Chapter 1

Computational Methods for Advancing Interfaces

J.A. Sethian
Dept. of Mathematics
Univ. of California, Berkeley
Berkeley, California 94720
sethian@math.berkeley.edu*

Abstract

A large number of computational problems and physical phenomena involve the motion of interfaces separating two or more regions. These can include problems in such areas as fluid mechanics, combustion, materials science, meteorology, and computer vision. In these problems, challenging issues often involve interfaces that change topology, form sharp corners and singularities, depend on delicate geometric quantities such as curvature and normal direction, and involve subtle feedback between the physics and chemistry off the interface and the position/motion of the front itself. In this paper, we will explain some of the issues involved in tracking interfaces, focus on a particular set of numerical techniques that arise from an implicit representation of the interface, and provide an overview of some of the applications that are possible with this view.

*Dedicated to Professor Steve Davis in thanks for having so generously welcomed me into the warm community of those celebrating his birthday and his work.

1.1 Characterizations of Moving Interfaces

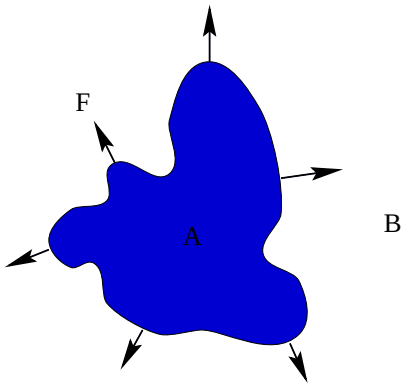

Fig. 1.1 Evolving Interface

Suppose we are given an interface separating two regions, and speed F in a direction normal to the interface (see Figure 1.1). Typically, this speed F can depend on the position of the front, the local geometry, and the solution to associated partial differential equations on either side of the interface, in addition to given jump conditions across the boundaries. Nonetheless, let us assume for now that the speed F is given.

1.1.1 *Mathematical Formulations*

There are at least three ways to characterize a moving interface, and none of them are new. Interestingly, each comes from its own branch of mathematics. For simplicity, we discuss the issues in two space dimensions, that is, a one-dimensional interface which is a simple closed curve $\Gamma(t)$ moving in two dimensions. Assume that a given velocity field $\vec{u} = (u, v)$ transports the interface. All three constructions carry over to three dimensions.

- **The Geometric View:** Suppose one parameterizes the interface, that is, $\Gamma(t) = x(s,t), y(s,t)$. Then one can write (see [23]) the equations of motion in terms of individual components (see Figure

1.2 $\vec{x} = (x, y)$ as

$$x_t = \quad u * \left(\frac{y_s}{(x_s^2 + y_s^2)^{1/2}} \right), \; y_t = -v * \left(\frac{x_s}{(x_s^2 + y_s^2)^{1/2}} \right) \tag{1.1}$$

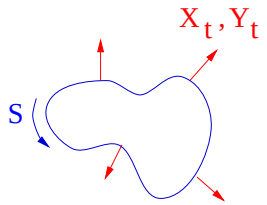

Fig. 1.2 Parameterized View

This is a differential geometry view; the underlying fixed coordinate system has been abandoned, and the motion is characterized by differentiating with respect to the parameterization variable s.

- **The Set Theoretic View**: Consider the characteristic function $\chi(x, y, t)$, where χ is one inside the interface Γ and zero otherwise (see Figure 1.3). Then one can write the motion of the characteristic function as

$$\chi_t = \vec{u} \cdot \nabla \chi \tag{1.2}$$

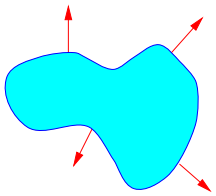

Fig. 1.3 Set-Theoretic View

In this view, all the points inside the set (that is, where the characteristic function is unity) are transported under the velocity field.

- **The Analysis View**: Consider the implicit function $\phi : R^2 \times [0, \infty) \rightarrow R$, defined so that the zero level set $\phi = 0$ corresponds to the evolving front $\Gamma(t)$ (see Figure 1.4).

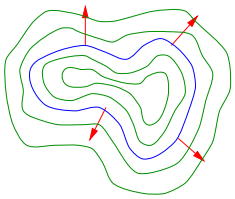

Fig. 1.4 The Implicit View

Then the equation for the evolution of this implicit function corresponding to the motion of the interface is given by

$$\phi_t + u \cdot \nabla \phi = 0 \tag{1.3}$$

1.1.2 *Discretizations*

Each of these views is perfectly reasonable, and each has spawned its own numerical methodology to discretize the equations of motion:

- Marker particle methods, also known as string methods and nodal methods, discretize the geometric view, and take a finite number of points to divide up the parameterization space S (see Figure 1.5).

