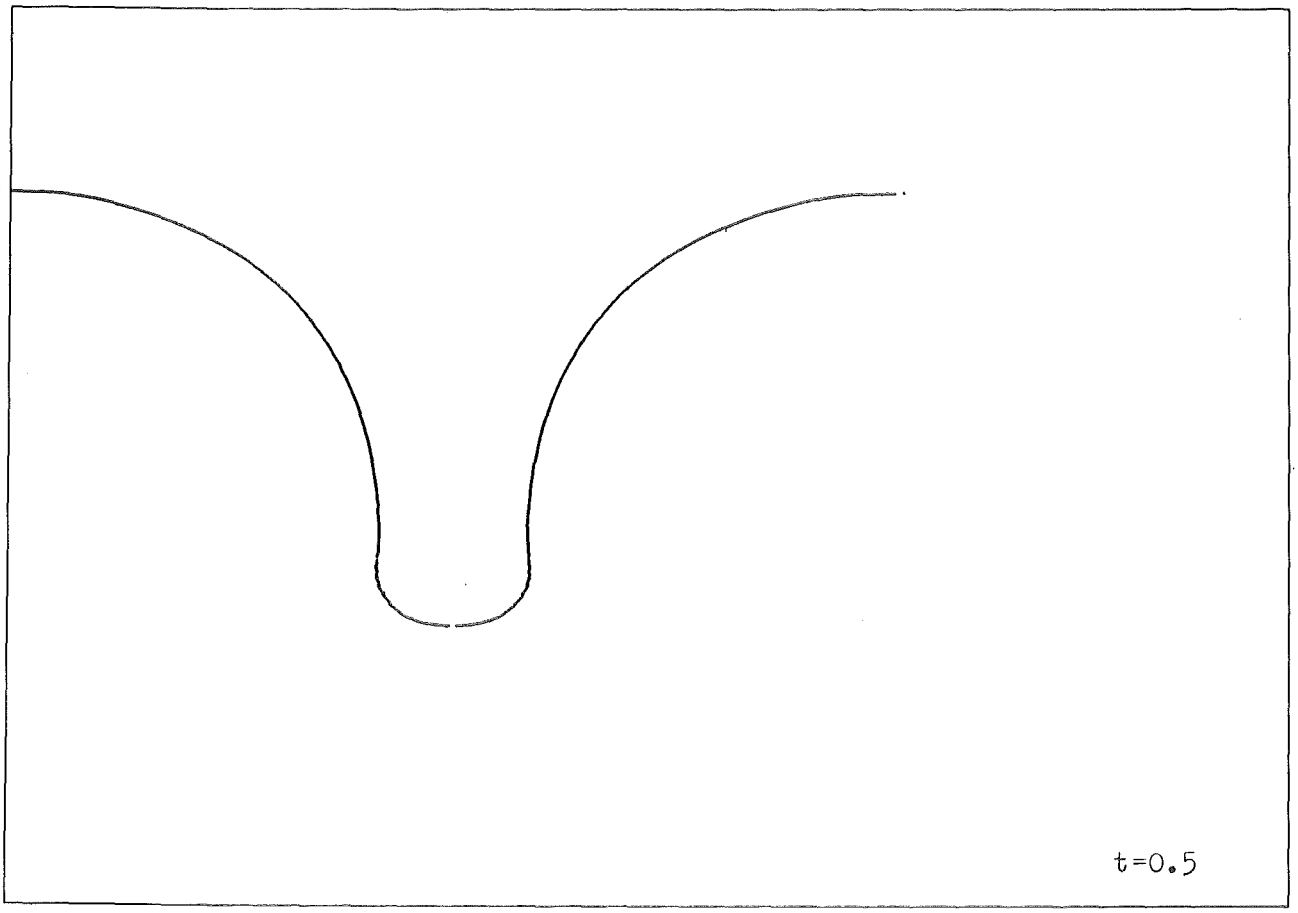
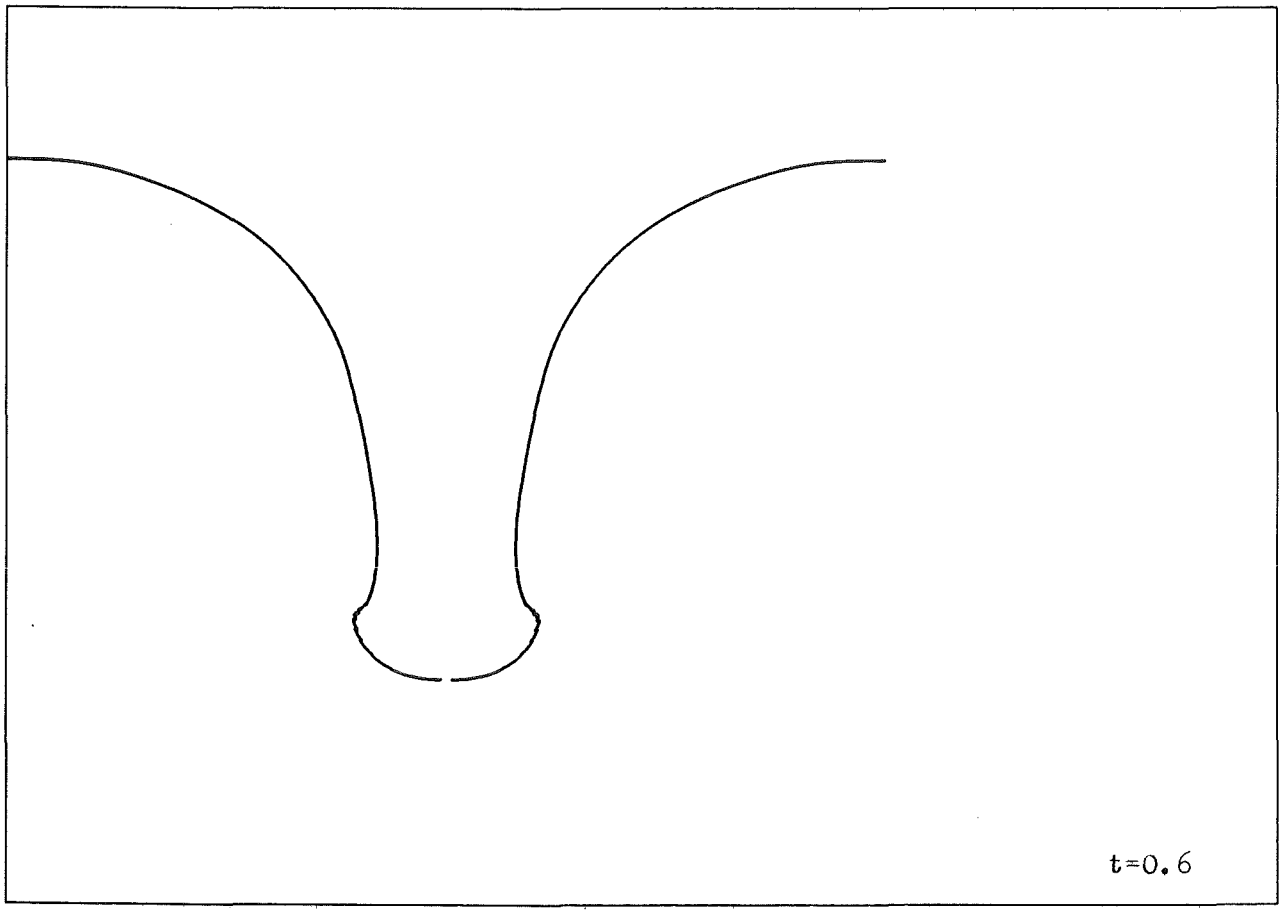


Fig. 6.15: Numerical simulation of a Rayleigh-Taylor instability at time t = 0.4, 0.5



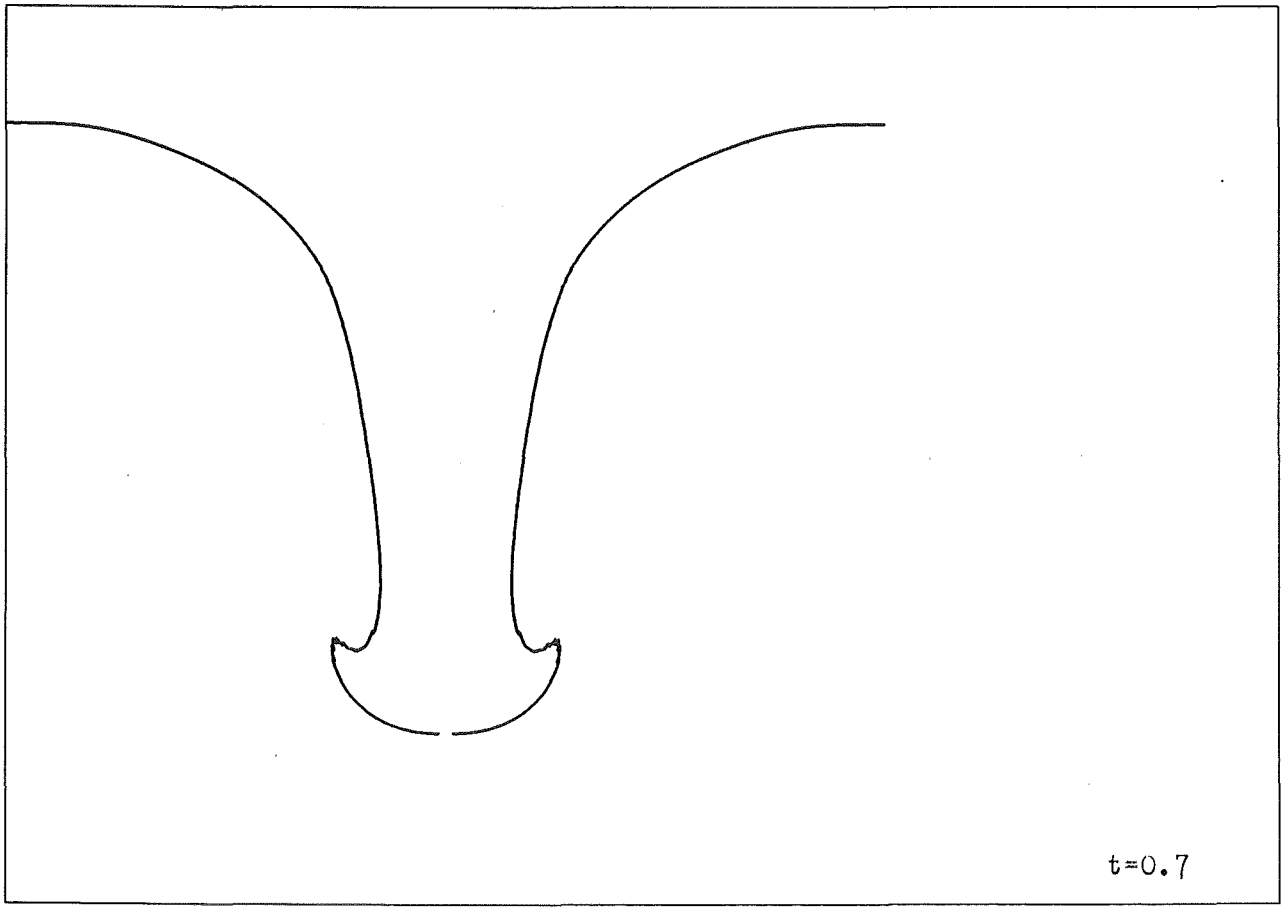
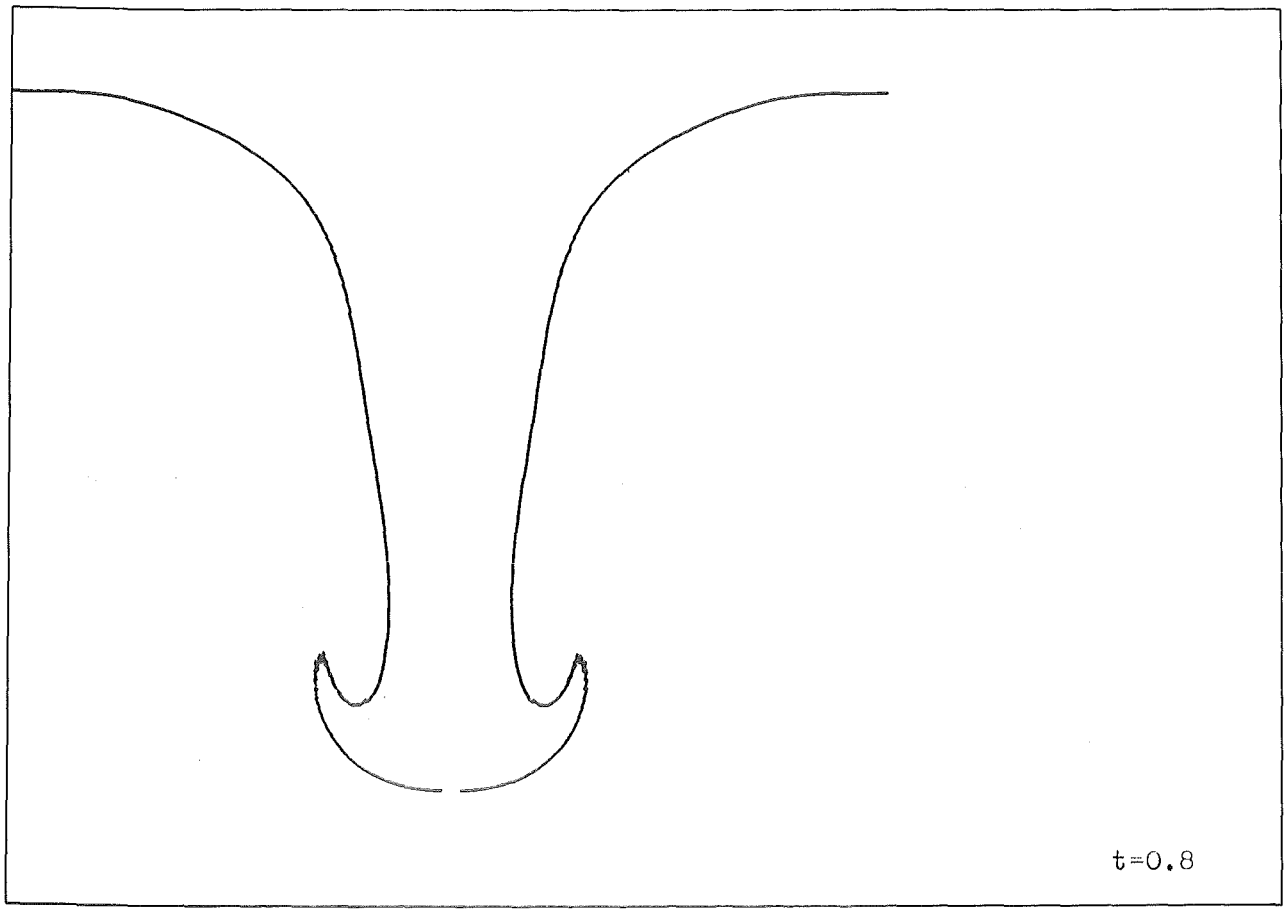


