Motion of Curves Constrained on Surfaces Using a Level Set Approach

Li-Tien Cheng *†
Paul Burchard *
Barry Merriman *
Stanley Osher *

August 31, 2000

Abstract

The level set method[21] has been successfully applied to a variety of problems that deal with curves in $\mathbf{R}^2$ or surfaces in $\mathbf{R}^3$. We present here a combination of these two cases, creating a level set representation for curves constrained to lie on surfaces. We will study primarily geometrically based motions of these curves on stationary surfaces while allowing for topological changes in the curves, i.e., merging and breaking, to occur. Applications include finding geodesic curves, shortest paths, and curve shortening on surfaces. Further applications can be arrived at by extending those for curves moving in $\mathbf{R}^2$ to surfaces. The problem of moving curves on surfaces can also be viewed as a simple constraint problem and may be useful in studying more difficult versions. Results show that our representation can accurately handle many geometrically based motions of curves on a wide variety of surfaces while automatically enforcing topological changes in the curves when they occur and automatically fixing the curves to lie on the surface. The method can also be easily extended to higher dimensions.

*Department of Mathematics, University of California Los Angeles, Los Angeles, California 90095. Research supported in part by ONR N00014-97-0027, NSF DMS 9706827, and ARO DAAG 55-98-1-0323.
†Research supported in part by an NDSEG Fellowship.

1 Introduction

The problem of moving curves on surfaces is important in many applications. It can be thought of as a model constraint problem. Also, since it is an extension of curve motion in $\mathbf{R}^2$, we may be able to extend the applications found there to surfaces. Such work includes those on geometrically based motions, image processing, two phase flow, and materials science (see [19]). We will mainly consider geometrically based motions here since it is an integral part of all of the other applications. Curves on surfaces are also useful in the study of surfaces. Geodesic curves and shortest paths are important elements of a surface and can be computed using certain geometrically based motions of curves. We allow for topological changes to occur in the curves as this is useful in many of the applications we have mentioned.

One obvious way to move curves on a surface is using a front tracking algorithm. This is where discrete points coming from a parametrization of the curves are evolved according to the flow (see, e.g., [11]). The main problem with this approach comes in finding when topological changes occur and enforcing the change when it does, a difficult problem to handle even for curves in $\mathbf{R}^2$. In this way, level set based methods are preferable since topological changes are automatically taken care of by the representation. Another issue in front tracking is in keeping the curve on the surface for all time. Thus, a great deal of work on moving curves on surfaces do not use this approach.

Previous work on moving curves on surfaces has mostly been confined either to specific motions or specific surface types. In [7], Chopp studied geodesic curvature motion of curves on manifolds by using a level set approach. This motion is also known as curve shortening or heat flow on isometric immersion (see [14]). Chopp's method involved computing on simply connected coordinate patches of the manifold projected into $\mathbf{R}^2$. This algorithm works for general manifolds as long as the patches are given. Finding the coordinate patches, however, is in general difficult. In [14], Kimmel also studied geodesic curvature flow using a different approach. He considered a surface given as a graph of a function and evolved the iso-gray level contours of a function representing a grayscale image on that surface as an application to image processing of images painted on surfaces. The algorithm, however, can only handle surfaces that can be represented as graphs of functions. Other methods similar to this one can be found in [16] and [15], and also in [10], which tackles the problem of grid generation. In [17], Kimmel and Sethian used a fast marching approach[27] on level set functions to compute geodesic paths on surfaces. In this method, the surface needs to be triangulated and also seems to also be restricted to those that can be represented as graphs of functions.

The method we construct will be able to handle a wide variety of motions for curves on more general surfaces as well as automatically enforce mergings and breakings in the curves. It can be viewed as an extension of [14] using a level set approach to more general motions and surface types. Also, Bertalmio, Sapiro, and Randall[2] have applied a method of the same flavor to the specific problem of region tracking. We will rederive and extend their results in Section 12. We now take a closer look at the standard level set method for curves in $\mathbf{R}^2$ and surfaces in $\mathbf{R}^3$ as we will use the ideas there to form our method.

2 Standard Level Set Method

The standard, or original, level set method[21] has been widely used to study curves in $\mathbf{R}^2$ and surfaces in $\mathbf{R}^3$. In this method, curves are represented by the zero level set of a real valued function on $\mathbf{R}^2$ and surfaces by the zero level set of a real valued function on $\mathbf{R}^3$. These particular functions are called level set functions. When studying moving curves and surfaces, the level set function is allowed to depend on time. Thus the zero level set at each time t represents the curve or surface at that time. Also, the motion of the zero level set can now be carried out by evolving the level set function. Usually this evolution is governed by a partial differential equation. One benefit of evolving the level set function instead of just its zero level set is that topological changes in the zero level set will be automatically enforced. Note the ability of a curve or surface to be represented as the zero level set of a function means the curve or surface must be the boundary of an open set. This limits the types of curves and surfaces a level set method can handle but does not seem to be overly restrictive and is a natural setup for problems such as two phase flow (see, e.g., [26]).