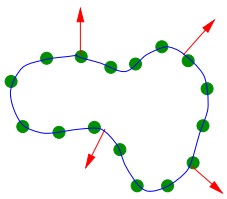

Fig. 1.5 Discretized parameterization into markers

- Volume-of-fluid methods, also known as cell methods and volume fraction methods, use a fixed underlying grid and discretize the

characteristic function, filling each cell with a number that reflects the amount of characteristic function contained in that cell (see Figure 1.6).

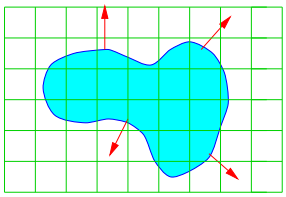

Fig. 1.6 Discretized characteristic function into cell fractions

- Level set methods, introduced by Osher and Sethian [14] approximate the partial differential equation for the time-dependent implicit function ϕ through a discretization of the evolution operators on a fixed grid (see Figure 1.7).

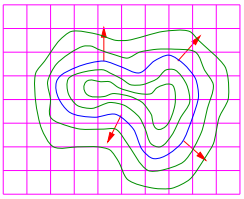

Fig. 1.7 Discretized implicit function onto grid

These discretizations contain keys to both the virtues and the drawbacks of the various approaches.

- The geometric/marker particle view keeps the definition of a front sharp. It requires special attention when marker particles collide; these can create corners and cusps, as well as changes in topology. These techniques often go by names such as contour surgery, reconnection algorithms, etc.; at their core, they reflect user-based decisions about the level of resolution. In addition, this discrete parameterized characterization of the interface can be intricate for two-dimensional surfaces moving in three dimensions.

- The characteristic/volume-of-fluid approach straightforwardly applies in multiple dimensions, and handles topological merger easily, since this results from Boolean operations on sets. It requires some method of differentiating the characteristic function χ; since by definition this object is discontinuous, one must devise an approximation to $\nabla\chi$ in order to perform the evolution update. This is typically done through algorithms which locally reconstruct the front from the volume or cell fractions, and then use this reconstruction to build the appropriate transport terms.
- The implicit/level set approach extends to multiple dimensions and handles topological changes easily. In addition, because the function ϕ is defined everywhere and smooth in many places, calculation of gradients in the transport term, as well as geometric quantities such as normal derivatives and curvature is straightforward. However, a numerical consistent and accurate way is required to produce the correct viscosity solution to the equations of interface propagation is required, as discussed below.

1.1.3 *Implicit Formulations of Interface Motion*

Our goal here is to discuss the implications and implementations of this implicit approach. We begin writing the implicit form for the equation of motion of a front evolving with speed F in its normal direction ([14]), namely

$$\phi_t + F|\nabla\phi| = 0 \tag{1.4}$$

In order to approximate this equation and solve for the evolving level set function ϕ, there are three central issues.

- First, an appropriate theory and strategy must be chosen in order to select the correct weak solution once the underlying smoothness is lost; this is provided by the work on the evolution of curves and surfaces and the link between hyperbolic conservation laws and propagation equations, see Sethian [17; 18; 19]; leading up to the introduction of level set methods by Osher and Sethian in [14].

- Second, the Osher-Sethian level set technique which discretizes the above requires an additional space dimension to carry the embedding, and hence is computationally inefficient for many problems. This is rectified through adaptive Narrow Band Method given by Adalsteinsson and Sethian in [1].
- Third, since both the level set function and the velocity are now defined away from the original interface, appropriate extensions of these values must be constructed. These extension velocities have been explicitly constructed for a variety of specific problems. One general technique for doing so for arbitrary physics and chemistry problems is given by Adalsteinsson and Sethian in [2] through the use of Fast Marching Methods to solve an associated equation which constructs these extensions.

1.1.4 *Interrelations Between Techniques*

It is important to state that each of the above techniques has evolved to the point where they provide practical, efficient, and accurate methodologies for computing a host of computational problems involving moving interfaces. Marker particles methods have been around for a very long time, and have been used in a collection of settings, including, for example, for example, Bunner and Tryggvason [4]. Volume-of-fluid techniques, starting with the initial work of Noh and Woodward [13], have been used to handle shock interactions and fluid interfaces (see, for example, Puckett [15]. Level set techniques have been applied to a large collection of problems; general reviews may be found in [22; 23]; and an introductory web page may be found at www.math.berkeley.edu/∼sethian/level_set.html.