Fig. 6.16: Numerical simulation of a Rayleigh-Taylor instability at time t = 0.6, 0.7



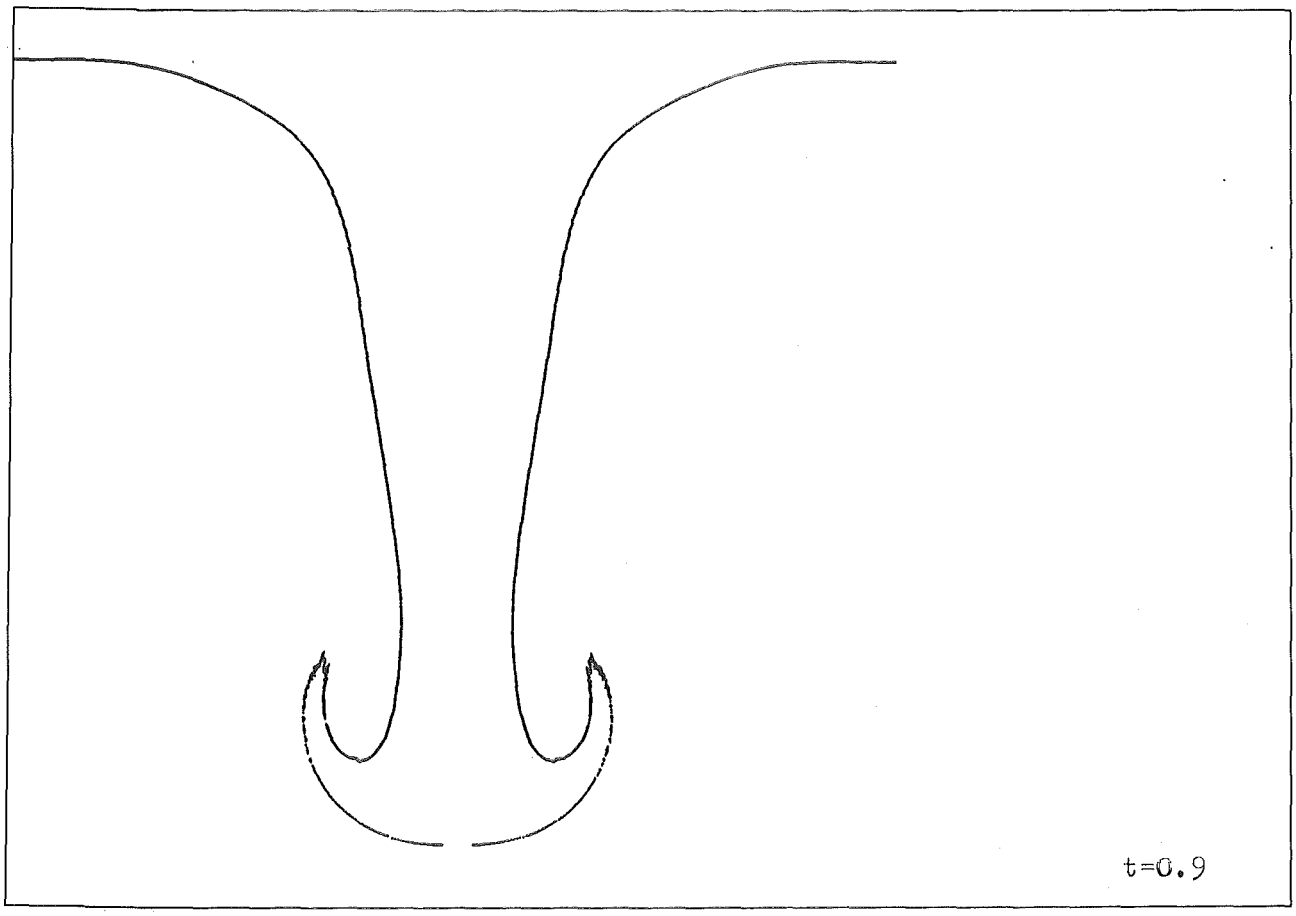


Fig. 6.17: Numerical simulation of a Rayleigh-Taylor instability at time t = 0.8, 0.9

Thus, for zero velocity the problem is in equilibrium. Sinusiodal perturbations are introduced by perturbation of the zero velocity. We adopted the initial values proposed by Daly /7/. The velocity components are given by the fomulas

$$u(x, y) = a\,\mathrm{sgn}(y)\sin(2\pi x)\, e^{-2\pi|y|} \qquad (6.6)$$

$$v(x, y) = a\cos(2\pi x)\, e^{-2\pi|y|} \qquad (6.7)$$

As initial values for the numerical scheme we prescribed the mean values in each grid zone. At the boundaries of the computational domain R we impose at the right and left side periodic conditions, at the top and the bottom those of a reflecting wall. For the numerical calculations we used a uniform grid with 100 x 100 grid zones. The amplitude of the perturbation (6.6), (6.7) is a = 0.5.

Figures 6.13 - 6.17 indicate the motion of the surface between the two fluids. The position of the marker particles are plotted at times 0.0, 0.1, ... 0.9. The results are similar to those of Daly /7/. A spike and a bubble arise. As expected, the bubble is much broader than the spike; the spike moves to the bottom at a higher propagation rate. At time t = 0.6 Kelvin-Helmholtz instabilities will occur on both sides of the spike. At later times rollup of the vortex sheets starts.

The next problem is the numerical simulation of a jet. The initial values are sketched in Fig. 6.18. A fluid to the left at velocity u = -0.5, v = 0.0 is separated by a small band flowing into the opposite direction. The density, pressure and velocity in y-direction are uniform in the whole domain: $\rho = 1.0$, $p = 1.0$, $v = 0.0$. The surfaces between the flows are given by

$$S_1: \; y = 0.05 + 0.01 \sin(4\pi x) \qquad (6.8)$$

$$S_2: \; y = -0.05 + 0.01 \sin(4\pi x) \qquad (6.9)$$

The computational domain is R = [-0.5, 0.5] x [-0.5, 0.5]. For the right and left sides of R we prescribed periodic boundary conditions, at the top and of the bottom those of a reflecting wall. The computational grid consists of 100 x 100 grid zones. As initial values for the numerical scheme we prescribed approximative mean values for each grid zone.

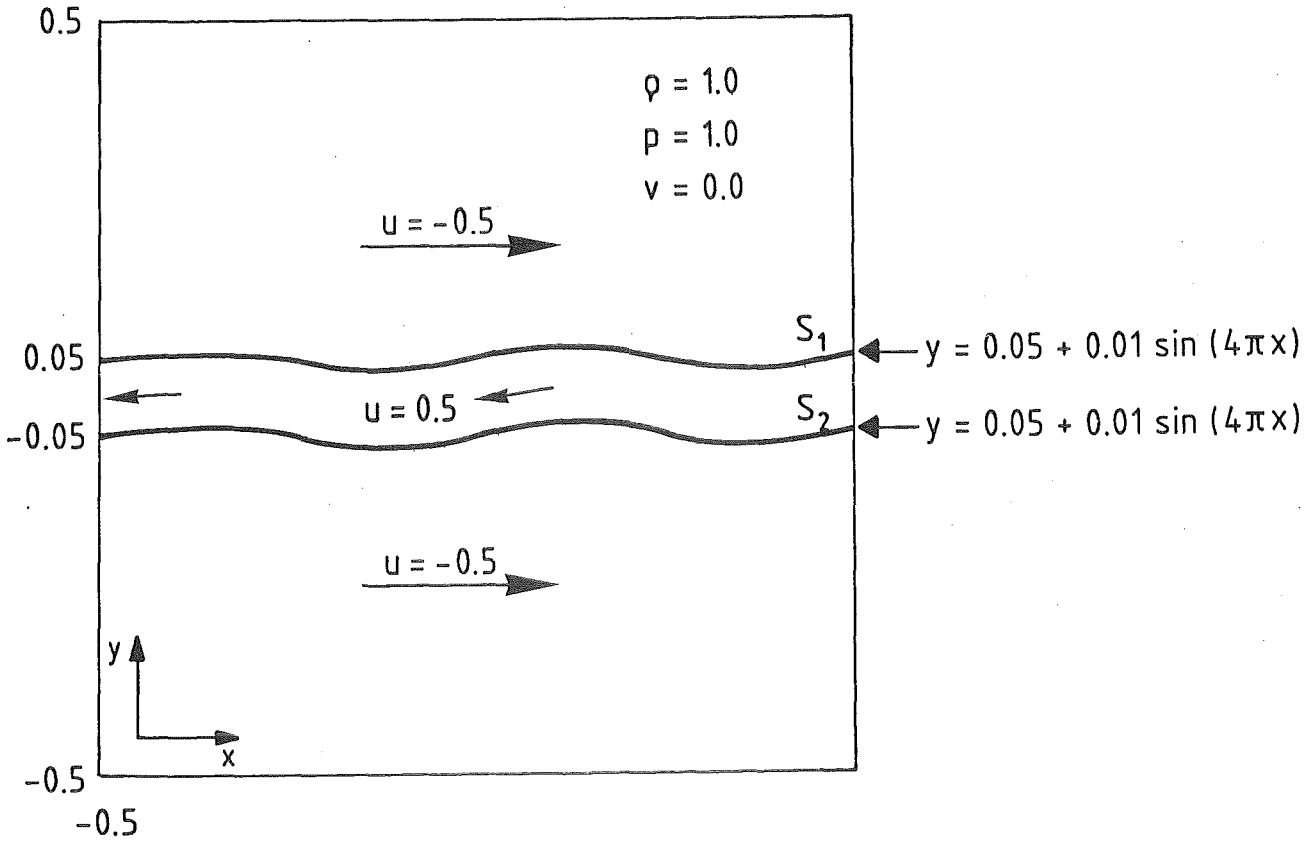
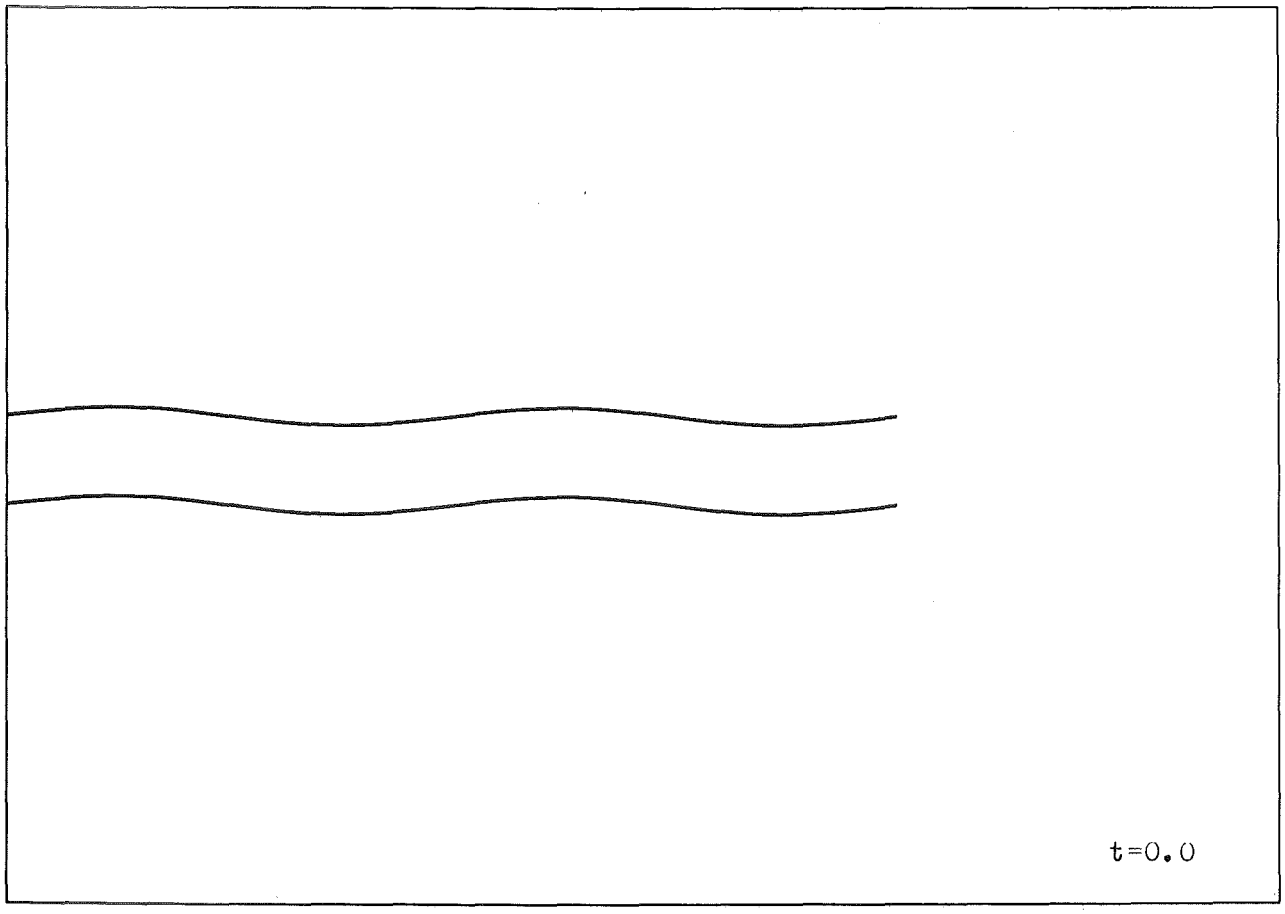


