Acta Math. Univ. Comenianae
Vol. LXVII, 1(1998), pp. 3–15

ADAPTIVE FAST MARCHING AND LEVEL SET METHODS FOR PROPAGATING INTERFACES

J. A. SETHIAN

Abstract. Adaptivity provides a way to construct optimal algorithms for tracking moving interfaces which arise in a wide collection of physical applications. Here, we summarize the development and interconnection between Narrow Band Level Set Methods and Fast Marching Methods, which provide efficient techniques for tracking fronts. We end with a small collection of examples to demonstrate the applicability of the techniques.

1. Introduction

Over the past ten years, a collection of numerical techniques have been developed to track propagating interfaces that arise in physical phenomena. These techniques allow for evolution under complex speed laws, including the effects of curvature and anisotropy, easily couple to the underlying physics, allow for natural topological change in the evolving interface, including splitting and merging, and are unchanged in three or more space dimensions. They take on a partial differential equations approach to the interface problem, casting the motion as either an initial value or boundary value partial differential equation, and rely on finite difference approximations to provide convergence, consistent numerical techniques of high order. Because of this reliance on finite difference schemes, the error of the solution is known at the start, and can be rigorously controlled.

At their core, these techniques hinge on the "viscosity solutions" view of the underlying equations, in which the correct weak solution is chosen which interprets the propagating front as a physical boundary between two regions. Two such techniques are **Level Set techniques**, introduced by Osher and Sethian [**6**], and **Fast Marching Methods**, introduced by Sethian in [**12**]. Both grew out of the theory of curve and surface evolution developed in [**9**], [**10**], [**11**], which develops the notion of weak solutions and entropy limits for evolving interfaces, and links upwind numerical methodology for hyperbolic conservation laws to front

Received November 17, 1997.
1980 *Mathematics Subject Classification* (1991 *Revision*). Primary 65M99; Secondary 68T10.
Key words and phrases. Level set methods, Fast Marching Methods.
Supported in part by the Applied Mathematics Subprogram of the Office of Energy Research under contract DE-AC03-76SF00098, and the National Science Foundation and DARPA under grant DMS-8919074.

propagation problems. Both become computationally efficient through the use of adaptive methodology.

In this review, we discuss the development of these techniques, describe the regimes in which each is appropriate, show how they are interrelated, and show how adaptivity leads to optimal techniques.

The outline of this paper is as follows. First, we discuss the two main interface perspectives: the initial value level set technique, and the boundary value stationary perspective. Next, we discuss the role of entropy-conditions and singularities in propagating interfaces, and the value of upwind schemes for approximating gradients to extract the correct entropy satisfying solution. Then, we discuss the role of adaptivity, leading to the Narrow Band Level Set Method and the Fast Marching Method. Finally, we end with a few examples to demonstrate the various techniques. For complete details and a review of the applications of Fast Marching and Level Set Methods, see [**13**].

2. Two Views of Propagating Interfaces

Given a moving closed hypersurface $\Gamma(t)$, that is, $\Gamma(t)\colon [0,1] \to R^N$, propagating with a speed F in its normal direction, we wish to produce a partial differential equations formulation for the motion of the hypersurface propagating along its normal direction with speed F, where F can be a function of various arguments, including the curvature, normal direction, etc. Our goal is an "Eulerian" formulation – that is, one in which the motion of the interface is described in terms of its action on an underlying fixed coordinate system.

Here, we imagine that the speed function F can be quite complex, depending not just on the geometry of the front, but also on the solution of various partial differential equations on either side of the interface, and which may include the interface as internal boundary conditions. For example, in crystal growth and dendritic solidification, (see [**14**]), the problem may dictate the solution of the heat equation on either side of the front, as well two internal boundary conditions which relate the speed of the interface to jumps in the heat and local surface tension.

Two possible ways to formulate this problem are given below, depending on the complexity of the speed function F.