Finally, we note that the strict delineations between various approaches is not meant to imply that the various techniques have not influenced each other. Modern level set methods often use a temporary marker representation of the front to help build the extension velocities; volume-of-fluid methods use differentiation ideas in level methods to help construct normal vectors and curvature values; and marker models often use an underlying fixed grid to help with topological changes. Good numerics is ultimately about getting things to work; the slavish and blind devotion to one approach above all others is usually a sign of unfamiliarity with the range of troubles and challenges presented by real applications.

1.2 Algorithms for Front Propagation: Level Set Methods and Fast Marching Methods

1.2.1 *The link with hyperbolic conservation laws*

In order to approximate the level set equation given by

$$\phi_t + F|\nabla\phi| = 0 \tag{1.5}$$

we must construct suitable approximations to the gradient term. As discussed in [17; 18], an evolving interface can develop corners as it evolves, and, once this happens, the correct weak solution must be constructed beyond the development of singularities in the curvature. In order to construct numerical schemes which build this correct solution, we briefly follow the argument given in [18] which links moving fronts to hyperbolic conservation laws.

Consider the initial front given by the graph of $f(x)$, with f and f' periodic on $[0, 1]$, and suppose that the propagating front remains a graph for all time. Let ψ be the height of the propagating function at time t, and thus $\psi(x, 0) = f(x)$. The tangent at (x, ψ) is $(1, \psi_x)$. The change in height V in a unit time is related to the speed F in the normal direction by

$$\frac{V}{F} = \frac{(1 + \psi_x^2)^{1/2}}{1}, \tag{1.6}$$

and thus the equation of motion becomes

$$\psi_t = F(1 + \psi_x^2)^{1/2}. \tag{1.7}$$

Use of the speed function $F(\kappa) = 1 - \epsilon\kappa$ and the formula $\kappa = -\psi_{xx}/(1 + \psi_x^2)^{3/2}$ yields

$$\psi_t \ - \ (1 + \psi_x^2)^{1/2} = \epsilon \frac{\psi_{xx}}{1 + \psi_x^2}. \tag{1.8}$$

This is a partial differential equation with a first order time and space derivative on the left side, and a second order term on the right. Differentiation of both sides of this equation yields an evolution equation for the slope $u = d\psi/dx$ of the propagating front, namely,

$$u_t + [-(1 + u^2)^{1/2}]_x = \epsilon \left[\frac{u_x}{1 + u^2} \right]_x . \tag{1.9}$$

Thus, as shown in [19], the derivative of the curvature-modified equation for the changing height ψ looks like some form of a viscous hyperbolic conservation law, with $G(u) = -(1 + u^2)^{1/2}$ for the propagating slope u. Hyperbolic conservation laws of this form have been studied in considerable detail and the strategy for picking the correct weak solution beyond the occurence of a singularity is equivalent to the one for propagating shocks in hyperbolic conservation laws (see [18; 19]).

1.2.2 *Link to Numerical Schemes for Hyperbolic Conservation Laws*

Given this connection, the next step in development of PDE-based interface advancement techniques was to in fact exploit the considerable numerical technology for hyperbolic conservation laws to tackle front propagation itself. In such problems, schemes are specifically designed to construct entropy-satisfying limiting solutions and maintain sharp discontinuities wherever possible; the goal is to keep fluid variables such as pressure from oscillating, and to make sure that discontinuities are not smeared out. This is equally important in the tracking of interfaces, in which one wants corners to remain sharp and to accurately track intricate development. Thus, the strategy laid out in [19] was to transfer this technology to front propagation problems, and led up to the level set method introduced in [14].

1.2.3 *Numerical Algorithms for Solving the Level Set Equation*

The above discussion focussed on curves which remain graphs. The numerical Osher-Sethian "level set method" recasts the front in one higher dimension, and uses the implicit analytic framework given above to tackle problems which do not remain graphs; in addition, that work developed multi-dimensional upwind schemes to approximate the relevant gradients. For the sake of completeness, and using the usual notation, a straightforward first order explicit advancement scheme for the level set equation is given by

$$\phi_{ijk}^{n+1} = \phi_{ijk}^{n} - \Delta t[\max(F_{ijk}, 0)\nabla^{+}\phi + \min(F_{ijk}, 0)\nabla^{-}\phi], \qquad (1.10)$$