Fig. 6.18: Initial values for a jet

The marker particles are located at the interfaces S_1, S_2. Figure 6.19 shows the position of the marker particles at the initial time and at time t = 0.4. The plot at t = 0.4 indicates that the amplitude of the sinusoidal perturbation (6.8), (6.9) is increased. Further small perturbations occur and the curves S_1, S_2 become a little bit wrinkled. Four of these small perturbation increase. These wrinkles become very obvious at time t = 0.6 (Fig. 6.20). They continue to inrease and at time t = 0.8 they look like four noses. During this process the amplitude of the sinusoidal perturbation further increased. In Figure 6.21 we see that the small perturbations grow and the shear layers start to rollup into a Karman vortex street. Figures 6.19 - 6.22 look quite similar to experimental results as presented by van Dyke (/8/, page 56). He shows the development of a Karman vortex behind



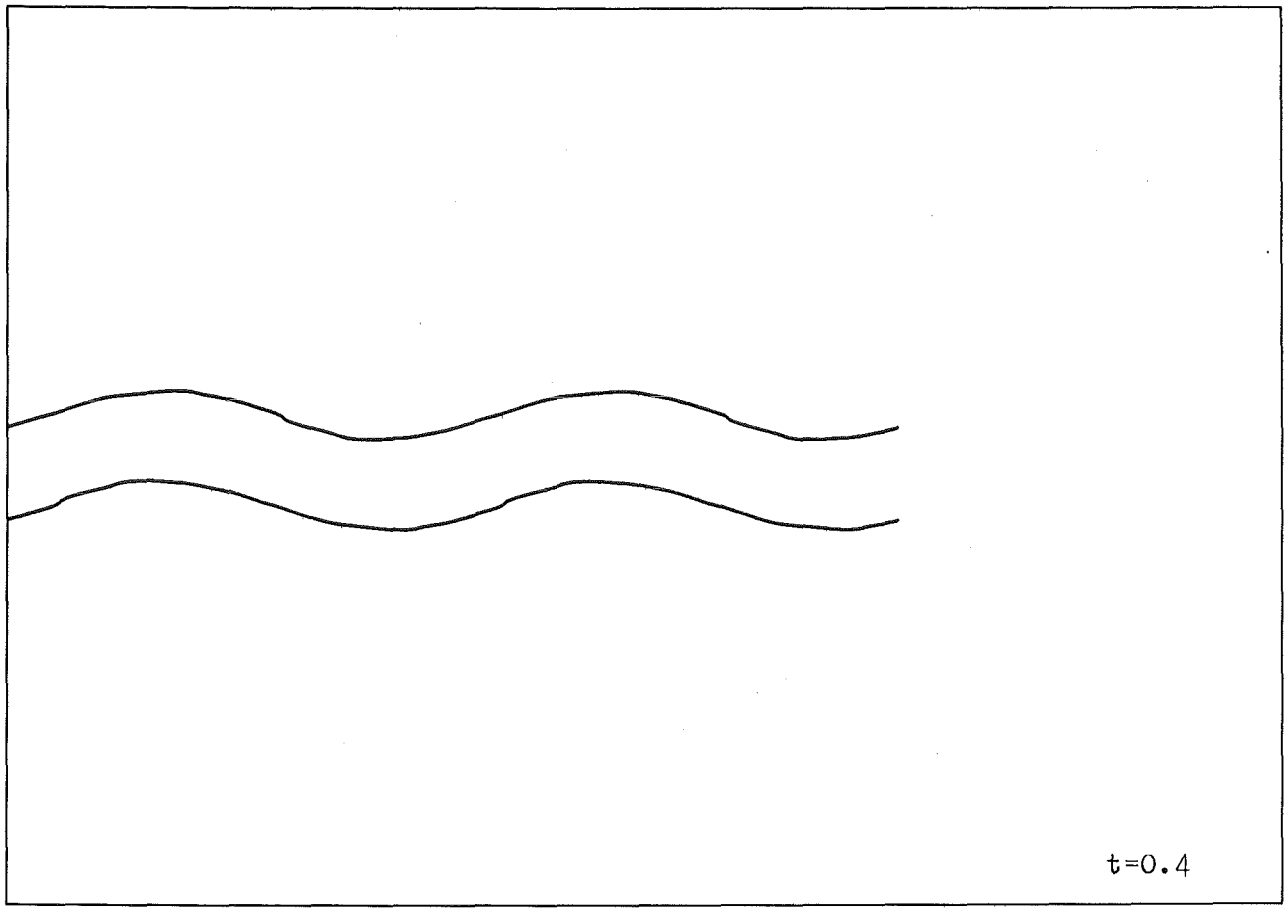
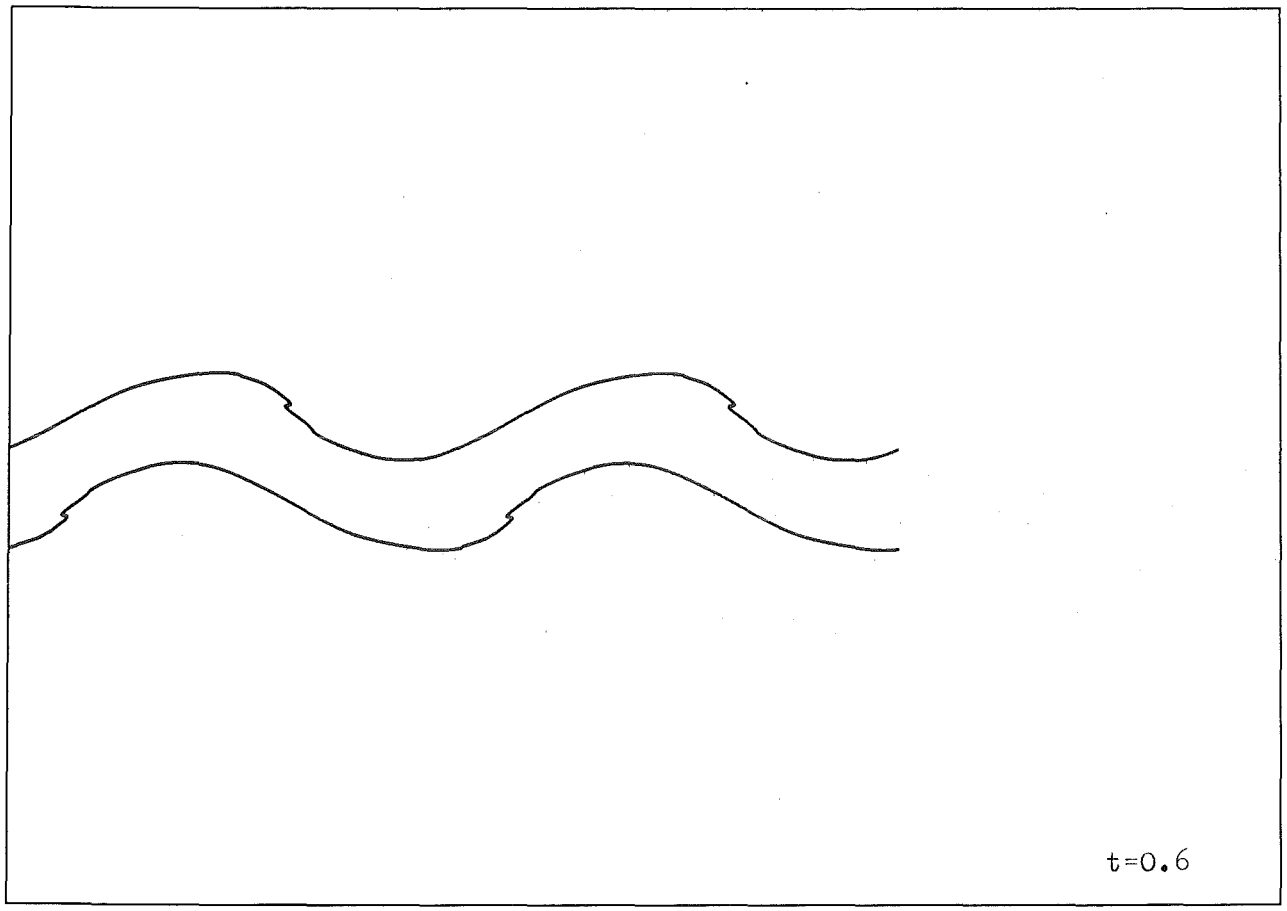


Fig. 6.19: Numerical simulation of a jet at time t = 0.0, 0.4



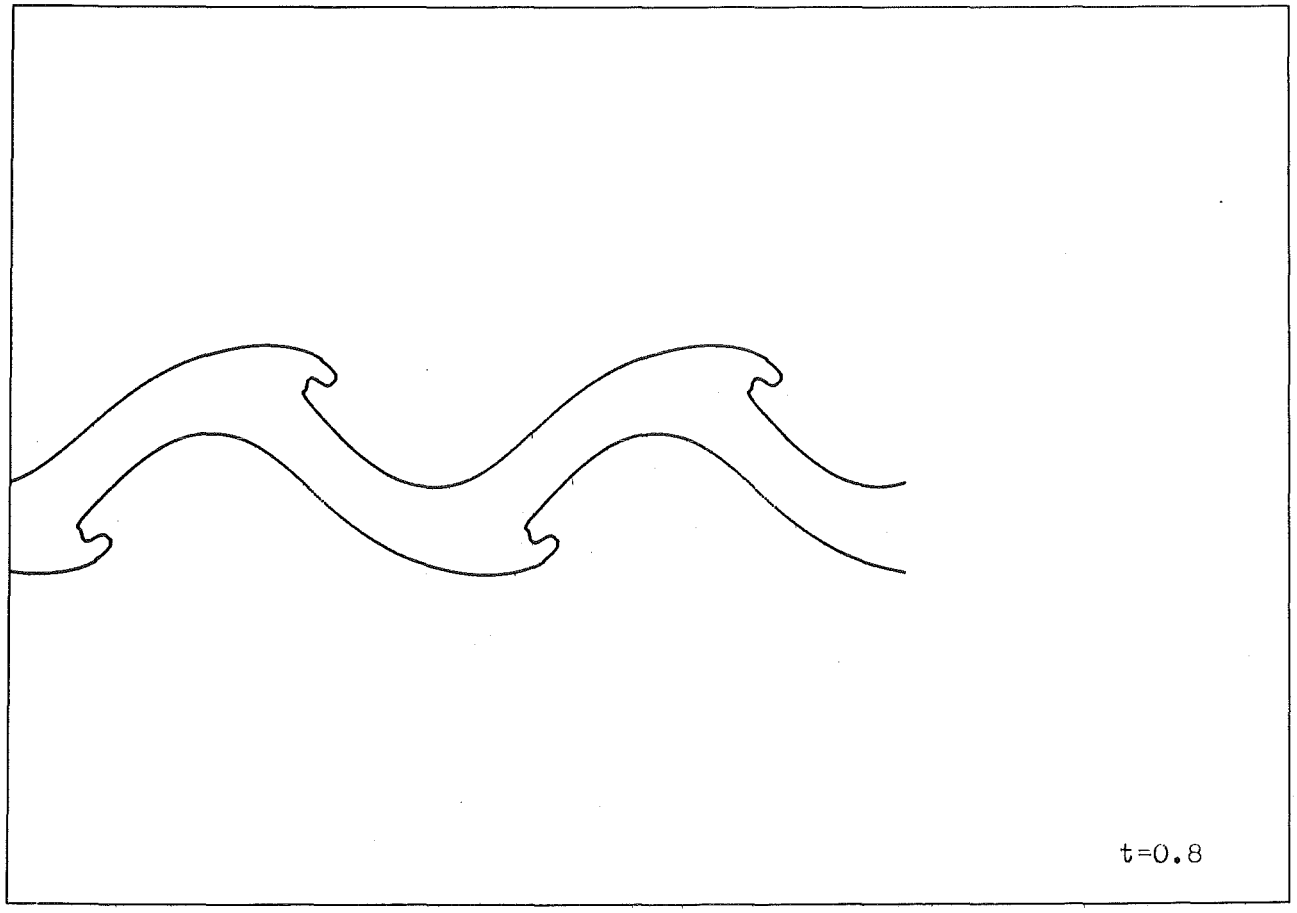
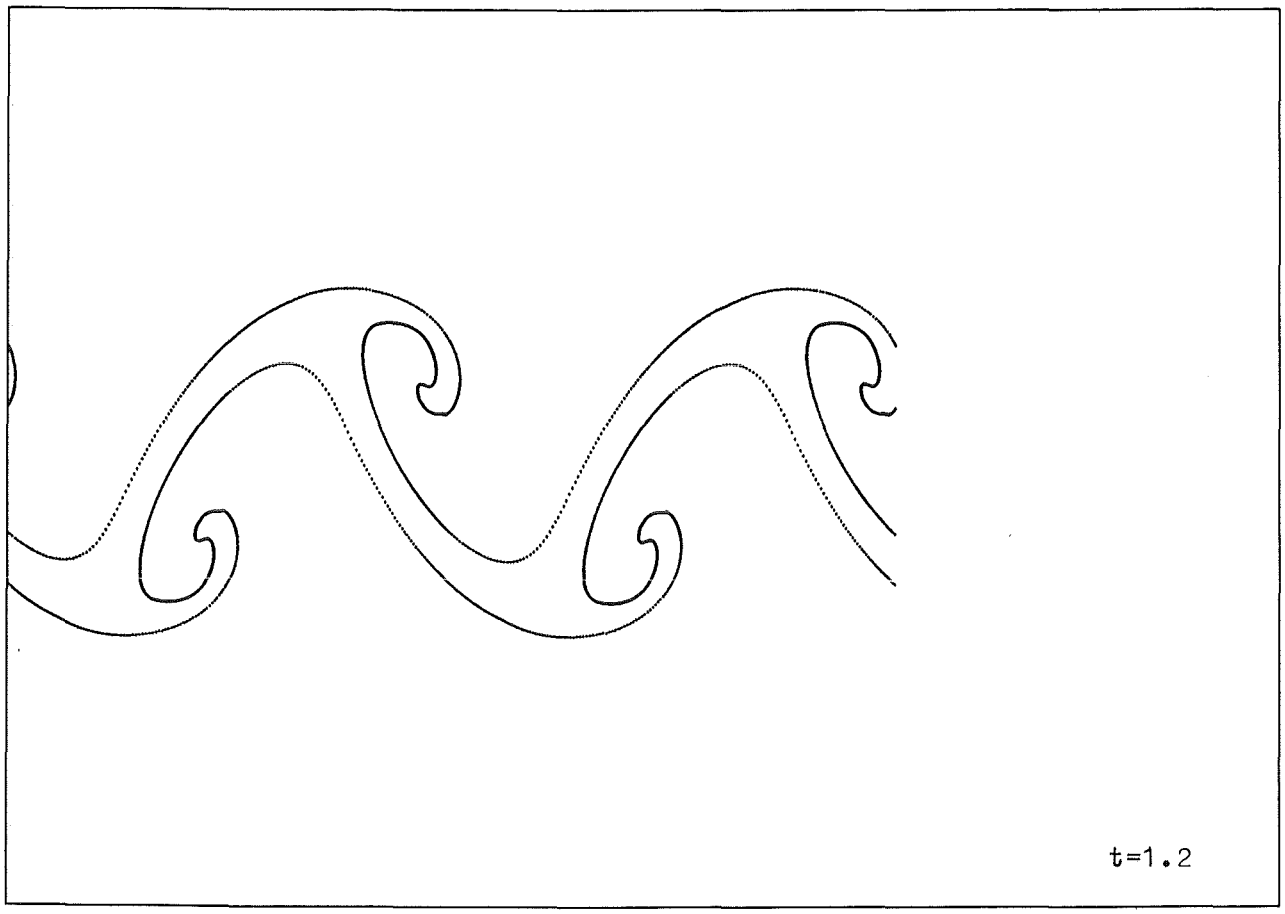