2.1 The Stationary Boundary Value Perspective

Suppose that the speed function F is always strictly positive[1]. In this case, we may cast the evolving interface problem as a stationary, boundary value partial differential equation, see Figure 2.1.

[1] or strictly negative

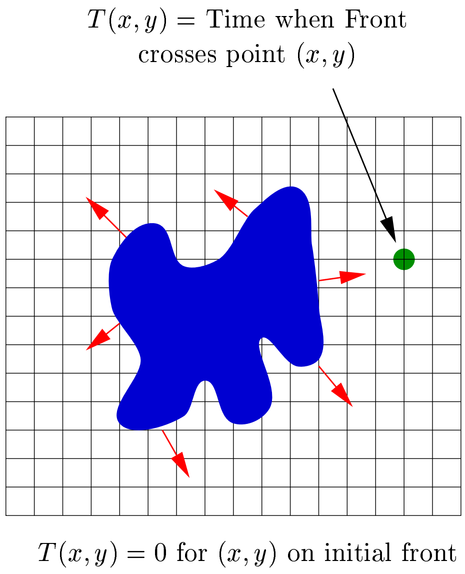


Figure 2.1. Transformation into Stationary Boundary value PDE.

Let $T(x, y)$ be the time at which the interface crosses the point (x, y); then we can write an equation for the solution T as

$$|\nabla T|F = 1$$
$$T = 0 \quad \text{on} \quad \Gamma .$$

We note that:

1. The above is a **boundary value** partial differential equation; the goal is to construct the solution surface $T(x, y)$ away from the value on the boundary curve.
2. If the speed function F depends only on position and first derivatives of the solution T, the resulting equation is a static Hamilton-Jacobi equation.
3. If the speed function F depends only on the position (x, y), then the resulting equation is the familiar Eikonal equation.
4. In any case, the solution T typically is multi-valued; although we require that the speed function F be strictly positive, this in itself does not ensure that the solution T only reflects a single crossing of the point (x, y). In fact, below we shall restrict our solution to the so-called viscosity solution which limits the solution to the first crossing time T.

2.2 The Initial Value Level Set Perspective

In the more general case of an arbitrary speed function F which may change sign, a different view is given by the Level Set Perspective, introduced by Osher and Sethian [**6**]. If we embed the propagating interface as the zero level set of a higher dimensional function ϕ, that is, let $\phi(x, t = 0)$, where $x \in R^N$ be defined by

$$\phi(x, t = 0) = \pm d; \tag{1}$$

where $\pm d$ is the signed distance to the interface from the point x, taken as positive if x is outside and negative if x is inside. An initial value partial differential equation can be obtained for the evolution of ϕ, namely

$$\phi_t + F|\nabla\phi| = 0 \tag{2}$$
$$\phi(x, t = 0) \quad \text{given} \tag{3}$$

This is known as the level set equation.

Note that the construction of the initial value PDE given in Eqn. (3) means that the velocity F is now defined for **all** the level sets, not just the zero level set corresponding the interface itself. We can be a little more precise about this by rewriting the level set equation as

$$\phi_t + F_{\text{ext}}|\nabla\phi| = 0 \tag{4}$$

where F_{ext} is some velocity field which, at the zero level set, equals the given speed F. In other words,

$$F_{\text{ext}} = F \text{ on } \phi = 0$$

This new velocity field F_{ext} is known as the "extension velocity". In many cases, construction of this extension velocity requires considerable effort, see [**3**] for fast techniques for doing so.

2.3 Advantages of the Eulerian PDE Perspective

The advantages of these perspectives include the following:

- As discussed in [**9**], [**10**], [**11**], shocks and rarefactions can develop in the slope, corresponding to corners and fans in the evolving interface, and numerical techniques designed for hyperbolic conservation laws can be exploited to construct upwind schemes which produce the correct, physically reasonable entropy solution. These are naturally captured in the above representations.
- The front is free to change topology as it evolves; no special care is required. In the stationary perspective, the front at time t is given by the set of all (x, y) such that $t = T(x, y)$. In the level set perspective, the front at time t is given

by the set of all points (x, y) such that $\phi(x, y, t) = 0$. In both cases, while the solution T and ϕ are single-valued function, the lower-dimensional set of points corresponding to the front may break, merge, and consist of multiple regions.
- Finite difference schemes may be employed to compute the solution to the pde's in a relatively straightforward manner.
- There are no differences in the above construction for hypersurfaces propagating in three or more space dimensions.