where

$$\nabla^+\phi = \left[\begin{array}{c} \max(D^{-x}_{ijk}\phi,0)^2 + \min(D^{+x}_{ijk}\phi,0)^2 + \\ \max(D^{-y}_{ijk}\phi,0)^2 + \min(D^{+y}_{ijk}\phi,0)^2 + \\ \max(D^{-z}_{ijk}\phi,0)^2 + \min(D^{+z}_{ijk}\phi,0)^2 \end{array} \right]^{1/2}$$

$$\nabla^-\phi = \left[\begin{array}{c} \max(D^{+x}_{ijk}\phi,0)^2 + \min(D^{-x}_{ijk}\phi,0)^2 + \\ \max(D^{+y}_{ijk}\phi,0)^2 + \min(D^{-y}_{ijk}\phi,0)^2 + \\ \max(D^{+z}_{ijk}\phi,0)^2 + \min(D^{-z}_{ijk}\phi,0)^2 \end{array} \right]^{1/2}$$

For information about these and higher order variations, see [14].

1.2.4 *Adaptivity and Efficiency*

Considerable computational speedup in the level set method comes from the use of the "Narrow Band Level Set Method", introduced by Adalsteinsson and Sethian in [1]. It is clear that performing calculations over the entire computational domain is wasteful. Instead, an efficient modification is to perform work only in a neighborhood (or "narrow band") of the zero level set. This drops the operation count in three dimensions to $O(kN^3)$, where k is the number of cells in the narrow band. This is a significant cost reduction; it also means that extension velocities need only be constructed at points lying in the narrow band, as opposed to all points in the computational domain.

1.2.5 *Fast Marching Methods*

There is a different view of propagating interfaces given by Fast Marching Methods [20], which exchanges the initial value perspective for a boundary value approach. Fast Marching Methods are finite difference techniques, more recently extended to unstructured meshes, for solving the Eikonal equation of the form

$$|\nabla T| F(x,y,z) = 1 \qquad\qquad T = 0 \text{ on } \Gamma.$$

This can be thought of as a front propagation problem for a front initially located at Γ and propagating with speed $F(x,y,z) > 0$. We note that this is a *boundary value* partial differential equation as opposed to an initial value problem given by level set methods, even though it describes a

moving interface. This Eikonal equation describes a large number of physical phenomena, including those from optics, wave transport, seismology, photolithography and optimal path planning, and Fast Marching Methods have been used to solve these and a host of other problems. We refer the reader to [24] and [23] for a large collection of applications based on this technique.

Fast Marching Methods are very fast ($O(N \log N)$) methods for solving the Eikonal equation, and rely on a marriage of upwind finite difference schemes, heap sort techniques, and a Dijkstra-like update ordering which reduces the problem to a single pass algorithm. We note only briefly two critical facts that lie at the core of Fast Marching Methods:

- An upwind difference scheme can be used to approximate the Eikonal equation in a viscosity-satisfying framework, namely

$$\left[\begin{array}{c} \max(D_{ijk}^{-x}u, -D_{ijk}^{+x}u, 0)^2 + \\ \max(D_{ijk}^{-y}u, -D_{ijk}^{+y}u, 0)^2 + \\ \max(D_{ijk}^{-z}u, -D_{ijk}^{+z}u, 0)^2 \end{array} \right]^{1/2} = \frac{1}{F_{ijk}}, \tag{1.11}$$

  (see Rouy and Tourin [16]).
- The order in which the grid values produced through these finite difference approximations are obtained is intimately connected to Dijkstra's method [7], which is a depth-search technique for computing shortest paths on a network. In that technique, the algorithm keeps track of the speed of propagation along the network links, and fans out along the network links to touch all the grid points. The Fast Marching Method exploits a similar idea in the context of a continuous finite difference approximation to the underlying partial differential equation, rather than discrete network links. The resulting technique is $O(N \log N)$, and hence is extraordinarily fast.

The Fast Marching Method evolved in part from examining the limit of the Narrow Band level set method as the band was reduced to one grid cell. Fast Marching Methods, by taking the perspective of the large body of work on higher order upwind, finite difference approximants from hyperbolic conservation laws, allow for higher order versions on both structured and unstructured meshes. The Fast Marching Method has been extended

to higher order finite difference approximations by Sethian in [24], first order unstructured meshes by Kimmel and Sethian [10], and higher order unstructured meshes by Sethian and Vladimirsky [26]. Some early applications include photolithography in [21], a comparison of this approach with volume-of-fluid techniques in [9], a fast algorithm for image segmentation in [12] and computation of seismic travel times by Sethian and Popovici [25]; see also [27] for a different Dijkstra-like algorithm which obtains the viscosity solution through a control-theoretic discretization which hinges on a causality relationship based on the optimality criterion.