Fig. 6.20: Numerical simulation of a jet at time t = 0.6, 0.8



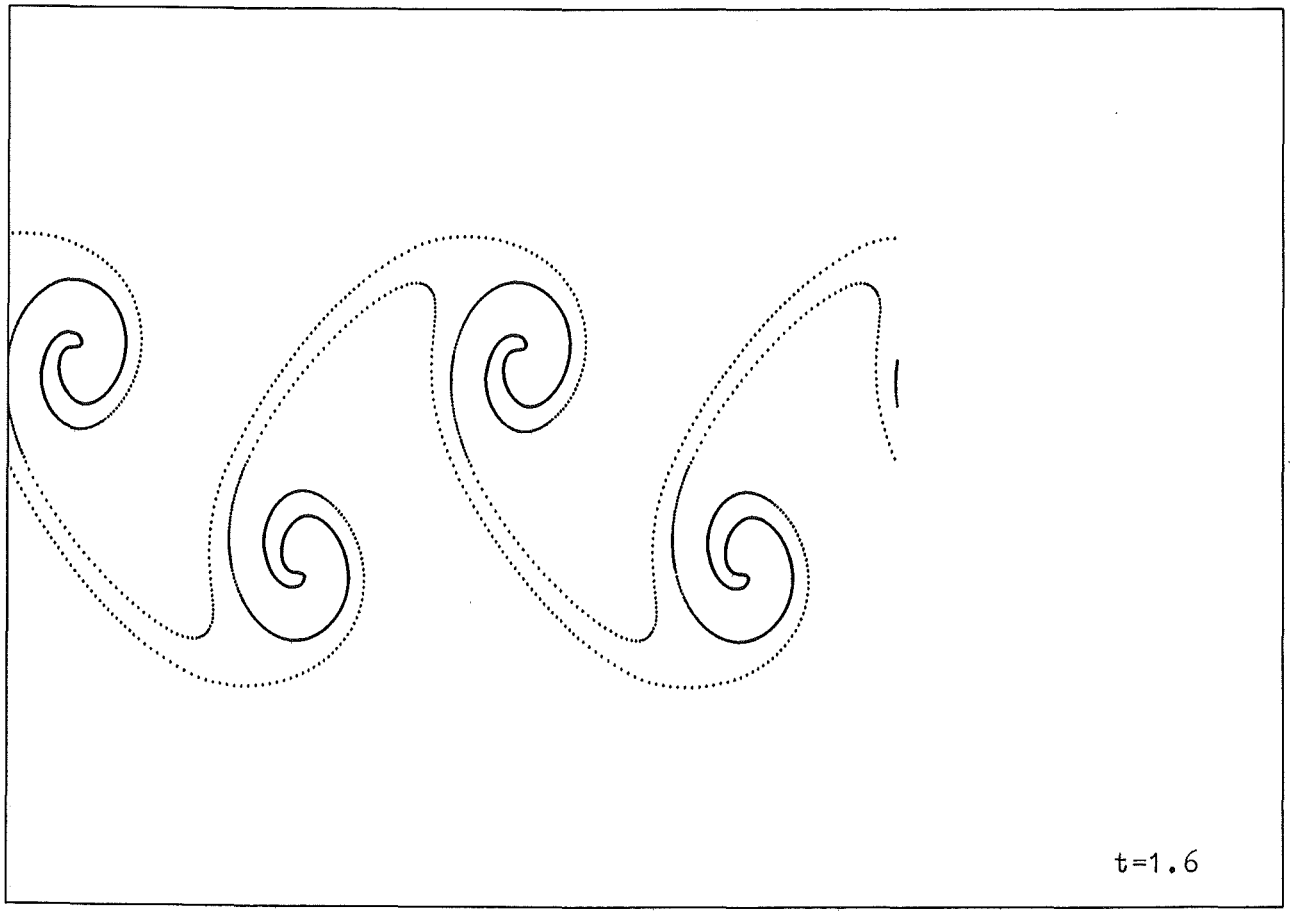
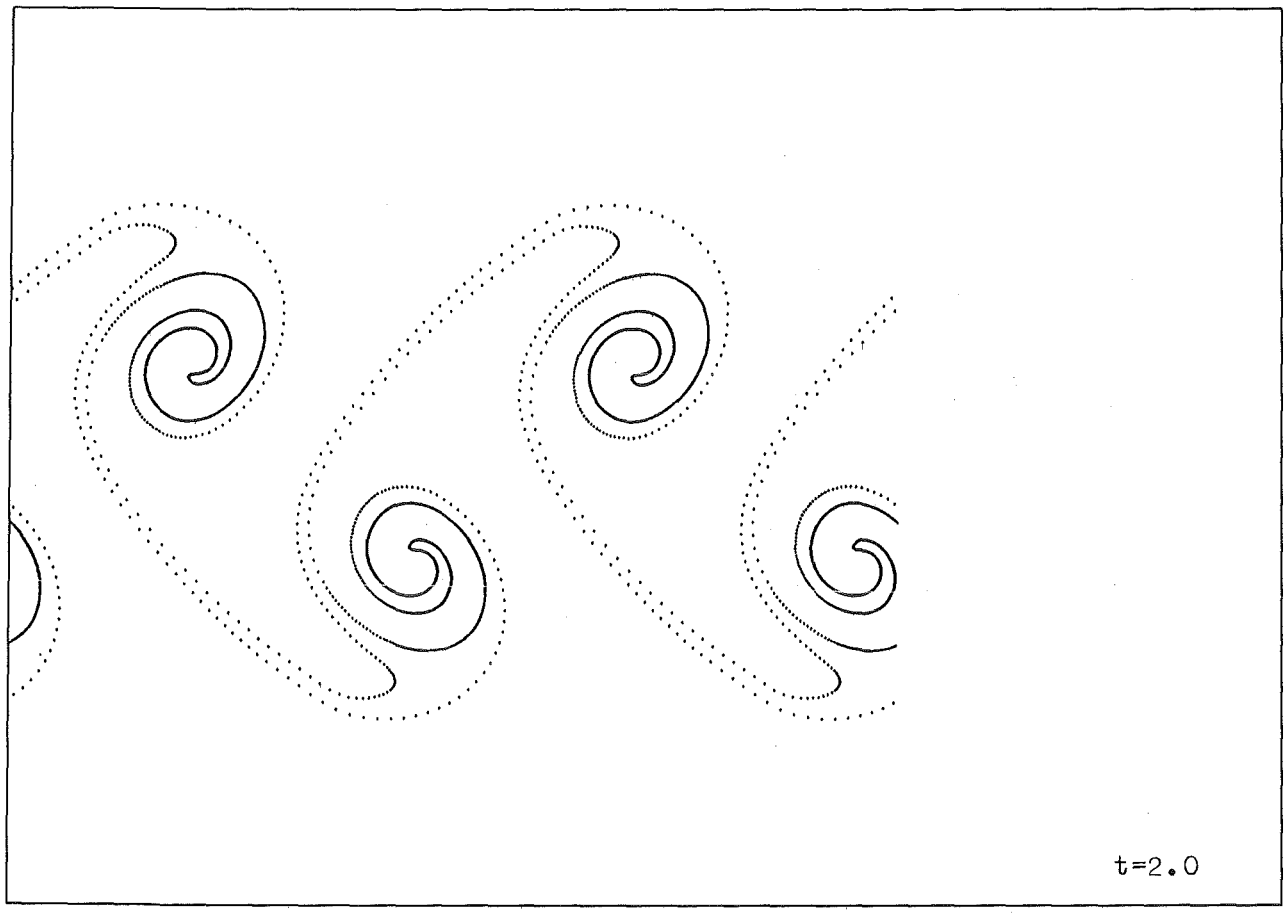


