UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Subscale Capturing in Computational Fluid Dynamics

Stanley Osher

March 1995

CAM Report 95-15

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

Subscale Capturing in Computational Fluid Dynamics

Stanley Osher*
Mathematics Department
University of California
Los Angeles, CA, USA
90095-1555

Abstract

We shall describe numerical methods which were devised for the purpose of computing small scale behavior in fluid dynamics without either fully resolving the whole solution or explicitly tracking certain singular parts of it. These include shock-capturing and front-capturing. These methods have recently proven useful in many fields of physics and engineering and even image processing and computer vision. This is an overview of many collaborations – key among them is joint work on ENO schemes with A. Harten, B. Engquist, S. Chakravarthy and C.-W. Shu, joint development of the level set technique with J. Sethian, and more recent work in this area with E. Harabetian, B. Merriman, P. Smereka, M. Sussman, T. Hou and graduate students M. Kang and S. Chen.

1 Introduction

In this paper we shall describe numerical methods which were devised for the purpose of computing small scale behavior in fluid dynamics without either fully resolving the whole solution or explicitly tracking certain singular parts of it. Techniques developed for this purpose include: shock-capturing and front-capturing. These methods have recently proven useful in many of fields of physics and engineering and even image processing and computer vision.

Shock capturing methods were devised for the numerical solution of nonlinear conservation laws. At the 1990 meeting of the International Congress of Mathematicians, Ami Harten [10] gave an overview of recent developments in that area, culminating in the construction of essentially nonoscillatory (ENO) schemes [11], [12]. We shall describe some of the ideas and results relating to this subject in section 3.

In 1987, together with J.A. Sethian [19] we devised a new numerical procedure for capturing fronts and applied it to curves and surfaces whose speeds depend on local curvature. The method uses a fixed (Eulerian) grid and finds the front as particular level set (moving with time) of a scalar function. The method applies to a very general class of problems.

The technique handles topological merging and breaking, works in any number of space dimensions, does not require that the moving surface be written as a function, captures sharp gradients and cusps in the front, and is relatively easy to program. Theoretical justification, involving the concept of viscosity solutions, has been given in [6], [4].

Many applications and extensions have recently been found. We shall describe the method and some applications in section 2. We also note that the motion of multiple junctions using related ideas has been studied in [17]. A particularly novel application and extension (done with E. Harabetian) is to the numerical study of unstable fronts – e.g. vortex sheets, in [9], [8]. This will

*Research supported by DARPA/ONR-N00014-92-J-1890, ARO DAAL03-91-G0162, and NSF DMS-91-03104

also be described in section 2. The level set formulation allows for the capturing of the front with minimal regularization because the zero level set of a continuous function can become quite complicated, even though the function itself is easy to compute. There is also a novel "topological regularization" inherent to the level set approach as described in [9], [8].

2 The Level Set Method for Capturing Moving Fronts

In a variety of physical phenomena, one wishes to follow the motion of a front whose speed is a function of the local geometry and an underlying flow field. Generally the location of the interface or front affects the flow field. Typically there have been two types of numerical algorithms employed in the solution of such problems. The first parameterizes the moving front by some variable and discretizes this parameterization into a set of marker points. The positions of these marker points are updated according to approximations of the equations of motion. For large complex motion, several problems occur. First, marker particles come together in regions where the curvature builds, causing numerical instability unless regridding is used. The regridding mechanism often dominates the real effects. Moreover the numerical methods tend to become quite stiff in these regions – see e.g. [23]. Secondly, such methods suffer from topological problems: e.g. when two regions merge or a single region splits, ad-hoc technologies are required.

Other algorithms commonly employed fall under the category of "volume of fluid" techniques which track the motion of the interior region e.g. [18], [1]. These are somewhat more adaptable to topological changes than the tracking methods but still lack the ability to easily compute geometrical quantities such as curvature of the front.

Both methods are difficult to implement in three space dimensional problems. Our idea, as first developed with J.A. Sethian in [19] is as follows. Given a region Ω in R^2 or R^3 (which could be multiply connected), and whose boundary is moving with time, we construct an auxiliary function $\varphi(\bar{x},t)$ which is Lipschitz continuous and has the property

$$\varphi(\bar{x},t) > 0 \quad \Leftrightarrow \quad \bar{x}\ \varepsilon\ \Omega \quad \text{at time } t \tag{2.1}$$

$$\varphi(\bar{x},t) < 0 \quad \Leftrightarrow \quad \bar{x}\ \varepsilon\ \Omega^c \quad \text{at time } t \tag{2.2}$$

$$\varphi(\bar{x},t) = 0 \quad \Leftrightarrow \quad \bar{x}\ \varepsilon\ \partial\Omega \quad \text{at time } t \tag{2.3}$$

On any level set of φ we have

$$\varphi_t + \vec{u}\cdot\nabla\varphi = 0 \tag{2.4}$$

where $\vec{u} = (\dot{x}(t), \dot{y}(t))$, the motion of the front and the set $\varphi \equiv 0$ characterizes $\partial\Omega$ at time t.

Generally, if the normal velocity $\vec{u}\cdot\vec{n}$ is a given function, f, of the geometry, the level set motion is governed by

$$\varphi_t + |\nabla\varphi| f = 0. \tag{2.5}$$

Typically (in 2 dimensions) f is a function of the curvature of the front, $f = f(\kappa) = f\left(\nabla\cdot\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right)\right)$. In this case we can replace (2.4) by an equation involving φ only

$$\varphi_t + |\nabla\varphi| f\left(\nabla\cdot\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right)\right) = 0. \tag{2.6}$$

Our algorithm is merely to extend (2.6) to be valid throughout space and just pick out the zero level set as the front at all later times. Equations of this type, for $f'(0) < 0$, have been analyzed in [6], [4] using the theory of viscosity solutions. In addition to well-posedness, it was shown that modulo a few exceptions, the level set method works. This means that the zero level set agrees with the classical motion for smooth, noninteracting curves. Moreover, the asymptotic behavior of certain fronts arising in reaction diffusion equations leads to this motion as the small parameter goes to zero [5].

In many applications involving multiphase flow in fluid dynamics the interface between any two regions can be represented by judiciously using delta functions as source terms in the equations of motion. This is true in particular for computing rising air bubbles in water, falling water drops in air, and in numerous other applications – see e.g. [30],[2], [29]. In fact surface tension often plays a role and this quantity is just proportional to curvature, here easy to compute. Thus an Eulerian framework is easily set up, using the level set approach, allowing phenomena such as merging of water drops, resulting in surface tension driven oscillations, and drops hitting the base and deforming [30], [29].

A key requirement here and elsewhere is that the level set function φ stay well behaved, i.e. $0 < c \leq |\nabla\varphi| \leq C$ for fixed constants (except for isolated points). In fact it would be desirable to set

$$|\nabla\varphi| = 1 \tag{2.7}$$

with the additional criteria (2.1, 2.2, 2.3). In other words, we wish to replace (at least near $\partial\Omega$) φ by d, signed distance to the boundary.

We can do this as described in [30], [29], through reinitialization after every discrete update of the system, in a very fast way by obtaining the viscosity solution of

$$d_\tau + (|\nabla d| - 1)H(\varphi) = 0 \tag{2.8}$$

for $\tau > 0$, in fact as $\tau \uparrow \infty$, with $d(\bar{x}, 0) = \varphi(x, t)$. Here $H(\varphi)$ is any smooth monotone function of φ with $H(0) = 0$.

ENO schemes for Hamilton-Jacobi equations, as defined in [19],[20] may be used to solve this. By the method of characteristics it is clear that, near $\partial\Omega$, which is the zero level set of φ, the steady state is achieved very quickly. We thus have a fast method of computing signed distance to an arbitrary set of closed curves in R^2 or surfaces in R^3.

We present some results on air bubbles in water, water drops in air, and an air, oil, water interface problem in section 4.

Another example of the use of this method in fluid dynamics involves area (or volume) preserving motion by mean curvature. This represents the simplified motion of foam and can be modelled simply by finding the zero level set of

$$\varphi_t = |\nabla\varphi| \left(\nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|} \right) - \bar{\kappa} \right) \tag{2.9}$$

where $\bar{k}$ is the average curvature of the interface. This last can be easily computed

$$\bar{\kappa} = \frac{\int\int_\Omega \left(\nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|} \right) \right) \delta(\varphi)|\nabla\varphi|}{\int\int_\Omega \delta(\varphi)|\nabla\varphi|}. \tag{2.10}$$

The distance reinitialization is used and the method easily yields merging and topological breaking, see [13]. More realistic models involving volume preserving acceleration by mean curvature are being developed and analyzed with the same group of people. We present a simple 3D example in section 4.

Another interesting example concerns Stefan problems. Earlier work was done using the level set formulation [24]. Our formulation seems to be quite simple and flexible. We solve for the temperature (in 2 or 3 dimensions)

$$\begin{aligned} T_t &= \nabla \cdot k(\bar{x}) \nabla T && (2.11)\\ k(\bar{x}) &= k_1 \;\text{ if } \bar{x} \;\varepsilon\; \Omega && (2.12)\\ k(\bar{x}) &= k_2 \;\text{ if } \bar{x} \;\varepsilon\; \Omega^c && (2.13)\\ T &= 0 \;\text{ for } \bar{x} \;\varepsilon\; \partial\Omega && (2.14) \end{aligned}$$

and the boundary of Ω moves with normal velocity

$$\vec{v} \cdot \vec{n} = \left[\frac{\partial T}{\partial n} \right] c_1 + c_2 \kappa \tag{2.15}$$

where $\kappa =$ curvature of the front.

We solve this using φ, the level set function, with reinitialization, by using

$$\varphi_t + \vec{u} \cdot \nabla\varphi = 0 \tag{2.16}$$

for u defined semi-numerically as

$$\begin{aligned} \vec{u} &= c_1 \left[\Delta x \Delta_+^x \Delta_-^x T,\; \Delta y \Delta_+^y \Delta_-^y T \right] && (2.17)\\ &+ c_2 \left[\nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|} \right) \right] \frac{\nabla\varphi}{|\nabla\varphi|} \end{aligned}$$

for Δ_+, Δ_- the usual undivided difference operators. The first term on the right is $O(\Delta x, \Delta y)$ except at the front.

We solve (2.11) by using the piecewise constant values k_1 or k_2 except when the discrete operators above cross the level set $\varphi = 0$. At such points we merely interpolate using the distance function to find the x and/or y value at which $T = 0$. We thus can get a one sided arbitrary high order approximation to $\Delta x T_{xx}$ and/or $\Delta y T_{yy}$ there. This is also used in (2.17). The results appear to be state of the art for this simple method. This is joint work with S. Chen, B. Merriman, and P. Smereka [3].

We demonstrate this algorithm on a supercooled liquid case in section 4.

Next, with E. Harabetian [9], [8] we consider an extension of the level set method where the normal velocity need not be intrinsic (solely geometry or position based) and for which the problem written in Lagrangian (moving) coordinates is Hadamard ill-posed. The main observation is that our approach provides an automatic regularization. There appear to be at least two reasons for this. The first is topological: a level set of a function cannot change its winding number – certain topological shapes based on the curve crossing itself are impossible. The second is analytical: the linearized problem is well-posed in the direction of propagation normal to the level set in this formulation; however it is ill-posed overall.

We shall describe the method in R^2. The three dimensional extension is relatively straightforward. Our two paradigms will be: (1) the initial value problem for the Cauchy-Riemann equations and (2) the motion of a vortex sheet in two dimensional, incompressible, inviscid fluid flow.

Our general problem is to move a curve Γ_0 : $(x_0(x), y_0(s))$, where s need not be arclength, through a system of partial differential equations

$$\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \vec{v}(x, y, x_s, y_s) \tag{2.18}$$

with initial conditions

$$\begin{pmatrix} x(s,0) \\ y(x,0) \end{pmatrix} = \begin{pmatrix} x_0(s) \\ y_0(s) \end{pmatrix}, \quad 0 \le s \le L \tag{2.19}$$

and periodic boundary conditions

$$\begin{aligned} x(0,t) &= x(L,t) \\ y(0,t) &= y(L,t). \end{aligned} \tag{2.20}$$

Here $\Gamma_0(s)$ (which might be multiply connected) divides R^2 into an inside Ω and ouside Ω^c. Also, v could depend on higher order derivatives (which it does in the curvature dependent case) or it could be nonlocal (as in the vortex sheet case).

In addition to the level set function φ, we define a conjugate function $\psi(x, y, t)$ with

$$\psi(x(0,s), y(0,s), 0) \equiv s \tag{2.21}$$

and

$$\nabla\varphi \cdot (\nabla\psi)^* = \varphi_x\psi_y - \varphi_y\psi_x \neq 0 \text{ at } t = 0 \text{ on } \Gamma_0. \tag{2.22}$$

We require an additional important condition on the conjugate function ψ

$$\psi(x(s,t), y(s,t), t) \equiv s \quad \text{for } t > 0. \tag{2.23}$$

Differentiating both equations (2.3) and (2.23) leads us to two equations on $\Gamma(s,t)$

$$\varphi_t + \vec{v} \cdot \nabla\varphi = 0 \tag{2.24}$$

$$\psi_t + \vec{v} \cdot \nabla\psi = 0. \tag{2.25}$$

It remains to define x_s, y_s in terms of $\nabla\varphi$ and $\nabla\psi$ within the arguments of v in (2.24), (2.25). We do this by differentiating (2.23) and (2.3) with respect to s, which leads us to

$$\begin{pmatrix} x_s \\ y_s \end{pmatrix} = [(\nabla\varphi) \cdot (\nabla\psi)^*]^{-1} \begin{pmatrix} -\varphi_y \\ \varphi_x \end{pmatrix}. \tag{2.26}$$

We replace (x_s, y_s) by this expression in the arguments of v in (2.24), (2.25), extend this to all space, and arrive at our formulation

$$\varphi_t + \vec{v}\left(x, y, \frac{-\varphi_y}{(\nabla\varphi \cdot (\nabla\psi)^*)}, \frac{\varphi_x}{(\nabla\varphi \cdot (\nabla\psi)^*)}\right) \cdot \nabla\varphi = 0 \tag{2.27}$$

$$\psi_t + \vec{v}\left(x, y, \frac{-\varphi_y}{(\nabla\varphi \cdot (\nabla\psi)^*)}, \frac{\varphi_x}{(\nabla\varphi \cdot (\nabla\psi)^*)}\right) \cdot \nabla\psi = 0. \tag{2.28}$$

At every time step φ is reinitialized to be signed distance. We also reinitialize ψ as follows. As described in [2], we can construct ψ initially so that $\nabla\varphi \cdot \nabla\psi = 0$ on and near Γ, i.e. we generate an orthonormal coordinate system.

We reinitialize ψ to have this property by solving to steady state near Γ

$$\tilde{\psi}_t + H(\varphi)\frac{\nabla\varphi \cdot \nabla\tilde{\psi}}{|\nabla\varphi|} = 0 \tag{2.29}$$

where H is defined as in (2.8).

An interesting example is the Cauchy-Riemann system

$$\begin{aligned} x_t &= y_s \\ y_t &= -x_s. \end{aligned} \tag{2.30}$$

The level set formulation is to find the set $\varphi \equiv 0$ where

$$\begin{aligned} \varphi_t &+ \frac{|\nabla\varphi|^2}{(\nabla\varphi)\cdot(\nabla\psi)^*} = 0 \qquad (2.31) \\ \psi_t &+ \frac{(\nabla\varphi)\cdot(\nabla\psi)}{(\nabla\varphi)\cdot(\nabla\psi)^*} = 0 \qquad (2.32) \end{aligned}$$

with the reinitialization described above. This formulation appears to stabilize the problem. Justification is given in [9].

In special cases when the velocity v is purely normal to Γ we have an alternative formulation of (2.27), (2.28). The system (2.16) can be rewritten

$$\begin{aligned} x_t &= gy_s \qquad (2.33) \\ y_t &= -gx_s \qquad (2.34) \end{aligned}$$

for $g = g(x, y, x_s, y_s)$. If we set $f = \sqrt{x_s^2 + y_s^2}$ (the arclength), then Γ is moving normal to itself with velocity fg. Differentiating f with respect to t gives us a system of two equations φ and f (rather than φ and ψ) in which curvature of level sets appears in a transparent way:

$$\begin{aligned} \varphi_t &+ gf\cdot|\nabla\varphi| = 0 \qquad (2.35) \\ f_t &+ \frac{gf}{|\nabla\varphi|}\nabla\varphi\cdot\nabla f = g\kappa f^2 \qquad (2.36) \end{aligned}$$

(for the Cauchy Riemann equations, $g \equiv 1$).