In order to compute the solution of the above boundary value and initial value equations, we need to exploit the use of upwind schemes in hyperbolic conservation laws, and the connection to theory of viscosity solutions developed by Crandall and Lions [**4**].

3. Entropy Solutions, Upwind Schemes, and Viscosity Solutions

It is well-known, (see [**10**], [**13**]), that both of the above equations of motion become non-differentiable for certain initial data, and appropriate weak solutions must be built. Several solutions are possible once a singularity occurs, including the swallowtail solution. The appropriate weak solution comes from satisfying the entropy condition introduced in [**10**]; which may be summarized as follows. Imagine the interface as a propagating flame front; with the requirement that once the front burns past a certain point it stays burnt. This selects a unique solution beyond the occurrence of singularities. There are several different ways to interpret this selection process:

- **A Curvature Regularization View**: In the presence of a curvature term, the solution remains smooth for all time, see [**10**]. The chosen solution is the limiting solution as the regularizing curvature term vanishes.
- **A Wave Front View**: The correct solution is obtained from Huyghen's principle, that is, the solution is the envelope of all disturbances located on the initial front and expanding isotropically with local speed F. Thus, the chosen weak solution corresponds to the first arrival time of information from the front.
- **An Optics View**: The chosen weak solution is the first term in the standard optics expansion corresponding the local Eikonal equation, and caustics that arrive from later waves are ignored.
- **The Viscosity Solutions Framework**: The solution to the equation of motion is defined in terms of the effect on smooth test functions on the solution, see [**4**]. In the case of a smooth solution, this "viscosity solution" is identical to the classical one; in the case of a non-differential viscosity solution, it is then **proved** that the solution is the viscous limit of the same equation with a diffusive smoothing term. For details, see [**4**], [**13**].

The viscosity solutions view is the most rigorous and precise, however, all lead to the same construction.

In order to approximate the equations of motion, the key idea is to select an approximation to the gradient operator ∇T or $\nabla \phi$ which correctly chooses this correct limiting weak solution. One of the simplest such upwind entropy satisfying approximations to a gradient ∇T was given in [**6**], namely

$$|\nabla T| \approx \left[\max(D_{ij}^{-x}T,0)^2 + \min(D_{ij}^{+x}T,0)^2 + \max(D_{ij}^{-y}T,0)^2 + \min(D_{ij}^{+y}T,0)^2\right]^{1/2}. \tag{5}$$

Here, we have used standard finite difference notation, namely that $D_{ij}^{-x}T = \frac{T_{i,j}-T_{i-1,j}}{\Delta x}$, where Δx is the space step; the other operators are defined similarly. The crucial point in this (or any such appropriate) numerical scheme is the correct direction of the upwinding and treatment of sonic points. For details and an extensive review, see [**13**].

Employing these upwind operators, we may now easily write down workable (though inefficient) schemes for both the stationary and level set perspectives:

- **Stationary Perspective**:
  Find T_{ij} such that

$$\left[\max(D_{ij}^{-x}T - D_{ij}^{+x}T,0)^2 + \max(D_{ij}^{-y}T - D_{ij}^{+y}T,0)^2\right]^{1/2} = 1./F_{ij}\,.$$

  where here we have chosen the upwind operator given in [**8**]. Our reason for doing so is that it provides an slightly less diffusive operator, which in fact is also easier to work with.
- **Level Set Perspective**:
  Compute the evolution of ϕ_{ij}, where

$$\phi_{ij}^{n+1} = \phi_{ij}^{n} + (\Delta t)(F_{ij})\left[\max(D_{ij}^{-x}\phi,0)^2 + \min(D_{ij}^{+x}\phi,0)^2 + \max(D_{ij}^{-y}\phi,0)^2 + \min(D_{ij}^{+y}\phi,0)^2\right]^{1/2}\,.$$

