Journal of Computational Physics **169,** 463–502 (2001)
doi:10.1006/jcph.2000.6636, available online at http://www.idealibrary.com on IDEAL®

Level Set Methods: An Overview and Some Recent Results[1]

Stanley Osher* and Ronald P. Fedkiw†

**Department of Mathematics, University of California, Los Angeles, Los Angeles, California 90095; and*
†Computer Science Department, Stanford University, Stanford, California 94305

Received March 1, 2000; revised September 11, 2000

The level set method was devised by S. Osher and J. A. Sethian (1988, *J. Comput. Phys*. **79**, 12–49) as a simple and versatile method for computing and analyzing the motion of an interface Γ in two or three dimensions. Γ bounds a (possibly multiply connected) region Ω. The goal is to compute and analyze the subsequent motion of Γ under a velocity field **v**. This velocity can depend on position, time, the geometry of the interface, and the external physics. The interface is captured for later time as the zero level set of a smooth (at least Lipschitz continuous) function $\varphi(\mathbf{x}, t)$; i.e., $\Gamma(t) = \{\mathbf{x} \mid \varphi(\mathbf{x}, t) = 0\}$. φ is positive inside Ω, negative outside Ω, and is zero on $\Gamma(t)$. Topological merging and breaking are well defined and easily performed. In this review article we discuss recent variants and extensions, including the motion of curves in three dimensions, the dynamic surface extension method, fast methods for steady state problems, diffusion generated motion, and the variational level set approach. We also give a user's guide to the level set dictionary and technology and couple the method to a wide variety of problems involving external physics, such as compressible and incompressible (possibly reacting) flow, Stefan problems, kinetic crystal growth, epitaxial growth of thin films, vortex-dominated flows, and extensions to multiphase motion. We conclude with a discussion of applications to computer vision and image processing.

1. INTRODUCTION

The original idea behind the level set method was a simple one. Given an interface Γ in R^n of codimension one, bounding a (perhaps multiply connected) open region Ω, we wish to analyze and compute its subsequent motion under a velocity field **v**. This velocity can depend on position, time, the geometry of the interface (e.g., its normal or its mean

[1] Research supported in part by ONR N00014-97-1-0027, DARPA/NSF VIP Grant NSF DMS9615854, AFOSR FQ8671-9801346, NSF DMS 9706827, and ARO DAAG 55-98-1-0323.

curvature), and the external physics. The idea, as devised in 1987 by S. Osher and J. A. Sethian [64] is merely to define a smooth (at least Lipschitz continuous) function $\varphi(x, t)$, that represents the interface as the set where $\varphi(x, t) = 0$. Here $x = x(x_1, \ldots, x_n)\ \varepsilon\ R^n$.

The level set function φ has the following properties:

$$\begin{aligned} &\varphi(x, t) > 0 \quad \text{for } x \in \Omega \\ &\varphi(x, t) < 0 \quad \text{for } x \notin \bar{\Omega} \\ &\varphi(x, t) = 0 \quad \text{for } x \in \partial\Omega = \Gamma(t). \end{aligned}$$

Thus, the interface is to be captured for all later time, by merely locating the set $\Gamma(t)$ for which φ vanishes. This deceptively trivial statement is of great significance for numerical computation, primarily because topological changes such as breaking and merging are well defined and performed “without emotional involvement.”

The motion is analyzed by convecting the φ values (levels) with the velocity field **v**. This elementary equation is

$$\frac{\partial\varphi}{\partial t} + \mathbf{v} \cdot \nabla\varphi = 0. \tag{1}$$

Here **v** is the desired velocity on the interface and is arbitrary elsewhere.

Actually, only the normal component of v is needed, $v_{\mathrm{N}} = \mathbf{v} \cdot \frac{\nabla\varphi}{|\nabla\varphi|}$, so (1) becomes

$$\frac{\partial\varphi}{\partial t} + v_{\mathrm{N}}|\nabla\varphi| = 0. \tag{2}$$

In Section 3 we give simple and computationally fast prescriptions for reinitializing the function φ to be signed distance to Γ, at least near the boundary [84], smoothly extending the velocity field v_{N} off of the front Γ [24] and solving Eq. (2) only locally near the interface Γ, thus lowering the complexity of this calculation by an order of magnitude [66]. This makes the cost of level set methods competitive with that of boundary integral methods, in cases where the latter are applicable; e.g, see [42].

We emphasize that all this is easy to implement in the presence of boundary singularities and/or topological changes and in two or three dimensions. Moreover, in the case which v_{N} is a function of the direction of the unit normal (as in kinetic crystal growth [62] and uniform density island dynamics [15, 36]), Eq. (2) becomes the first-order Hamilton–Jacobi equation

$$\frac{\partial\varphi}{\partial t} + |\nabla\varphi|\gamma(\mathbf{N}) = 0, \tag{3}$$

where $\gamma = \gamma(\mathbf{N})$ is a given function of the normal, $\mathbf{N} = (\nabla\varphi)/|\nabla\varphi|$.

High-order accurate, essentially nonoscillatory discretizations to general Hamilton–Jacobi equations including (3) were obtained in [64]; see also [43, 65].

Theoretical justification of this method for geometric-based motion came through the theory of viscosity solutions for scalar time-dependent partial differential equations [23, 30]. The notion of viscosity solution (see e.g., [8, 27])—which applies to a very wide class of these equations, including those derived from geometric-based motions—enables users to have confidence that their computer simulations give accurate, unique solutions. A particularly interesting result is in [29], where motion by mean curvature, as defined by Osher and Sethian in [64], is shown to be essentially the same motion as is obtained from the asymptotics in the phase field reaction diffusion equation. The motion in the level set method involves no superfluous stiffness as is required in phase field models. As was proven in [53], this stiffness due to a singular perturbation involving a small parameter ϵ will lead to incorrect answers as in [48] without the use of adaptive grids [59]. This is not an issue in the level set approach.

The outline of this paper is as follows: In Section 2 we present recent variants, extensions, and a rather interesting selection of related fast numerical methods. This section might be skipped at first, especially by newcomers to this subject. Section 3 contains the key definitions and basic level set technology, as well as a few words about the numerical implementation. Section 4 describes applications in which the moving interfaces are coupled to external physics. Section 5 concerns the variational level set approach with applications to multiphase (as opposed to two phase) problems. Section 6 gives a very brief introduction to the ever-increasing use of level set methods and related methods in image analysis.

2. RECENT VARIANTS, EXTENSIONS, AND RELATED FAST METHODS

2.1. *Motion of Curves in Three Spatial Dimensions*

In this section we discuss several new and related techniques and fast numerical methods for a class of Hamilton–Jacobi equations. These are all relatively recent developments and less experienced readers might skip this section at first.

As mentioned above, the level set method was originally developed for curves in R^2 and surfaces in R^3. Attempts have been made to modify it to handle objects of high codimension. Ambrosio and Soner [5] were interested in moving a curve in R^3 by curvature. They used the squared distance to the curve as the level set function, thus fixing the curve as the zero level set, and evolved the curve by solving a PDE for the level set function. The main problem with this approach is that one of the most significant advantages of the level set method, the ability to easily handle merging and pinching, does not carry over. A phenomenon called “thickening” emerges, where the curve develops an interior.

Attempts have also been made in other directions, e.g., front tracking (see [41]), where the curve is parameterized and then numerically represented by discrete points. The problem with this approach lies in finding when merging and pinching will occur and in reparameterizing the curve when they do. The representation we derived in [13] makes use of two level set functions to model a curve in R^3, an approach Ambrosio and Soner also suggested but did not pursue because the theoretical aspects became very difficult. In this formulation, a curve is represented by the intersection between the zero level sets of two level set functions ϕ and ψ, i.e., where $\phi = \psi = 0$. From this, many properties of the curve can be derived, such as the tangent vectors, $\mathbf{T} = \nabla\psi \times \nabla\phi / |\nabla\psi \times \nabla\phi|$, the curvature vectors, $\kappa\mathbf{N} = \nabla\mathbf{T} \cdot \mathbf{T}$, and even the torsion, $\tau\mathbf{N} = -\nabla\mathbf{B} \cdot \mathbf{T}$, where $\mathbf{N}$ and $\mathbf{B}$ are the normal and binormal respectively.

Motions of the curve can then be studied under the appropriate system of PDE’s involving the two level set functions. The velocity can depend on external physics, as well as on the geometry of the curve (as in the standard level set approach). The resulting system of PDEs for ψ and ϕ is

$$\begin{aligned} \phi_{\mathrm{t}} &= -\mathbf{v} \cdot \nabla\phi \\ \psi_{\mathrm{t}} &= -\mathbf{v} \cdot \nabla\psi. \end{aligned}$$

A simple example involves moving the curve according to its curvature vectors, for which

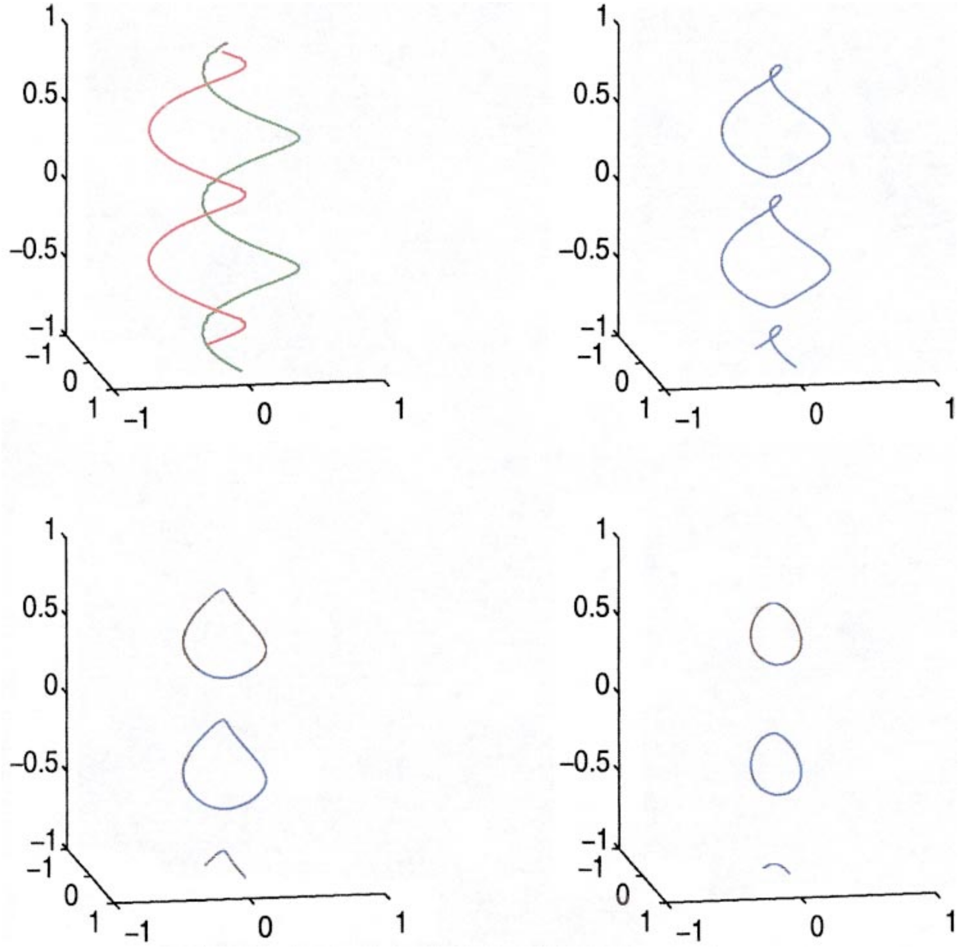


FIG. 1. Merging and pinching of curves in R^3 moving by mean curvature. Reprinted from [13].

$\mathbf{v} = \kappa\mathbf{N}$. We have shown that this system can also be obtained by applying a gradient descent algorithm minimizing the length of the curve,

$$L(\phi, \psi) = \int_{R^3} |\nabla\psi \times \nabla\phi|\delta(\psi)\delta(\phi)\, d\mathbf{x}.$$

This follows the general procedure derived in [88] for the variational level set method for codimension one motion, also described in [90]. Numerical simulations performed in [13] on this system of PDEs, and shown in Fig. 1 and 2, show, that merging and pinching off are handled automatically and follow curve-shortening principles.

We repeat the observation made above that makes this sort of motion easily accessible to this vector-valued level set method. Namely, all geometric properties of a curve Γ which is expressed as the zero level set of the vector equation

$$\begin{aligned} \phi(x, y, z, t) &= 0 \\ \psi(x, y, z, t) &= 0 \end{aligned}$$

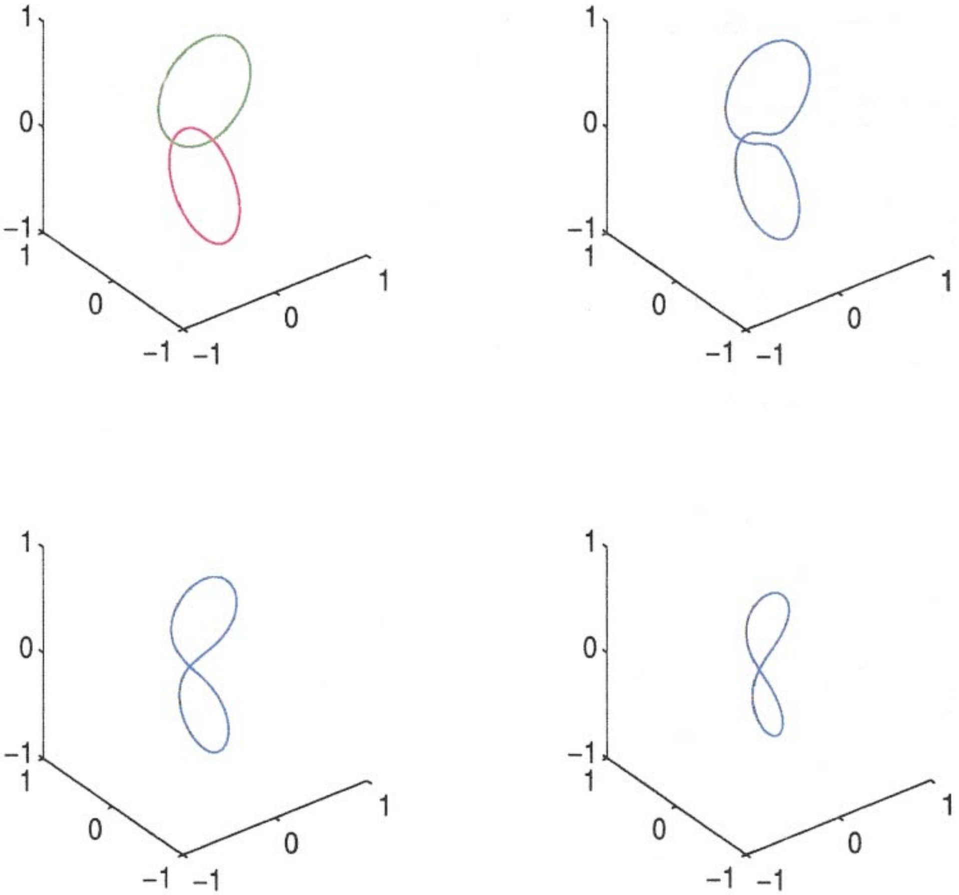


FIG. 2. Merging and pinching of curves in R^3 moving by mean curvature. Reprinted from [13].

can easily be obtained numerically by computing discrete gradients and higher derivatives of the functions ϕ and ψ restricted to their common zero level set.

This method will be used to simulate the dynamics of defect lines as they arise in heteroepitaxy of nonlattice notched materials; see [79, 80] for Lagrangian calculations.

An interesting variant of the level set method for geometry-based motion was introduced in [53] as diffusion-generated motion, and has now been generalized to forms known as convolution-generated motion or threshold dynamics. This method splits the reaction–diffusion approach into two highly simplified steps. Remarkably, a vector-valued generalization of this approach, as in the vector-valued level set method described above, gives an alternative way [74] to easily compute the motion (and merging) of curves moving normal to themselves in three dimensions with velocity equal to their curvature.