The second equation is almost a Riccati equation for the arclength f. Illposedness is reflected in the blow up of f or of $f \to 0$, depending on the sign of the curvature κ.

An ill-posed problem of great physical interest is the motion of a vortex sheet in the incompressible Euler equations. We have a velocity vector field $\vec{v}$ which is incompressible

$$\nabla\cdot\vec{v} = 0 \qquad (2.37)$$

and which satisfies

$$\nabla\times\vec{v} = \omega \qquad (2.38)$$

where the vorticity $\omega(x, y, 0)$ is a singular distribution which can be written, using the level set function φ

$$\omega(x,y,0) = R(s)\delta(\varphi)|\nabla\varphi| = R(x(s,0),y(s,0))\delta(\varphi)|\nabla\varphi| \qquad (2.39)$$

where $R(s)$ is the strength along the initial vortex sheet $(x(s,0), y(s,0))$.

The vorticity moves according to the advection equation:

$$\omega_t + \vec{v}\cdot\nabla\omega = 0. \qquad (2.40)$$

Rather than evolving the vortex sheet by the well-known Birkhoff-Rott equation (see e.g. [14]), we shall use a new desingularization technique [8] which simplifies our procedure still further.

If the vorticity can be written

$$\omega = P(\varphi) \qquad (2.41)$$

for some smooth level set function, then since φ advects with $\vec{v}$ we can easily solve this problem.

Simply set

$$\vec{v} = (-\gamma_y, \gamma_x)^T \qquad (2.42)$$

with

$$\Delta\gamma = \omega = P(\varphi). \qquad (2.43)$$

We may use off-the-shelf Poisson solvers to do this, together with ENO schemes to solve

$$\varphi_t + \vec{v}\cdot\nabla\varphi = 0. \qquad (2.44)$$

ENO is needed because $\vec{v}$ has a tangential discontinuity on the set $\{(x,y)|\varphi(x,y) = 0\}$.

For vortex sheets we simply choose a level set function so that, at $t = 0$, $\{(x,y)|\varphi(x,y) = 0\}$,

$$\frac{\partial\varphi}{\partial n} = |\nabla\varphi| = \frac{1}{R}. \qquad (2.45)$$

The resulting smooth flow generates the vortex sheet [9], where $P(\varphi) = \delta(\varphi)$ for later time. For results see [9] and section 4. This method also works for vortex patches, point vortices and sheets in 3D, [9].

For vortex sheets in 2D, we were able to compute the roll-up of a vortex sheet past the time of singularity as computed by Krasny in [14]. We do not do any explicit filtering in the Fourier frequencies, nor do we use blobs to smooth out the flow as in [14].

Finally we mention that complicated motion of multiple junctions can be rather simply implemented by using as many level set functions as there are regions – see [17]. Also, in the special case of mean curvature motion, the simple heat equation together with a projection may be used [17].

3 Shock Capturing Methods

There is a vast literature on this subjection, also see [10] for a recent review article at the 1990 International Congress of Mathematicians. The fundamental problem is that the solution to the initial value problem for a system of hyperbolic conservation laws generally develops discontinuities (shocks) in finite time, no matter how smooth the initial data is. Weak solutions must be computed. The goal is to develop numerical methods which "capture" shocks automatically. Reasonable design principles are:

(1) Conservation form (defines shock capturing – see [7], [16]).

(2) No spurious overshoots, wiggles near discontinuities, yet sharp discrete shock profiles.

(3) High accuracy in smooth regions of the flow.

(4) Correct physical solution, i.e. satisfaction of the entropy conditions in the convergent limit [15].

Conventional methods had trouble with combining (1) and (3). It should be noted that wiggles can pollute the solution causing e.g. negative densities and pressures and other instabilities.

We have developed with Harten, Engquist and Chakravarthy [12], [11] and later simplified with Shu [26], [27] a class of shock capturing algorithms designed to satisfy (1-4).

These methods are called essentially nonoscillatory (ENO) schemes. They resemble their predecessors – total variation diminishing (TVD) schemes in that the stencil is adaptive, however the total variation of the solution of the approximation to a one space dimensional scalar model might increase, but only at a rate $O((\text{grid size})^p)$, for p the order of the method, up to discontinuities, and the order of accuracy can be made arbitrary in regions of smoothness. TVD schemes traditionally degenerate to first order at isolated extrema (see [22] for extensions up to second order).

The basic idea is to extend Godunov's [7] ingenious idea past first order accuracy. This was first done up to second order accuracy by van Leer [31]. A key step, and the only one we have space to describe here, is the construction of a piecewise polynomial of degree m, which interpolates discrete data w given at grid x_j. In each cell $d_j = \{(x)x_j \le x \le x_{j+1}\}$ we construct a polynomial of degree m which interpolates $w(x)$ at $m+1$ successive points $\{x_i\}$ including x_j and x_{j+1}.

The idea is to avoid creating oscillations by choosing the points using the "smoothest" values of w. (This is a highly nonlinear choice, as it must be). One way of doing this is to use the Newton interpolating polynomials and the associated coefficients. We start with a linear interpolant in each cell

$$q_{1,j+\frac{1}{2}} = w[x_j] + (x - x_j)w[x_j, x_{j+1}] \tag{3.1}$$

using the Newton coefficients

$$w[x_i] = w(x_i) \tag{3.2}$$

$$\begin{aligned} &w[x_i, \ldots, x_{i+k}] = \\ &= (x_{i+k} - x_i)^{-1}(w[x_{i+1}, \ldots, x_k] - w[x_i, \ldots, x_{k-1}]). \end{aligned} \tag{3.3}$$

We get two candidates for $q_{2,j+\frac{1}{2}}$; which interpolate w at x_j, x_{j+1} and either x_{j-1}, or x_{j+2}

$$\begin{aligned} &q_{2,j+\frac{1}{2}} = q_{1,j+\frac{1}{2}} + \\ &+(x - x_j)(x - x_{j+1})[w[x_{j-1}, x_j, x_{j+1}] \\ &\text{or } w[x_j, x_{j+1}, x_{j+2}]]. \end{aligned} \tag{3.4}$$

Since we are trying to minimize oscillations by taking information from regions of smoothness, we pick the coefficient which is *smaller* in magnitude. We store this choice and proceed inductively up to degree m. The result is a method which is exact for piecewise polynomials of degree $\le m$ and which is nonoscillatory (i.e. essentially monotone) across jumps. See [11] for further discussions.

Other choices are possible, in fact it seems advantageous to minimize truncation error by biasing the choice of stencil towards the center – see [25], [21]By now, ENO schemes for compressible flows have proven their worth – see e.g. [28]. We give an example involving compressible isotropic turbulence in section 4.

4 Results

Our first set of figures 1-7 concern the motion of air bubbles in water or water drops in air. The calculations were performed by Mark Sussman, see ref [29], [30]. The algorithm is based on the level set method and is quite simple to implement. It should be stressed that no special work is needed at pinchoff, merging, development of kinks, etc. and the algorithm is very easy to implement on a fixed, uniform grid. All these figures should be read left to right, top to bottom.

Figure 1 shows the evolution of a 2D rising bubble with high Reynolds number and low surface tension.

Figure 2 shows the evolution of a large water drop with no surface tension which deforms as it hits the base.

Figure 3 shows the evolution with surface tension. The drop remains circular as it hits the base.

Figure 4 shows the evolution of two water drops colliding with each other. The combined drop experiences surface tension driven oscillations.

Figure 5 shows (essentially) conservation of mass for the merging drop problem. A new idea of E. Fatemi and Sussman (private communication) is used to preserve mass in the distance reinitialization step.

Figure 6 demonstrates the rupture of a gas bubble at an air water interface. The surface tension effects cause a high speed jet of water to form.

Finally, in figure 7 we see an air bubble rising through a water/oil interface. Surface tension effects are ignored. Ideas developed in [17] for the motion of multiple junctions, were used here.

Figures 8a,b,c,d,e from [13] show volume preserving motion by mean curvature velocity. The topological change presents no difficulty. See [13] for merging and physically realistic acceleration cases.

Figure 9 from [3] shows a Stefan problem with supercooled liquid. The initial solutions are 2 square seeds (temperature $T = 0$) in a surrounding bath of undercooled liquid ($T = -.8$). No surface temperature or kinetic undercooling terms are incorporated (i.e. $T = 0$ on the boundary). Time levels shown are increasing in intervals of $t = 1.2$ up to a time of $t = 24.6$.

Figures 10(a-f) from [9] show the results of our simple vortex sheet evolution algorithm. This is done using a third order ENO scheme on the advection step on a 128^2 grid.

Figures 11(a-e) show the roll-up using 1024^2 points. (Figures 10 and 11 were computed by Professor Chi-Wang Shu).

Figures 12 and 13 from [28] contrast spectral methods versus third order ENO for compressible isotropic turbulence. Figure 12 shows density and vorticity contours for spectral, Figure 13 for ENO. The oscillations in Figure 11 disappear with ENO. For details see [28].

References

[1] J. Brackbill, D. Kothe, and C. Zemach (1992) A Continuum Method for Modeling Surface Tension, J. Comput. Phys., vol.100 pp.335-353

[2] Y.C. Chang, T.Y. Hou, B. Merriman, and S. Osher (1994) A Level Set Formulation of Eulerian Interface Capturing Methods for Incompressible Fluid Flows, to appear, J. Comput. Phys.

[3] S. Chen, B. Merriman, P. Smereka, and S. Osher (1994) A Fast Level Set Based Algorithm for Stefan Problems, preprint

[4] Y.G. Chen, Y. Giga, and S. Goto (1986) Uniqueness and Existence of Viscosity Solutions of Generalized Mean Curvature Flow Equations, J. Diff. Geom., vol.23 pp.749-785

[5] L.C. Evans, M. Soner, and P. Souganidis (1992) Phase Transitions and Generalized Motion by Mean Curvature, Comm. Pure Appl. Math., vol.45 pp.1097-1123

[6] L.C. Evans and J. Spruck (1986) Motion of Level Sets by Mean Curvature, I, J. Diff. Geom., vol.23 pp.69-96

[7] S. Godunov (1959) A Difference Scheme for Computation of Discontinuous Solutions of Equations of Fluid Dynamics, Math. Sbornik, vol.47 pp.271-306

[8] E. Harabetian and S. Osher (1995) A Level Set Based Approach to Vortex Dynamics, preprint.

[9] E. Harabetian and S. Osher (1994) Stabilizing Ill-Posed Problems Via the Level Set Approach, preprint

[10] A. Harten (1990) Recent Developments in Shock Capturing Schemes, Proceedings of the ICM, Kyoto 1990, pp.1549-1573

[11] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy (1987) Uniformly High Order Accurate Essentially Nonoscillatory Schemes, II, J. Comp. Phys., vol.71 pp.231-303

[12] A. Harten and S. Osher (1987) Uniformly High-Order Accurate Nonoscillatory Schemes, I, SINUM, vol.24 pp.279-304

[13] M. Kang, P. Smereka, B. Merriman, and S. Osher (1994) On Moving Interfaces by Volume Preserving Velocities or Accelerations, preprint

[14] R. Krasny (1990) Computing Vortex Sheet Motion, Proceedings of the ICM, Kyoto 1990, pp.1573-1583

[15] P.D. Lax (1954) Weak Solutions of Nonlinear Hyperbolic Equations and their Numerical Computation, Comm. Pure Appl. Math., vol.7 pp.159-193

[16] P.D. Lax and B. Wendroff (1960) Systems of Conservation Laws, Comm. Pure Appl. Math., vol.13 pp.217-237

[17] B. Merriman, J. Bence, and S. Osher (1994) Motion of Multiple Junctions: A Level Set Approach, J. Comput. Phys., vol.112 pp.334-363

[18] W. Noh and P. Woodward (1970) A Simple Line Interface Calculation, Proceeding, Fifth Int'l. Conf. on Fluid Dynamics, A.I. van de Vooran and D. J. Zandberger, eds., Springer-Verlag

[19] S. Osher and J.A. Sethian (1988) Fronts Propagating with Curvature Dependent Speed, Algorithms Based on a Hamilton-Jacobi Formulation, J. Comp. Phys., vol.79 pp.12-49

[20] S. Osher and C.-W. Shu (1991) High-Order Essentially Nonoscillatory Schemes for Hamilton-Jacobi Equations, SINUM, vol.28 pp.907-922

[21] A. Rogerson, E. Meiburg (1990) A Numerical Study of Convergence Properties of ENO Schemes, J. Scientific Computing vol.5 pp.151-167

[22] R. Sanders (1988) A Third Order Accurate Variation Nonexpansive Difference Scheme for a Single Nonlinear Conservation, Math. Comp., vol.51, pp.535-558

[23] J.A. Sethian (1985) Curvature and the Evolution of Fronts, Comm. Math. Phys. vol.101 pp.487-499

[24] J. Sethian and J. Strain (1992) Crystal Growth and Dendrite Solidification, J. Comp. Phys., vol.98 pp.231-253

[25] C.-W. Shu (1990) Numerical Experiments on the Accuracy of ENO and Modified ENO Schemes, J. Scientific Computing, vol.5 pp.127-150

[26] C.-W. Shu and S. Osher (1988) Efficient Implementation of Essentially Nonoscillatory Schemes I, J. Comput. Phys., vol.77 pp.439-471

[27] C.-W. Shu and S. Osher (1989) Efficient Implementation of Essentially Nonoscillatory Schemes II, J. Comput. Phys., vol.83 pp.32-78

[28] C.-W. Shu, T.A. Zang, G. Erlebacher, D. Whitaker, and S. Osher (1992) High Order ENO Schemes Applied to Two and Three Dimensional Compressible Flow, Appl. Num. Math., vol.9 pp.45-71

[29] M. Sussman (1994) Ph.D. Thesis, UCLA CAM report #94-13, UCLA, Mathematics

[30] M. Sussman, P. Smereka, and S. Osher (1994) A Level Set Approach for Computing Solutions to Incompressible Two Phase Flow, to appear, J. Comput. Phys.

[31] B. Van Leer (1979) Towards the Ultimate Conservative Difference Scheme V. A Second Order Sequel to Godunov's Method, J. Comp. Phys., vol.32 pp.101-136

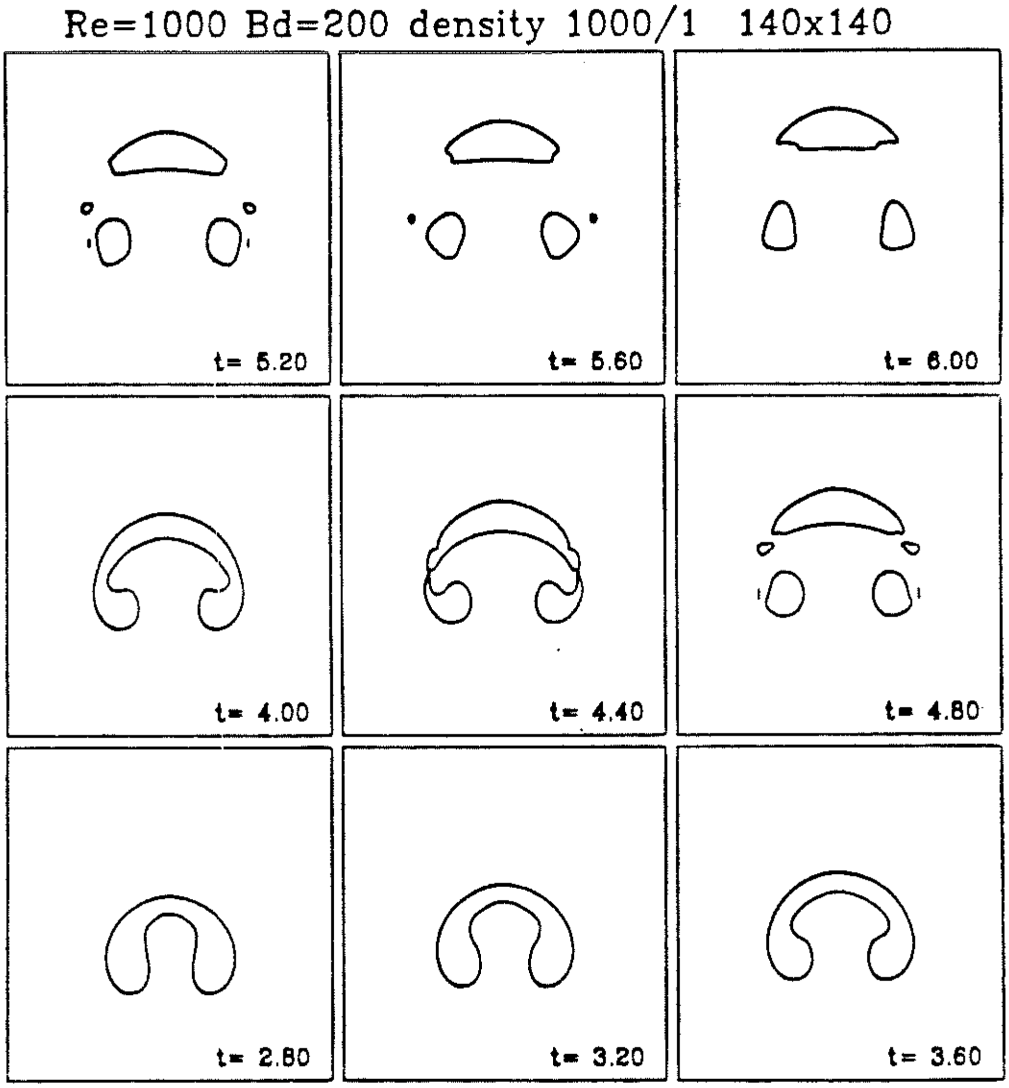

Figure 1: Evolution of rising bubble with high Reynolds number and low surface tension.