Here, we have employed standard finite difference notation. The first scheme requires an iteration to construct the solution; one starts with an initial guess, and iterates until convergence. This, for example, is the approach taken in [**8**]. The second scheme is an explicit, time-marching algorithm which requires one to compute the evolution of **all**, the level sets, not just the zero level set corresponding to the interface itself. In fact, both schemes can be made efficient through adaptivity; this is the topic of the next section.

4. Efficient Front Propagation Schemes through Adaptivity

4.1. Adaptivity for Level Set Methods: the Narrow Band Approach
The straightforward level set method introduced in [**6**] is a somewhat time-consuming algorithm. We can make a rough operation count as follows; consider a curve propagating in two-dimensions with speed $F = 1$, and suppose one chooses a time step that exactly matches the CFL condition, so that $\Delta t/\Delta x = 1$. Then with N points in each space direction, a total of N^2 points are updated every time step, with roughly N time steps required for the interface to propagate its way from the center out to the edge of the computational domain, yielding an $O(N^3)$ algorithm.

The narrow band method, introduced in [**1**], is an adaptive level set method that limits computational labor to a grid points located in a narrow band around the front. Grid points around the front are kept in a one-dimensional array, and updated using the level set equation until the the interface nears the edge of this narrow band, at which point a new narrow band is re-initialized. Figure 4.1 shows how the narrow band tags a collection of nearby grid points.

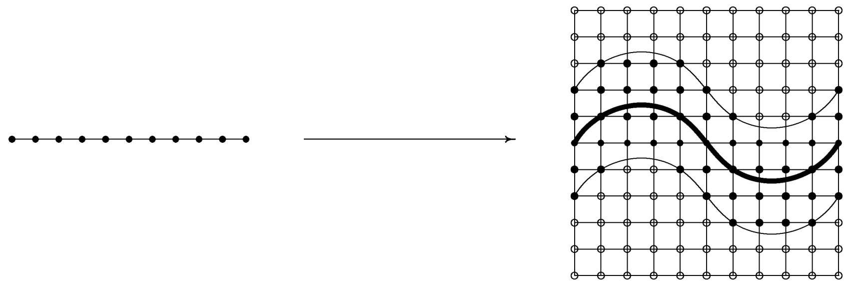


Figure 4.1.

By employing this technique, the computational labor for a curve propagating in two dimensions drops to $O(kN^2)$, where k is the width of the narrow band. An extensive discussion about narrow band methods, choice of sizes, accuracy, and other details may be found in [**1**].

4.2 Adaptivity for the Stationary Approach: Fast Marching Methods
As described above, the solution to the stationary equation typically requires iteration to construct the solution surface. In fact, as developed in [**12**], the key observation in Fast Marching Methods is to exploit the fact that use of an upwind difference operator prescribes an ordering of the points so that iteration is not required, and the solution may be constructed in a single pass. Information propagates "one way", that is, from smaller values of T to larger values. Hence, the

fast marching algorithm rests on "solving" Equation (3) by building the solution outward from the smallest T value. The idea is to sweep the front ahead in an upwind fashion by considering a set of points in narrow band around the existing front, and to march this narrow band forward, freezing the values of existing points and bringing new ones into the narrow band structure. The key is in the selection of which grid point in the narrow band to update.