2.2. *Dynamic Surface Extension (DSE)*

Another fixed grid method for capturing the motion of self-intersecting interfaces was obtained in [73]. This is a fixed grid, interface-capturing formulation based on the dynamic surface extension (DSE) method of Steinhoff *et al.* [82]. The latter method was devised as an alternative to the level set method of Osher and Sethian [64] which is needed to evolve wavefronts according to geometric optics. The problem is that the wavefronts in this case are supposed to pass through each other—not merge as in the viscosity solution case. Ray-tracing can be used but the markers tend to diverge, which leads to loss of resolution and aliasing.

The original (ingenious) DSE method was not well suited to certain fundamental self-intersection problems such as formation of swallowtails. In [73] we extended the basic DSE scheme to handle this fundamental problem, as well as all other complex intersections.

The method is designed to track moving sets Γ of points of arbitrary (perhaps changing) codimension; moreover there is no concept of “inside” or “outside.” The method is, in some sense, dual to the level set method. In the latter, the distance representation is constant tangential to a surface. In the DSE method, the closest point to a surface is constant in directions orthogonal to the surface.

The version of DSE presented in [73] can be described as follows:

For each point in R^n, set the tracked point $\mathrm{TP}(\mathbf{x})$ equal to $\mathrm{CP}(\mathbf{x})$, the closest point (to $\mathbf{x}$) on the initial surface Γ_0. Set $\mathbf{N}$ equal to the surface normal at the tracked point $\mathrm{TP}(\mathbf{x})$. Let $\mathbf{v}(\mathrm{TP}(\mathbf{x}))$ be the velocity of the tracked point.

Repeat for all steps:

1. Evolve the tracked point $\mathrm{TP}(\mathbf{x})$ according to the local dynamics $\mathrm{TP}(\mathbf{x})_{\mathrm{t}} = \mathbf{v}(\mathrm{TP}(\mathbf{x}))$.

2. Extend the surface representation by resetting each tracked point $\mathrm{TP}(\mathbf{x})$ equal to the true closest point $\mathrm{CP}(\mathbf{x})$ on the updated surface Γ, where Γ is defined to be the locus of all tracked points; i.e., $\Gamma = \{\mathrm{TP}(\mathbf{x}) \mid \mathbf{x} \varepsilon R^n\}$.

Replace each $\mathbf{N}(\mathbf{x})$ by the normal at the updated $\mathrm{TP}(\mathbf{x})$.

This method treats self-intersection by letting moving sets pass through each other. This is one of its main virtues in the ray-tracing case. However, it has other virtues—namely the generality of the moving set—and curves can end or change dimension.

An important extension is motivated by considering first arrival times. This enables us to easily compute swallowtails, for example, and other singular points. We actually use a combination of distance and direction of motion. One interesting choice arises when nodal values of $\mathrm{TP}(\mathbf{x})$ are set equal to the “minimizing point”

$$\mathrm{MP}(\mathbf{x}) = \min_{\mathbf{y}\,\varepsilon\,\text{Interface}} \beta|(\mathbf{x}-\mathbf{y}) \cdot \mathbf{N}^{\perp}(\mathbf{y})| + \|\mathbf{x}-\mathbf{y}\|^2$$

for $\beta > 0$ (rather than $\mathrm{CP}(\mathbf{x})$), since a good agreement with the minimal arrival time representation is found near the surface. Recall that the minimal arrival time at a point $\mathbf{x}$ is the shortest time it takes a ray emanating from the surface to reach $\mathbf{x}$. Using this idea gives a very uniform approximation and naturally treats the prototype swallowtail problem.

For the special case of curvature-dependent motion we may use an elegant observation of DeGiorgi [28]. Namely the vector mean curvature for a surface of arbitrary codimension is given by $\kappa\mathbf{N} = -\Delta\nabla(d^2/2)$ where κ is the local mean curvature and d is the distance to the surface. Using the elementary, but basic fact that

$$d\nabla d = \mathbf{x} - \mathrm{CP}(\mathbf{x}),$$

where $\mathrm{CP}(\mathbf{x})$ is the closest point to $\mathbf{x}$ on the surface, we obtain a very simple expression for vector mean curvature:

$$\kappa\mathbf{N} = -\Delta(\mathbf{x} - \mathrm{CP}(\mathbf{x})) = \Delta\mathrm{CP}(\mathbf{x}).$$

Thus motion by a function F, of mean curvature for surfaces of arbitrary codimension can be achieved by using $\mathbf{v}(\mathrm{TP}(\mathbf{x})) = \Delta\mathrm{CP}(\mathbf{x})$. Then curvature-dependent velocities are possible by using

$$\mathbf{v} = F(\Delta\mathrm{CP}(\mathbf{x})|_{\mathrm{TP}(\mathbf{x})} \cdot \mathbf{N})\mathbf{N}.$$

Numerical experiments in [73] have validated these algorithms to some degree.

A variety of interesting topics for future research are still open. In particular, adjustments need to be made if merging is desired. Moreover we can move objects with more complex topology and geometry, such as surfaces with boundaries (or curves with endpoints), objects of composite topology (such as a filament attached to a sheet), and surfaces on curves with triple point junctions (see [53, 88] and Section 5 of this paper for successful level-set-based and diffusion-generated–based approaches for the codimension one case).

Further work in the area of curvature-dependent motions is also possible. Computationally the construction of fast extension methods and localization as in [66] for the level set method would be of great practical importance. It would be particularly interesting to determine if surfaces fatten (or develop interiors) when mergers occur. See [9] for a detailed discussion of this phenomenon.

Additionally, in [73], we successfully calculated a geometric optics expansion by retaining the wave-front curvature. Thus this method has the possibility of being quite useful in electromagnetic calculations. We hope to investigate its three-dimensional performance and include the effects of diffraction.

2.3. *A Class of Fast Hamilton–Jacobi Solvers*

Another important set of numerical algorithms involves the fast solution of steady (time-independent) Hamilton–Jacobi equations. We also seek methods which are faster than the globally defined schemes originally used to solve Eq. (2). The level set method of Osher and Sethian [64] for time-dependent problems can be localized. This means that the problem

$$\varphi_{\mathrm{t}} + \mathbf{v} \cdot \nabla\varphi = 0$$

with $\Gamma(t) = \{\mathbf{x} \mid \varphi(\mathbf{x}, t) = 0\}$ as the evolving front, can be solved locally near $\Gamma(t)$. Several algorithms exist for doing this; see [2, 66]. These both report an $O(N)$ algorithm where N is the total number of grid points on or near the front. However, the algorithm in [66] has $O(N\log(N))$ complexity because a partial-differential-equation-based reinitialization step requires $\log(\frac{1}{\Delta x}) \approx \log(N)$ steps to converge (we are grateful to Bjorn Engquist for pointing this out). The algorithm in [2] claims $O(N)$ complexity, but this is not borne out by the numerical evidence presented there.

However, for some special Hamilton–Jacobi equations, there is a fast method whose formal complexity is $O(N\log(N))$, but which, in our experience, is around one order of magnitude faster than these general local methods.

The idea is as follows:

For an equation of the form

$$\tilde{H}(\mathbf{x}, \nabla\psi) = 0,$$

given $\psi = 0$ on a noncharacteristic set S,

$$\nabla\psi \cdot \tilde{H}_{\nabla\psi} \neq 0,$$

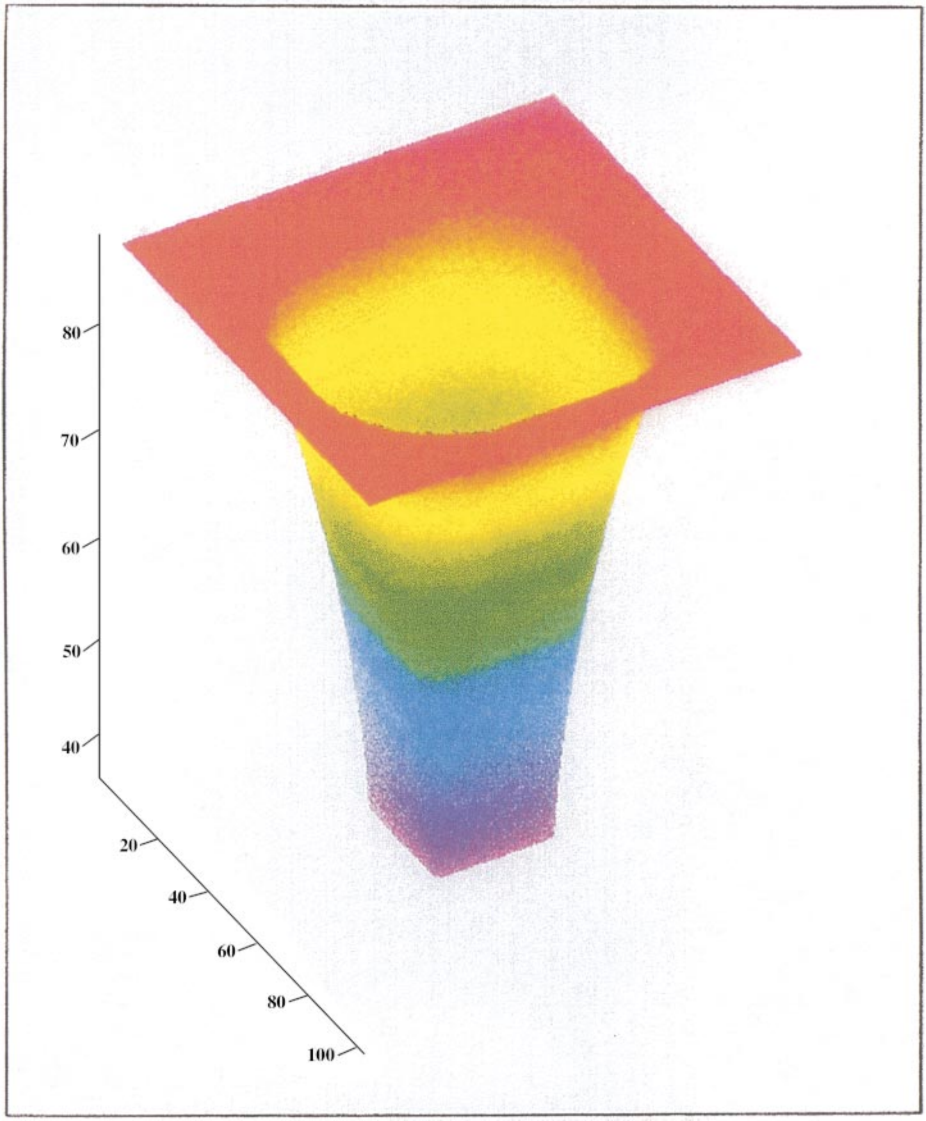


FIG. 3. Three-dimensional etching using a fast algorithm. Reprinted from [61].

we proved in [63] that the t level set

$$\{\mathbf{x} \mid \psi(\mathbf{x}) = t\} = \Gamma'(t)$$

is the same as the zero level set $\Gamma(t)$ of $\varphi(\mathbf{x}, t)$, for $t > 0$, where φ satisfies

$$\tilde{H}\left(\mathbf{x}, -\frac{\nabla\varphi}{\varphi_{\mathrm{t}}}\right) = 0.$$

This means that the viscosity solutions of both problems have level sets which correspond to each other. (This was also suggested in the original level set paper of Osher and Sethian [64].) Thus, one would like to find $\Gamma(t)$, the zero level set of $\varphi(x, t)$, as $\Gamma'(t)$, the t level set of $\psi(x)$.

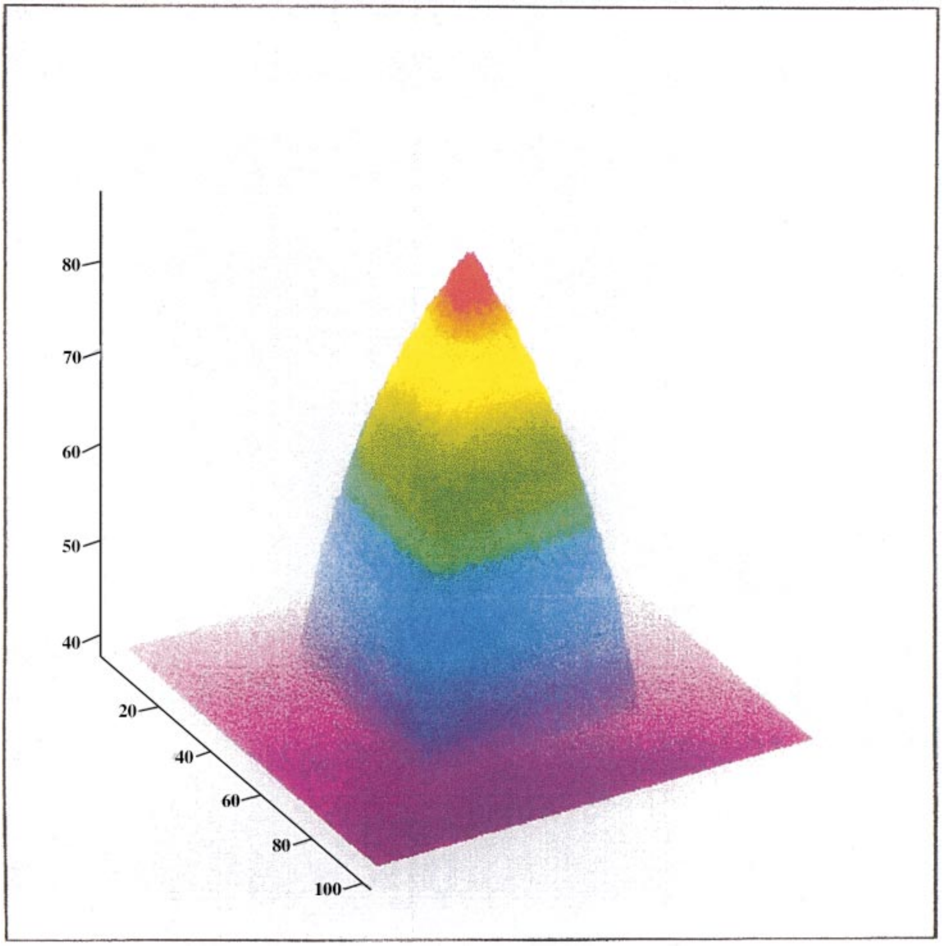


FIG. 4. Three-dimensional etching using a fast algorithm. Reprinted from [61].

A canonical example is the eikonal equation

$$\varphi_t + c(\mathbf{x})|\nabla\varphi| = 0, \qquad c(\mathbf{x}) < 0,$$

which can be replaced by

$$|\nabla\psi| = -\frac{1}{c(\mathbf{x})} = a(\mathbf{x}) > 0.$$

So we find first arrival times instead of zero level sets.

In [86] J. N. Tsitsiklis devised a fast algorithm for the eikonal equation. He obtained the viscosity solution using ideas involving Dijkstra's algorithm, adapted to the eikonal equation, heap sort, and control theory. From a numerical PDE point of view, however, Tsitsiklis had an apparently nonstandard approximation to $|\nabla\psi|$ on a uniform Cartesian grid.

In (1995) Sethian [76] and Helmsen *et al.* [40] independently published what appeared to be a simpler algorithm making use of the Rouy–Tourin algorithm to approximate $|\nabla\varphi|$. This has become known as the “fast marching method.” However, together with Helmsen [61], we have proven that Tsitsiklis’s approximation is the usual Rouy–Tourin [69] version of Godunov’s monotone upwind scheme. That is, the algorithm in [40, 76] is simply Tsitsiklis’s algorithm with a different (simpler) exposition.