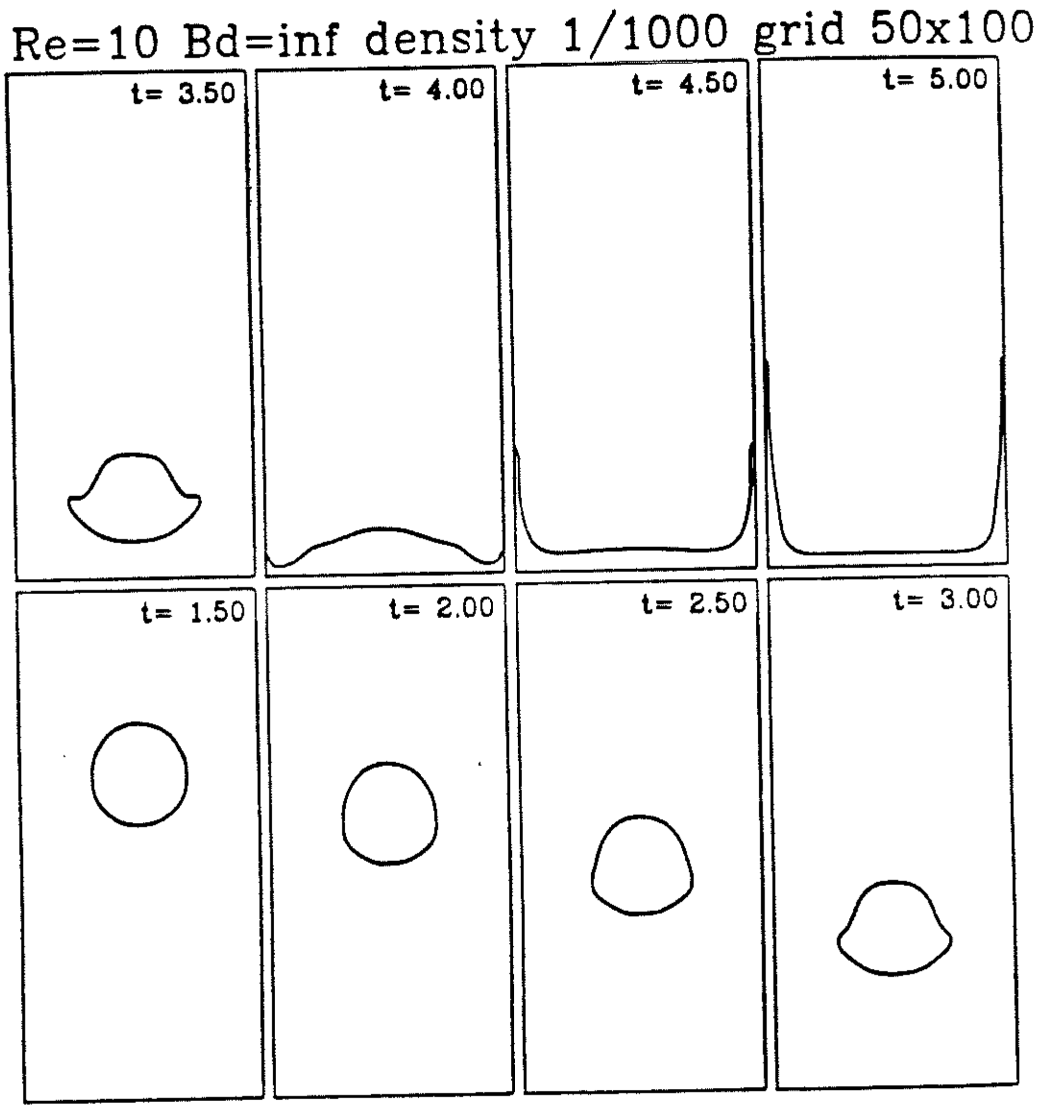

Figure 2: Evolution of a large water drop (no surface tension). Drop deforms as it hits the base.

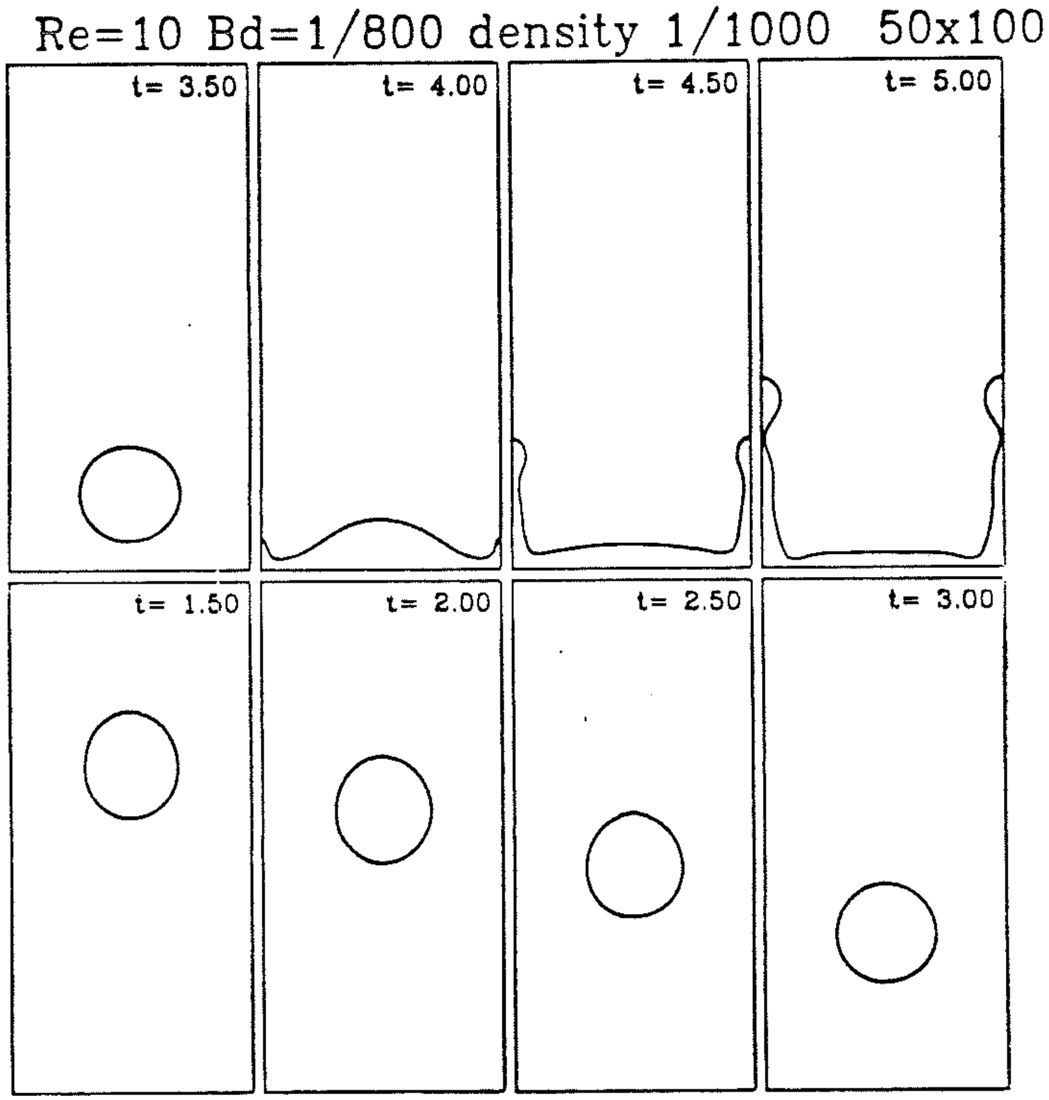

Figure 3: Evolution of a water drop with surface tension. Drop remains circular as it hits the base.

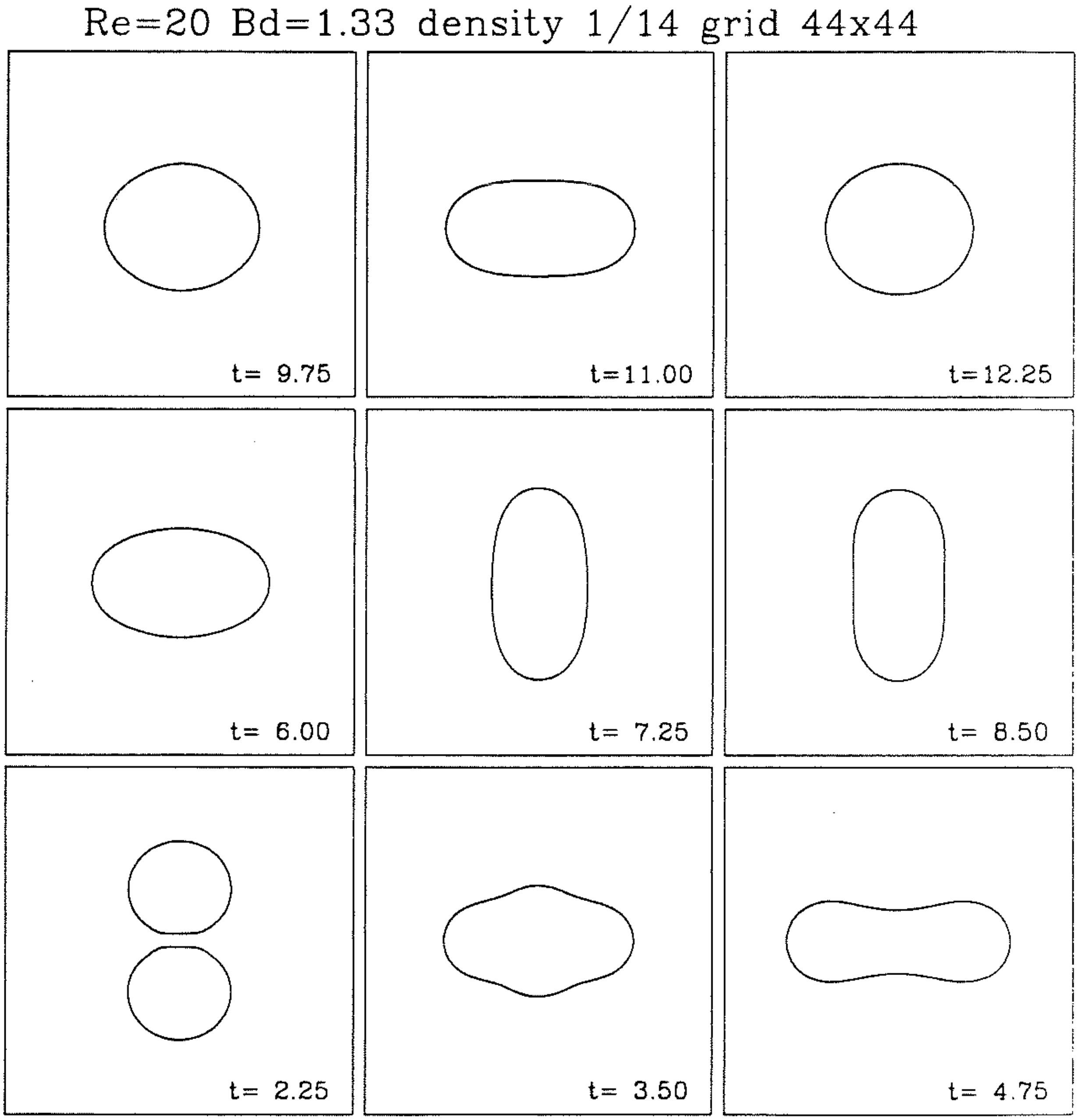

Figure 4: Evolution of two water drops colliding with each other. The combined drops experiences surface tension driven oscillation.

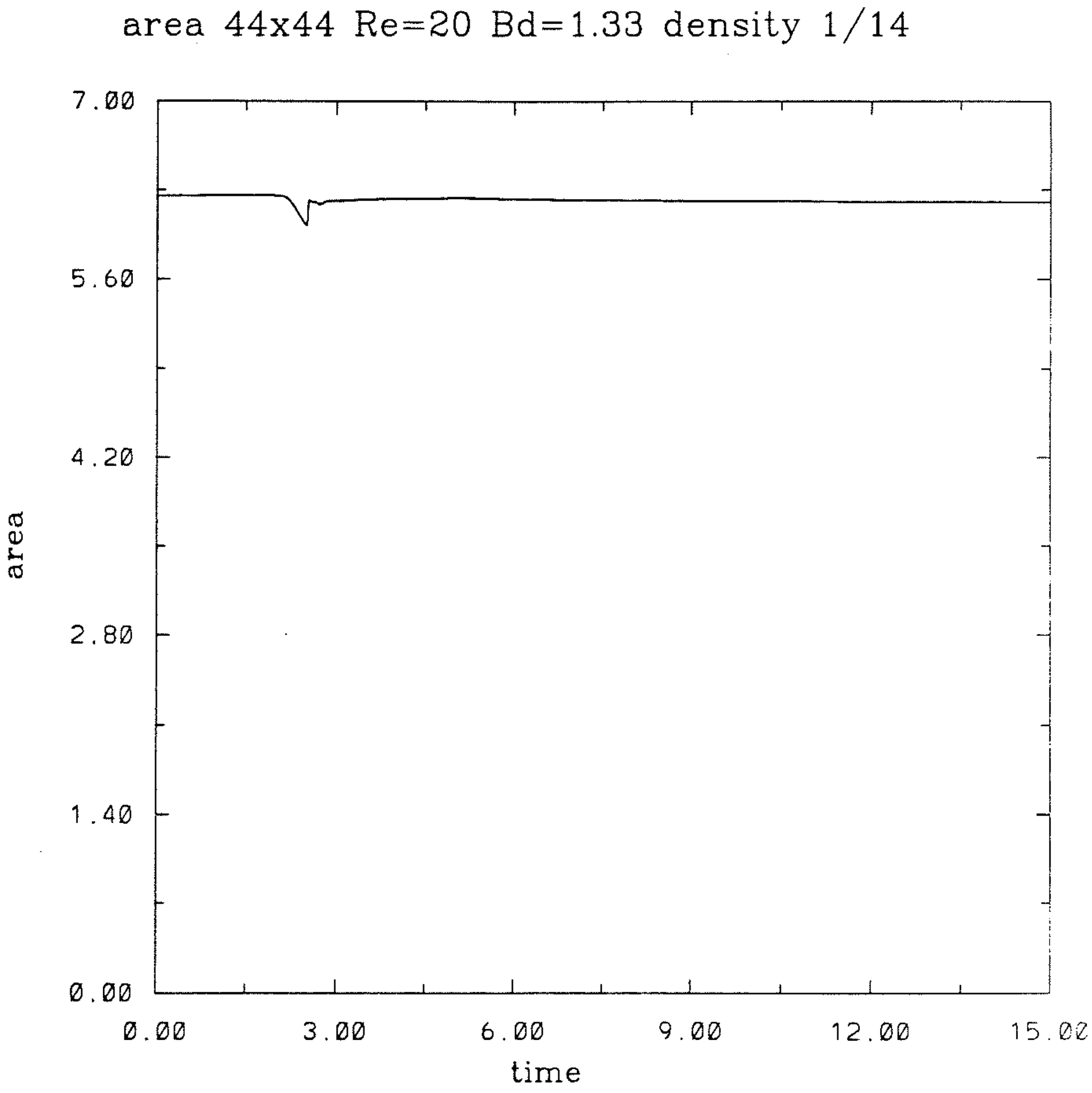

Figure 5: Conservation of mass for the merging drop problem.