Fig. 6.21: Numerical simulation of a jet at time t = 1.2, 1.6



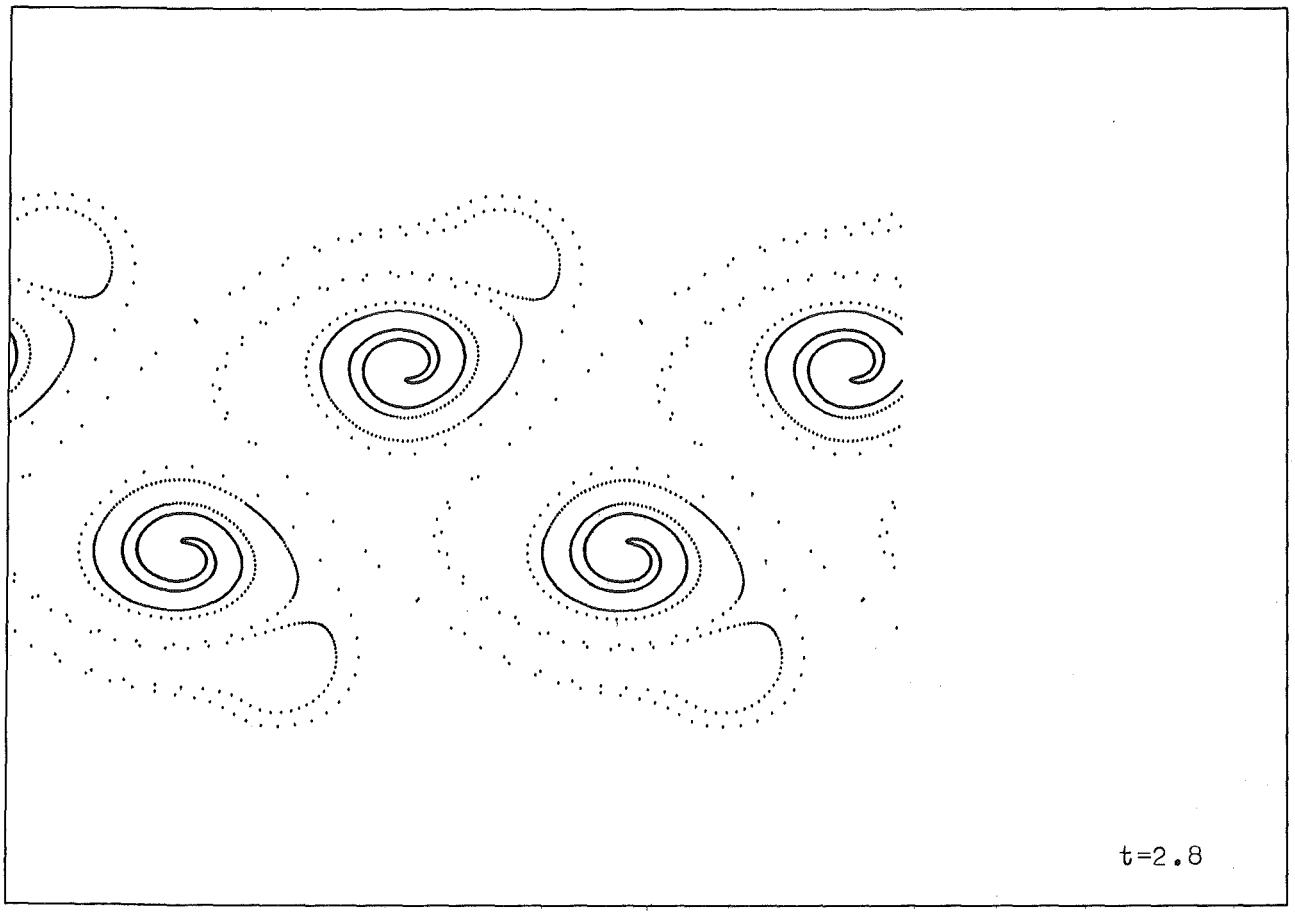
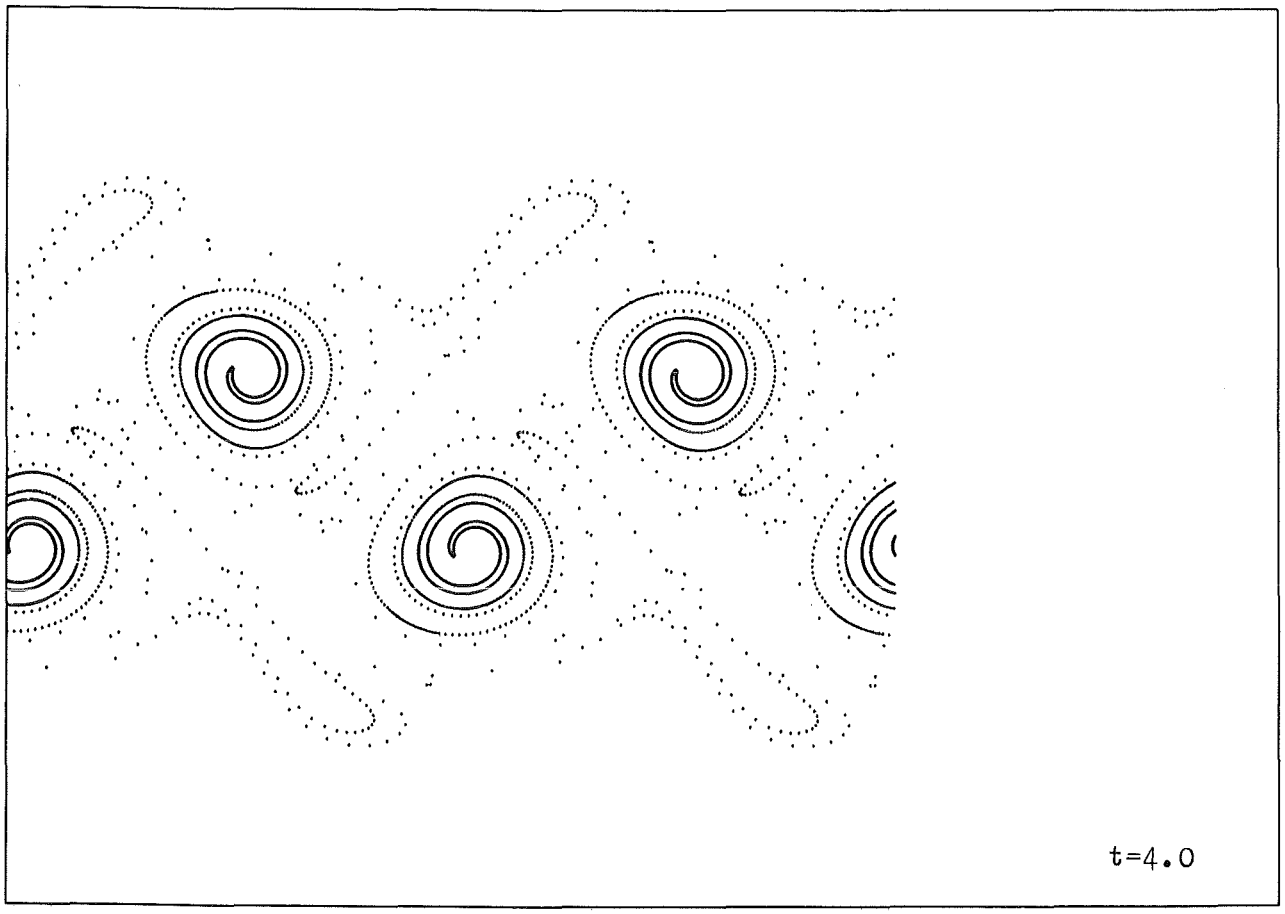


Fig. 6.22: Numerical simulation of a jet at time t = 2.0, 2.8



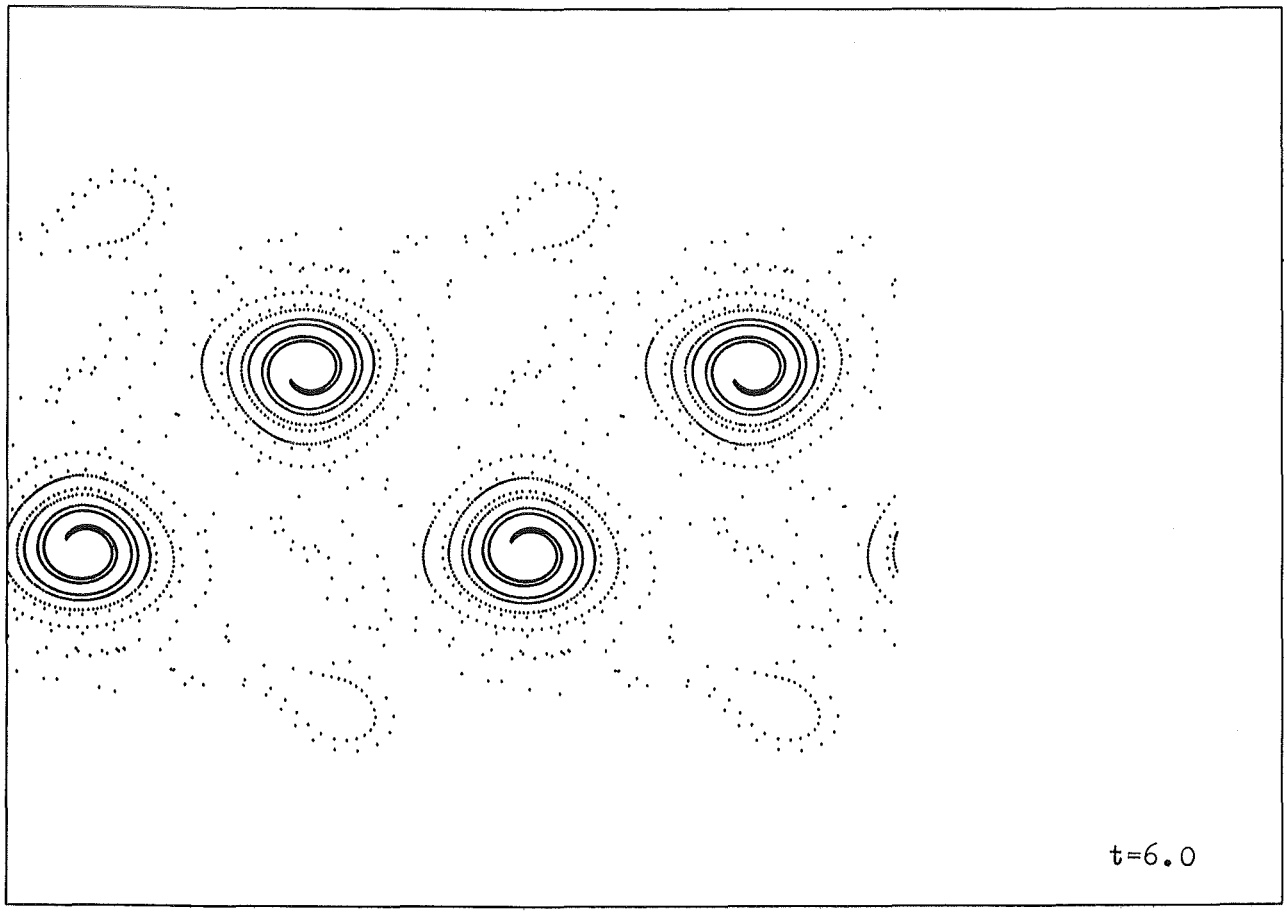


Fig. 6.23: Numerical simulation of a jet at time t = 4.0, 6.0

a circular cylinder. Figure 6.23 shows the behavior of the Karman vortex street after larger times t = 4.0 and t = 6.0.

We obtained almost the same results using another perturbation of the initial data. The diagram of the initial values is outlined in Fig. 6.24. The surfaces of the flows are now given by

$$S_1 : y = 0.05 \tag{6.10}$$

$$S_2 : y = -0.05 \tag{6.11}$$

The fluid between these lines flows to the right at velocity u = -0.5, while the other part propagates into the opposite direction. The density and pressure are uniform in the whole domain. The sinusoidal perturbations of the initial values are introduced via perturbations of the velocity component v into y-direction. In the region B the velocity v depends on x:

$$v = v(x) = a \sin(4 \pi x) \tag{6.12}$$

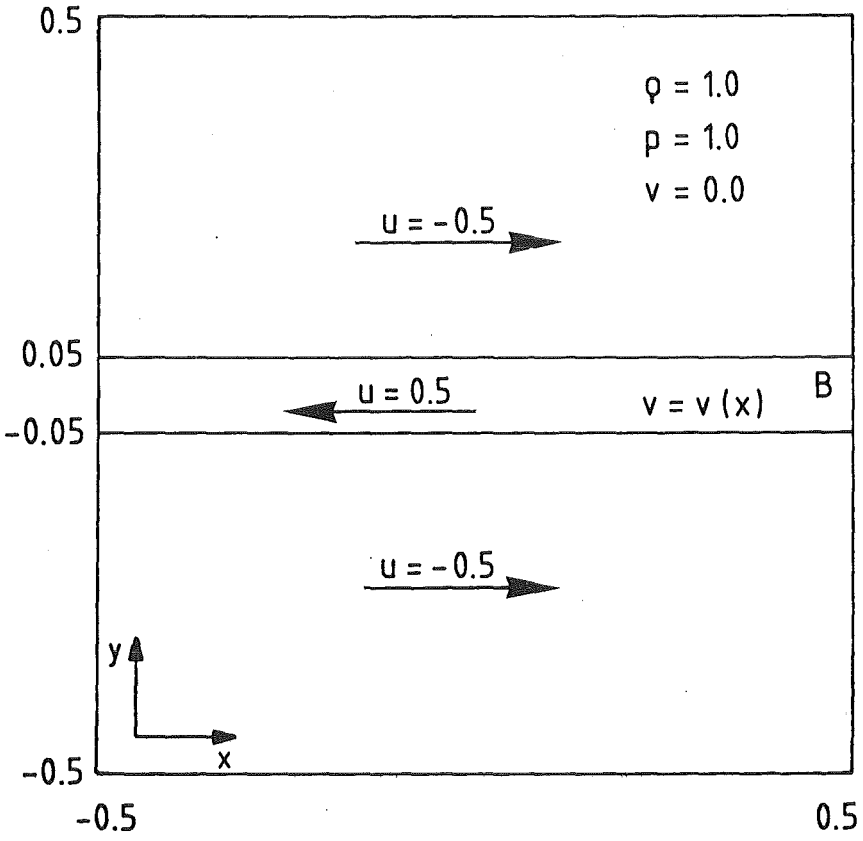
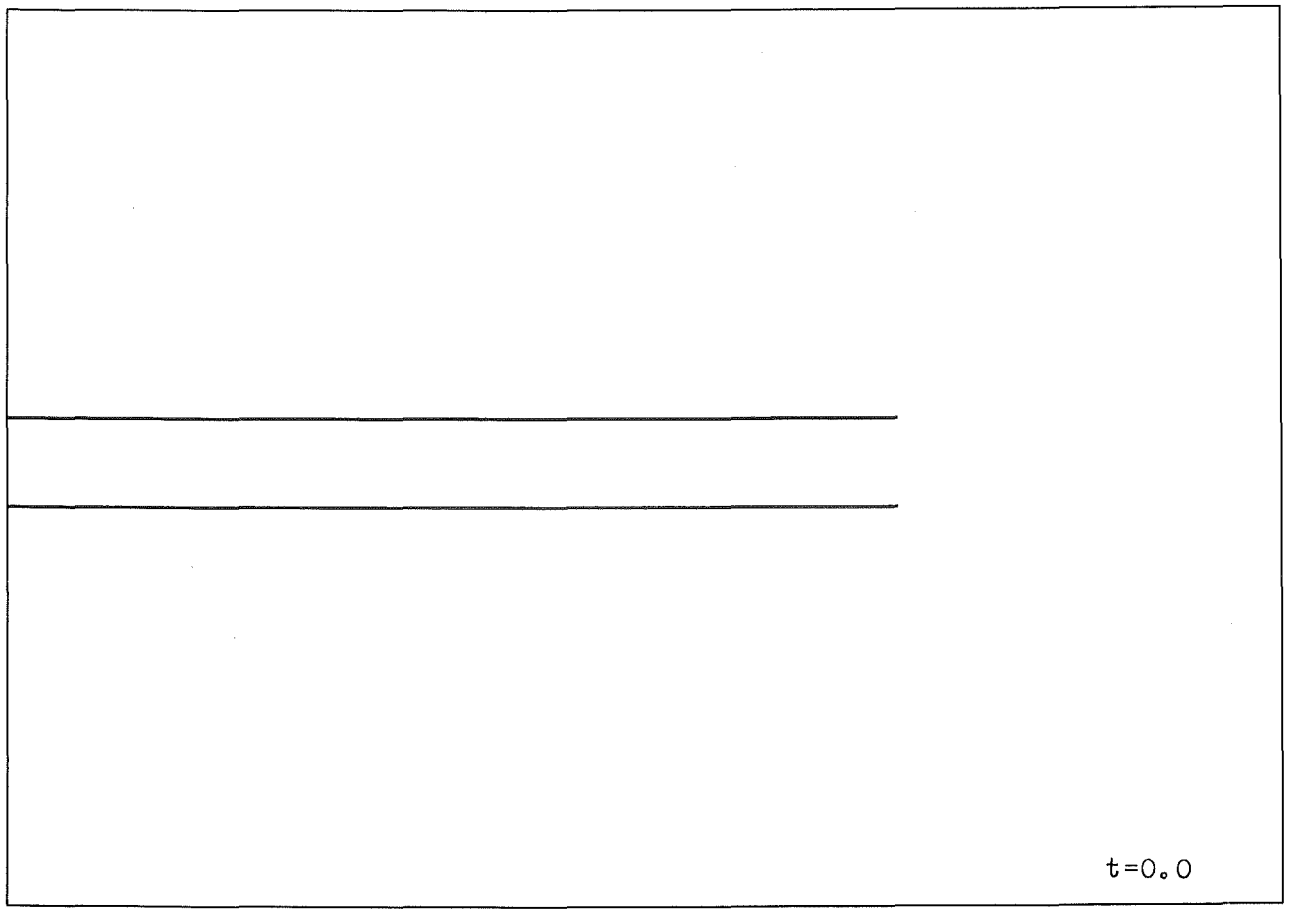