1.3 Applications

The range of applications of level set and Fast Marching Methods is vast. Figure 1.8 gives a perspective on how some of these topics are related. There are many other aspects in the evolution of these ideas; the chart is meant to give perspective on how the theory, algorithms, and applications have evolved. The text and bibliography of [23] gives a somewhat more complete sense of the literature and the range of work underway.

In this paper, we discuss only one application, namely sintering and flow under surface diffusion. Sintering (see [11; 28]) is the process under which a compact consisting of many particles is heated to such a high temperature that the particles become a viscous creeping fluid, and the particles begin to coalesce together. One of the oldest technological examples involves the production of bricks; other examples include formation of rock strata from sandy sediments and the motion of thin films of metals in the microfabrication of electronic components.

At issue is the solution of the equations for creeping flow, in which the body forces on the boundary of the materials depend on the tangential stress derivative on the boundary. In one component of this model, the interface speed F in its normal direction depends on the second derivative of the curvature, where the derivatives are taken with respect to arc length α. Thus, in our level set framework, one wants (in two dimensions) to follow a curve propagating with speed $F = -\epsilon\kappa_{\alpha\alpha}$. Thus,

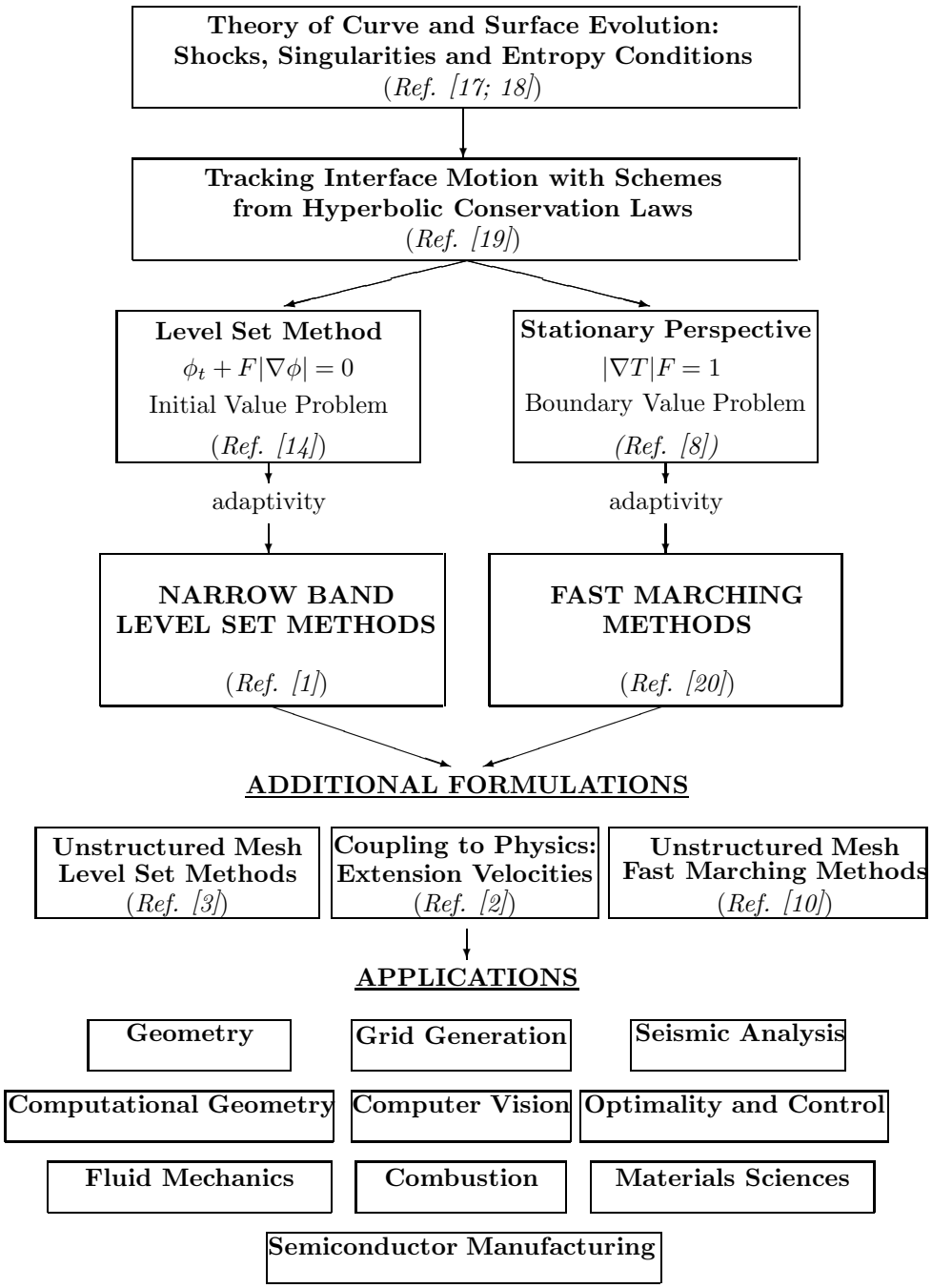

Fig. 1.8 Algorithms and Applications for Interface Propagation

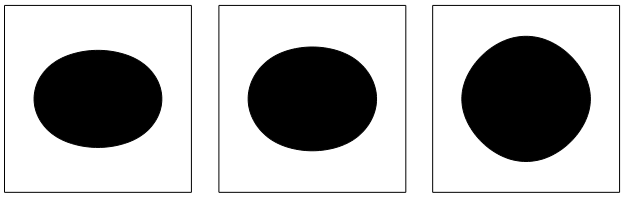

Fig. 1.9 Motion of ellipse under speed $F = \kappa_{\alpha\alpha}$.

$$\phi_t + \epsilon\kappa_{\alpha\alpha}|\nabla\phi| = \phi_t + \epsilon\left[\nabla\cdot\frac{\nabla\phi}{|\nabla\phi|}\right]_{\alpha\alpha}|\nabla\phi| = 0,$$

$$\phi_t = \left[-\epsilon\nabla\left[\nabla\left[\nabla\cdot\frac{\nabla\phi}{|\nabla\phi|}\right]\cdot\frac{(\phi_y,-\phi_x)}{|\nabla\phi|}\right]\cdot\frac{(\phi_y,-\phi_x)}{|\nabla\phi|}\right]|\nabla\phi|. \qquad (1.12)$$

This means that the speed depends on the *fourth derivative* of the level set function. We immediately note that a circle is a stable object, since the curvature is constant. A little examination leads one to think that an ellipse undergoes a restoring force which brings it back into a circle.