Our goal here is to extend the applicability of this idea from the eikonal equation to any geometrically based Hamiltonian. By this we mean a Hamiltonian satisfying the properties

$$H(\mathbf{x}, \nabla\psi) > 0, \qquad \text{if } \nabla\psi \neq \mathbf{0} \tag{4}$$

and

$$H(\mathbf{x}, \nabla\psi) \text{ is homogeneous of degree one in } \nabla\psi. \tag{5}$$

We wish to obtain a fast algorithm to approximate the viscosity solution of

$$\tilde{H}(\mathbf{x}, \nabla\psi) = H(\mathbf{x}, \nabla\psi) - a(\mathbf{x}) = 0. \tag{6}$$

The first step is to set up a monotone upwind scheme to approximate this problem. Such a scheme is based on the idea of Godunov used in the approximation of conservation laws. In Bardi and Osher [7; see also 65], the following was obtained (for simplicity we exemplify using two space dimensions and ignore the explicit **x** dependence in the Hamiltonian),

$$\begin{aligned} H(\psi_x, \psi_y) &\approx H^{\mathrm{G}}(D_+^x\psi, D_-^x\psi_j; D_+^y\psi, D_-^y\psi) \\ &= \operatorname{ext}_{u\ \varepsilon\ I_{(u^-, U^+)}} \operatorname{ext}_{v\ \varepsilon\ I_{(v^-, v^+)}} H(u, v), \end{aligned}$$

where

$$I(a, b) = [\min(a, b), \max(a, b)]$$

$$\operatorname{ext}_u I(a, b) = \begin{cases} \min_{a \leq u \leq b} & \text{if } a \leq b \\ \max_{b \leq u \leq a} & \text{if } a > b \end{cases}$$

$$u_\pm = D_\pm^x \psi_{ij} = \pm\frac{(\psi_{i\pm 1,j} - \psi_{ij})}{\Delta x}, \quad v_\pm = D_\pm^y \psi_{ij} = \pm\frac{(\psi_{i,j\pm 1} - \psi_{ij})}{\Delta y}.$$

(Note that the order may be reversed in the ext operations above—we always obtain a monotone upwind scheme which is often, but not always, order invariant [65].)

This is a monotone upwind scheme which is obtained through the Godunov procedure involving Riemann problems, extended to general Hamilton– Jacob equations [7, 65].

If we approximate

$$H(\nabla\varphi) = a(x, y)$$

by

$$H^{\mathrm{G}}(D_+{}^x\varphi, D_-{}^x\varphi; D_+{}^x\psi; D_+{}^y\psi, D_-{}^y\psi) \tag{7}$$

for Hamiltonians satisfying (4), (5) above, then there exists a unique solution for $\psi_{i,j}$ in terms of $\psi_{i\pm1,j}$, $\psi_{i,j\pm1}$, and $\psi_{i,j}$. Furthermore $\psi_{i,j}$ is a nondecreasing function of all these variables.

However, the fast algorithim needs to have property $\mathcal{F}$: The solution to (7) depends on the neighboring $\psi_{\mu,\nu}$ only for $\psi_{\mu,\nu} < \psi_{i,j}$. This gives us a hint as to how to proceed.

For special Hamiltonians of the form $H(u, v) = F(u^2, v^2)$, with F nondecreasing in these variables, we have the result [61]

$$H^{\mathrm{G}}(u_+, u_-; v_+, v_-) = F(\max((u_+^-)^2, (u_-^+)^2); \max((v_+^-)^2, (v_-^+)^2)), \tag{8}$$

where $x^+ = \max(x, 0)$, $x^- = \min(x, 0)$. It is easy to see that this numerical Hamiltonian has the property $\mathcal{F}$ described above. This formula, as well as the one obtained in Eq. 10 below, enables us to extend the fast marching method algorithm to a much wider class than before. For example, using this observation we were able to solve an etching problem, also considered in [3], where the authors did not use a fast marching method algorithm, but instead used a local narrow band approach and schemes devised in [64]. The Hamiltonian was

$$H(\varphi_x, \varphi_y, \varphi_z) = \sqrt{\varphi_z^2}\left(1 + \left(4\left(\varphi_x^2 + \varphi_y^2\right)\right)\big/\left(\varphi_x^2 + \varphi_y^2 + \varphi_z^2\right)\right).$$

We are able to use the same heap-sort technology as for the eikonal equation for problems of this type. See Figs. 3 and 4. These Figures represent the level contours of an etching process whose normal velocity is a function of the direction of the normal. The process moves down in Fig. 3 and up in Fig. 4.

More generally, for $H(u, v)$ having the property

$$uH_1 \geq 0, \quad vH_2 \geq 0, \tag{9}$$

we also proved [61]

$$H^{\mathrm{G}}(u_+, u_-; v_+, v_-) = \max[H(u_+^-, v_+^-), H(u_-^+, v_+^-), H(u_+^-, v_-^+), H(u_-^+, v_-^+)], \tag{10}$$

and property $\mathcal{F}$ is again satisfied.

Again in [61], we were able to solve a somewhat interesting and very anisotropic etching problem with this new fast algorithm. Here we took

$$H(\varphi_x, \varphi_y) = |\varphi_y|\left(1 - a(\varphi_y)\varphi_y\big/\left(\varphi_x^2 + \varphi_y^2\right)\right),$$

where

$$\begin{aligned} a &= 0 \qquad \text{if } \varphi_y < 0 \\ a &= .8 \qquad \text{if } \varphi_y > 0 \end{aligned}$$

and observed merging of two fronts. See Figs. 5 and 6. These figures show a two-dimensional etching process resulting in a merger.

The fast method originating in [86] is a variant of Dijkstra's algorithm and as such involves the tree-like heap-sort algorithm to compute the smallest of a set of numbers. Recently Boué

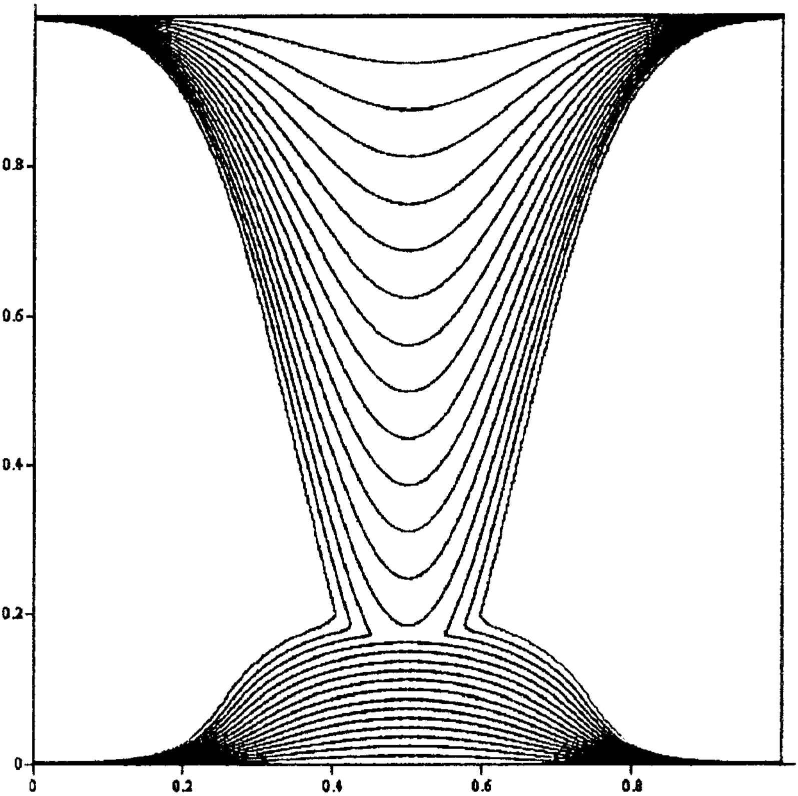


FIG. 5. Two-dimensional etching with merging using a fast algorithm. Reprinted from [61].

and Dupuis [11] have proposed an extremely simple fast algorithm for a class of convex Hamiltonians including those which satisfy (4) and (5) above. Basically, their statement is that the standard Gauss–Seidel algorithm, with a simple ordering, converges in a *finite* number of iterations for Eq. (7). This would give an $O(N)$, not $O(N \log N)$ operations, with an extremely simple-to-program algorithm—no heap sort is needed. Moreover, for the eikonal equation with $a(x, y) = 1$, the algorithm would seem to converge in $2^{\mathrm{d}}N$ iterations in R^{d}, $d = 1, 2, 3,$ which is quite fast. This gives a very simple and fast redistancing algorithm. For more complicated problems we have found more-iterations to be necessary, but have still obtained promising results, together with some theoretical justification. See [85] for details, which also include results for a number of nonconvex Hamiltonians. We call this technique the "fast sweeping method" in [85]. We refer to it in Section 3 when we discuss the basic distance reinitialization algorithm.

3. LEVEL SET DICTIONARY, TECHNOLOGY, AND NUMERICAL IMPLEMENTATION

We list key terms and define them by their level set representation.

1. The interface boundary $\Gamma(t)$ is defined by $\{\mathbf{x} \mid \varphi(\mathbf{x}, t) = 0\}$. The region $\Omega(t)$ is bounded by $\Gamma(t) : \{\mathbf{x} \mid \varphi(\mathbf{x}, t) > 0\}$ and its exterior is defined by $\{\mathbf{x} \mid \varphi(\mathbf{x}, t) < 0\}$.

2. The unit normal $\mathbf{N}$ to $\Gamma(t)$ is given by

$$\mathbf{N} = -\frac{\nabla\varphi}{|\nabla\varphi|}.$$

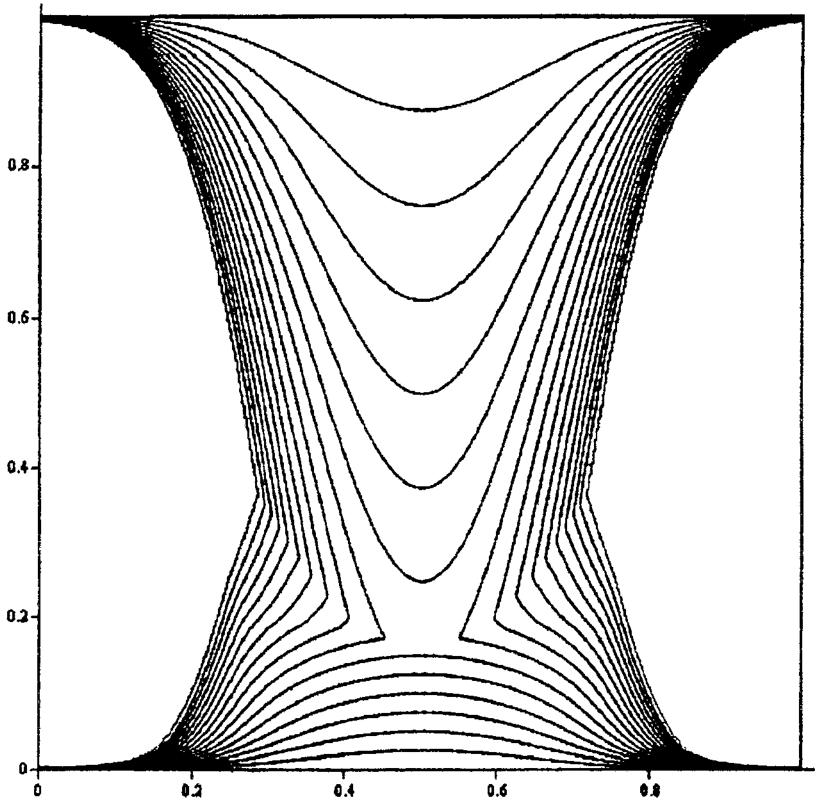


FIG. 6. Two-dimensional etching with merging using a fast algorithm. Reprinted from [61].

3. The mean curvature κ of $\Gamma(t)$ is defined by

$$\kappa = -\nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|}\right).$$

4. The Dirac delta function concentrated on an interface is

$$\delta(\varphi)|\nabla\varphi|,$$

where $\delta(x)$ is a one-dimensional delta function.

5. The characteristic function χ of a region $\Omega(t)$ is

$$\chi = H(\varphi),$$

where

$$\begin{aligned} H(x) &\equiv 1 \qquad \text{if } x > 0 \\ H(x) &\equiv 0 \qquad \text{if } x < 0 \end{aligned}$$

is a one-dimensional Heaviside function.

6. The surface (or line) integral of a quantity $p(\mathbf{x}, t)$ over Γ is

$$\int_{R^n} p(\mathbf{x}, t)\delta(\varphi)|\nabla\varphi|\, d\mathbf{x}.$$

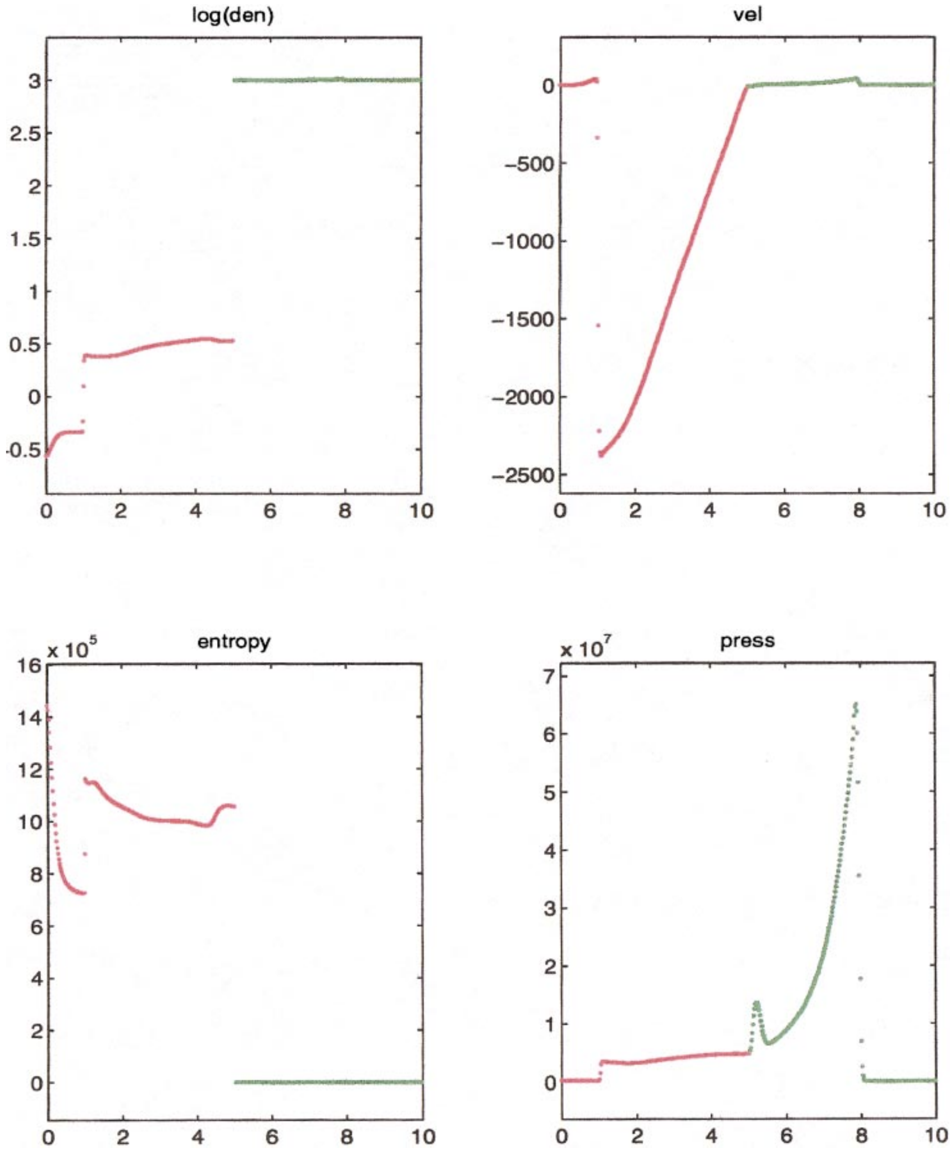


FIG. 7. Two-phase compressible flow calculated with the ghost fluid method. Air on the left and water on the right. Reprinted from [32].

7. The volume (or area) integral of $p(\mathbf{x}, t)$ over Ω is

$$\int_{R^n} p(\mathbf{x}, t) H(\varphi)\, d\mathbf{x}.$$

Next we describe three key technological advances which are important in many, if not most, level set calculations.