Fig. 6.24: Initial values for a jet



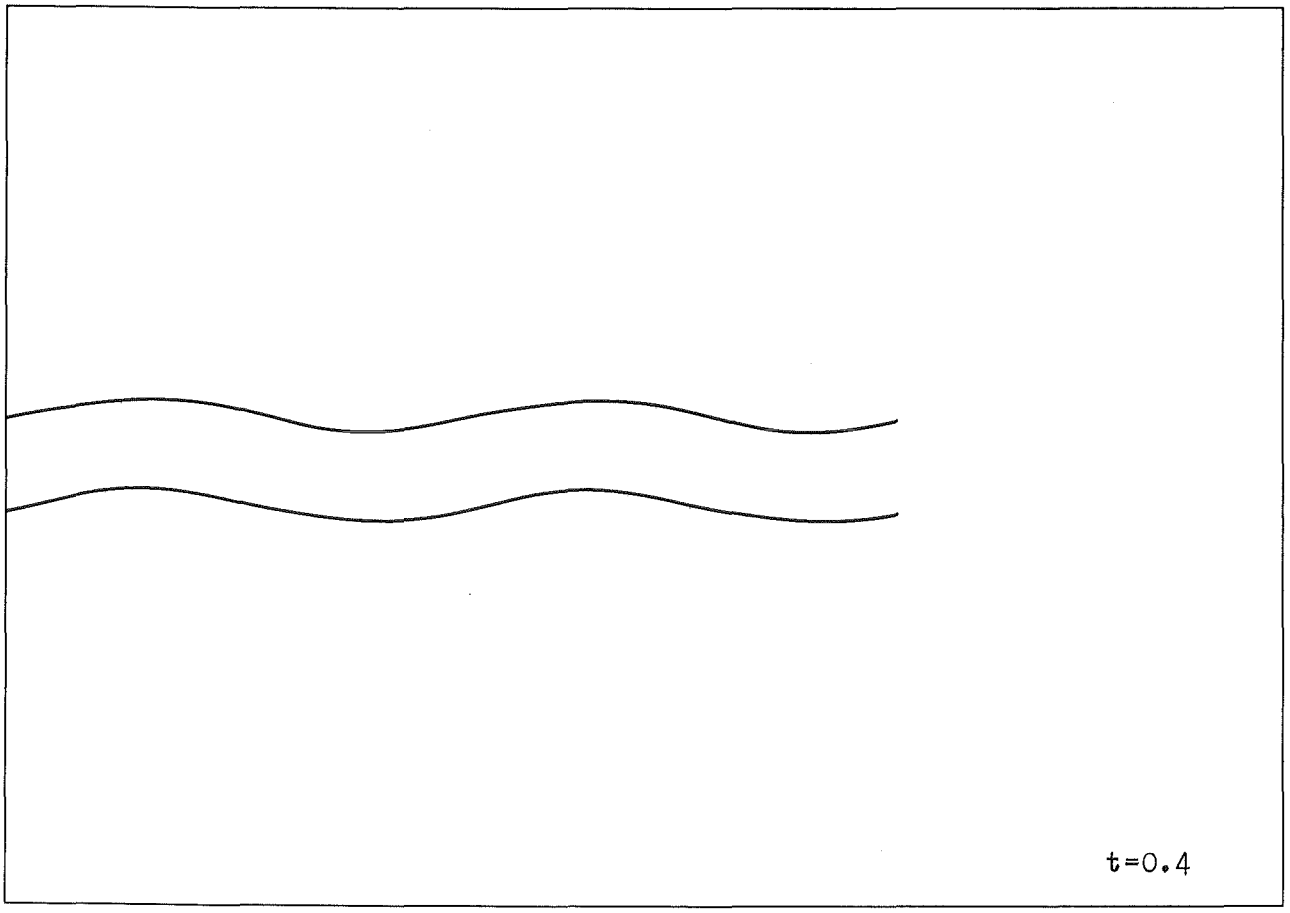
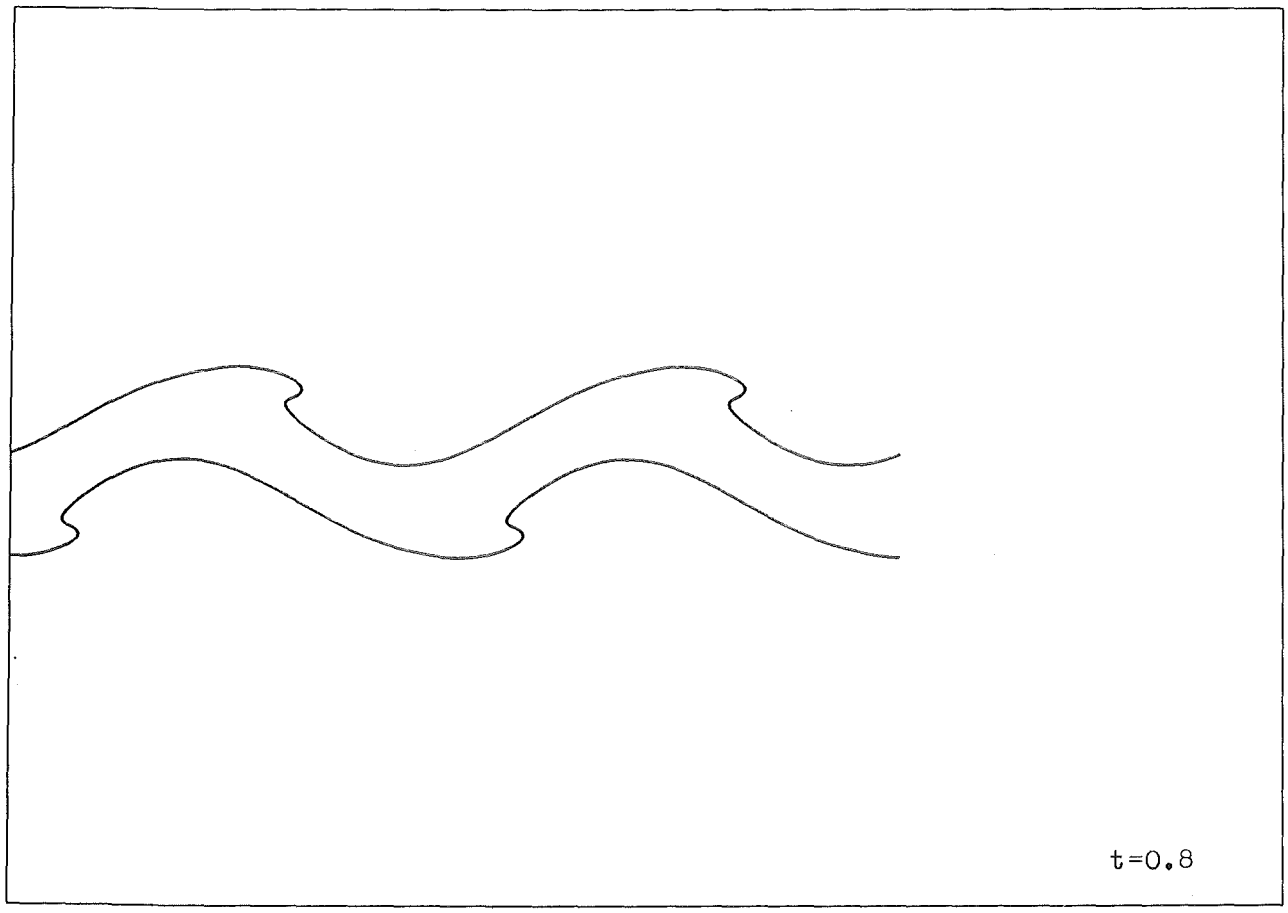


Fig. 6.25: Numerical simulation of a jet at time t = 0.0, 0.4



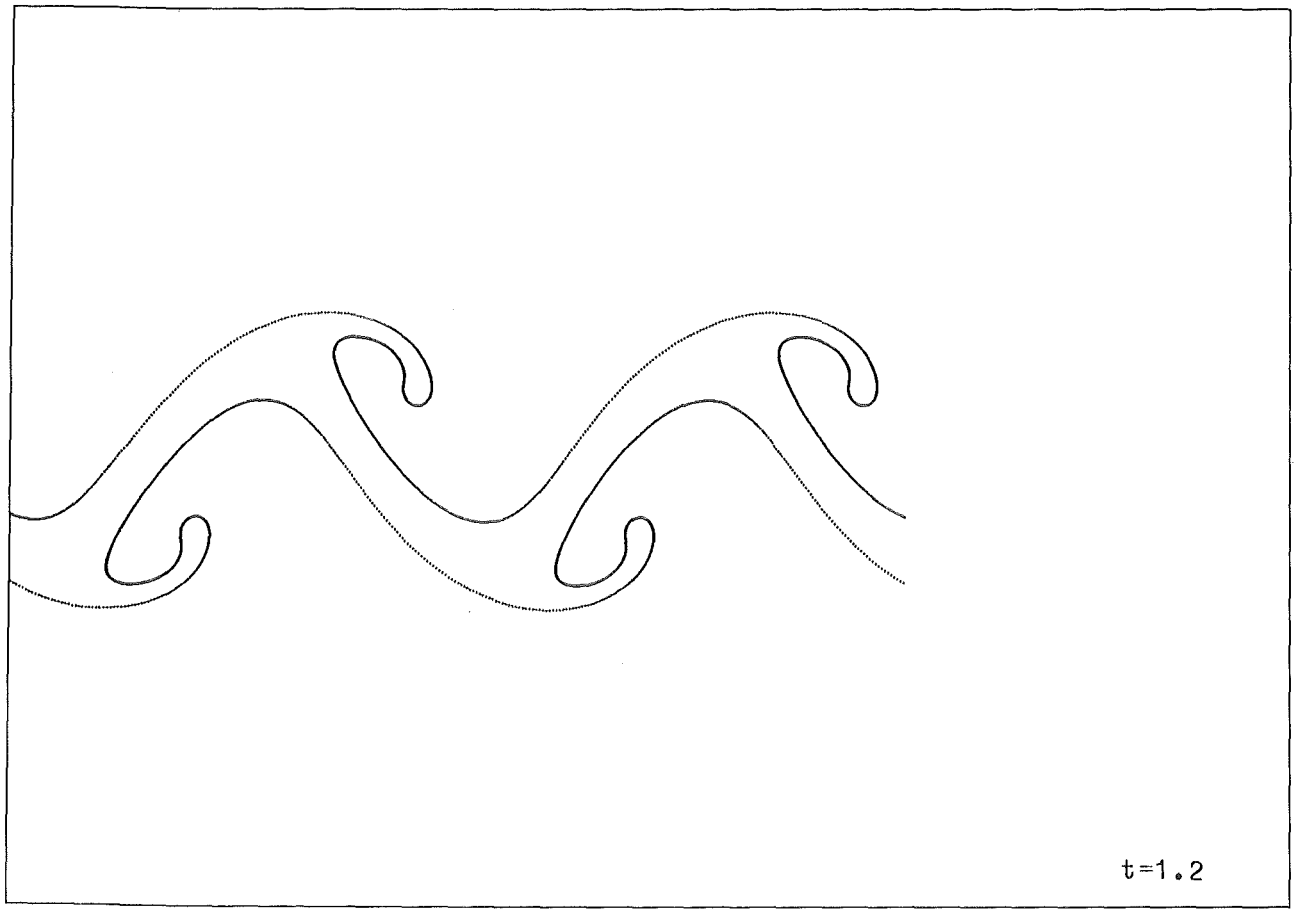
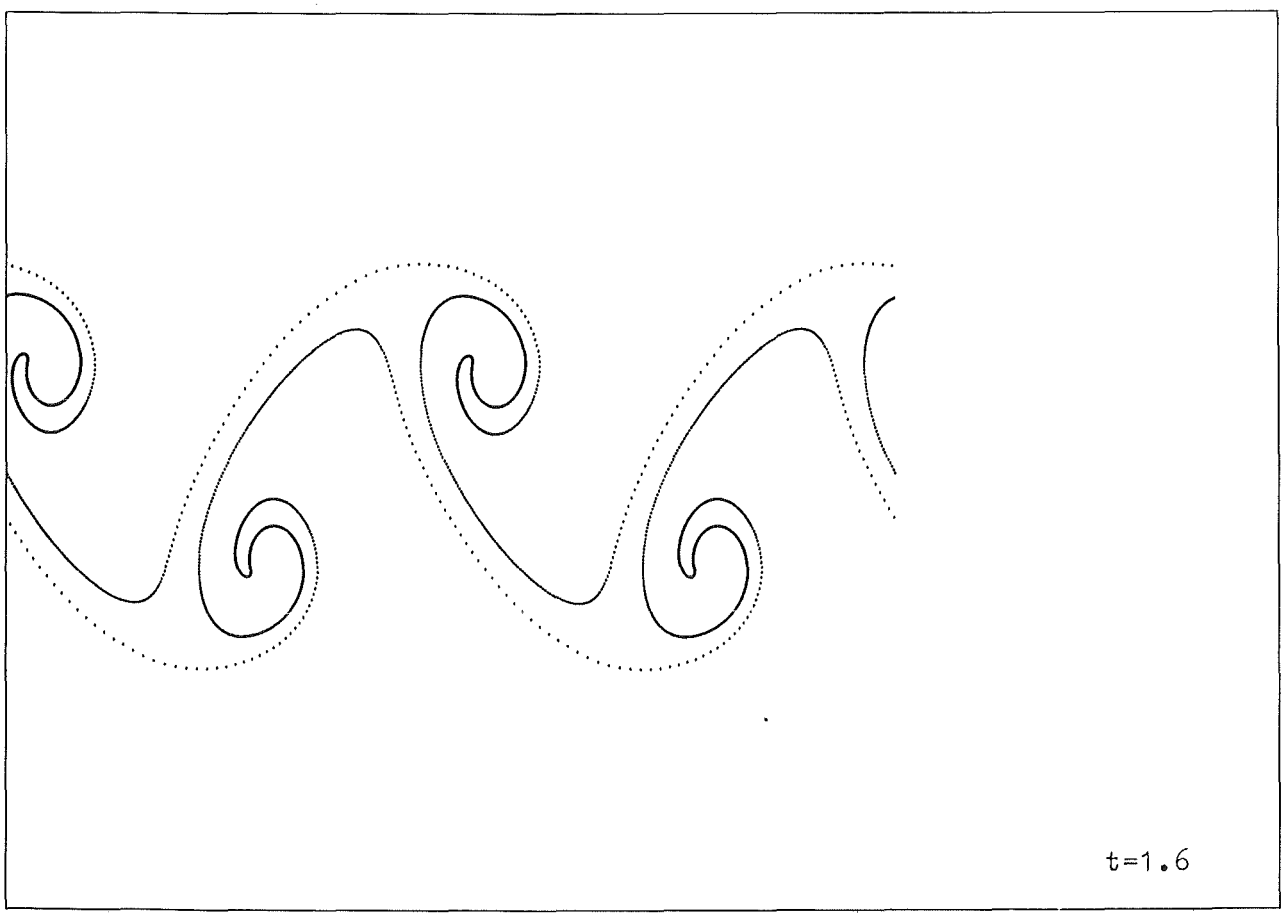