What about more complex shapes? The problem is quite subtle. Numerical experiments are notoriously unstable when they involve computing fourth derivatives†, and are eloquently described by Van de Vorst [28]; he uses marker particle schemes together with elaborate remeshing strategies to keep the calculation alive.

A level set approach to this problem was developed by Chopp and Sethian [5]. In that work, the individual derivatives in the above expression were approximated by central difference approximations and are used to study the motion of a sequence of closed curves to analyze flow under the second derivative of curvature given by Eqn. 1.12. Here, we summarize some of the results in [5]. First, in Figure 1.9, we show the evolution of a simple ellipse under this motion. The transformation shows the elliptical initial state on the left, followed by the evolution into a circle, which then remains fixed after a large number of calculations.

This might seem to indicate that a convex shape remains convex as it flows under this evolution equation. This in fact is not true, as seen by

†For example, computing the solution to the biharmonic equation is delicate.

Iter=1 Iter=100 Iter=200

Fig. 1.10 Motion of elongated ellipse under speed $F = \kappa_{\alpha\alpha}$.

examining the motion of a slightly more elongated ellipse in Figure 1.10. At points of high curvature, the interface moves inward, leaving a bulge which propagates around to the flatter sides until the interface balances itself out. In the case of sharp corners, the effect is more pronounced, as seen in Figure 1.11. Figure 1.11 shows the evolution of several non-convex initial shapes, all of which approach the stable state of a circle. The curves are shown at uneven times, and the flows are not completed.

Second derivative flow becomes even more murky in the face of topological change. Imagine two ellipses, each with a large ratio between the major and minor axes. If they are put side by side (rather than end to end), the flatter sides will cross over each other, and one expects (at least in many physical situations) some sort of merger, as in Figure 1.12.

This example underlies the difference between curvature flow ($F = -\kappa$) and flow by the second derivative of curvature. In the former, a maximum principle ensures that two separate closed curves will always remain separate under this flow. This allows the sort of natural embedding prescribed by a level set interpretation. In contrast, flow by the second derivative of curvature has no such property, as demonstrated in the previous example. Hence, the notion of embedding the motion of the two ellipses in a single "level set function" requires thought.