The algorithm is as follows: First, we tag points in the initial conditions as **Alive**. We then tag as **Close** all points one grid point away. Finally, we tag as **Far** all other grid points. Then the loop is:

1. Begin Loop: Let **Trial** be the point in **Close** with the smallest value for u.
2. Add the point **Trial** to **Alive**; remove it from **Close**
3. Tag as **Close** all neighbors of **Trial** that are not **Alive**. If the neighbor is in **Far** remove it from that list and add it to the set **Close**.
4. Recompute the values of u at all neighbors according to Eqn. (3) by solving the quadratic equation, only using values for points that are **Alive**.
5. Return to top of Loop;

This algorithm works because the process of recomputing the u values at upwind neighboring points cannot yield a value smaller than any of the accepted points. Thus, we can march the solution outward, always selecting the narrow band grid point with minimum trial value for u, and readjusting neighbors, (see Figure 4.2).

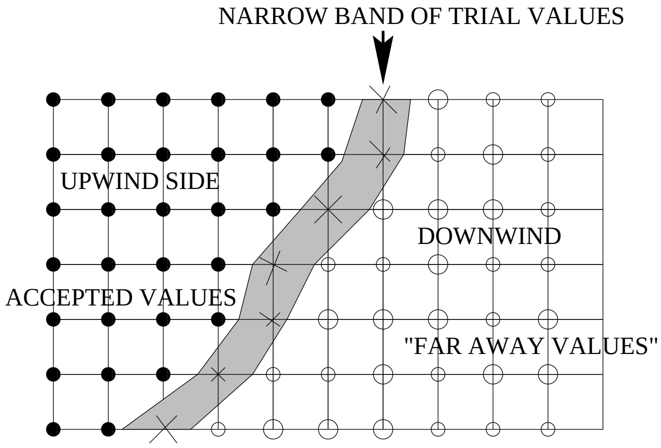


Figure 4.2.

Another way to look at this is that each minimum trial value begins an application of Huyghen's principle, and the expanding wave front touches and updates all others. The speed of the algorithm comes from a heapsort technique to efficiently locate the smallest element in the set **Close**. Suppose that there are M

points in the computational domain. Then each point is visited once; the heap requires $\log M$ operations to keep its structure. The resulting algorithm is thus $O(M \log M)$, which is, to our knowledge, the fastest of all possible algorithms. For more details, see [**12**], [**13**].

4.3 Summary of the Two Techniques

The linking between these two techniques is summarized in Figure 4.3.