Fig. 6.26: Numerical simulation of a jet at time t = 0.8, 1.2



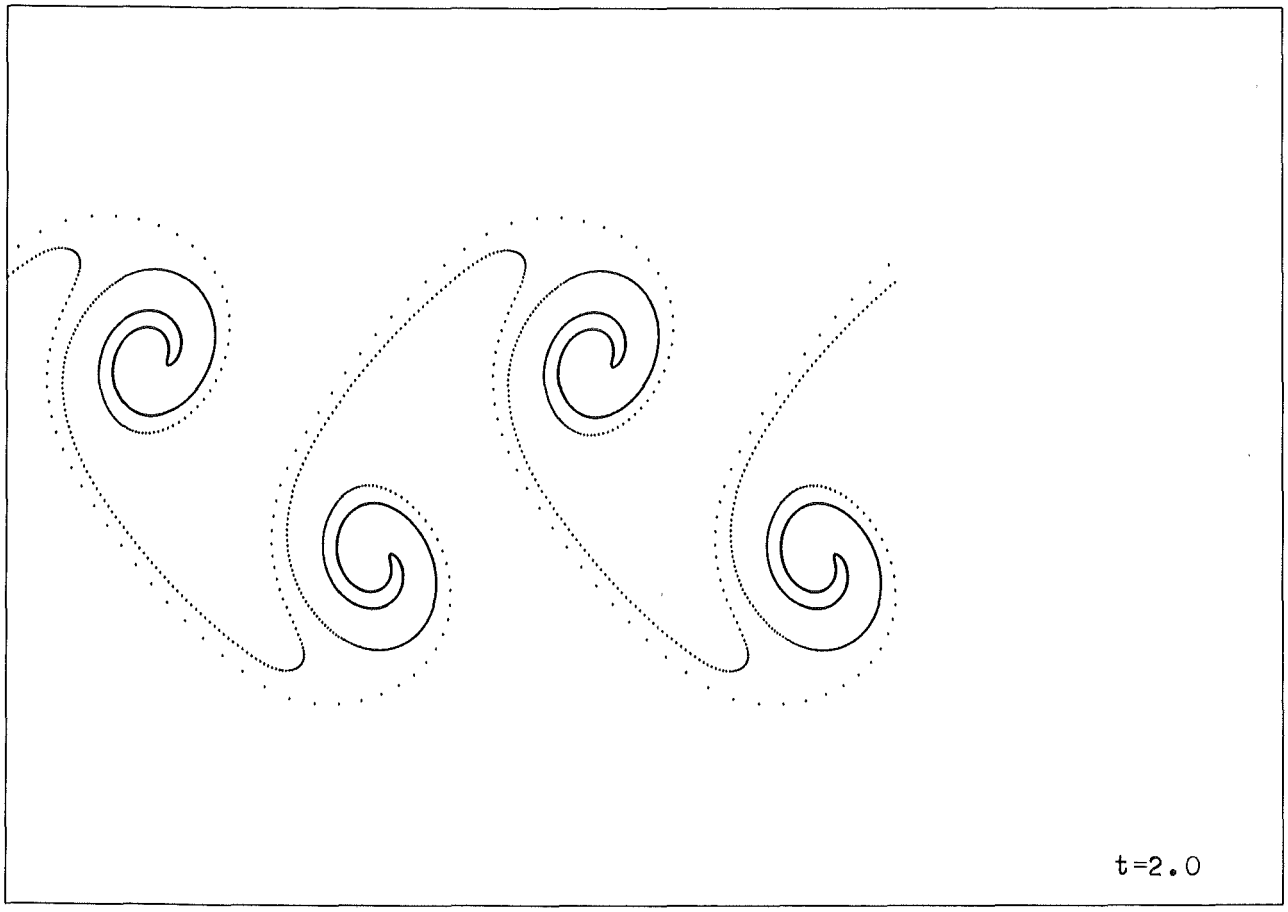
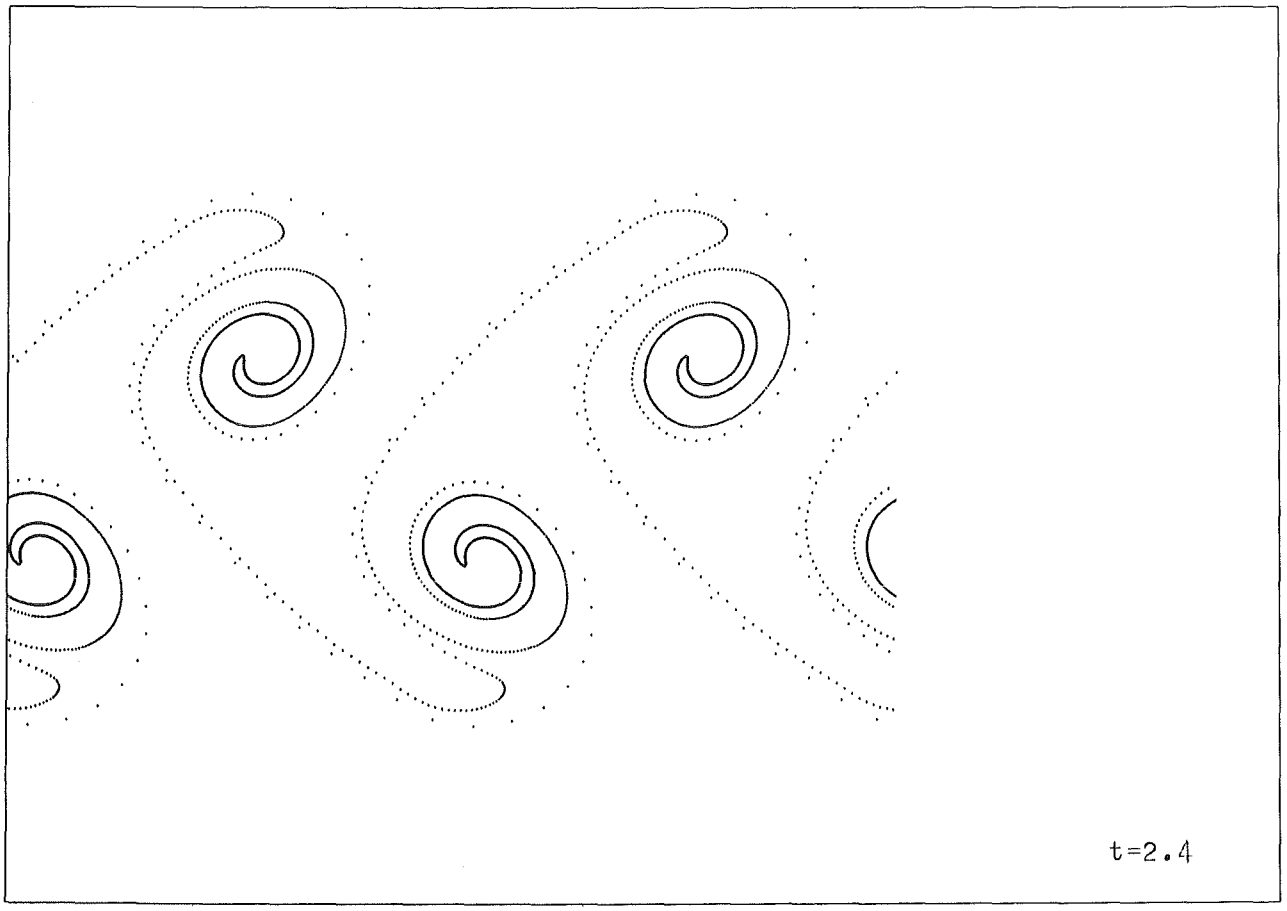