8. The distance reinitialization procedure replaces a general level set function $\varphi(\mathbf{x}, t)$ by $d(\mathbf{x}, t)$ which is the value of the distance from $\mathbf{x}$ to $\Gamma(t)$, positive outside and negative inside. This assures us that φ does not become too flat or too steep near $\Gamma(t)$. Let $d(\mathbf{x}, t)$ be

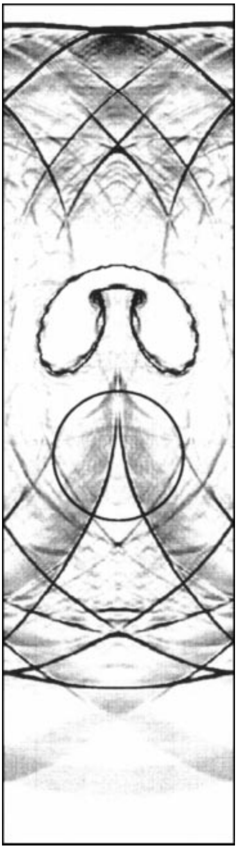


FIG. 8. Mach 1.22 air shock collapse of a helium bubble. Reprinted from [32].

signed distance of **x** to the closest point on Γ. The quantity $d(\mathbf{x}, t)$ satisfies $|\nabla d| = 1$, $d > 0$ in Ω, $d < 0$ in $(\bar{\Omega})^c$, and is the steady state solution (as $\tau \to \infty$) to

$$\begin{aligned} \frac{\partial \psi}{\partial \tau} + \operatorname{sgn}(\varphi)(|\nabla \psi| - 1) = 0 \\ \psi(\mathbf{x}, 0) = \varphi(\mathbf{x}, t), \end{aligned} \tag{11}$$

where $\operatorname{sgn}(x) = 2H(x) - 1$ is the one-dimensional signum function. This procedure was designed in [84]. The key observation is that in order to define d in a band of width ϵ around Γ, we need solve (11) only for $\tau = O(\epsilon)$. It can easily be shown that this can be used globally to construct distance (with arbitrary accuracy) in $O(N \log N)$ iterations [66]. Alternatively, we may use Tsitsiklis's fast algorithm [86], which is also $O(N \log N)$, with a much smaller constant, but which is only first-order accurate. A locally second-order accurate (in the high-resolution sense) fast marching method was proposed in [77]. While this method has a much lower local truncation error than a purely first-order accurate method, it is still globally, first-order accurate except for special cases. Finally, we might also use the fast sweeping method from [11, 85] as described in the last section, which appears to have $O(N)$ complexity and which is also only first-order accurate, although this complexity estimate has not been rigorously justified.

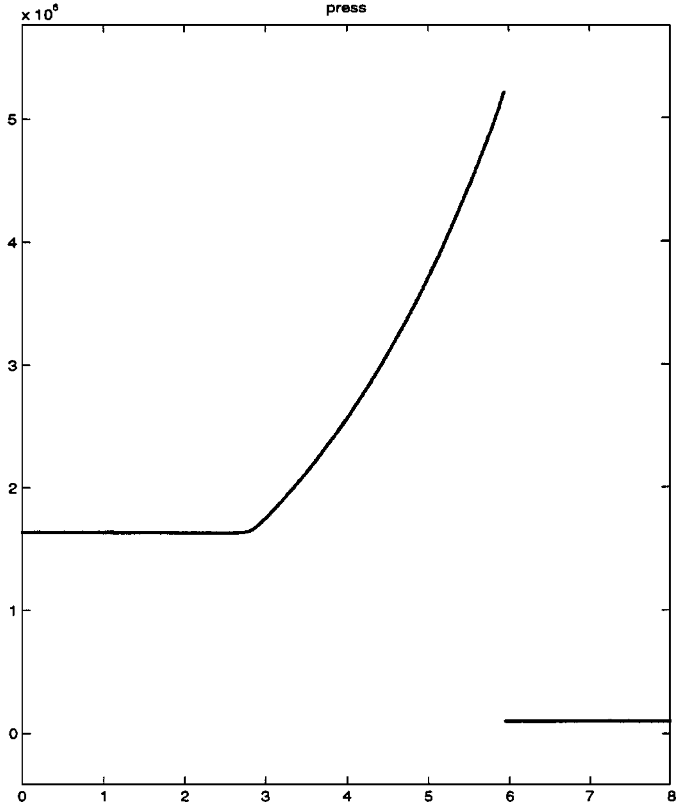


FIG. 9. Nonsmeared detonation wave traveling away from a solid wall. Reprinted from [33].

9. Smooth extension of a quantity, e.g., v_{n} on Γ to a neighborhood of Γ. Let the quantity be $p(\mathbf{x}, t)$. Solve to steady state $(\tau \rightarrow \infty)$

$$\frac{\partial q}{\partial \tau} + \operatorname{sgn}(\varphi)\left(\frac{\nabla\varphi}{|\nabla\varphi|} \cdot \nabla q\right) = 0$$
$$q(\mathbf{x}, 0) = p(\mathbf{x}, t).$$

Again, we need only solve this for $\tau = O(\epsilon)$ in order to extend p to be constant in the direction normal to the interface in a tube of width ϵ. This was first suggested and implemented in [24], analyzed carefully in [88], and further discussed and implemented in both [32], and [66]. A computationally efficient algorithm based on heap sort technology and fast marching methods was devised in [1]. There are many reasons to extend a quantity off of Γ, one of which is to obtain a well-conditioned normal velocity for level contours of φ close to $\varphi = 0$ [24]. Others involve implementation of the ghost fluid method of [32] discussed in the next section.

10. The basic level set method concerns a function $\varphi(\mathbf{x}, t)$ which is defined throughout space. Clearly this is wasteful if one only cares about information near the zero level set. The local level set method defines φ only near the zero level set. We may solve (2) in a neighborhood of Γ of width $m\Delta x$, where m is typically 5 or 6. Points outside of this neighborhood need not be updated by this motion. This algorithm works in "φ" space—so not too much intricate computer science is used. For details see [66]. Thus this local method

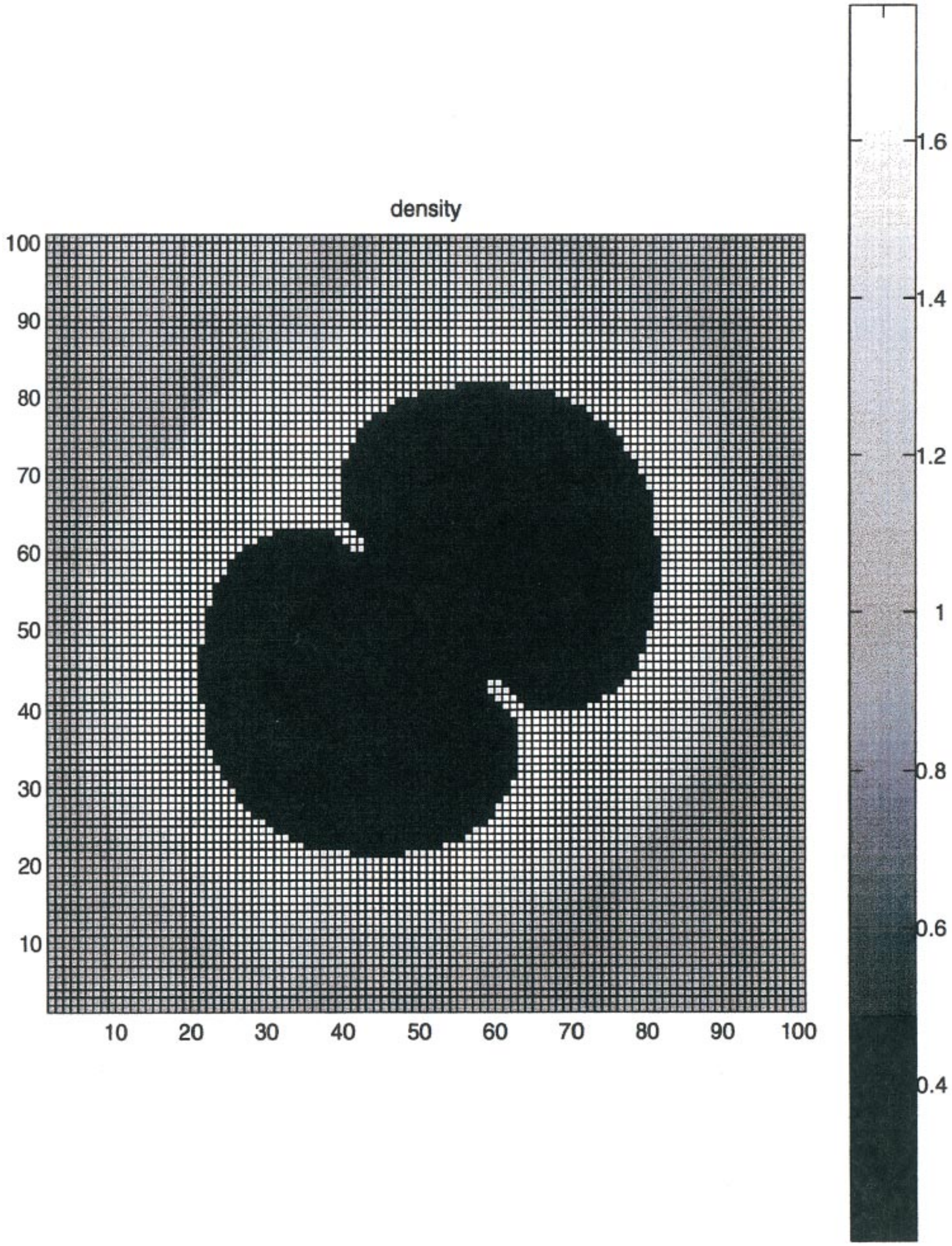
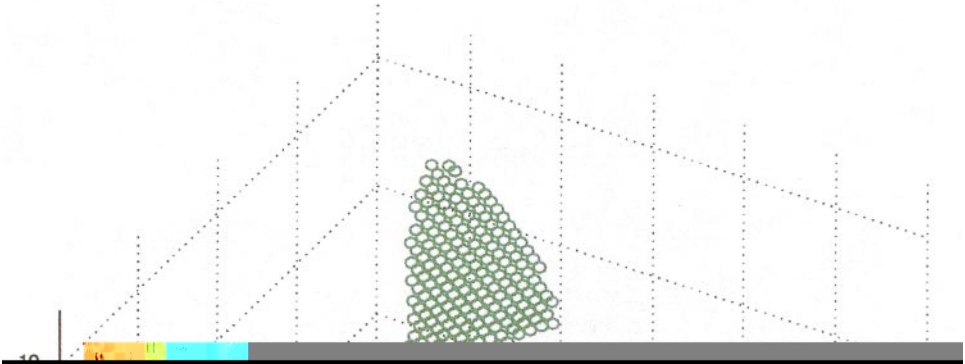


FIG. 10. Two deflagration fronts depicted shortly after merging. Reprinted from [33].

works easily in the presence of topological changes and for multiphase flow. An earlier local level set approach called "narrow banding" was devised in [2].

Finally, we repeat that, in the important special case where v_N in Eq. 2 is a function only of $\mathbf{x}$, t, and $\nabla\varphi$ (e.g., $v_N = 1$), Eq. 2 becomes a Hamilton–Jacobi equation whose solutions generally develop kinks (jumps in derivatives). We seek the unique viscosity solution. Many good references exist for this important subject; see, e.g., [8, 27]. The appearance of these singularities in the solution means that special, but not terribly complicated, numerical



[redacted]

FIG. 11. Two spatial dimensions, $\nabla \cdot (\frac{1}{\rho}\nabla p) = f(x, y), [p] = g(x, y), [\frac{1}{\rho}\nabla p \cdot \mathbf{N}] = h(x, y)$. Reprinted from [49].

methods have to be used, usually on uniform Cartesian grids. This was first discussed in [64] and numerical schemes developed there were generalized in [43, 65]. The key ideas involve monotonicity, upwind differencing, essentially nonoscillatory (ENO) schemes, and weighted essentially nonoscillatory (WENO) schemes. See [43, 64, 65] for more details.

4. COUPLING OF THE LEVEL SET METHOD WITH EXTERNAL PHYSICS

Interface problems involving external physics arise in various areas of science. The computation of such problems has a very long history. Methods of choice include front

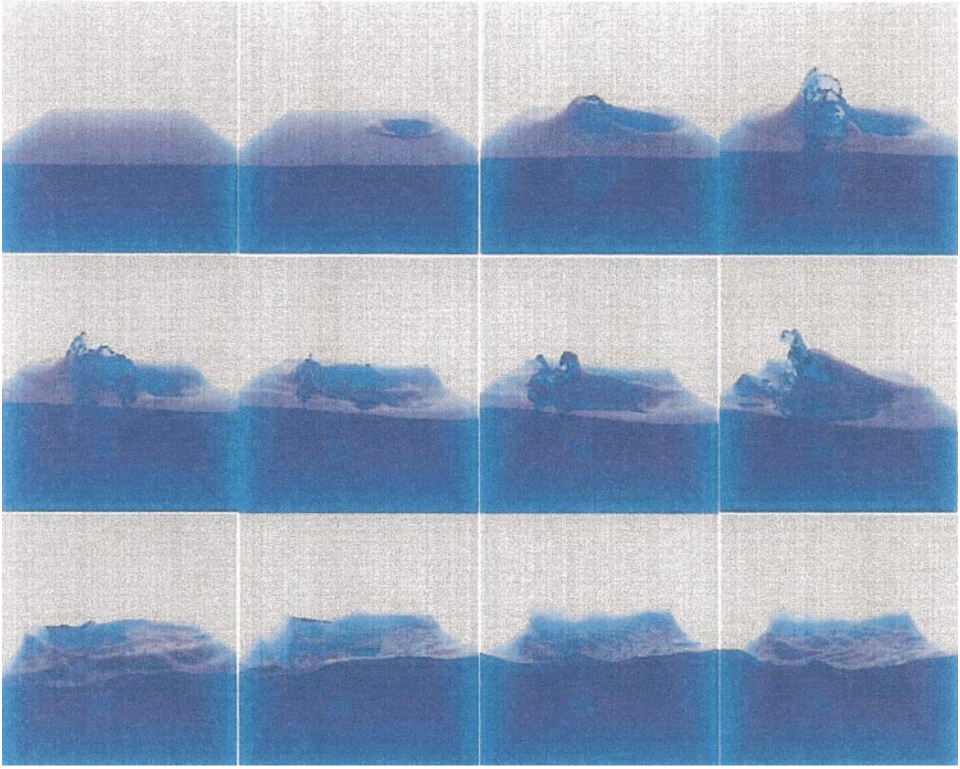


FIG. 12. Water waves generated by the impact of an (invisible) solid object. Reprinted from [44].

tracking—see, e.g., [41, 87]; phase-field methods—see, e.g., [48, 59]; and the volume of fluid (VOF) approach—see, e.g., [12, 60]. The level set method has had major successes in this area. Much of the level set technology discussed in the previous two sections was developed with such applications in mind.

Here, we shall describe level set approaches to problems in compressible flow, incompressible flow, flows having singular vorticity, Stefan problems, kinetic crystal growth, and a relatively new island dynamics model for epitaxial growth of thin films. We shall also discuss a recently developed technique, the ghost fluid method (GFM), which can be used (1) to remove numerical smearing and unphysical oscillations in flow variables near the interface and (2) to simplify the numerical linear algebra arising in some of the problems in this section and elsewhere.

4.1. *Compressible Flow*

Chronologically, the first attempt to use the level set method in this area came in two-phase inviscid compressible flow [55]. There, to the equations of conservation of mass, momentum, and energy, we appended Eq. (1), which we rewrote in conservation form as

$$(\rho\varphi)_t + \nabla\cdot(\rho\varphi\mathbf{v}) = 0 \tag{12}$$

using the density of the fluid ρ.

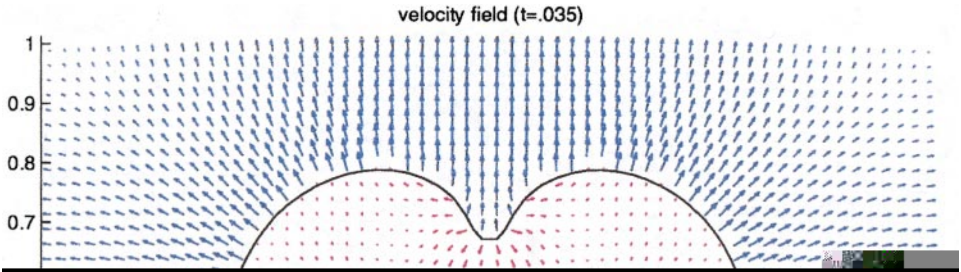


FIG. 13. Two-phase incompressible flames depicted shortly after merging (two spatial dimensions). Reprinted from [57].