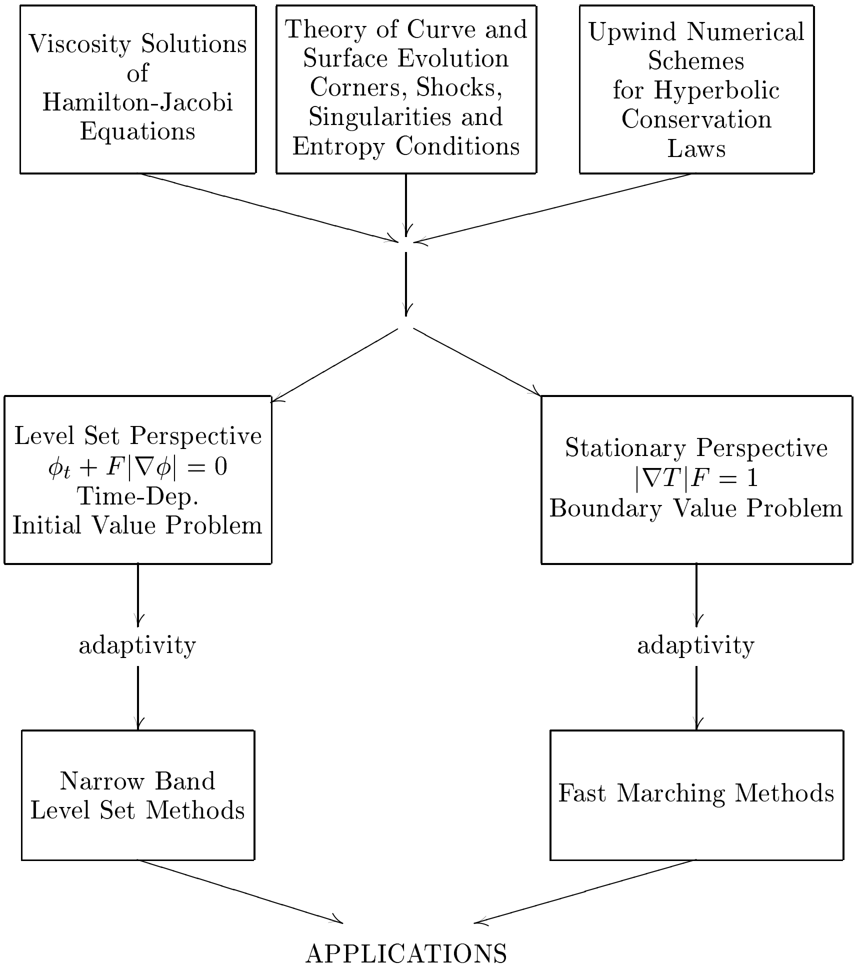


Figure 4.3.

5. Examples

We end with few examples to demonstrate the techniques. The first is the calculation of seismic travel times, see [**7**]. Here, the Fast Marching Method is used to construct the first arrival times of seismic disturbance. The contours indicate the equi-arrival lines, see Figure 5.1. The slowness function F is supplied as data, and the result computes the three-dimensional arrival surface.

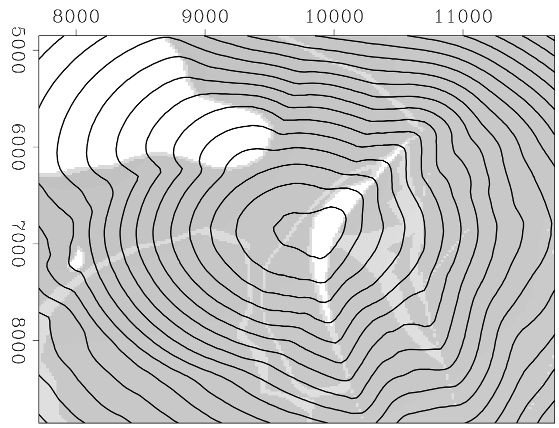


Figure 5.1. Calculation of First Arrivals.

The second application shows the reconstruction of the cortical surface of the brain from a scan (see Figure 5.2). Here, the image gradient is used to synthesize a speed function, which is the given to a propagating seed. As the interface nears the edge of the desired shape, the interface slows in response to the large image gradient, segmenting the image. The algorithm is a hybrid of Fast Marching Methods to quickly reach the edge of the desired shape, coupled to Narrow Band Level Set techniques to produce the fine structure. For details, see [**5**].

Finally, we end with an example from semiconductor manufacturing, in which the goal is to simulate etching and deposition processes during microchip fabrication. We show the evolution of a saddle surface during ion-milling, which is an etching process in which the optimal etching angle occurs not with a direct beam from the normal, but in fact from a glancing blow from the side. The angle θ is the angle between the normal to the surface and the vertical. The resulting Hamilton-Jacobi equation contains a non-convex flux function, which gives rises to faceting and sharp edges in the resulting solution. In Figure 5.3, we show the time motion of such a surface, demonstrating these faceting effects on the evolving shape. For details, see [**2**].