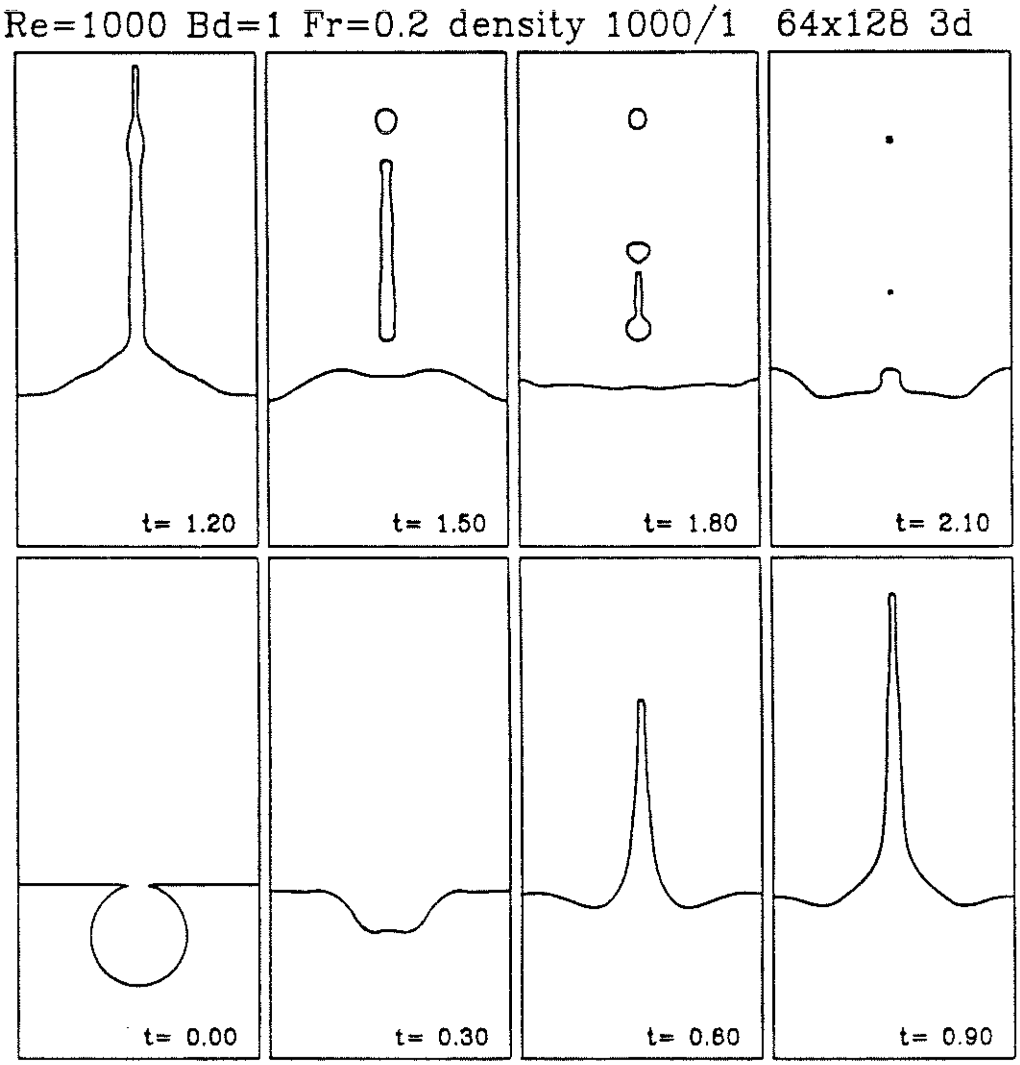

Figure 6: Upon rupture of a gas bubble at an air/water interface, surface tension effects cause a high speed jet of water to form.

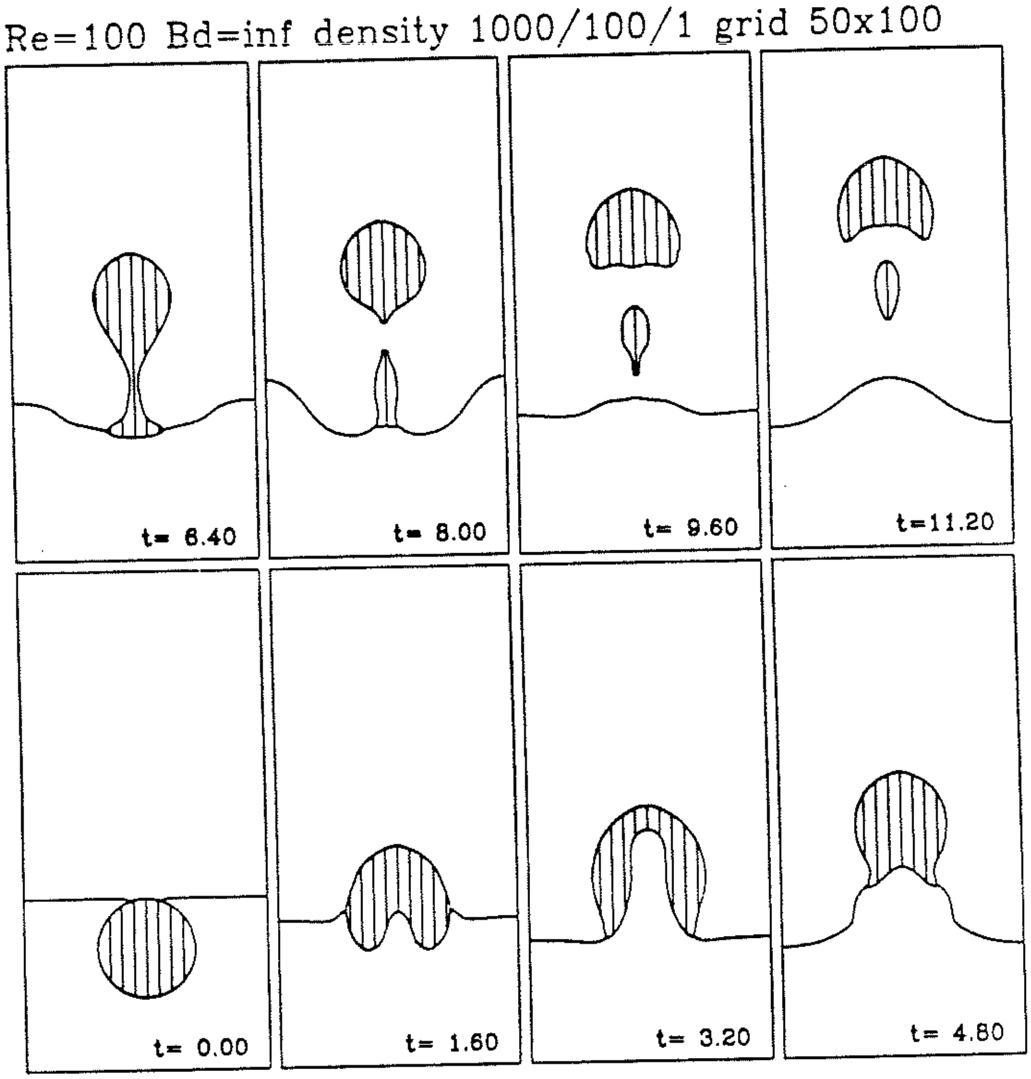

Figure 7: Air bubble rises through a water/“oil” interface. “Oil” is a less dense but more viscous fluid. Surface tension effects are ignored.

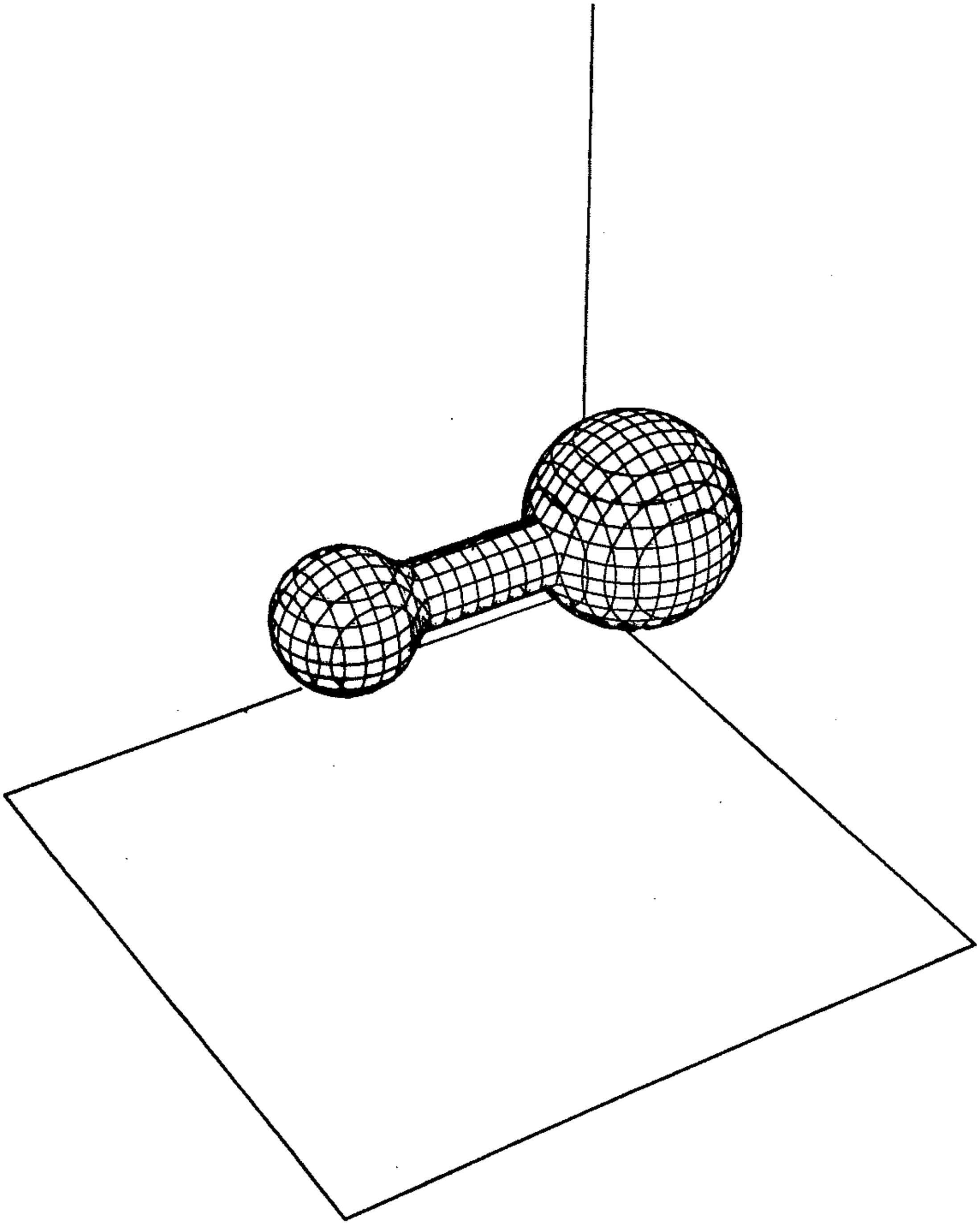

Figure 8a: 3D volume preserving mean curvature velocity.

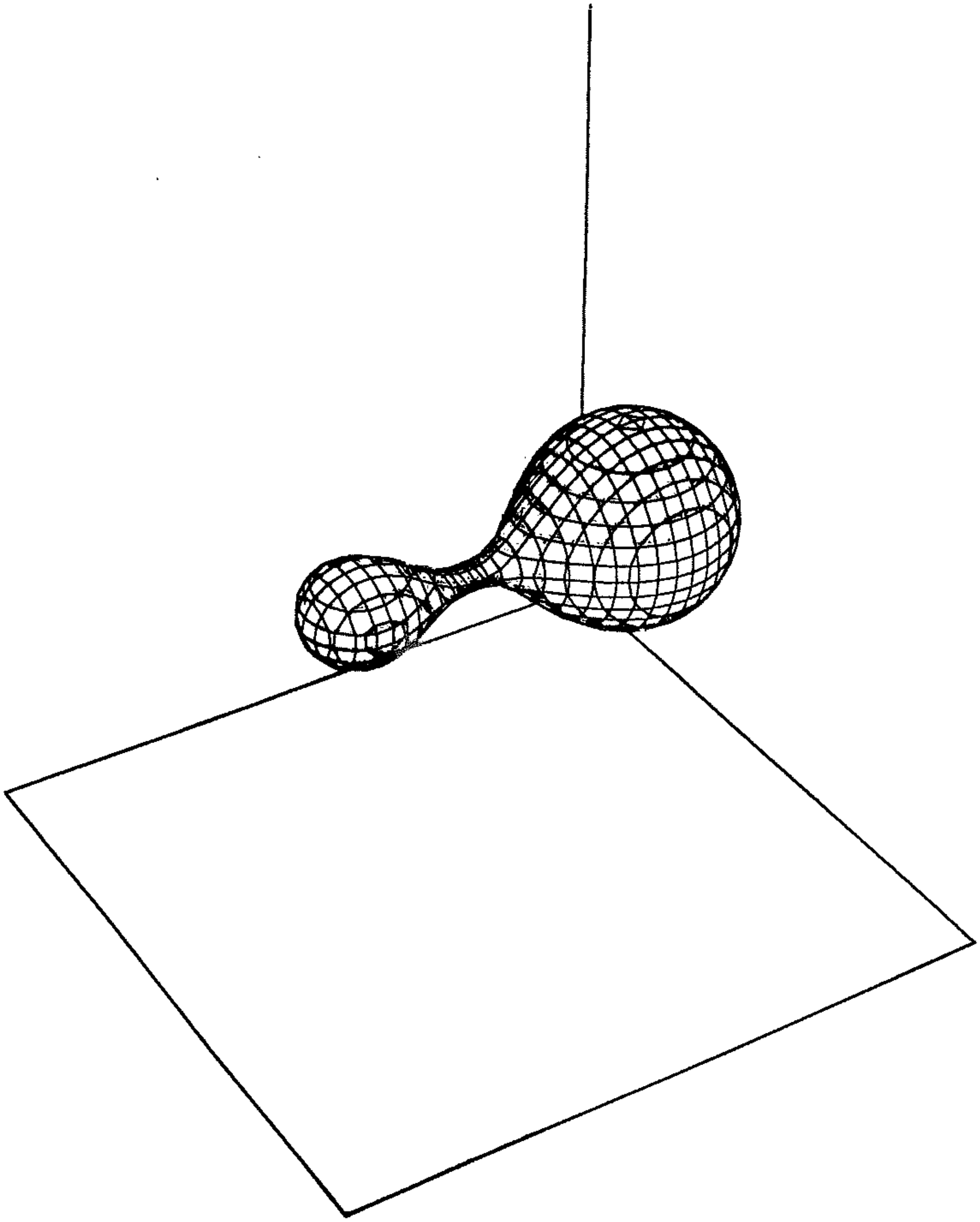

Figure 8b: 3D volume preserving mean curvature velocity.