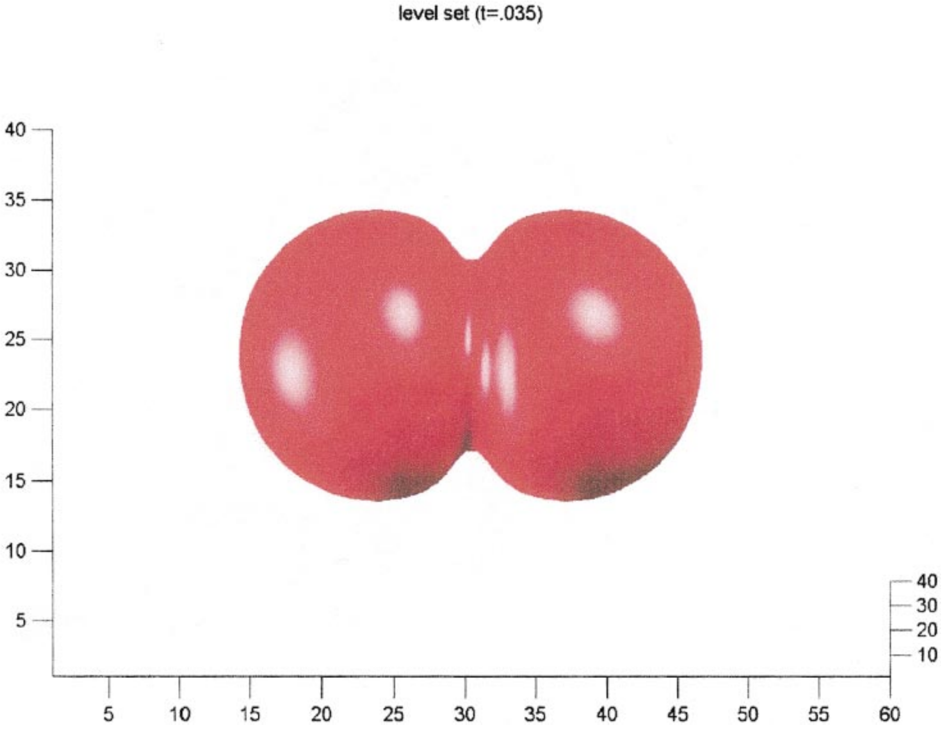


FIG. 14. Two-phase incompressible flames depicted shortly after merging (three spatial dimensions). Reprinted from [57].

The sign of φ is used to identify which gas occupied which region, so it determines the local equation of state. This (naive) method suffered from spurious pressure oscillations at the interface, as shown in [45, 46]. These papers proposed a new method which reduced these errors using a nonconservative formulation near the interface. However, [45, 46] still smear out the density across the interface, leading to terminal oscillations for many equations of state.

A major breakthrough in this area came in the development of the ghost fluid method (GFM) in [32]. This enables us to couple the level set representation of discontinuities to finite difference calculations of compressible flows. The approach was based on using the jump relations for discontinuities which are tracked using Eq. (1) (for two-phase compressible flow). What the method amounts to (in any number of space dimensions) is to populate cells next to the interface with "ghost values," which, for two-phase compressible flow, retain their usual values of pressure and normal velocity (quantities which are continuous across the interface), with extrapolated values of entropy and tangential velocity (which jump across the interface). These quantities are used in the numerical flux when "crossing" an interface.

An important aspect of the method is its simplicity. There is no need to solve a Riemann problem normal to the interface, to consider the Rankine–Hugoniot jump conditions, or solve an initial–boundary value problem. Another important aspect is its generality. The philosophy appears to be as follows: At a phase boundary, use a finite difference scheme which takes only values which are continuous across the interface, using the natural values whenever possible. Of course, this implies that the tangential velocity is treated in the same fashion as the normal velocity and the pressure when viscosity is present. The same holds true for the temperature in the presence of thermal conductivity.

Figure 7 shows results obtained for two-phase compressible flow using the GFM together with the level set method. Air, with density around 1 kg/m^3, is to the left of the interface and water, with density around 1000 kg/m^3, is to the right of the interface. Note that there is no numerical smearing of the density at the interface itself, which is fortunate, as water cavitates at a density above 999 kg/m^3 leading to a host of unphysical problems near the interface. Note too, that the pressure and velocity are continuous across the interface, although there are kinks in both of these quantities. A more complicated multidimensional calculation is shown in Fig. 8, where a shock wave in air impinges upon a helium droplet. See [32] for more details.

While the GFM was originally designed for multiphase compressible flow, it can be generalized to treat a large number of flow discontinuities. In [33], we generalized this method to treat shocks, detonations, and deflagrations in a fashion that removed the numerical smearing of the discontinuity. Figure 9 shows the computed solution for a detonation wave. Note that there is no numerical smearing of the leading wave front, which is extremely important in the elimination of spurious wave speeds for stiff source terms on coarse grids, as first pointed out by [26]. While shocks and detonations have associated Riemann problems, the Riemann problem for a compressible-flow deflagration discontinuity is not well posed unless the speed of the deflagration is given. Luckily, there is a large amount of literature on the G-equation for flame discontinuities, which was originally proposed in [50]. The G-equation represents the flame front as a discontinuity in the same fashion as the level set method so that one can easily consult the abundant literature on the G-equation to obtain deflagration speeds for the GFM. Figure 10 shows two initially circular deflagration fronts that have just recently merged together. Note that the light-colored region surrounding the deflagration fronts is a precursor shock wave that causes the initially circular deflagration waves to deform as they attempt to merge.

The GFM was extended in [34] in order to treat the two-phase compressible viscous Navier–Stokes equations in a manner that allows a large jump in viscosity across the interface. This paper spawned the technology needed to extend the GFM to multiphase incompressible flow including the effects of viscosity, surface tension, and gravity, as discussed in the next section.

4.2. *Incompressible Flow*

The earliest real success in coupling the level set method to problems involving external physics came in computing two-phase Navier–Stokes incompressible flow [22, 84]. The equations can be written as

$$\mathbf{u}_t + \mathbf{u}\cdot\nabla\mathbf{u} + \frac{\nabla p}{\rho} = \mathbf{g} + \frac{\nabla\cdot(2\mu D)}{\rho} + \frac{\delta(\varphi)\sigma\kappa\mathbf{N}}{\rho}$$

$$\nabla\cdot\mathbf{u} = 0,$$

where $\mathbf{u} = (u, v, w)$ is the fluid velocity, p is the pressure, $\rho = \rho(\varphi)$, and $\mu = \mu(\varphi)$ are the piecewise constant fluid densities and viscosities, g is the gravitational force, D is the viscous stress tensor, σ is the surface tension coefficient, κ is the curvature of the interface, $\mathbf{N}$ is the unit normal, and $\delta(\varphi)$ is a delta function. See [12, 87] for earlier front tracking and VOF methods (respectively) using a similar formulation. This equation is coupled to the front motion through the level set evolution equation (1) with $\mathbf{v} = \mathbf{u}$. This involves defining the interface numerically as having a finite width of approximately three to five grid cells. Within this smeared-out band, the density, viscosity, and pressure are modeled as continuous functions. Then the $\sigma\kappa\mathbf{N}/\rho$ term is used to approximate the surface tension forces which are lost when using a continuous pressure [84]. Successful computations using this model were performed in [22, 84]. Problems involving area loss were observed and significant improvements were made in [83].

As mentioned above, the technology from [34] motivated the extension of the ghost fluid method to this two-phase incompressible-flow problem. First, a new boundary-condition-capturing approach was devised and applied to the variable coefficient Poisson equation to solve problems of the form

$$\nabla\left(\frac{1}{\rho}\nabla p\right) = f,$$

where the jump conditions $[p] = g$ and $[\frac{1}{\rho}\nabla p\cdot\mathbf{N}] = h$ are given and ρ is discontinuous across the interface. This was accomplished in [49]. A sample calculation from [49] is shown in Fig. 11, where one can see that both the solution, p, and its first derivatives are sharp across the interface without numerical smearing. Next, this new technique was applied to multiphase incompressible flow in [44]. Here, since one can model the jumps in pressure directly, there is no need to add the $\sigma\kappa\mathbf{N}/\rho$ source term to the right-hand side of the momentum equation in order to capture the surface tension forces. Instead surface tension is modeled directly by imposing a pressure jump across the interface. In addition, [44] allows exact jumps in both ρ and μ so that the nonphysical finite width smeared-out interface in [84] can be replaced by a sharp interface. A three-dimensional calculation of an (invisible) solid sphere impacting water and causing a splash is shown in Fig. 12. Here the air has density near 1 kg/m^3 while the water has density near 1000 kg/m^3.

Recently, in [57], this boundary-condition-capturing technology was extended to treat two-phase incompressible flames where the normal velocity is discontinuous across the interface as well. Figure 13 shows an example calculation where two flames have just merged. Note that the velocity vectors in Fig. 13 clearly indicate that the velocity is kept discontinuous across the flame front. Two-phase incompressible flames were considered in [39], as well, and a method was proposed that keeps the interface sharp and removes numerical smearing. Unfortunately, the method proposed in [39] cannot treat topological changes in the flame front. Our method improves upon [39], allowing flame-front discontinuities to merge, as in Fig. 13, or pinch off. Figure 14 shows two flame fronts shortly after merging in three spatial dimensions.

4.3. *Topological Regularization*

In [37, 38], it is shown that the level set formulation provides a novel way to regularize certain ill-posed equations of interface motion by blocking interface self-intersection. We computed two- and three-dimensional unstable vortex motion without regularization other than that in the discrete approximation to $\delta(\varphi)$—this is done over a few grid points. The key observation is that viewing a curve or surface as the level set of a function, and then evolving it with a perhaps unstable velocity field, prevents certain types of blow-up from occuring. This is denoted as "topological regularization." For example, a tracked curve can develop a figure eight pattern, but a level-set-captured curve will pinch off and stabilize before this happens. For the setup (involving two functions), see [37], where we perform calculations involving the Cauchy–Riemann equations. The motions agree until pinch off, when the topological stabilization develops.

As an example, we considered the two-dimensional incompressible Euler equations, which may be written as

$$\begin{aligned} \omega_{\mathrm{t}} + \mathbf{u} \cdot \nabla\omega &= 0 \\ \nabla \times \mathbf{u} &= \omega \\ \nabla \cdot \mathbf{u} &= 0. \end{aligned}$$

We are interested in situations in which the vorticity is initially concentrated on a set characterized by the level set function φ as follows:

$$\begin{aligned} \text{vortex parth: } \omega &= \mathrm{H}(\varphi) \\ \text{vortex sheet: } \omega &= \delta(\varphi), \left(\text{strength of sheet is } \frac{1}{|\nabla\varphi|}\right) \\ \text{vortex sheet dipole: } \omega &= \frac{d}{d\varphi}\delta(\varphi) = \delta'(\varphi). \end{aligned}$$

The key observation is that φ also satisfies a simple advection equation and $\mathbf{u}$ and ω can be easily recovered. For example, for the vortex sheet case we solve

$$\begin{aligned} \varphi_{\mathrm{t}} + \mathbf{u} \cdot \nabla\varphi &= 0 \\ \mathbf{u} &= \begin{pmatrix} -\partial y \\ \partial x \end{pmatrix} \Delta^{-1}\delta(\varphi). \end{aligned}$$

Standard Laplace solvers may be used. See [38] for results involving two- and three-dimensional flows. In [66] we added reinitialization and extension to this procedure and obtained improved results in the two-dimensional case.

4.4. *Stefan Problem*

Another classical field concerns Stefan problems [24]; see also [78] for an earlier, but much more involved level-set-based approach. Here we wish to simulate melting ice or freezing water, or more complicated crystalline growth, as in the island dynamics model discussed below.

We begin with a simplified, nondimensionalized model (see [47] for an extension as mentioned below),

$$\begin{aligned} \frac{\partial T}{\partial t} &= \nabla^2 T, \qquad \mathbf{x} \notin \partial\Omega = \Gamma(t) \\ v_{\mathrm{N}} &= [\nabla T \cdot N], \qquad \mathbf{x}\,\varepsilon\,\Gamma(t), \end{aligned}$$

where $[\cdot]$ denotes the jump across the boundary, and

$$T = -\bar{\varepsilon}_{\mathrm{c}}\kappa(1 - A\cos(\kappa_A\theta + \theta_0)) + \bar{\varepsilon}_{\mathrm{v}} v_{\mathrm{n}}(1 - A\cos(\kappa_A\theta + \theta_0))$$

on $\Gamma(t)$ and where κ is the curvature, $\theta = \cos^{-1}\varphi_x/|\nabla\varphi/|$, and the constants $A, \kappa_A, \theta_0, \bar{\varepsilon}_{\mathrm{c}}$ and $\bar{\varepsilon}_{\mathrm{v}}$ depend on the material being modeled.

We directly discretize the boundary conditions at Γ: To update T at grid nodes near the boundary, if the stencil for the heat equation would cross Γ (as indicated by nodal sign change in φ), we merely use dimension by dimension one-sided interpolation and the given boundary T value at an imaginary node placed at $\varphi = 0$ (found by interpolation on φ) to compute T_{xx} and/or T_{yy} (never interpolating across the interface), rather than the usual three-point central stencils. The level set function φ is updated and then reinitialized to be equal to the signed distance to Γ. Note that the level set update uses v_{N} that has been extended off the interface. See [24] for details.

We note that one can easily extend this to

$$\frac{\partial T}{\partial t} = \nabla \cdot (\kappa \nabla T),$$

where κ is a different positive constant inside and outside of Ω and

$$v_{\mathrm{N}} = [\kappa \nabla T \cdot \mathbf{N}], \qquad \mathbf{x}\,\varepsilon\,\Gamma(t),$$

as was recently done in [47].

An important observation is that our finite differencing at the interface leads to a non-symmetric matrix inversion when implicit discretization in time, is applied, although the method does have nice properties such as second-order accuracy and a maximum principle. This lack of symmetry is a bit problematic for a fast implementation, especially for very large values of κ. Fortunately, an extension of GFM can be used to derive a different spatial discretization, producing a symmetric matrix that can be inverted rather easily using fast methods. This was originally proposed by Fedkiw [31] and is described below.

It is sufficient to explain how the spatial derivatives are derived with respect to one variable, since there are no mixed partial derivative terms. Suppose the interface point, x_{f}, falls in between two grid points x_i and x_{i+1}. From ϕ, the distances between x_i, x_{i+1}, and x_{f} can be estimated by

$$x_{\mathrm{f}} - x_i \approx (-\phi_i \Delta x)/(\phi_{i+1} - \phi_i) = \theta_1 \Delta x \tag{13}$$

$$x_{i+1} - x_{\mathrm{f}} \approx (-\phi_{i+1} \Delta x)/(\phi_{i+1} - \phi_i) = \theta_2 \Delta x. \tag{14}$$

To avoid numerical errors caused by division by 0, θ_1 or θ_2 are not used if either is less than Δx^2. If $\theta_1 < \Delta x^2$, then x_{f} is assumed equal to x_i. If $\theta_2 < \Delta x^2$, then x_{f} is assumed equal to x_{i+1}. Either assumption is effectively a second-order perturbation of the interface location leading to second-order accurate spatial discretization. The nonsymmetric second-order accurate discretization for T_{xx} given in [24] is

$$(T_{xx})_i \approx \left(\left(\frac{T_{\mathrm{f}} - T_i}{\theta_1 \Delta x}\right) - \left(\frac{T_i - T_{i-1}}{\Delta x}\right)\right) \Big/ \frac{1}{2}(\theta_1 \Delta x + \Delta x) \tag{15}$$

$$(T_{xx})_{i+1} \approx \left(\left(\frac{T_{i+2} - T_{i+1}}{\Delta x}\right) - \left(\frac{T_{i+1} - T_{\mathrm{f}}}{\theta_2 \Delta x}\right)\right) \Big/ \frac{1}{2}(\Delta_x + \theta_2 \Delta x), \tag{16}$$

where T_{f} denotes the value of T at x_{f} and is determined from the boundary condition. Instead of using the nonsymmetric equations (15) and (16), Fedkiw [31] proposed using

$$(T_{xx})_i \approx \left(\frac{T_{\mathrm{f}} - T_i}{\theta_1 \Delta x} - \frac{T_i - T_{i-1}}{\Delta x}\right) \Big/ \Delta x \tag{17}$$

$$(T_{xx})_{i+1} \approx \left(\frac{T_{i+2} - T_{i+1}}{\Delta x} - \frac{T_{i+1} - T_{\mathrm{f}}}{\theta_2 \Delta x}\right) \Big/ \Delta x, \tag{18}$$

which leads to a symmetric linear system when using implicit time discretization. Equation (17) is derived using linear extrapolation of T from one side of the interface to the other, obtaining

$$T_{\mathrm{G}} = T_{\mathrm{f}} + (1 - \theta_1)\left(\frac{T_{\mathrm{f}} - T_{\mathrm{i}}}{\theta_1}\right) \tag{19}$$

as a ghost cell value for T at x_{i+1}. The standard second-order discretization of $\frac{\partial^2 T}{\partial x^2}$ at x_i using T_{G} at x_{i+1} is

$$(T_{xx})_i \approx \left(\frac{T_{\mathrm{G}} - T_i}{\Delta x} - \frac{T_i - T_{i-1}}{\Delta x}\right) \Big/ \Delta x, \tag{20}$$

and the substitution of Eq. (19) into Eq. (20) leads directly to (17). Equation (18) is derived similarly.