In order to execute true merger of two regions moving under the second derivative of curvature, care must be taken. In Fig. 1.13, taken from [5], we show the merger of two rectangular regions. The results show how the regions come together. Finally, in Figure 1.14, also taken from [5], we show an example of three-dimensional flow under the Laplacian of curvature,

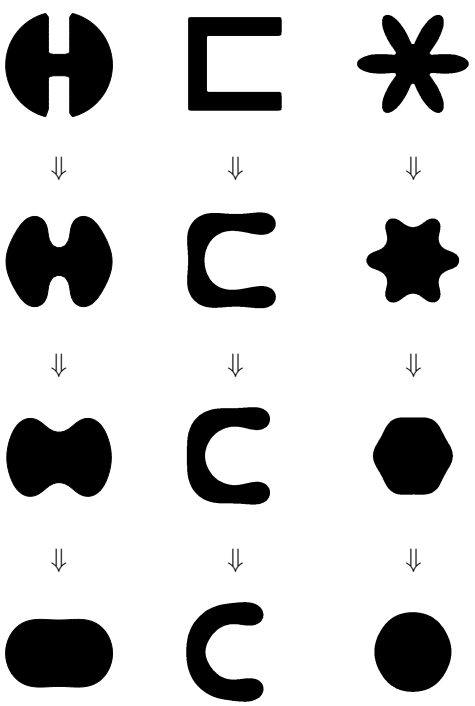

Fig. 1.11 Motion of non-convex curves under speed $F = \kappa_{\alpha\alpha}$.

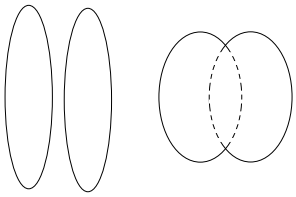

Fig. 1.12 Motions of two ellipses under the second derivative of curvature.

revealing the smoothing effects of this flow. For details, see [5].

Finally, we point out that the above algorithm is slow, because it is restricted by the time step due to the Courant condition. In [6], an alternative technique is presented that uses an approximation to the equations of motion that allows for a much larger time step.

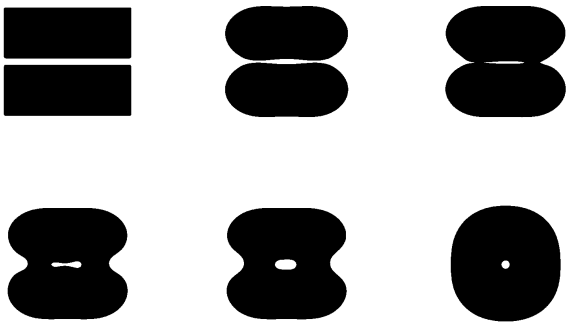

Fig. 1.13 Merger of two separate regions under $F = \kappa_{\alpha\alpha}$.

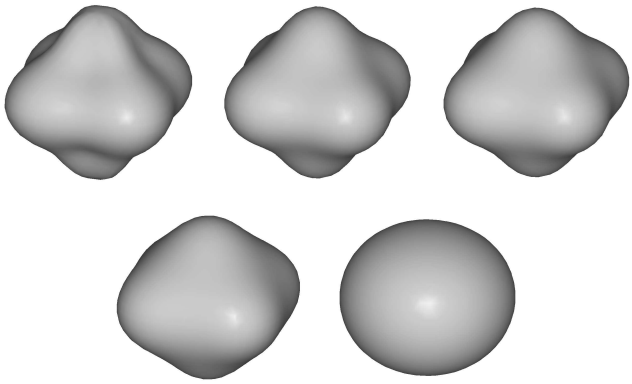

Fig. 1.14 Flow under Laplacian of Curvature

Acknowledgements

All calculations were performed at the University of California at Berkeley and the Lawrence Berkeley Laboratory. All of the work on sintering is joint with Professor David Chopp of Northwestern University. A detailed web page on level set methods and Fast Marching Methods may be found at www.math.berkeley.edu/∼sethian/level_set.html.

Bibliography

Adalsteinsson, D., and Sethian, J.A., *A Fast Level Set Method for Propagating Interfaces*, J. Comp. Phys., 118, 2, pp. 269–277, 1995.

Adalsteinsson, D., and Sethian, J.A., *The Fast Construction of Extension Velocities in Level Set Methods*, J. Comp. Phys., 148, 1999, pp. 2-22.

Barth, T.J., and Sethian, J.A., *Numerical Schemes for the Hamilton-Jacobi and Level Set Equations on Triangulated Domains*, J. Comp. Phys., 145, 1, pp. 1–40, 1998.