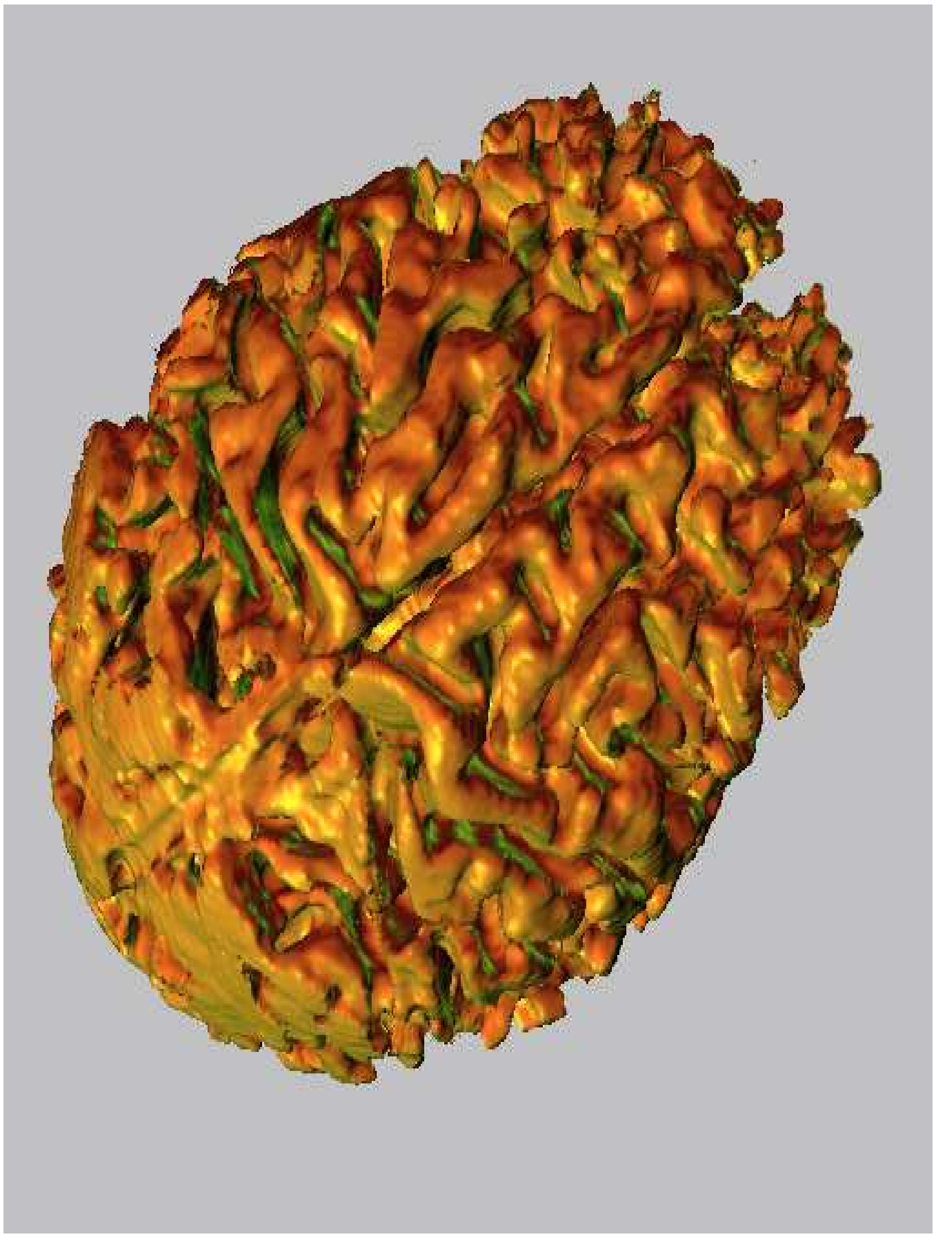
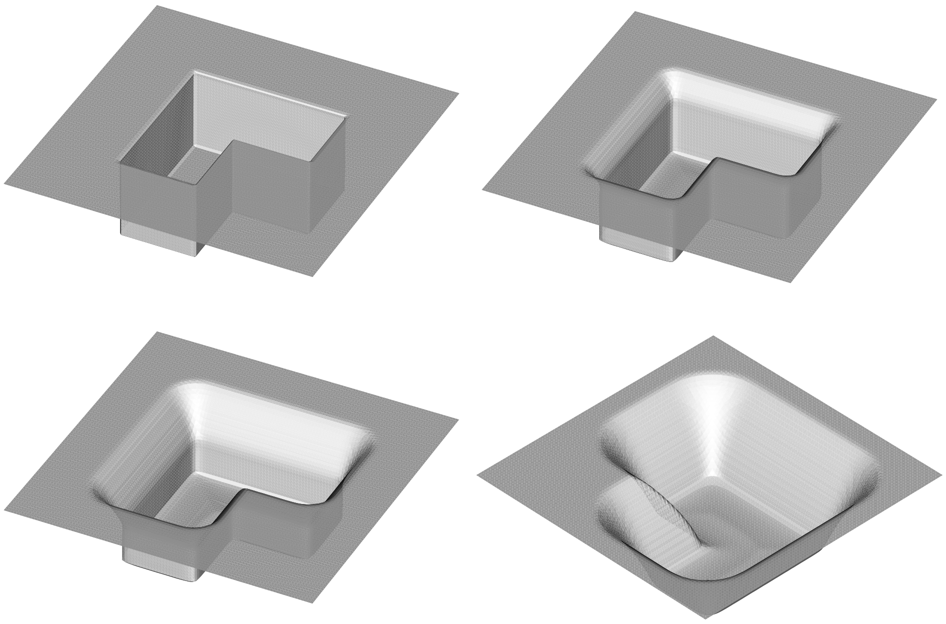