Formulas (17) and (18) have $O(1)$ errors using formal truncation error analysis. However, they are second-order accurate on a problem where the interface has been perturbed by $O(\Delta x^2)$, making them second-order accurate in the interface location. Assume that the standard second-order accurate discretization is used to obtain the standard linear system of equations for T at every grid point except for those adjacent to the interface, that is except for x_i and x_{i+1}. Since the linear system of equations for the nodes to the left and including x_i is independent of the system for the nodes to the right including x_{i+1}, only the linear system to the left is discussed here. Equation (20) is used to write a linear equation for T_i introducing a new unknown T_G, and the system is closed with Eq. (19) for T_G. In practice, Eqs. (19) and (20) are combined to obtain Eq. (17) and a symmetric linear system of equations. This linear system of equations results in well-determined values (up to some prescribed tolerance near roundoff error levels) of T at each grid node, as well as a well-determined value of T_G (from Eq. (19)). For the sake of reference, designate **T** as the solution vector containing the values of T at each grid point to the left and including x_i as well as the value of T_G at x_{i+1}which are obtained by solving this symmetric linear system. Below, **T** is shown to be a second-order accurate solution to our problem by showing that it is the second-order accurate solution to a modified problem where the interface location has been perturbed by $O(\Delta x^2)$.

Consider the modified problem where a Dirichlet boundary condition of $T = T_\mathrm{G}$ is specified at x_{i+1}, where T_G is chosen to be the value of T_G from **T** defined above. This modified problem can be exactly discretized to second-order accuracy everywhere using the standard discretization at every node except x_i, where Eq. (20) is used. We note that Eq. (20) is the standard second-order accurate discretization when a Dirichlet boundary condition of $T = T_\mathrm{G}$ is applied at x_{i+1}. This new linear system can be discretized and solved in a standard fashion to obtain a second-order accurate solution at each grid node. Then the realization that **T** is an exact solution to *this* linear system implies that **T** is a second-order accurate solution to this modified problem. Next consider the interface location dictated by the modified problem. Since **T** is a second-order accurate solution to the modified problem, **T** can be used to obtain the interface location to second-order accuracy. The linear interpolant that uses T_i at x_i and T_G at x_{i+1} predicts an interface location of *exactly* x_f, which is the true interface location. Since higher order interpolants (higher than linear) can contribute at most an $O(\Delta x^2)$ perturbation of the interface location, the interface location dictated by the modified problem is at most an $O(\Delta x^2)$ perturbation of the true interface location, x_f.

In [25], we used this strategy to obtain a second-order accurate symmetric discretization of the variable coefficient Poisson equation.

$$\nabla(k\nabla T) = f$$

on irregular domains in as many as three spatial dimensions. Then, in a straightforward way, we obtained second-order accurate symmetric discretizations of the heat equation on irregular domains using backward Euler time stepping with $\Delta t = (\Delta x)^2$ and Crank–Nicolson time stepping with $\Delta t = \Delta x$.

4.5. *Kinetic Crystal Growth*

For an initial state consisting of any number of growing crystals in R^d, d arbitrary, moving outward with given normal growth velocity $\mathbf{v}(\mathbf{N}) > 0$ which depends on the angle of the unit surface normal **N**, the asymptotic growth shape is a single (kinetic) Wulff-construct crystal. This result was first conjectured by Gross in (1918) [35]. This shape is also known to minimize the surface integral of $\mathbf{v}(\mathbf{N})$ for a given volume. We gave a proof of this result [62], see also [81], using the level set formulation and the Hopf–Bellman formulas [6] for the solution of a Hamilton–Jacobi equation. Additionally, with the help of the Brunn–Minkowski inequality, we showed that if we evolve a convex surface under the motion described in (3), then the ratio to be minimized monotonically decreases to its minimum as time increases, which provides a new proof that the Wulff construction solves the generalized isoperimetric problem as well. Thus there is a new link between this hyperbolic surface evolution and this (generally nonconvex) energy minimization. This also provides a convenient framework for numerically computing anisotropic kinetic crystal growth [67]. The theoretical and numerical results of this work are illustrated in the uniform density island dynamics models of [15, 36]. That model describes crystals growing in two dimensions with an anisotropic velocity.

An interesting spinoff of this work came in [67], in which we proved that any two-dimensional Wulff shape can be interpreted precisely as the solution of a Riemann problem for a scalar conservation law—contact discontinuities correspond to jumps in the angle of the normal to the shape, smoothly varying nonflat faces correspond to rarefaction waves, and planar facets correspond to constant states, which develop because of kinks in the conservation law's flux function. These kinks are also seen in the convexified Wulff energy.

4.6. *Epitaxial Growth of Thin Films*

A new continuum model for the epitaxial growth of thin films has been developed. Molecular beam epitaxy (MBE) is a method for growing extremely thin films of material. The essential aspects of this growth process are as follows: Under vacuum conditions a flux of atoms is deposited on a substrate material, typically at a rate that grows one atomic monolayer every several seconds. When deposition flux atoms hit the surface, they bond weakly rather than bounce off. These surface "adatoms" are relatively free to hop from lattice site to lattice site on a flat (atomic) planar surface. However, when they hop to a site at which there are neighbors at the same level, they form additional bonds which hold them in place. This bonding could occur at the "step edge" of a partially formed atomic monolayer, contributing to the growth of that monolayer. Or it could occur when two adatoms collide with each other. If the critical cluster size is one, the colliding adatoms nucleate a new partial monolayer "island" that will grow by trapping other adatoms at its step edges.

By these means, the deposited atoms become incorporated into the growing thin film. Each atomic layer is formed by the nucleation of many isolated monolayer islands, which then grow in area, merge with nearby islands, and ultimately fill in to complete the layer. Because the deposition flux is continually raining down on the entire surface, including the tops of the islands, a new monolayer can start growing before the previous layer is completely filled. Thus islands can form on top of islands in a "wedding cake" fashion, and the surface morphology during growth can become quite complicated.

The island dynamics model is a continuum model designed to capture the longer-length-scale features that are likely to be important for the analysis and control of monolayer thin film growth. It is also intended to model the physics relevant to studying basic issues of surface morphology, such as the effects of noise on growth, the long time evolution of islands, and the scaling relationships between surface features (mean island area, step edge length, etc.) in various growth regimes (precoalescence, coalescence). Refer to the classic work of [14] for useful background on the modeling of the growth of material surfaces. Our present discussion of the island dynamics model is an abridged version of what was discussed in [54]. We shall present this new model in some detail because, although it has many of the features of the Stefan problem, it also requires some new level set technology. This includes a wedding cake formulation involving several level sets of the same function, nucleation of new islands, and nontrivial numerical treatment of the interface to obtain rapid convergence of implicit time marching schemes.

In the island dynamics model, we treat each of the islands present as having a unit height, but a continuous (step edge) boundary on the surface. This represents the idea that the films are atomic monolayers, so that height is discrete, but they cover relatively large regions on the substrate, so x and y are continuum dimensions. The adatoms are modeled by a continuous adatom density function on the surface. This represents the idea that they are very mobile, and so they effectively occupy a given site for some fraction of the time, with statistical continuity, rather than discretely.

Thus, the domain for the model is the x–y region originally defined by the substrate, and the fundamental dynamical variables for this model are

- The island boundary curves $\Gamma_i(t),\ i = 1, 2, \ldots, N$
- The adatom density on the surface $\rho(x, y, t)$.

The adatom density ρ obeys a surface diffusive transport equation, with a source term for the deposition flux

$$\frac{\partial \rho}{\partial t} = \nabla \cdot (D \nabla \rho) + F,$$

where $F = F(x, y, t)$ is specified. During most phases of the growth, it is simply a constant. This equation may also include additional small loss terms reflecting adatoms lost to the nucleation of new islands, or lost to de-absorption off the surface. This equation must be supplemented with boundary conditions at the island boundaries. In the simplest model of irreversible aggregation, the binding of adatoms to step edges leaves the adatom population totally depleted near island boundaries, and the boundary condition is

$$\rho|_{\Gamma} = 0.$$

More generally, the effects of adatom detachment from boundaries, as well as the energy barriers present at the boundary, lead to boundary conditions of the form

$$\left[A\rho + B\frac{\partial \rho}{\partial n} \right] = C,$$

where C is given and $[\cdot]$ denotes the local jump across the boundary. In particular, note that ρ itself can have a jump across the boundary, even though it satisfies a diffusive transport equation. This simply reflects the fact that the adatoms on top of the island are much more likely to incorporate into the step edge than to hop across it and mix with the adatoms on the lower terrace, and vice versa.

The island boundaries Γ_i move with velocities $\mathbf{v} = v_{\mathrm{N}}\mathbf{N}$, where the normal velocity v_{N} reflects the island growth. This is determined simply by local conservation of atoms: the total flux of atoms to the boundary from both sides times the effective area per atom, a^2 must equal the local rate of growth of the boundary, v_{N},

$$v_{\mathrm{N}} = -a^2[\mathbf{q} \cdot \mathbf{N}]$$

(this assumes there is no particle transport along the boundary; more generally, there is a contribution from this as well), where $\mathbf{q}$ is the surface flux of adatoms to the island boundary and **N** is the local outward normal. In general, the net atom flux **q** can be expressed in terms of the diffusive transport, as well as attachment and detachment probabilities, all of which can be directly related to the parameters of kinetic Monte Carlo models. In the special case of irreversible aggregation, **q** is simply the surface diffusive flux of adatoms

$$\mathbf{q} = -D\nabla\rho.$$

To complete the model we include a mechanism for the nucleation of new islands. If islands nucleate by random binary collisions between adatoms (and if the critical cluster size is one), we expect the probability that an island is nucleated at a time t, at a site (x, y), scales as

$$P[dx, dy, dt] = \epsilon\rho(x, y, t)^2\, dt\, dx\, dy.$$

The model describes nucleation as a site-by-site, timestep-by-timestep random process. A simplifying alternative is to assume the nucleation occurs at the continuous rate obtained by averaging together the probabilistic rates at each site. In this case, if we let $n(t)$ denote the total number of islands nucleated prior to time t, we have the deterministic equation

$$\frac{dn}{dt} = \langle\epsilon\rho^2\rangle,$$

where $\langle\cdot\rangle$ denotes the spatial average. In this formulation, at each time when $n(t)$ reaches a new integer value, we nucleate a new island in space. This is carried out by placing it randomly on the surface with a probability weighted by ρ^2, so that the effect of random binary collisions is retained.

This basic model also has natural extensions to handle more complex thin film models. For example, additional continuum equations can be added to model the dynamics of the density of kink sites on the island boundaries, which is a microstructural property that significantly influences the local adatom attachment rates (see [15]). Also, we can couple this model to equations for the elastic stress that results from the “lattice mismatch” between the size of the atoms in the growing layers and the size of the atoms in the substrate.

Conversely, the above model has a particular interesting extreme simplification. We can go to the limit where the adatoms are so mobile on the surface ($D \to \infty$) that the adatom density is spatially uniform, $\rho(x, y, t) = \rho(t)$. In this case, the loss of adatoms due to the absorbing boundaries is assumed to take on a limiting form proportional to the adatom density and the total length L of all the island boundaries, which can be written as a simple sink term

$$\frac{d\rho}{dt} = F - \lambda L\rho.$$

(This equation can be derived systematically from the conservation law for the total number of adatoms, $\int\rho$, that follows from the adatom diffusion equation. The above loss term is just a simplified model for the net loss of adatoms to the island boundaries.) Further, it is assumed the velocity takes on a given normal-dependent limiting form, $v_{\mathrm{N}} = v_{\mathrm{N}}(\mathbf{N})$ (which implies the growing islands will rapidly assume the associated “Wulff shape” for this function $v_{\mathrm{N}}(\mathbf{N})$ (as in [62])). We have used this “uniform density” model to prototype the numerical methods and to develop an understanding of how the island dynamics models are related to the continuum "rate equation" models that describe island size distribution evolution while using no information at all about the spatial interactions of the islands.

Much of the above model is formally a Stefan problem and many of the level set techniques required for this were developed in [24] and can similarly be applied here. In addition, the internal boundary condition discretization of the adatom diffusion equation can be implemented using the symmetric matrix version of the discretization proposed by Fedkiw [31] and discussed above.

5. A VARIATIONAL APPROACH WITH APPLICATIONS TO MULTIPHASE MOTION

In many situations, e.g., crystal growth, a material is composed of three or more phases. The interfaces between the phases move according to some law. If the material is a metal and its grain orientation is different in each region, then an isotropic surface energy means that the velocity is the mean curvature of the interface. Or the velocities of the interfaces may depend on the pair of phases in contact, e.g., a different constant velocity on each interface.

Several fixed grid approaches to this problem have been used. Merriman, Bence, and Osher [53] have extended the level set method to compute the motion of multiple junctions. Also in that paper, and in [51, 52], a simple method based on the diffusion of characteristic functions of each set Ω_i, followed by a certain reassignment step, was shown to be appropriate for the motion of multiple junctions in which the bulk energies are zero (and hence, the constants $e_i = 0, i = 1, \ldots, n$) and the $f_{i,j}$ are all equal to the same positive constant, i.e., pure mean curvature flow. See Eq. (21) below.

Another method using an "influence matrix" was designed in [75]. However, as cautioned by the author, the method is expensive and complex.

More general motion involving somewhat arbitrary functions of curvature, perhaps different for each interface, was proposed in [53] as well. This was implemented basically by decoupling the motions and then using a reassignment step. Again each region has its own private level set function. This function moves each level set with a normal velocity depending on the proximity to the nearest interface; thus vacuum and overlapping regions generally develop. Then a simple reassignment step is used, removing all the spurious regions. For details see [53]. In that paper there was no restriction to gradient flows. However, the general method in [53] lacks (so far) a clean theoretical basis to guide the design of numerical algorithms. These difficulties were rectified by the following method.

In [88] we developed the variational level set approach inspired by [68]. Given a disjoint family Ω_i of regions in R^2 with the common boundary between Ω_i and Ω_j denoted by $\Gamma_{i,j}$, we associate to this geometry an energy function of the form

$$\begin{aligned} E &= E_1 + E_2 \\ E_1 &= \sum_{1 \le i \le j \le n} f_{i,j} \text{ length } (\Gamma_{i,j}) \\ E_2 &= \sum_{1 \le i \le n} e_i \text{area}(\Omega_i), \end{aligned} \tag{21}$$

where E_1 is the energy of the interface (surface tension), E_2 is bulk energy, and n is the number of phases. The gradient flow induces motion such that the normal velocity of each interface is defined in (22). At triple points (which can be seen geometrically by the triangle inequality to be the only stable junctions if all the $f_{i,j} > 0$), the angles are determined by (23) throughout the motion.

$$\text{Normal velocity of } \Gamma_{i,j} = (v_{\mathrm{N}})_{i,j} = f_{i,j}\kappa_{i,j} + (e_i - e_j) \text{ and} \tag{22}$$

$$\frac{\sin\theta_1}{f_{2,3}} = \frac{\sin\theta_2}{f_{3,1}} = \frac{\sin\theta_3}{f_{1,2}}. \tag{23}$$

This could be rewritten as

$$\begin{aligned} E &= E_1 + E_2 \\ E_1 &= \sum_{i=1}^{n} \gamma_i \int\!\!\int \delta(\varphi_i(x,y,t))|\nabla\varphi_i(x,y,t)|\,dx\,dy \\ E_2 &= \sum_{i=1}^{n} e_i \int\!\!\int H(\varphi_i(x,y,t))\,dx\,dy, \end{aligned} \tag{24}$$

where

$$f_{i,j} = \gamma_i + \gamma_j, \qquad 1 \le i < j \le n.$$

In the (most interesting) case when $n = 3$ we can solve uniquely for the γ_i.