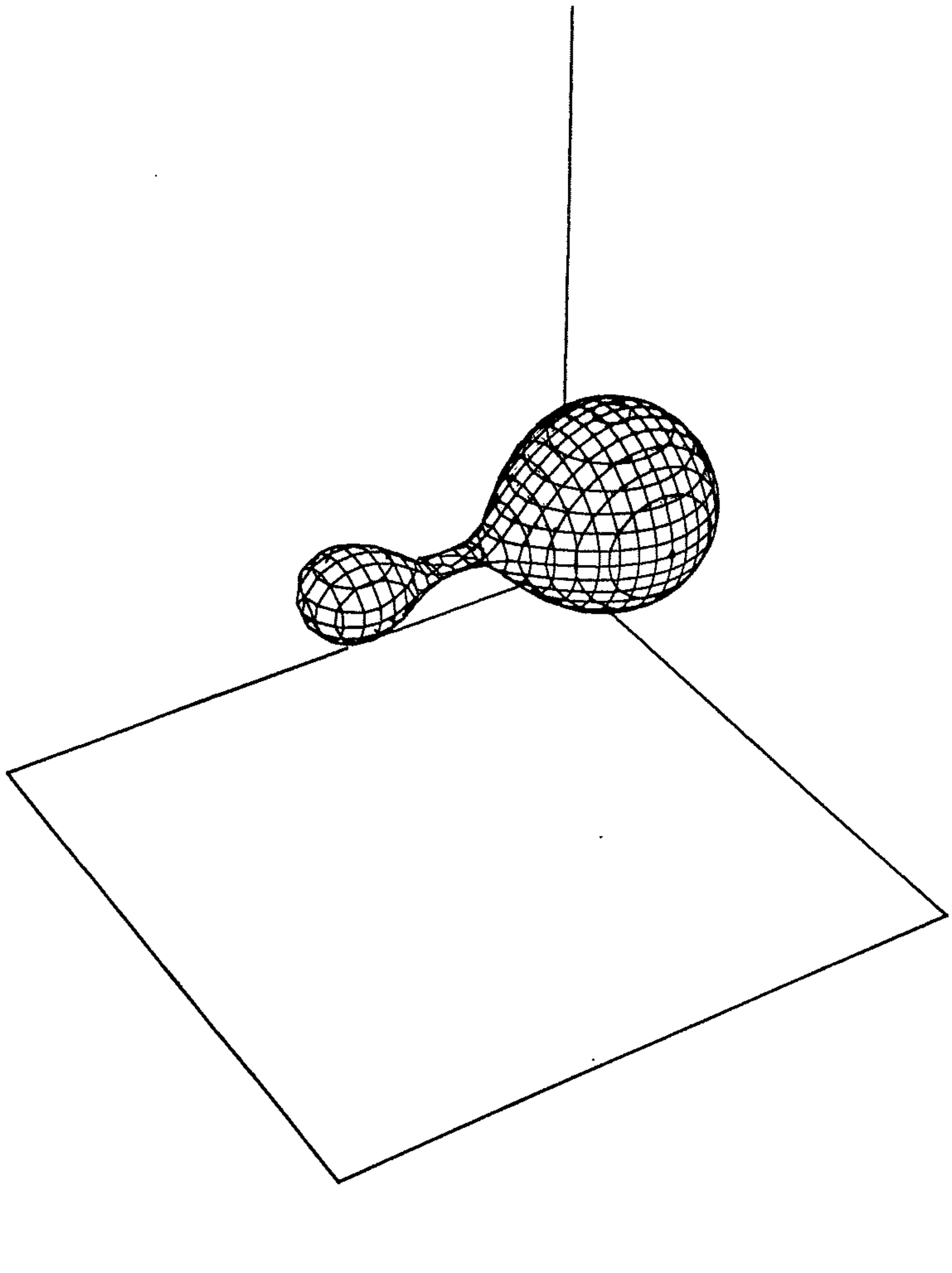

Figure 8c: 3D volume preserving mean curvature velocity.

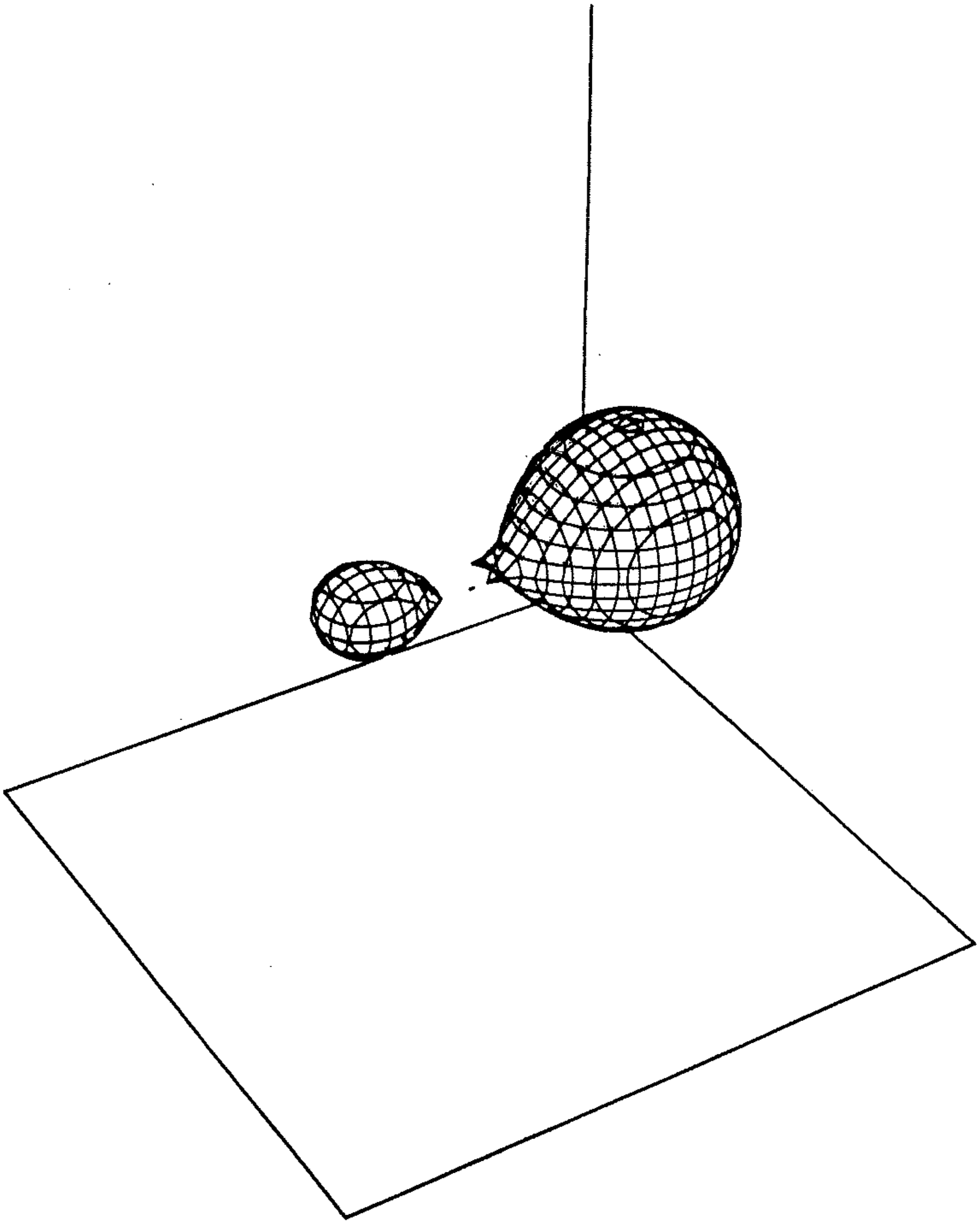

Figure 8d: 3D volume preserving mean curvature velocity.

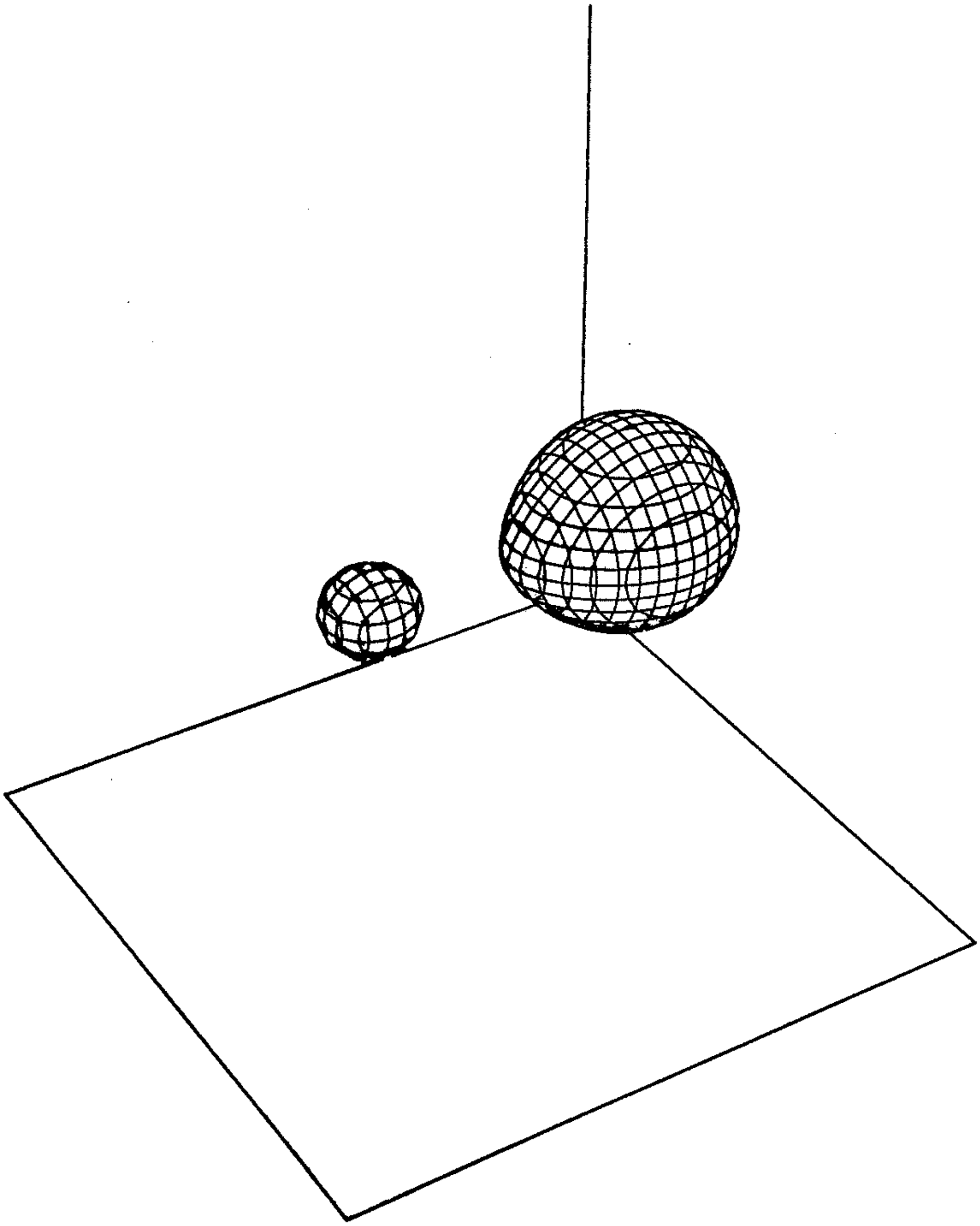

Figure 8e: 3D volume preserving mean curvature velocity.

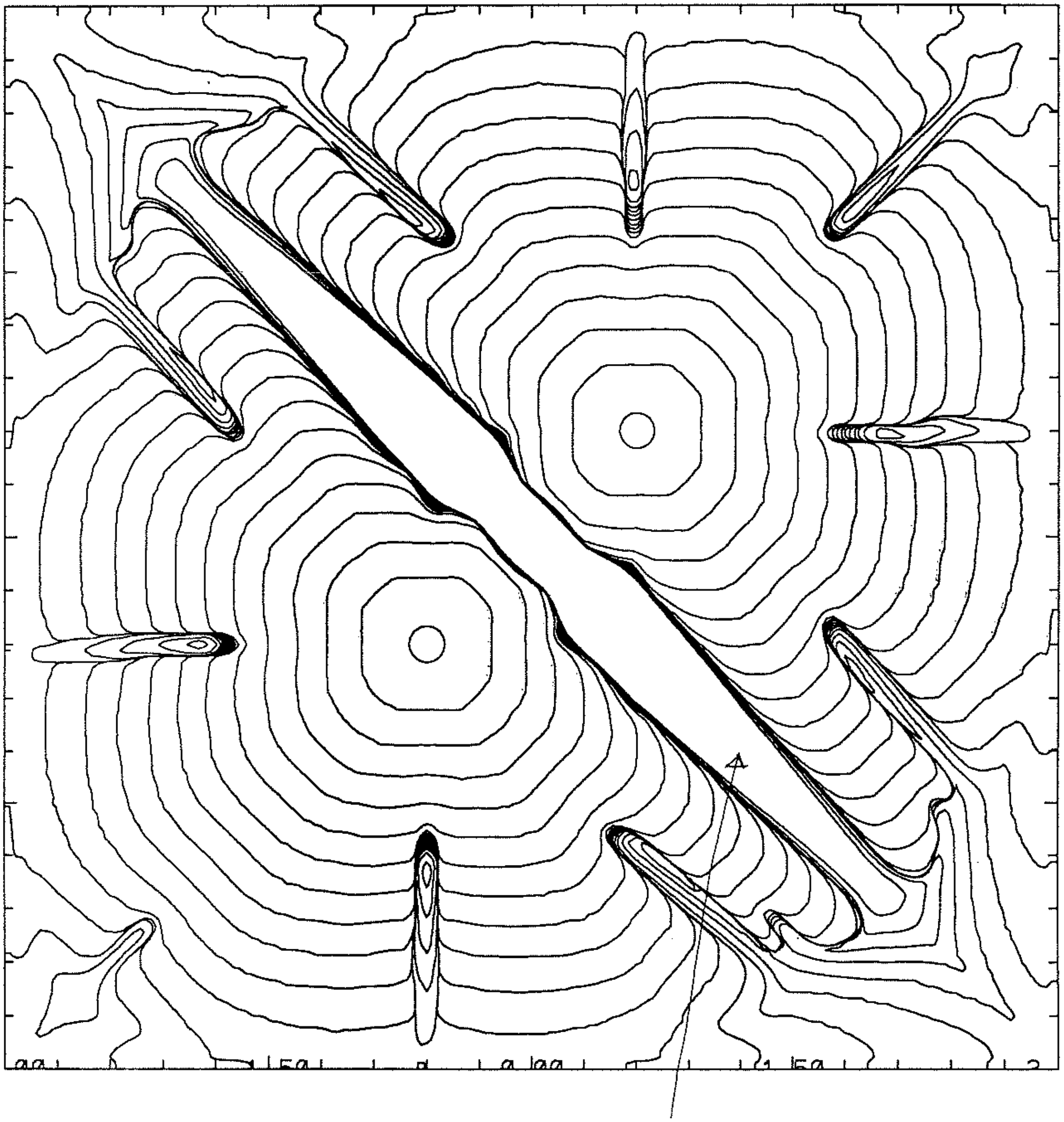

Figure 9: Region of trapped supercooled liquid.

$t = 1 \qquad ENO3 \qquad 128^2$

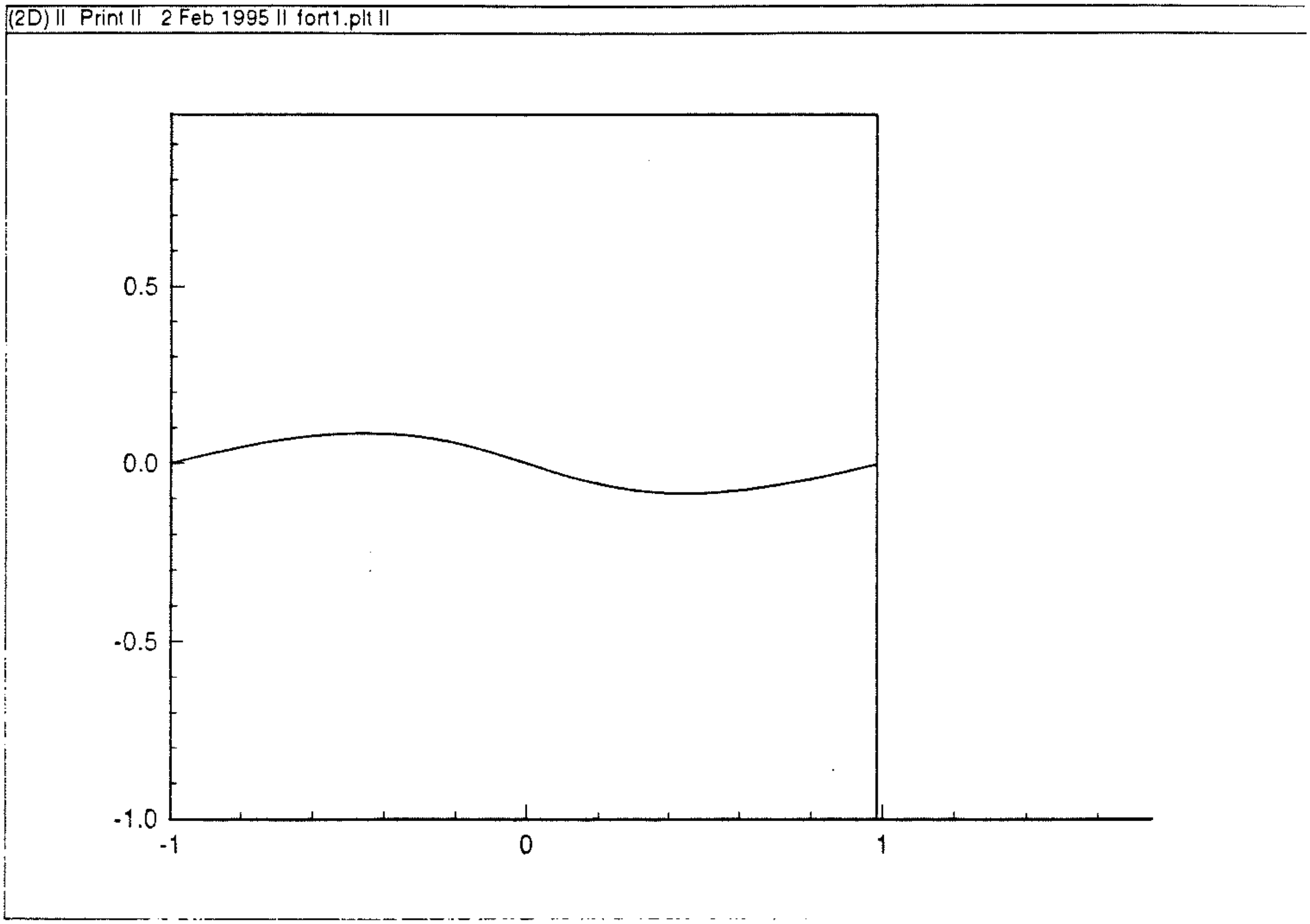

Figure 10a: Roll up of a vortex sheet.