Bunner, B.; Tryggvason, G., *Direct numerical simulations of three-dimensional bubbly flows*, Physics of Fluids, 11, 8, pp. 1967-9, 1999.

Chopp, D.L., and Sethian, J.A., *Motion by Intrinsic Laplacian of Curvature*, CPAM Report PAM-746, Dept. of Mathematics, Univ. of California, Interfaces and Free Boundaries, 1, 1999.

Chopp, D.L., Tongen, A., and Sethian, J.A., *Fast Approximations of Surface Diffusion*, submitted for publication, J. Comp. Phys., Jan., 2000.

Dijkstra, E.W., *A Note on Two Problems in Connection with Graphs*, Numerische Mathematic, 1:269–271, 1959.

Garabedian, P., Partial Differential Equations, Wiley, New York, 1964.

Helmsen, J., Puckett, E.G., Colella, P., and Dorr, M., *Two new methods for simulating photolithography development*, SPIE 1996 International Symposium on Microlithography, SPIE, v. 2726, June, 1996.

Kimmel, R., and Sethian, J.A., *Fast Marching Methods on Triangulated Domains*, Proc. Nat. Acad. Sci., 95, pp. 8341-8435, 1998.

Kuiken, H.K., *Viscous Sintering: the Surface-tension-driven Flow of a Liquid Form under the Influence of Curvature Gradients at its Surface*, J. Fluid Mech., 214, pp. 503–515, 1990.

Malladi, R., and Sethian, J.A., *An $O(N \log N)$ Algorithm for Shape Modeling*, Proc. Nat. Acad. Sci., Vol. 93, pp. 9389-9392, 1996.

Noh, W., and Woodward, P., *A Simple Line Interface Calculation.* Proceedings, Fifth International Conference on Fluid Dynamics, Eds. A.I. vn de Vooran

and P.J. Zandberger, Springer-Verlag, 1976.

Osher, S., and Sethian, J.A., *Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton–Jacobi Formulations*, Journal of Computational Physics, 79, pp. 12–49, 1988.

Puckett, E.G., *A Volume-of-Fluid Interface Tracking Algorithm with Applications to Computing Shock Wave Refraction*, Proceedings of the 4th International Symposium on Computational Computational Fluid Dynamics, Davis, California, 1991.

Rouy, E. and Tourin, A., *A Viscosity Solutions Approach to Shape-From-Shading*, SIAM J. Num. Anal, 29, 3, pp. 867–884, 1992.

Sethian, J.A., *An Analysis of Flame Propagation*, Ph.D. Dissertation, Dept. of Mathematics, University of California, Berkeley, CA, 1982.

Sethian, J.A., *Curvature and the Evolution of Fronts*, Comm. in Math. Phys., 101, pp. 487–499, 1985.

Sethian, J.A., *Numerical Methods for Propagating Fronts*, in Variational Methods for Free Surface Interfaces, Eds. P. Concus and R. Finn, Springer-Verlag, NY, 1987.

Sethian, J.A., *A Fast Marching Level Set Method for Monotonically Advancing Fronts*, Proc. Nat. Acad. Sci., 93, 4, pp.1591–1595, 1996.

Sethian, J.A., *Fast Marching Level Set Methods for Three-Dimensional Photolithography Development*, Proceedings, SPIE 1996 International Symposium on Microlithography, Santa Clara, California, June, 1996.

Sethian, J.A., *A Review of the Theory, Algorithms, and Applications of Level Set Methods for Propagating Interfaces*, Acta Numerica, Cambridge University Press, 1996.

Sethian, J.A., Level Set Methods and Fast Marching Methods, Cambridge University Press, 1999.

Sethian, J.A., *Fast Marching Methods*, SIAM Review, 41, July, pp. 199-235, .1999.

Sethian, J.A., and Popovici, M., *Fast Marching Methods Applied to Computation of Seismic Travel Times*, Geophysics, 64, 2, 1999.

Sethian, J.A., and Vladimirsky, A., *Extensions to Triangulated Fast Marching Methods*, submitted for publication, Proc. Nat. Acad. Sci., Nov. 1999.

Tsitsiklis, J.N., *Efficient Algorithms for Globally Optimal Trajectories*, IEEE Transactions on Automatic Control, Volume 40, pp. 1528-1538, 1995.

Van de Vorst, G.A.L., *Modeling and Numerical Simulation of Viscous Sintering*, PhD. Thesis, Eindhoven University of Technology, Febodruk-Enschede, The Netherlands, 1994.