Now our problem becomes the following:

Minimize E subject to the constraint that

$$\sum_{i=1}^{n} H(\varphi_i(x,y)) - 1 \equiv 0. \tag{25}$$

This infinite set of constraints prevents the development of overlapping regions and/or vacuum. It requires that the level curves $\{(x,y) \mid \varphi_i(x,y,t) = 0\}$ match perfectly.

The implementation of (24) with the infinite set of constraints (25) is computationally demanding. Instead we try to replace the constraint (25) by a single constraint

$$\int\!\!\int \frac{\left(\sum H(\varphi_i(x,y,t)) - 1\right)^2}{2}\,dx\,dy = \epsilon, \tag{26}$$

where $\epsilon > 0$ is as small as we can manage numerically.

The gradient projection method leads us to an interesting coupled system which involves motion of level contours of each φ with normal velocity $a + b\kappa$ together with a term enforcing the no-overlap/vacuum constraint. We find that $\epsilon \approx \Delta x$ in real calculations. See [88] for details.

We have used this technique to reproduce the general behavior of complicated bubble and droplet motions in two and three dimensions [90]. The problems included soap bubble colliding and merging drops falling or remaining attached to a generally irregular ceiling (see Fig. 15), liquid penetrating through an asymmetric funnel opening (see Fig. 16), and mercury sitting on the floor (see Fig. 17).

This variational approach has also been found to have many applications in computer vision—this will be discussed in the next section.

6. APPLICATIONS TO COMPUTER VISION AND IMAGE PROCESSING

The use of PDEs and level set motion in image analysis and computer vision has exploded in recent years. Good references include [18, 58].

One basic idea is to view an image as $u_0(x, y)$, a function defined on a square, and obtain a (usually second-order) flow equation of the form

$$\begin{aligned} u_t &= F(u, Du, D^2u, x, t) \\ u(x, y, 0) &= u_0(x, y), \end{aligned} \tag{27}$$

which, for positive t, processes the image.

For example, if one solves the heat equation with $F(u, Du, D^2u, x, t) = \Delta u$, then $u(x, y, t)$ is the same as convolution of u_0 with a Gaussian of variance t.

L. I. Rudin, in his Ph.D. thesis [70], made the point that images are largely characterized by singularities, edges, boundaries, etc., and thus nonlinearity, especially ideas related to shock propagation, should play a role. This led to the very successful total variation–based image restoration algorithms of [71, 72]. Briefly, if we are presented with a noisy blurred image

$$u_0 = j * u + n, \tag{28}$$

where j is a given convolution kernel, and the mean and variance of the noise are given, we wish to obtain the "best" restored image. This leads us (see [71, 72]) to the evolution equation

$$u_t = \nabla \cdot \frac{\nabla u}{|\nabla u|} - \lambda j * (j * u - u_0) \tag{29}$$

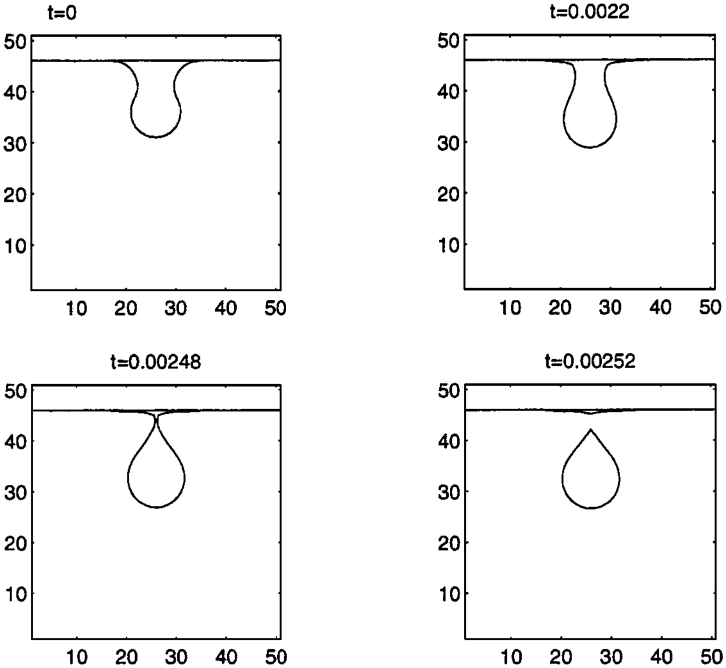


FIG. 15. Three-dimensional drop falling from ceiling. Reprinted from [90].

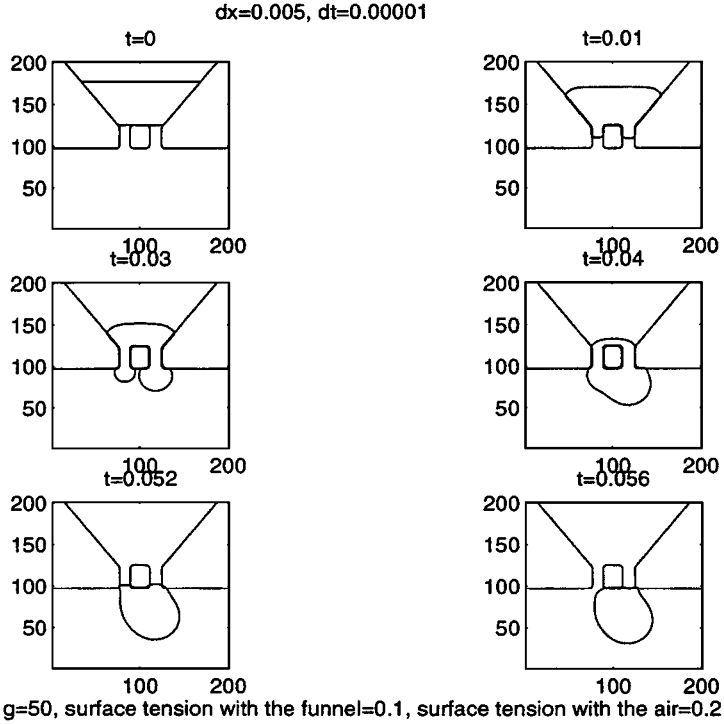


FIG. 16. Liquid falling through funnel opening. Reprinted from [90].

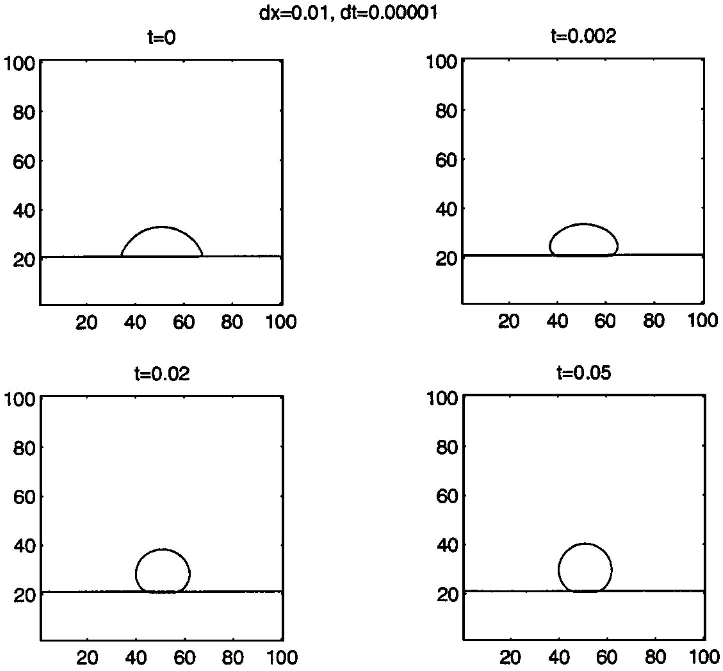


FIG. 17. Mercury droplet responding to surface tension. Reprinted from [90].

to be solved for $t > 0$, where $u(x, y, 0)$ is given, and $\lambda(t) > 0$ is obtained as a Lagrange multiplier, or is set to be a fixed constant. If $j * u = u$, this becomes a pure denoising problem. The (very interesting) geometric interpretation of this procedure is that each level contour of u is moved normal to itself with velocity equal to its curvature, divided by the norm of the gradient of u, then “pulled back” in an attempt to deconvolve (28). The results are state-of-the-art for many problems. Noisy regions can be thought of as corresponding to contours having very high curvature, while edges have finite curvature and infinite gradients.

Here the motion of level sets is just used to interpret the dynamics. In [4], it was shown that reasonable axioms of image processing lead to the remarkable fact that motion of level contours by a function of curvature is fundamental to the subject. The artificial time t is actually the scale parameter [4].

We would like to describe a few new applications of this set of ideas. In [10], we have considered the problem of processing of images defined on manifolds. The technique actually can be used to solve a wide class of elliptic equations on manifolds, without triangulation, using only a local Cartesian grid, for very general situations.

Given a manifold in R^3, defined by $\psi(x, y, z) = 0$, we can define the projection matrix

$$P_{\nabla\psi} = I - \frac{\nabla\psi}{|\nabla\psi|} \otimes \frac{\nabla\psi}{|\nabla\psi|}. \tag{30}$$

If u is an image defined on $\psi = 0$ we can use our level set calculus to extend it constant normal to the manifold, in some neighborhood of the manifold.

If u_0 is the original noisy image, the energy to be minimized is

$$E(u) = \int_{R^3} |P_{\nabla\psi} \nabla u| \delta(\psi) |\nabla\psi| \, d\mathbf{x} + \frac{\lambda}{2} \int (u - u_0)^2 \delta(\psi) |\nabla\psi| \, d\mathbf{x}.$$

Using the gradient descent algorithm, i.e., following the general procedure of [72, 88], leads us to

$$u_{\mathrm{t}} = \frac{1}{|\nabla\psi|} \nabla \cdot \left(\frac{P_{\nabla\psi} \nabla u}{|P_{\nabla\psi} \nabla u|} |\nabla\psi| \right) - \lambda(u - u_0).$$

This corresponds to total variation denoising. This is done using the local level set method [66] which allows great flexibility in geometry, while always using a Cartesian grid. See [10] for denoising and deblurring results.

The technique is quite general—both variational problems and PDEs defined on manifolds can be solved in a reasonably straightforward fashion, without restrictions on the manifold and without complicated triangulation, just by using a fixed Cartesian grid.

Another basic image processing task is to detect objects hidden in an image u_0. A popular technique is called active contours or snakes, in which one evolves a curve, subject to constraints, until the curve surrounds the image.

The level set method was first used in [16] as a very convenient tool to follow the motion of active contours in order to surround hidden objects. This was an important step since topological changes could easily be handled, a variational approach could easily be used [17], and stable, easy-to-program algorithms resulted. The curve is moved with a velocity which vanishes when the object is surrounded. Thus edge detectors are traditionally used to stop the evolving curve. For example, one might use

$$g(|\nabla u_0|) = (1/(1 + |\nabla j_\sigma * u_0|))^2,$$

where j_σ is a Gaussian of variance σ.

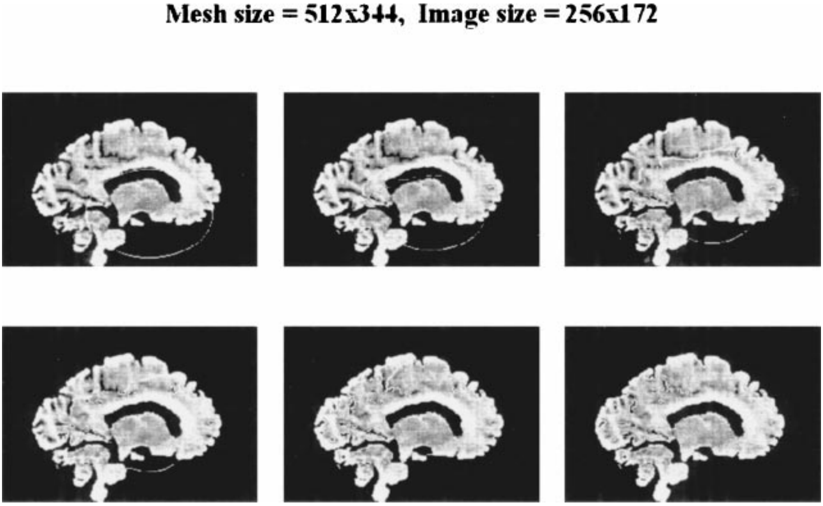


FIG. 18. Active contour segmentation of an MRI brain image from its background. Reprinted from [19].

In [20] the authors developed a model which was not based on edges, using a scale parameter, based on a simplification of the Mumford–Shah [56] energy-based segmentation. The implementation is done through the variational level set approach [88] and the results are remarkable. The method has a denoising capability as well as the ability to perform a multiscale segmentation. See [20, 21] for details. Here we just present the evolution equation for the level set function φ,

$$\varphi_{\mathrm{t}} = |\nabla\varphi| \left[\mu \nabla \cdot \frac{\nabla\varphi}{|\nabla\varphi|} - \nu - \lambda(u_0 - c_1)^2 + \lambda(u_0 - c_2)^2 \right],$$

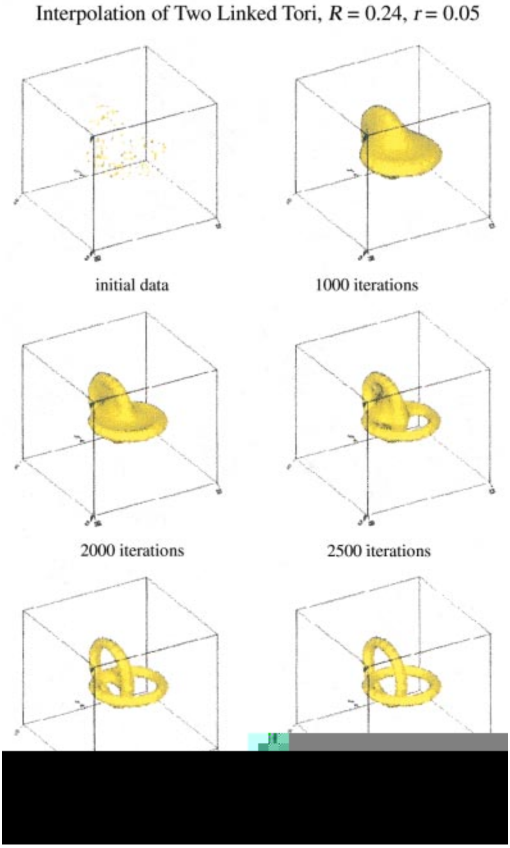
for parameters $\mu, \nu, \lambda \geq 0$, where c_1 and c_2 are the averages of u_0 over the regions for which $\varphi \geq 0$ and $\varphi \leq 0$, respectively. ν corresponds to the bulk energy of the area for which $\varphi \geq 0$; μ corresponds to the surface tension of the interface; and λ is the penalty for the L^2 error between u_0 and its mean over each region. Figure 18 shows an active contour segmenting a MRI brain image from its background.

A somewhat related problem as discussed in [89] is the following. Given a collection of unorganized points, and/or curves, and/or surface patches, find a surface which can be regarded as its shape. This is a fundamental visualization problem which arises in computer graphics, visualization, and simulation. No assumptions about the ordering, connectivity, or topology of the data sets or of the true shape are given. The input is the general distance to the data set which is given on a (usually logically rectangular) grid. Additionally, we may input the values of the normal to the surface at the same or different data points.

The key idea is to find a function φ whose zero level set is the interpolating surface; φ changes sign as one goes from inside to outside the surface. The output is the discrete values of φ, which can be reinitialized to be signed distance to this surface.

We set up a variational problem, which basically minimizes the integral over the unknown surface, of the pth power of distance to the data set. We may include information about the normals in analogous fashion.