Figure 5.2. Reconstructing Cortical Structure from Scan.

Acknowledgments. All calculations were performed at the University of California at Berkeley and the Lawrence Berkeley Laboratory.



$F = [1 + 4\sin^2(\theta)]\cos(\theta)\ T = 4$ Rotated

Figure 5.3. Sputter Etching of Concave/Convex Saddle Surface.

References

1. Adalsteinsson D. and Sethian J. A., *A fast level set method for propagating interfaces*, Jour. Comp. Phys. **118** (1995), 269–277.
2. ———, *A Level Set Approach to a Unified Model for Etching, Deposition, and Lithography II: Three-Dimensional Simulations*, Jour. Comp. Phys. **122**(2) (1995), 348–366.
3. Adalsteinsson D. and Sethian J. A., *Fast Marching Methods for Reinitialization and Extension Velocities in Level Set Methods*, in progress..
4. Crandall M. G. and Lions P-L., *Viscosity Solutions of Hamilton-Jacobi Equations*, Tran. AMS **277** (1983), 1–43.
5. Malladi R. and Sethian J. A., *A Unified Approach to Noise Removal, Image Enhancement, and Shape Recovery*, IEEE Transactions on Image Processing **5**(11) (1996), 1154–1168,.
6. Osher S. and Sethian J. A., *Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation,*, Jour. Comp. Phys. **79** (1988), 12–49.
7. Sethian J. A. and Popovici M., *Three dimensional traveltimes computation using the Fast Marching Method, submitted for publication,*, Geophysics (1997).
8. Rouy E. and Tourin A., *A Viscosity Solutions Approach to Shape-from-shading*, SIAM. J. Numer. Anal. **29**(3) (1992), 867–884.
9. Sethian J. A., *An Analysis of Flame Propagation*, Ph.D. Dissertation, Mathematics, University of California, Berkeley, 1982..
10. ———, *Curvature and the evolution of fronts*, Commun. in Math. Physics **101** (1985), 487–499.
11. ———, *Numerical methods for propagating fronts*, Variational methods for free surface interfaces (P. Concus and R. Finn, eds.), Springer-Verlag, New Work, 1987.

12. ———, *CPAM Report, Dept. of Mathematics, Univ. of California, A Fast Marching Level Set Method for Monotonically Advancing Fronts*, Proc. Nat. Acad. Sci. **93**(4) (1996).

13. ———, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Material Science*, in press, Cambridge University Press, 1996.

14. Sethian J. A. and Strain J. D., *Crystal Growth and Dendritic Solidification*, J. Comp. Phys. **98** (1992), 231–253.

J. A. Sethian, Department of Mathematics, University of California, Berkeley and Lawrence Berkeley Laboratory, University of California, Berkeley, California 94720