Fig. 6.27: Numerical simulation of a jet at time t = 1.6, 2.0



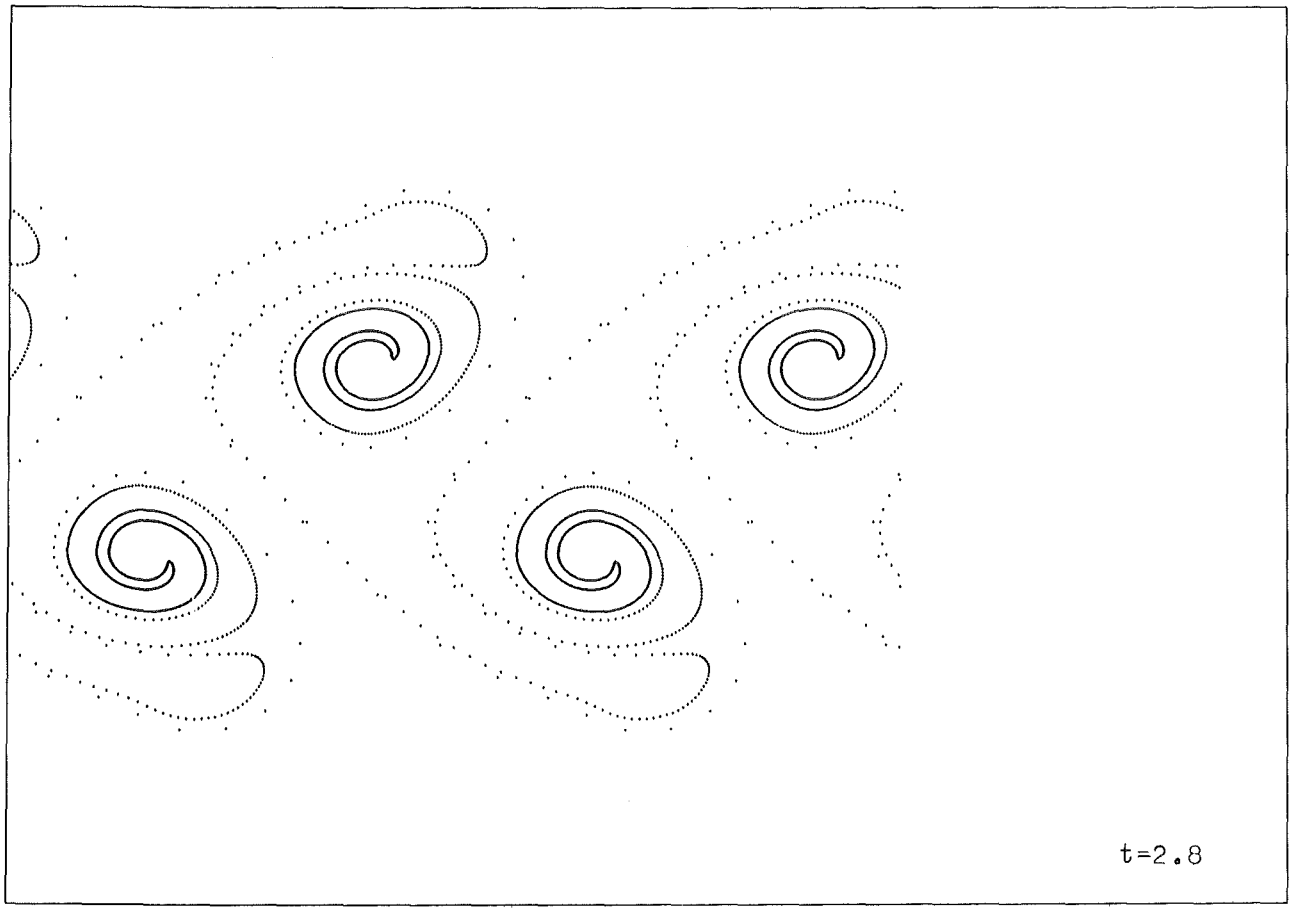
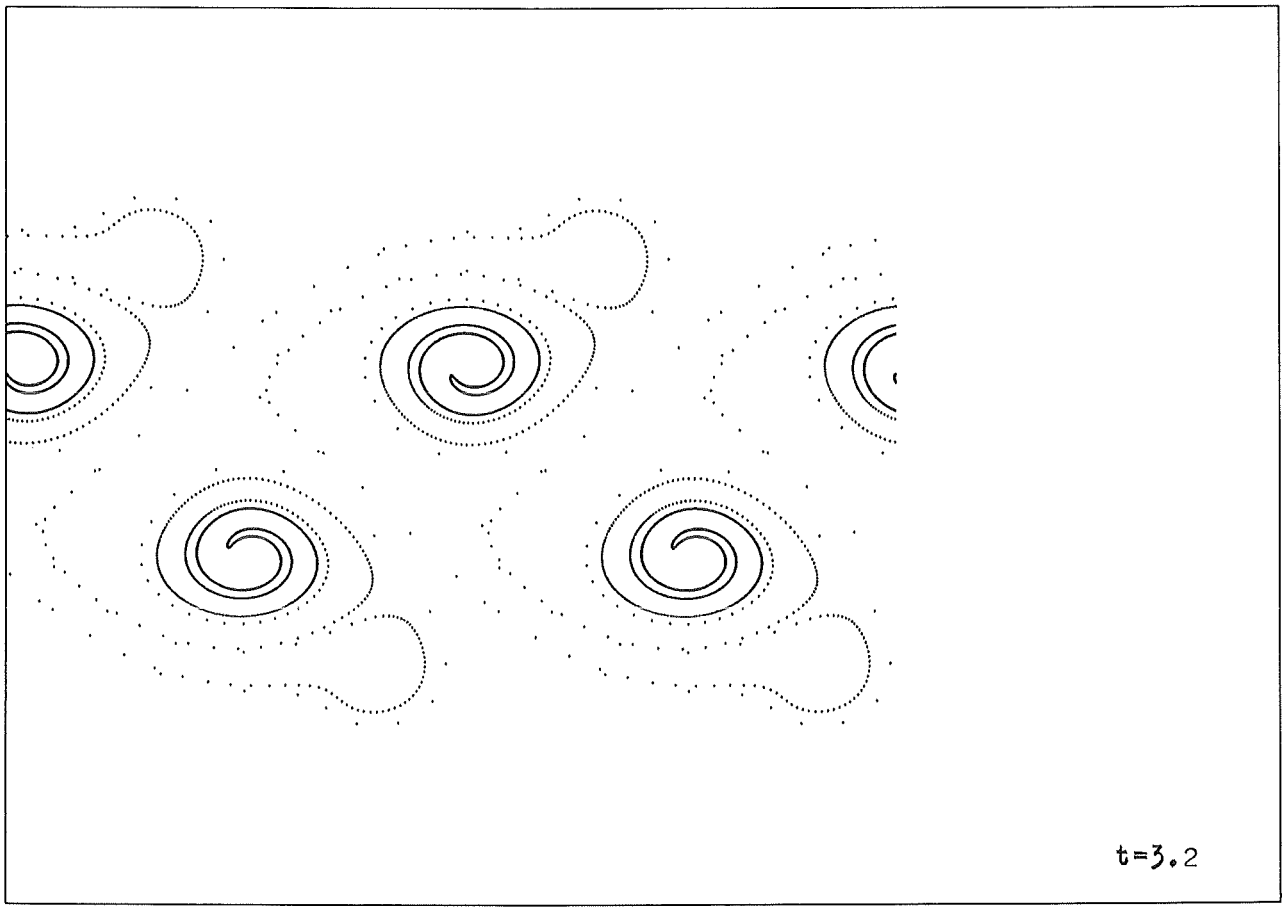


Fig. 6.28: Numerical simulation of a jet at time t = 2.4, 2.8



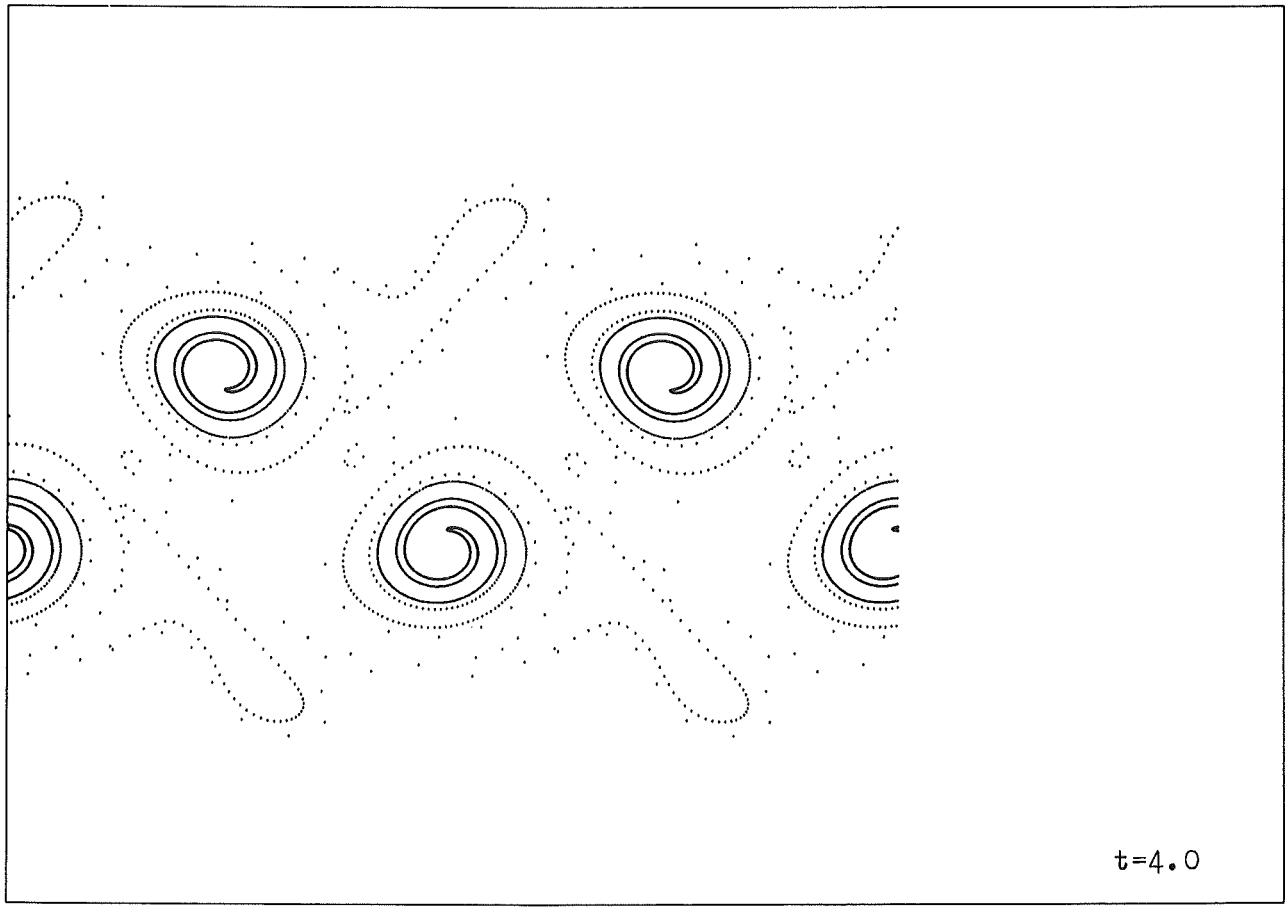


Fig. 6.29: Numerical simulation of a jet at time t = 3.2, 4.0

As in the calculations of a jet above the computational domain is $R = [-0.5, 0.5] \times [-0.5, 0.5]$. This domain is divided into 100 x 100 grid zones. The boundary conditions are periodic or reflecting wall conditions, respectively. The marker particles are again located at the interfaces S_1, S_2. As in the previous calculations we used here 2000 marker particles. The Figures 6.25 - 6.28 show the numerical results for the amplitude $a = 0.1$ of the initial perturbation (6.10). The numerical results are almost identical to the results obtained in the case of sinusoidal perturbation (6.8), (6.9) of the initial data.

7. Conclusions

The results show that a high resolution scheme for the Euler equations combined with a Langrangian tracking of interfaces gives a very efficient method for the numerical simulation of hydrodynamic instabilities. The high resolution scheme, presented here, is designed for an efficient implementation on a vector computer. By that we are able to perform large scale computations on fine grids. For zero viscosity the two-dimensional instabilities, considred here, are ill posed in the sense that small initial disturbances introduced by the grid will rapidly increase. The perturbations which destroy the rollup of the shear layers do not appear if a small amount of viscosity is added to the Euler equations. In our calculations the viscosity is introduced by numerical dissipation. The rollup of the shear layers is stabilized by the numerical dissipation.

Several questions remain open and will be investigated in the future. We have started to compare our results with those obtained by schemes which possess les numerical dissipation but where physical viscosity is introduced.By that we can estimate the Reynolds number corresponding to the numerical dissipation of the results presented in this paper. In a next paper we will compare these calculations for different Reynolds numbers with fluid mechanical experiments and theory.

References

/1/ Birkhoff, G.: Helmholtz and Taylor instability, in Hydrodynamic Instability, G. Birkhoff, R. Bellman, C.C. Lin (Eds.), AMS 1962, 55 - 76.

/2/ Birkhoff, G. and Fisher, J.: Do vortex sheets roll up? Rend. Circ. Mat. Palermo (2) vol. 8 (1959), 77 - 90.

/3/ Chern, I-L., Glimm, J., Mc Bryan, O., Plohr, B. and Yaniv, S.: Front tracking for gas dynamics, J. Comput. Phys. 62 (1986), 83 - 110.

/4/ Colella, P.: A direct Eulerian MUSCL scheme for gas dynamics, SIAM J. Sci. Stat. Comput. 6 (1985), 104 - 117.

/5/ Colella, P. and Glaz, H.M.: Efficient solution algorithms for the Riemann problem for real gases, J. Comput. Phys. 59 (1985), 264 - 289.

/6/ Colella, P. and Woodward, P.R.: The piecewise parabolic method (PPM) for gas dynamical simulation, J. Comput. Phys. 54 (1984), 174 - 201.

/7/ Daly, B.J.: Numerical study of two fluid Rayleigh-Taylor instability, Physics of Fluids 10 (1967), 297 - 307.

/8/ Dyke, M. van: An Album of Fluid Motion, Parabolic Press, Stanford, California 1982.

/9/ Einfeldt, B.: On Godunov-type methods for gas dynamics, to appear in SIAM J.Numer. Anal. 25 (1988).

/10/ Einfeldt, B. and Munz, C.D.: Die numerische Behandlung zweidimensionaler Strömungen mit starken Stoßwellen, KfK-Bericht 4191, Kernforschungszentrum Karlsruhe 1987.

/11/ Harten, A.: High resolution schemes for hyperbolic conservation laws, J. Comput. Phys. 49 (1983), 357 - 393.

/12/ Harten, A.: The artificial compression method for computation of shocks and contact discontinuities. I. Single conservation laws, Comm. Pure Appl. Math. 30 (1977), 611 - 638.

/13/ Harten, A., Lax, P.D. and van Leer, B.: On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25 (1983), 35 - 62.

/14/ Huang, L.C.: Pseudo-unsteady difference schemes for discontinuous solutions of steady-state, one-dimensional fluid dynamics problems, J. Comput. Phys. 42 (1981), 195 - 211.

/15/ Lax, P.D. and Wendroff, B.: Systems of conservation laws, Comm. Pure App. Math. 13 (1960), 217 - 237.

/16/ Leer, B. van: On the relation between the upwind-differencing schemes of Godunov, Engquist-Osher and Roe, SIAM J. Sci. Stat. Comput. 5 (1984), 1 - 21.

/17/ Leer, B. van: Flux-vector splitting for the Euler equations, Lecture Notes in Physics, Vol. 170 (1982), 507 -- 512.

/18/ Leer, B. van: Towards the ultimate conservative difference scheme, V. A second-order sequel to Godunov's method, J. Comput. Phys. 32 (1979), 101 - 136.

/19/ Moore, D.W., Numerical and analytical aspects of Helmholtz instability, in Theoretical and Applied Mechanics, F.I. Niordson, N. Olhoff (Eds.), Elsevier Science Publishers (North Holland) 1985, 263 - 274.

/20/ Munz, C.D.: On the numerical dissipation of high resolution schemes for nonlinear hyperbolic conservation laws, to appear in J. Comput. Phys.

/21/ Munz, C.D.: On the construction and comparison of two-step schemes for the Euler equations, in Finite Approximations in Fluid Mechanics 14, Vieweg 1986, 195 - 217.

/22/ Roe, P.L.: Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (1981), 357 - 372.

/23/ Seldner, D. and Westermann, T.: Numerische Algorithmen für zweidimensionale Teilchen-Simulationsmodelle in technisch relevanten Geometrien, KfK-Bericht 4282, Kernforschungszentrum Karlsruhe 1987.

/24/ Steger, J.L. and Warming, R.F.: Flux vector splitting of the inviscid gas dynamics equations with application to finite difference methods, J. Comput. Phys. 40 (1981), 263 - 293.

/25/ Stoer, J., Bulirsch, R.R.: Einführung in die Numerische Mathematik II, Springer Verlag, Berlin/Heidelberg/New York 1978.

/26/ Strang, G.: On the construction and comparison of difference schemes, SIAM J. Numer. Anal. 5 (1986), 506 - 517.

/27/ Woodward, P. and Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. J. Comput. Phys. 54 (1984), 115 - 173.