Gradient descent (as in the image restoration and active contour problems) gives us a weighted motion by curvature plus convection algorithm. The results are very promising, as shown in Fig. 19. For more details, see [89].



initial data

1000 iterations

2000 iterations

2500 iterations

[redacted]

FIG. 19. Interpolation of two linked tori. Reprinted from [89].

7. CONCLUSION

The idea of using a level set to represent an interface is a very old one. The level set method itself has antecedents, for example, in the G-equation approach of Markstein [50]. What is new is the level set method technology, theoretical justification through viscosity solutions, and the enormous number of wide-ranging applications that are now available, with new applications developing quite frequently.

REFERENCES

1. D. Adalsteinsson and J. A. Sethian, The fast construction of extension velocities in level set methods, *J. Comput. Phys.* **148**, 2 (1999).
2. D. Adalsteinsson and J. A. Sethian, A fast level set method for propagating interfaces, *J. Comput. Phys.* **118**, 269 (1995).
3. D. Adalsteinsson and J. A. Sethian, A level set approach to a unified model for etching, deposition, and lithography. II. Three-dimensional simulations, *J. Comput. Phys.* **122**, 348 (1995).
4. L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel, Axioms and fundamental equations of image processing, *Arch. Ration. Mech. Anal.* **123**, 199 (1993).
5. L. Ambrosio and H. M. Soner, Level set approach to mean curvature flow in arbitrary codimension, *J. Differential Geom.* **43**(4), 693 (1996).

6. M. Bardi and L. C. Evans, On Hopf's formulas for solutions of Hamilton–Jacobi equations, *Nonlinear Anal. TMA* **8**, 1373 (1984).
7. M. Bardi and S. Osher, The nonconvex multidimensional Riemann problem for Hamilton–Jacobi equations, *SIAM J. Anal.* **22**, 344 (1991).
8. G. Barles, *Solutions de Viscosité des Equations de Hamilton–Jacobi* (Springer-Verlag, Berlin, 1996).
9. G. Bellettini, M. Novaga, and M. Paolini, An example of three dimensional fattening for linked space curves evolving by curvature, *Comm. Partial Differential Equations*, in press.
10. M. Bertalmio, L. T. Cheng, S. Osher, and G. Sapiro, *Variational Problems and Partial Differential Equations on Implicit Surfaces: The Framework and Examples in Image Processing and Pattern Formation*, CAM Report 00-23 (University of California, Los Angeles), submitted for publication.
11. M. Boué and P. Dupuis, Markov chain approximations for deterministic control problems with affine dynamics and quadratic cost in the control, *SIAM J. Numer. Anal.* **36**(3), 667 (1999).
12. J. U. Brackbill, D. B. Kothe, and C. Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* **100**, 335 (1992).
13. P. Burchard, L.-T. Cheng, B. Merriman, and S. Osher, *Motion of Curves in Three Spatial Dimensions Using a Level Set Approach*, CAM Report 00-29 (University of California, Los Angeles), submitted for publication.
14. W. K. Burton, N. Cabrera, and F. C. Frank, The growth of crystals and the equilibrium structure of their surfaces, *Philos. Trans. R. Soc. London Ser. A*, 243 (1951).
15. R. E. Caflisch, M. Gyure, B. Merriman, S. Osher, C. Ratsch, D. Vvedensky, and J. Zinck, Island dynamics and the level set method for epitaxial growth, *Appl. Math. Lett.* **12**, 13 (1999).
16. V. Caselles, F. Catté, T. Coll, and F. Dibo, A geometric model for active contours in image processing, *Numer. Math.* **66**, 1 (1993).
17. V. Caselles, R. Kimmel, and G. Sapiro, Geodesic active contours, *Int. J. Comput. Vision* **22**, 61 (1997).
18. V. Caselles, J.-M. Morel, G. Sapiro, and A. Tannenbaum, Eds. *Special Issue on Partial Differential Equations and Geometry-Driven Diffusion in Image Processing and Analysis, IEEE Trans. Image Process.* **7**, 269 (1998).
19. T. Chan, R. Fedkiw, M. Kang, and L. Vese, Improvements in the Efficiency and Robustness of Active Contour Algorithms, in preparation.
20. T. Chan and L. Vese, *Active Contours without Edges*, CAM Report 98-53 (University of California, Los Angeles, 1998).
21. T. Chan and L. Vese, An active contour model without edges, in Lecture Notes in Computer Science, edited by M. Neilsen, P. Johansen O. F. Olsen, and J. Weickert (Springer-Verlag, Berlin/New York, 1999), Vol. 1687, p. 141.
22. Y. C. Chang, T. Y. Hou, B. Merriman, and S. Osher, A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, *J. Comput. Phys.* **124**, 449 (1996).
23. Y. G. Chen, Y. Giga, and S. Goto, Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations, *J. Differential Geom.* **33**, 749 (1991).
24. S. Chen, B. Merriman, S. Osher, and P. Smereka, A simple level set method for solving Stefan problems, *J. Comput. Phys.* **135**, 8 (1997).
25. L. T. Cheng, R. P. Fedkiw, F. Gibou, and M. Kang, *A Symmetric Method for Implicit Time Discretization of the Stefan Problem*, CAM Report 00–37 (University of California, Los Angeles, 2000).
26. P. Colella, A. Majda, and V. Roytburd, Theoretical and numerical structure for reacting shock waves, *SIAM J. Sci. Stat. Comput.* **7**(4), 1059 (1986).
27. M. G. Crandall, H. Ishii, and P.-L. Lions, User's guide to viscosity solutions of second order partial differential equations, *Am. Math. Soc. Bull.* **27**, 1 (1992).
28. E. DeGiorgi, Barriers, boundaries, motion of manifolds, lectures presented in Pavia, Italy, 1994.
29. Y. C. Evans, H. M. Soner, and P. E. Souganidis, Phase transitions and generalized motion by mean curvature, *Commun. Pure Appl. Math.* **65**, 1097 (1992).
30. Y. C. Evans and J. Spruck, Motion of level sets by mean curvature, I, *J. Differential Geom.* **33**, 635 (1991).
31. R. Fedkiw, *A symmetric spatial discretization for implicit time discretization of Stefan type problems*, unpublished.

32. R. Fedkiw, T. Aslam, B. Merriman, and S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), *J. Comput. Phys.* **152**(2), 457 (1999).

33. R. Fedkiw, T. Aslam, and S. Xu, The ghost fluid method for deflagration and detonation discountinuities, *J. Comput. Phys.* **154**(2), 393 (1999).

34. R. Fedkiw and X.-D. Liu, The ghost fluid method for viscous flows, in *Progress in Numerical Solutions of Partial Differential Equations*, edited by M. Hafez (Arcachon, 1998).

35. R. Gross, Zur theorie des washstrums und losungsforganges kristalliner materic, *Abh. Math.-Phys. Kl. Saechs. Akad. Wiss.* **35**, 137 (1918).

36. M. Gyure, C. Ratsch, B. Merriman, R. E. Caflisch, S. Osher, J. Zinck, and D. Vvedensky, Level set methods for the simulation of epitaxial phenomena, *Phys. Rev. E* **59**, R6927 (1998).

37. E. Harabetian and S. Osher, Regularization of ill-posed problems via the level set approach, *SIAM J. Appl. Math.* **58**, 1689 (1998).

38. E. Harabetian, S. Osher, and C.-W. Shu, An Eulerian approach for vortex motion using a level set approach, *J. Comput. Phys.* **127**, 15 (1996).

39. B. T. Helenbrook, L. Martinelli, and C. K. Law, A numerical method for solving incompressible flow problems with a surface of discontinuity, *J. Comput. Phys.* **148**, 366 (1999).

40. J. Helmsen, E. Puckett, P. Colella, and M. Dorr, Two new methods for simulating photolithography development in 3D, *Proc. SPIE* **2726**, 253 (1996).

41. T. Hou, Numerical solutions to free boundary problems, *Acta Numer.* **4**, 335 (1995).

42. T. Hou, Z. Li, S. Osher, and H.-K. Zhao, A hybrid method for moving interface problems with application to the Hele–Shaw flow, *J. Comput. Phys.* **134**, 236 (1997).

43. G.-S. Jiang and D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, *SIAM J. Sci. Comput.* **21**, 2126 (2000).

44. M. Kang, R. Fedkiw, and X.-D. Liu, *A Boundary Condition Capturing Method for Multiphase Incompressible Flow*, CAM Report 99-21 (University of California, Los Angeles), submitted for publication.

45. S. Karni, Hybrid multifluid algorithms, *SIAM J. Sci. Comput.* **17**(5), 1019 (1996).

46. S. Karni, Multicomponent flow calculations by a consistent primitive algorithm, *J. Comput. Phys.* **112**, 31 (1994).

47. Y.-T. Kim, N. Goldenfeld, and J. Dantzig, Computation of dendritic microstructures using a level set method, *Phys. Rev. E* **62**, (2000).

48. R. Kobayashi, Modeling and numerical simulations of dendritic crystal growth, *Physica D* **63**, 410 (1993).

49. X.-D. Liu, R. P. Fedkiw, and M. Kang, A boundary condition capturing method for Poisson's equation on irregular domains, *J. Comput. Phys.* **160**, 151 (2000).

50. G. H. Markstein, *Nonsteady Flame Propagation* (Pergamon, Oxford, 1964).

51. P. Mascarenhas, *Diffusion Generated Motion by Mean Curvature*, CAM Report 92-33 (University of California, Los Angeles, 1992).

52. B. Merriman, J. Bence, and S. Osher, Diffusion generated motion by mean curvature, in *AMS Select Lectures in Mathematics: The Computational Crystal Grower's Workshop*, edited by J. Taylor (Am. Math. Soc., Providence, 1993), p. 73.

53. B. Merriman, J. Bence, and S. Osher, Motion of multiple junctions: A level set approach, *J. Comput. Phys.* **112**, 334 (1994).

54. B. Merriman, R. Cafiisch, and S. Osher, Level set methods with an application to modelling the growth of thin flims, in *Free Boundary Value Problems, Theory and Applications*, edited by I. Athanasopoulos, G. Makrikis, and J. F. Rodriguez (CRC Press, Boca Raton, FL, 1999), p. 51.

55. W. Mulder, S. Osher, and J. A. Sethian, Computing interface motion in compressible gas dynamics, *J. Comput. Phys.* **100**, 209 (1992).

56. D. Mumford and J. Shah, Optimal approximation by piecewise smooth functions and associated variational problems, *Commun. Pure Appl. Math.* **42**, 577 (1989).

57. D. Nguyen, R. P. Fedkiw, and M. Kang, *A Boundary Condition Capturing Method for Incompressible Flame Discontinuities*, CAM Report 00-19 (University of California, Los Angeles), submitted for publication.

58. M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, Eds., *Scale Space Theories in Computer Vision*, Lecture Notes in Computer Science (Springer-Verlag, Berlin, 1999), Vol. 1682.

59. R. H. Nochetto, M. Paolini, and C. Verdi, An adaptive finite element method for two phase Stefan problems in two space dimensions. II. Implementation and numerical experiments, *SIAM J. Sci. Comput.* **12**, 1207 (1991).

60. W. F. Noh and P. R. Woodward, *SLIC (Simple Line Interface Construction)*, Lecture Notes in Physics, edited by A. van de Vooren and P. J. Zandbergen (Springer–Verlag, Berlin, 1976), Vol. 59, p. 330.

61. S. Osher and J. Helmsen, A generalized fast algorithm with applications to ion etching, in progress.

62. S. Osher and B. Merriman, The Wulff shape as the asymptotic limit of a growing crystalline interface, *Asian J. Math.* **1**(3), 560 (1997).

63. S. Osher, A level set formulation for the solution of the Dirichlet problem for Hamilton–Jacobi equations, *SIAM J. Anal.* **24**, 1145 (1993).

64. S. Osher and J. A. Sethian, Fronts propagating with curvature dependent speed: Algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* **79**, 12 (1988).

65. S. Osher and C. W. Shu, High order essentially non-oscillatory schemes for Hamilton–Jacobi equations, *SIAM J. Numer. Anal.* **28**(4), 907 (1991).

66. D. Peng, B. Merriman, S. Osher, H.-K. Zhao, and M. Kang, A PDE-based fast local level set method, *J. Comput. Phys.* **155**, 410 (1999).

67. D. Peng, S. Osher, B. Merriman, and H.-K. Zhao, The geometry of Wulff crystal shapes and its relations with Riemann problems, in Contemporary Mathematics, edited by G.-Q. Chen and E. DeBenedetto (Am. Math. Soc., Providence, RI, 1999), Vol. 238, p. 251.

68. F. Reitich and H. M. Soner, Three phase boundary motions under constant velocities. I. The vanishing surface tension limit, *Proc. R. Soc. Edinburgh Ser. A* **126**, 837 (1996).

69. E. Rouy and A. Tourin, A viscosity solutions approach to shape-from-shading, *SIAM J. Numer. Anal.* **29**(3), 867 (1992).

70. L. I. Rudin, *Images, Numerical Analysis of Singularities, and Shock Filters*, Ph.D. thesis, Computer Science Dept., Caltech, #5250:TR:87, Pasadena, CA, 1987.

71. L. I. Rudin and S. Osher, Total variation based restoration with free local constraints, in *Proceedings of the ICIP, IEEE International Conference on Image Processing, Austin, TX, 1994*, p. 31.

72. L. I. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D* **60**, 259 (1992).

73. S. Ruuth, B. Merriman, and S. Osher, *A Fixed Grid Method for Capturing the Motion of Self-Interesecting Interfaces and Related PDEs*, *J. Comput. Phys.* **163**, 21 (2000).

74. S. Ruuth, B. Merriman, J. Xin, and S. Osher, *Diffusion-Generated Motion for Mean Curvature of Filaments*, CAM Report 98-47 (University of California, Los Angeles, 1998), submitted for publication.

75. J. A. Sethian, Algorithm for tracking interfaces in CFD and materials science, *Annu. Rev. Comput. Fluid Mech.* (1995).

76. J. A. Sethian, Fast marching level set methods for three dimensional photolithography development, *Proc. SPIE* **2726**, 261 (1996).

77. J. A. Sethian, Fast marching methods, *SIAM Rev.* **41**, 199 (1999).

78. J. A. Sethian and J. Strain, Crystal growth and dendritic solidification, *J. Comput. Phys.* **98**, 231 (1992).

79. K. W. Schwarz, Simulations of dislocations on the mesoscopic scale. I. Methods and examples, *J. Appl. Phys.* **85**, 108 (1999).

80. K. W. Schwarz, Simulations of dislocations on the mesoscopic scale. II. Application to strained-layer relaxation, *J. Appl. Phys.* **85**, 120 (1999).

81. P. Soravia, Generalized motion of a front propagating along its normal direction: A differential games approach, *Nonlinear Anal. TMA* **22**, 1247 (1994).

82. J. Steinhoff, M. Fan, and L. Wang, A new Eulerian method for the computation of propagating short acoustic and electromagnetic pulses, *J. Comput. Phys.* **157**, 683 (2000).

83. M. Sussman, E. Fatemi, P. Smereka, and S. Osher, An improved level set method for incompressible two-phase flow, *Comput. and Fluids* **27**, 663 (1998).

84. M. Sussman, P. Smereka, and S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.* **114**, 146 (1994).

85. R. Tsai, H.-K. Zhao, and S. Osher, Fast sweeping algorithms for a class of Hamilton–Jacobi equations, in preparation.

86. J. N. Tsitsiklis, Efficient algorithms for globally optimal trajectories, *IEEE Trans. Autom. Control* **40**(9), 1528 (1995).

87. S. O. Unverdi and G. Tryggvason, A. front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.* **100**, 25 (1992).

88. H.-K. Zhao, T. Chan, B. Merriman, and S. Osher, A variational level set approach to multiphase motion, *J. Comput. Phys.* **127**, 179 (1996).

89. H.-K. Zhao, B. Merriman, S. Osher, and M. Kang, *Implicit Nonparametric Shape Reconstruction from Unorganized Points Using a Variational Level Set Method*, *Comput. Vision and Image Understanding* **80**, 295 (2000).

90. H.-K. Zhao, B. Merriman, S. Osher, and L. Wang, Capturing the behavior of bubbles and drops using the variational level set approach, *J. Comput. Phys.* **143**, 495 (1998).