$t = 2 \qquad ENO3 \qquad 128^2$

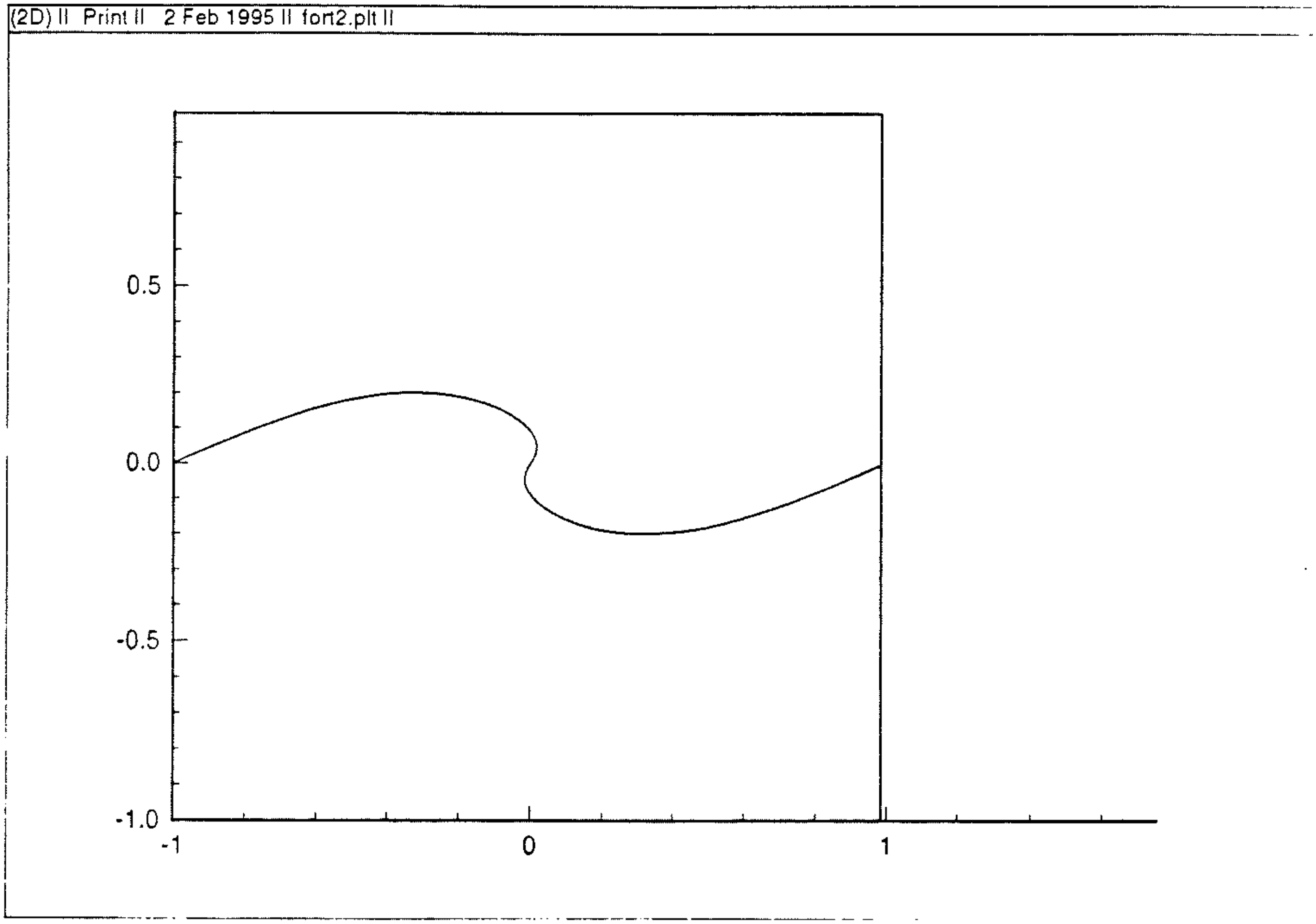

Figure 10b: Roll up of a vortex sheet.

$t = 3 \qquad ENO3 \qquad 128^2$

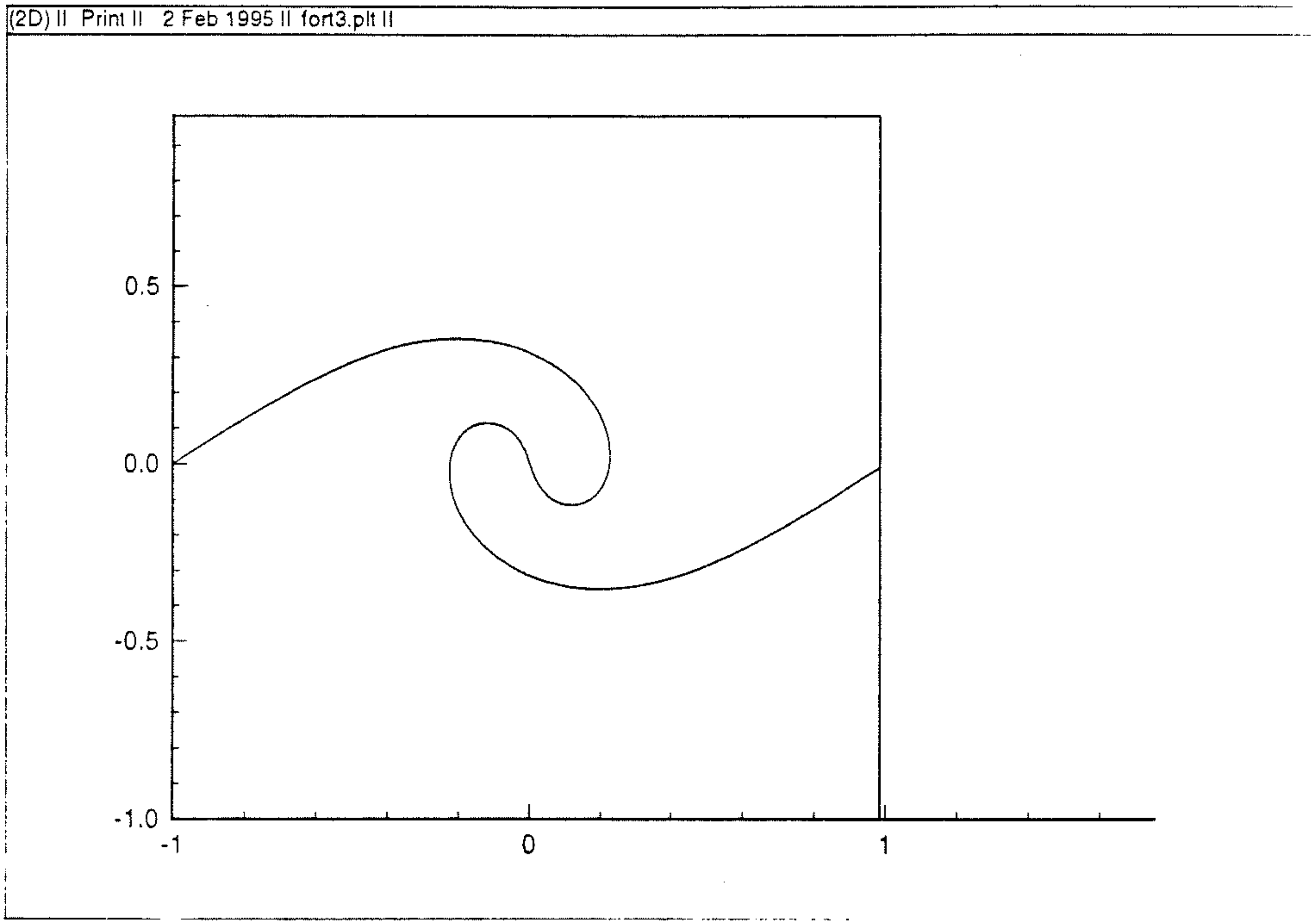

Figure 10c: Roll up of a vortex sheet.

$t = 4 \qquad ENO3 \qquad 128^2$

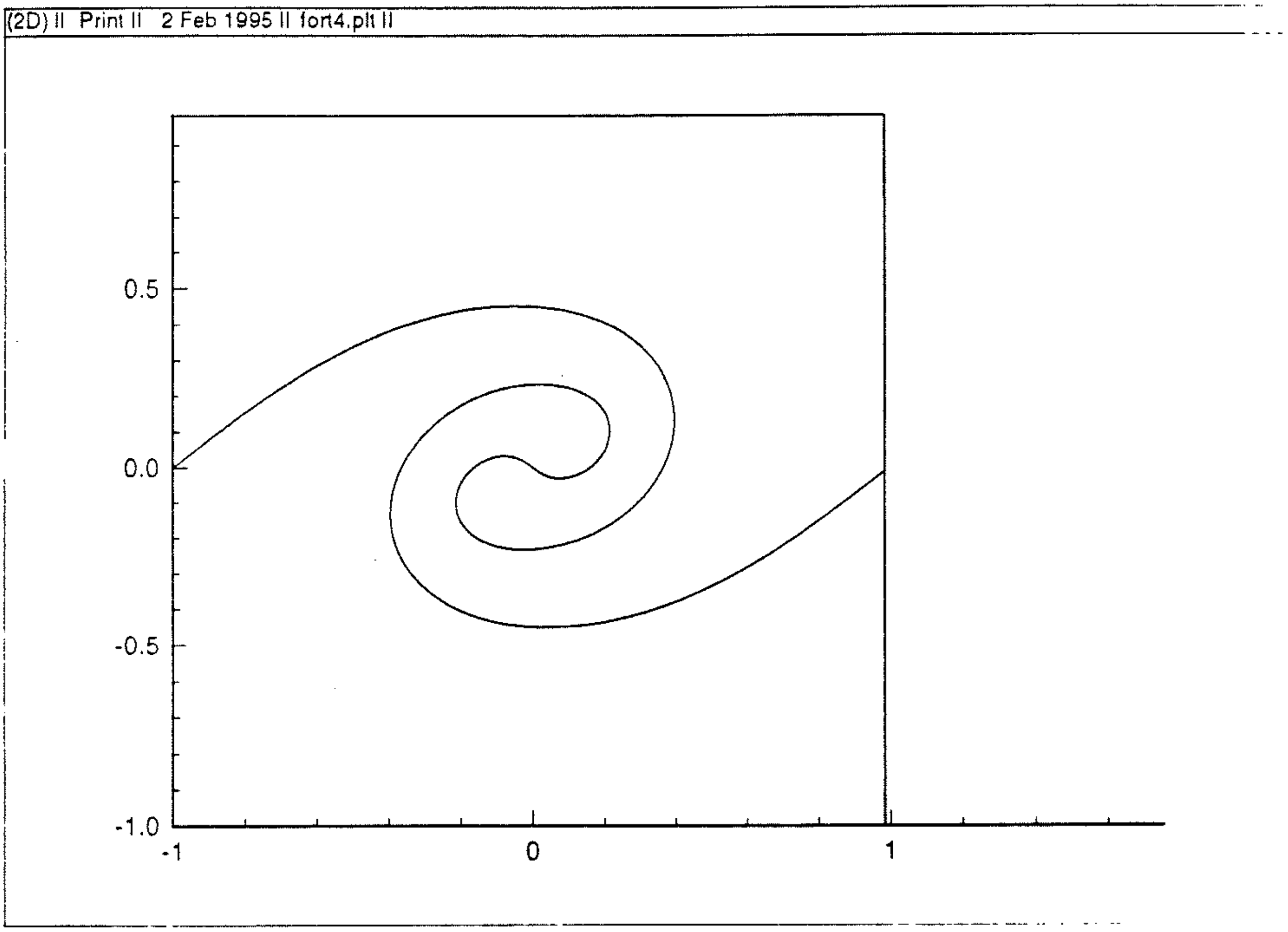

Figure 10d: Roll up of a vortex sheet.

$t = 5 \qquad ENO3 \qquad 128^2$

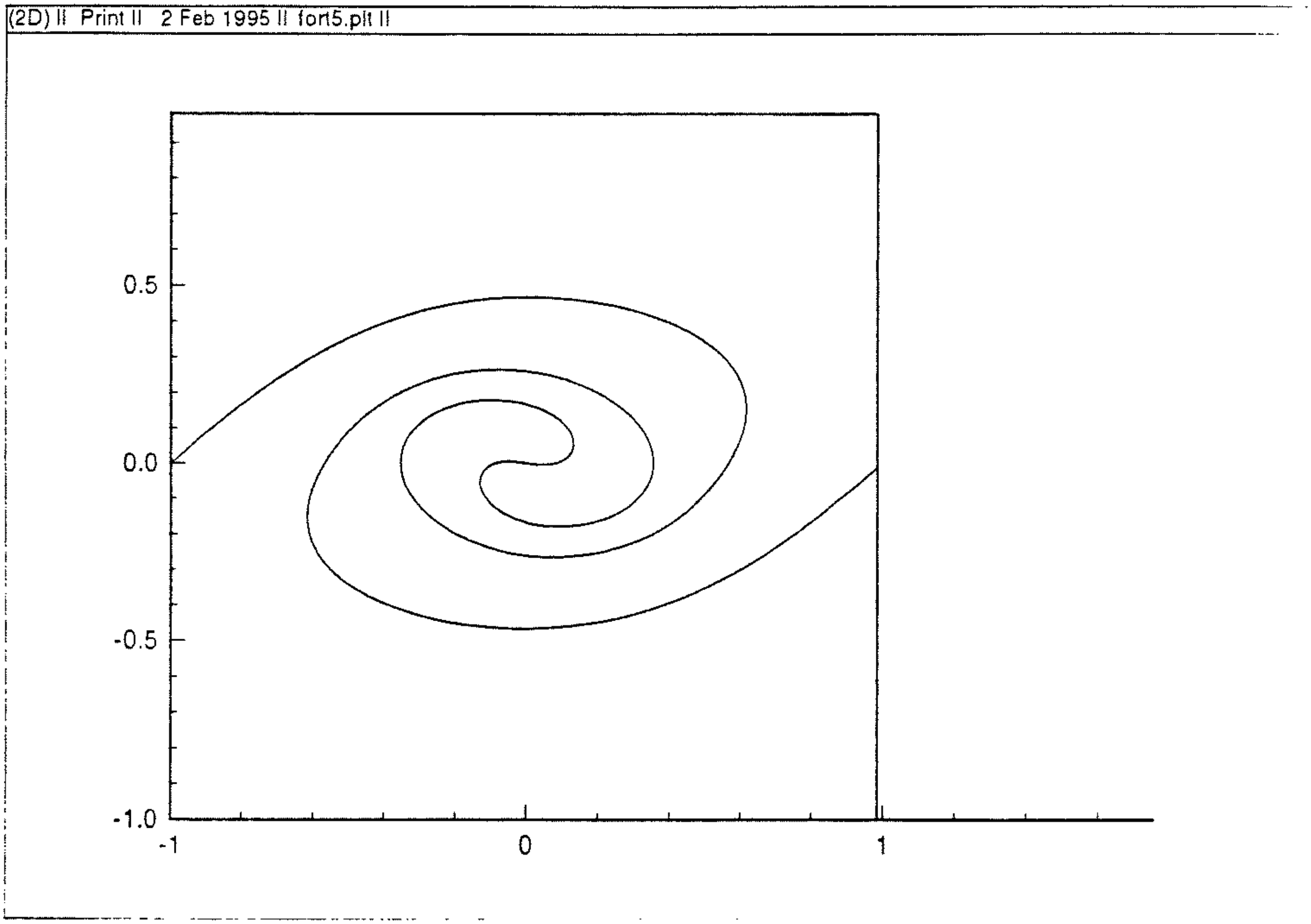

Figure 10e: Roll up of a vortex sheet.

$t = 6 \qquad ENO3 \qquad 128^2$

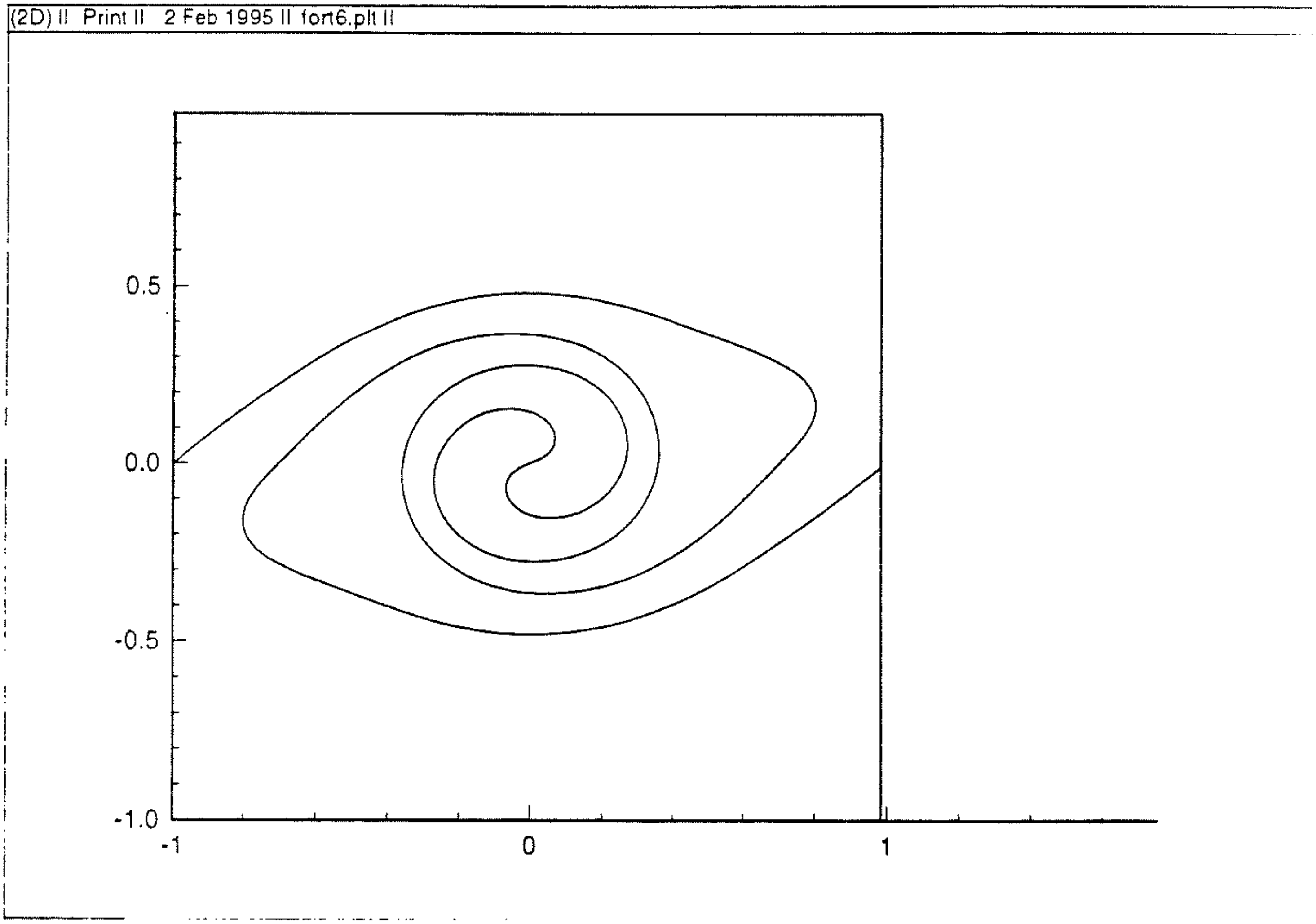

Figure 10f: Roll up of a vortex sheet.

$$t = 2, \qquad ENO3, \qquad 1024^2 \qquad \delta - \text{width} = 48\Delta x$$

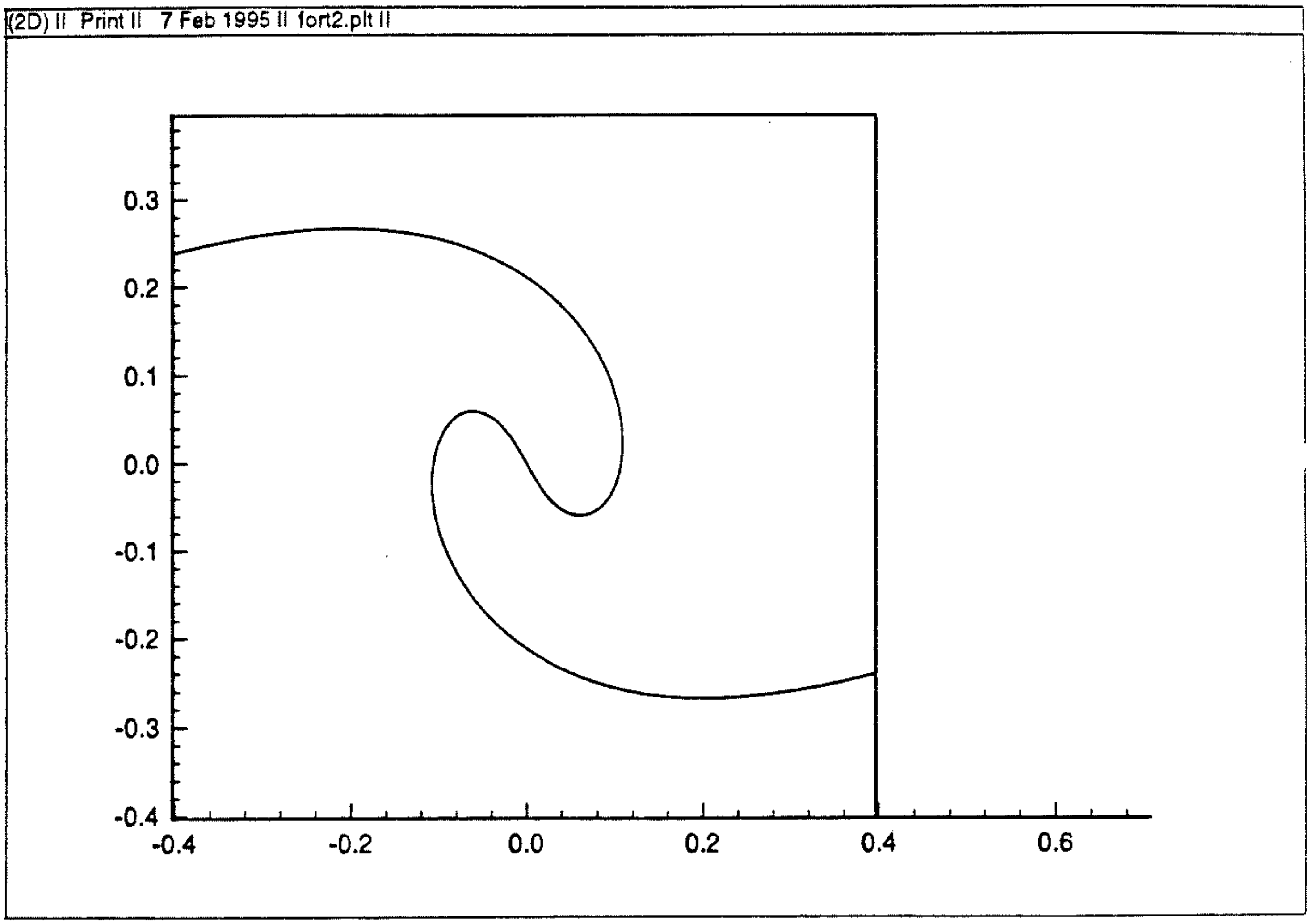

Figure 11a: Roll up of vortex sheet.

$t = 3, \qquad ENO3, \qquad 1024^2 \qquad \delta - \text{width} = 48\Delta x$

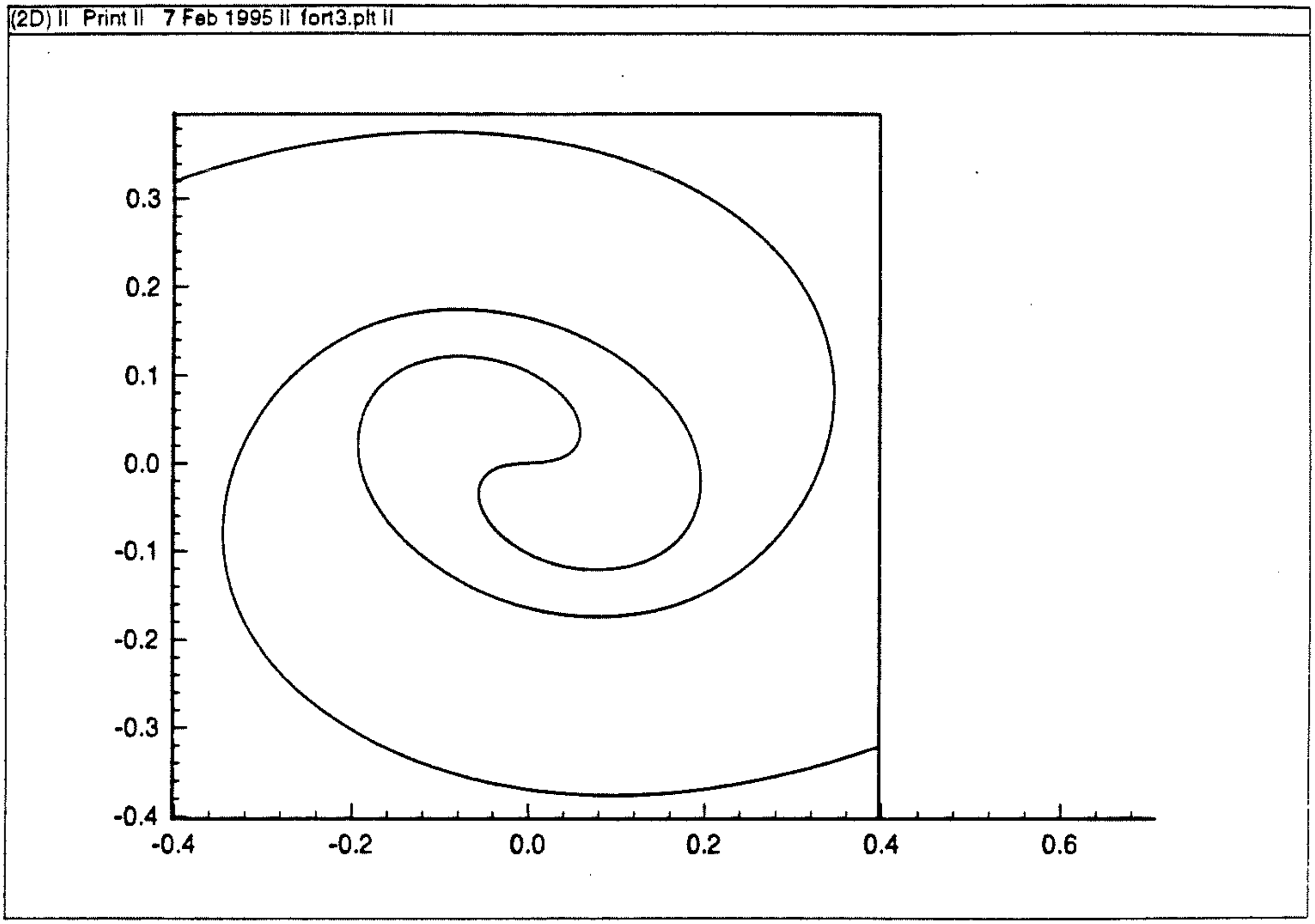

Figure 11b: Roll up of vortex sheet.

$t = 4, \qquad ENO3, \qquad 1024^2 \qquad \delta - \mathrm{width} = 48\Delta x$

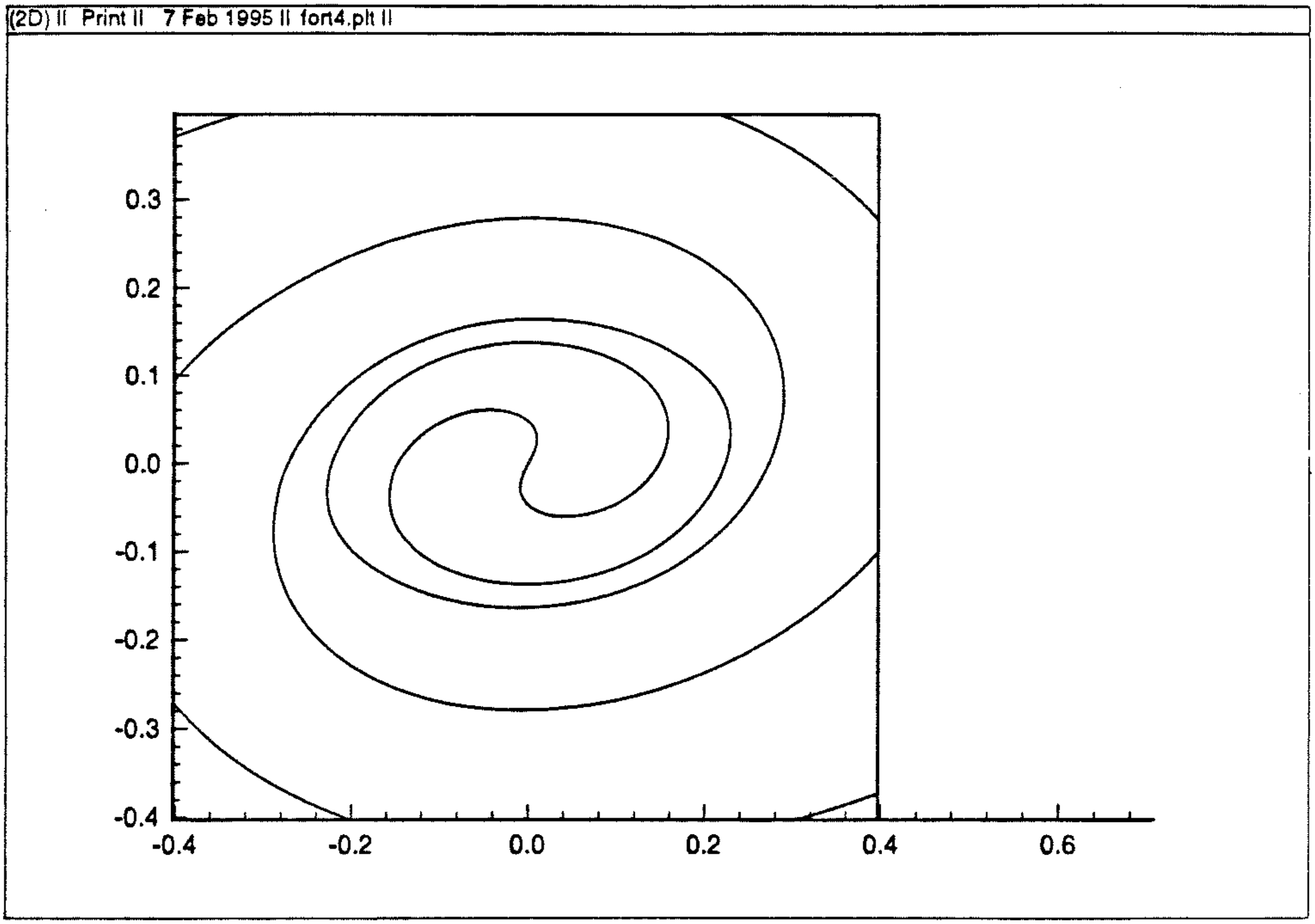

Figure 11c: Roll up of vortex sheet.

$t = 5, \qquad ENO3, \qquad 1024^2 \qquad \delta - \text{width} = 48\Delta x$

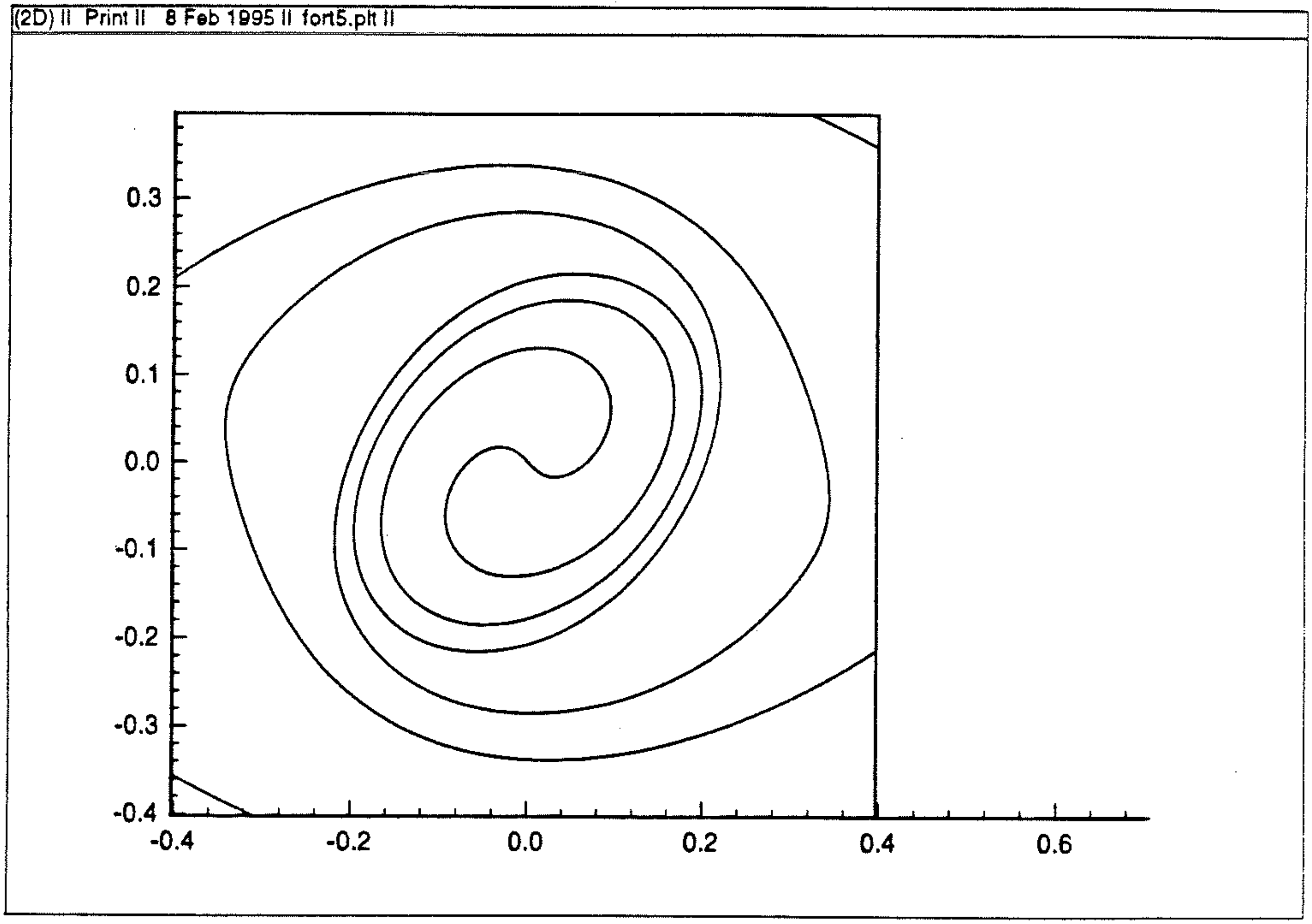

Figure 11d: Roll up of vortex sheet.

$t = 5.5, \qquad ENO3, \qquad 1024^2 \qquad \delta - \text{width} = 48\Delta x$

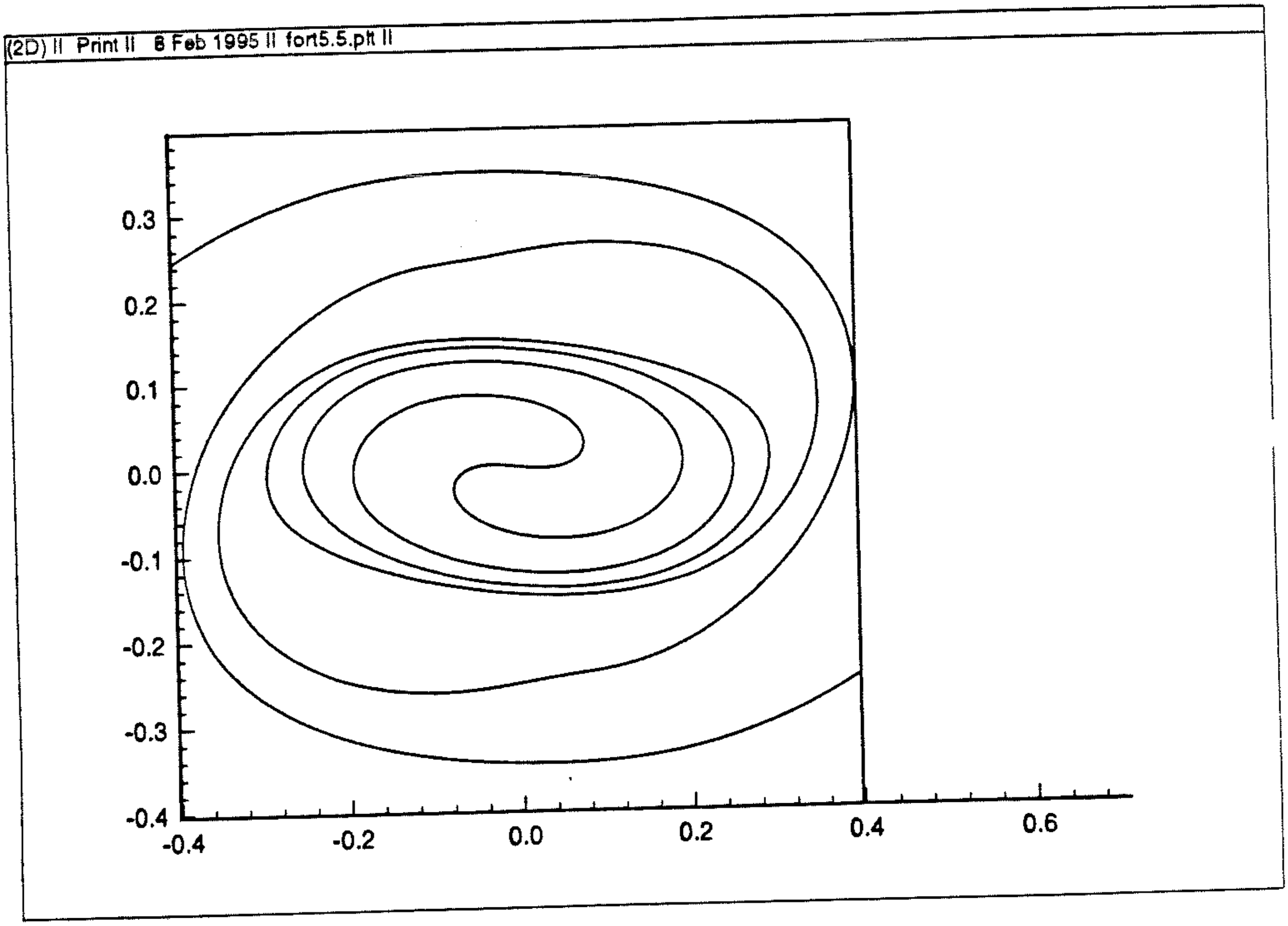

Figure 11e: Roll up of vortex sheet.

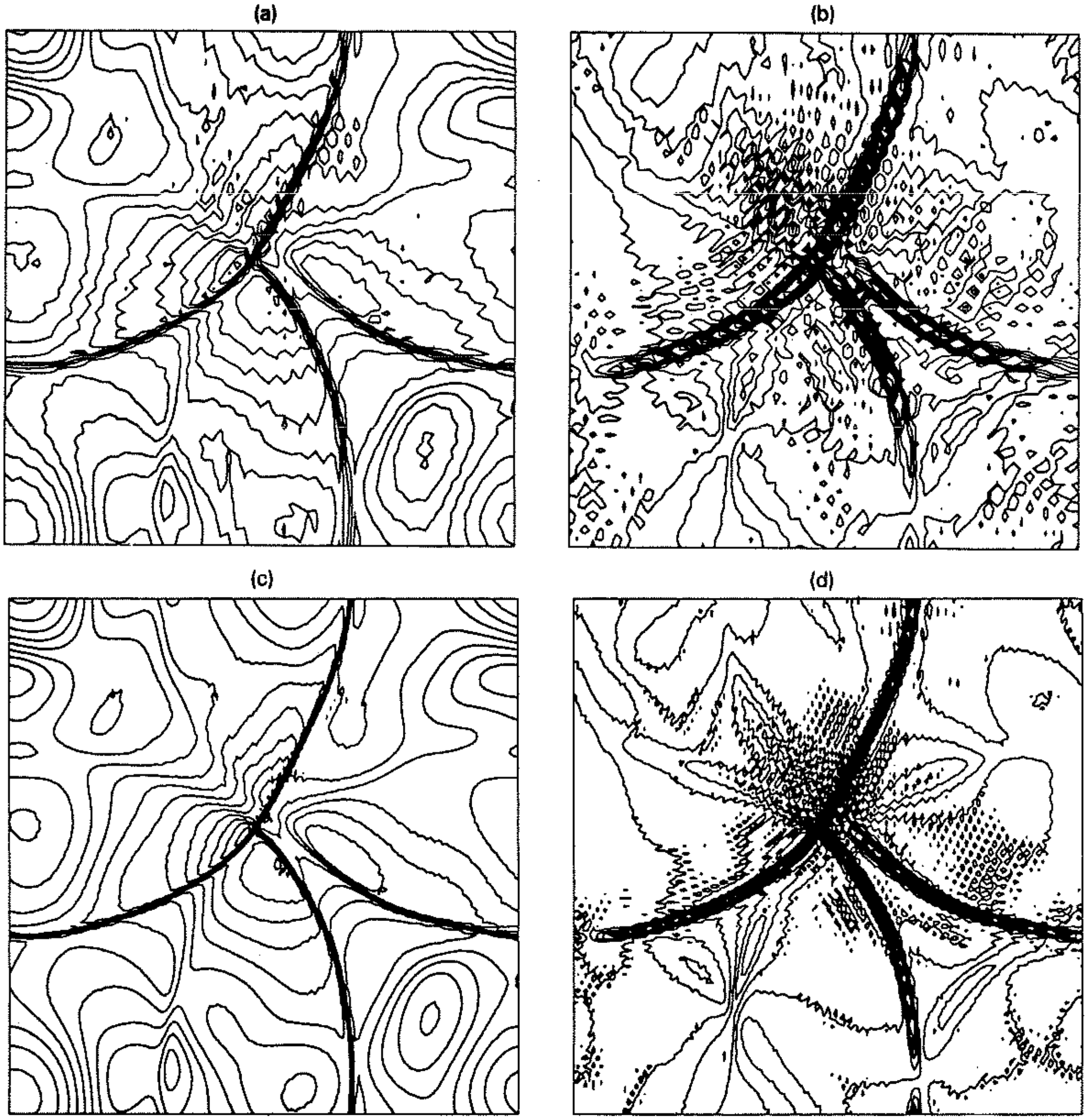

Figure 12: Density (left) and vorticity (right) contours for the spectral scheme with 64^2 (top) and 128^2 (bottom) points.

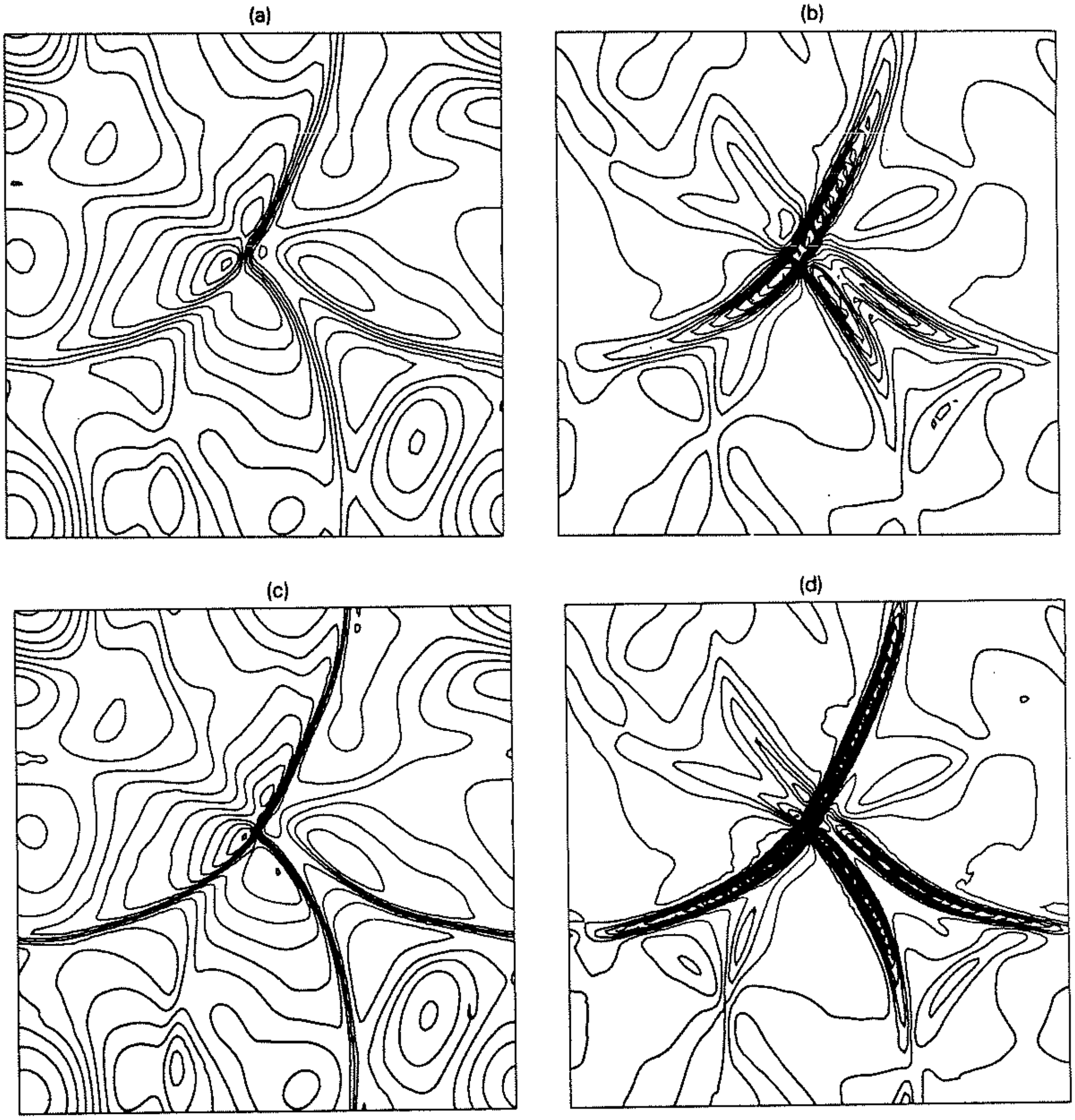

Figure 13: Density (left) and vorticity (right) contours for the third-order ENO-LF with 64^2 (top) and 128^2 (bottom) points.