Solving the evolution equation associated to the level set function usually needs to be done numerically. For this, a uniform grid (usually) is placed in all of space, $\mathbf{R}^2$ for curves and $\mathbf{R}^3$ for surfaces. Finite difference schemes are then used to discretize the evolution equation. An advantage of this approach comes in the use of high order numerical discretizations on uniform grids. Efficiency both in memory and speed can still be preserved by only storing data and computing near the front[22] (see also [1]), though sometimes at the price of a loss of accuracy.

The main advantage of level set methods, however, is that mergings and breakings in the curves are automatically handled by the representation. The time of this happening does not need to be computed and no extra work is required to enforce the topological changes, unlike with front tracking methods. The evolution equation is simply solved in the same way at every time step up to the desired time. The curve can then be interpolated from the level set function at the end of a run, when the curve is then plotted. During the run, the curve location is not needed and can remain uncomputed. Although the ease in handling topological changes is one of the main reasons for using level set methods, they are nonetheless attractive even when topological changes do not occur because of the simple and accurate finite difference schemes on uniform grids that can be used. Thus the level set method can be easily programmed and used. More on the level set method can be found in [19]. All this naturally leads us to attempt to use a level set based method for our problem of curves on surfaces.

2.1 Setup

In tackling the problem of moving curves constrained on surfaces, we begin by using a level set formulation to represent the curves and surfaces. Thus given a collection of surfaces M in $\mathbf{R}^3$ and curves γ on those surfaces, we represent M by the zero level set of a real valued function ψ on $\mathbf{R}^3$ and γ by the intersection of the zero level set of a real valued function ϕ on $\mathbf{R}^3$ with the zero level set of ψ. As before, we will call these functions level set functions. Since we will mainly consider the case where M is static in time, ψ will not depend on time. On the other hand, in order to study moving curves on surfaces, we let ϕ depend on time. Thus the time evolution of ϕ allows us to follow the moving curves, keeping in mind that the curves at time t are the intersection of the zero level set of ϕ at time t and the zero level set of ψ. In our representation, only a specific class of surfaces, boundaries of open sets in $\mathbf{R}^3$, can be handled by this method. Similarly, only a specific class of curves on the surfaces, boundaries of open sets on M, can be considered. This implies that there is a notion of the inside and outside of the curves or surfaces and we take, for definiteness, the inside to be where the level set function is negative and the outside to be where it is positive. Once again, this is especially natural, for example, for curves denoting the interface between two fluids on M. We also note that ψ and ϕ need only be defined in a neighborhood of the curves and not necessarily in all of $\mathbf{R}^3$. However, for simplicity of exposition, we will continue treating them as functions over all of $\mathbf{R}^3$. The only concern will be when we need to slightly modify the method to obtain optimal efficiency both in speed and memory usage. This will be discussed in Subsection 8.1. Finally, we can study constrained flows in other spatial dimensions by taking ψ and ϕ to be functions in $\mathbf{R}^n$.

Note our setup is basically the same as in [3] except ψ is now held fixed in time. Thus the constrained problem of moving curves on surfaces turns out to be easier, using our setup, than the unconstrained problem of moving curves in $\mathbf{R}^3$. We now develop various notation and tools for our representation to help simplify and clarify future calculations.

3 Preliminaries

Given a vector w in $\mathbf{R}^3$, let P_w be the orthogonal projection matrix defined by

$$P_w = I - \frac{w \otimes w}{|w|^2},$$

where I is the identity matrix. Thus the components of the matrix are

$$(P_w)_{ij} = \delta_{ij} - \frac{w_i w_j}{|w|^2},$$

where δ_{ij} is the Kronecker delta function. Note for x in M and ν the normal vector in $\mathbf{R}^3$ of M at x, P_ν projects vectors onto M at x, i.e., P_ν projects vectors onto the tangent plane of M at x. Now for X a vector field in $\mathbf{R}^3$ we define the differential operator $P_X \nabla$ by its components,

$$(P_X \nabla)_i = \sum_{j=1}^{3} \left(\delta_{ij} - \frac{X_i X_j}{|X|^2} \right) \partial_{x_j}.$$

Note this is just the projection matrix P_X multiplying the gradient vector operator in $\mathbf{R}^3$. In fact, given a real valued function u on $\mathbf{R}^3$,

$$(P_X \nabla) u = P_X \nabla u,$$

and given a vector field Y on $\mathbf{R}^3$,

$$P_X \nabla \cdot Y = \sum_{i=1}^{3} (P_X \nabla)_i Y_i.$$

We will constantly use this notation with the vector field $X = \nabla \psi$, which is parallel at each point in $\mathbf{R}^3$ to the normal vector of the level set surface of ψ that passes through that point. So given a point x in R^3, $P_{\nabla \psi}$ projects vectors onto the level set surface of ψ passing through x. Therefore, if $x \in M$, $P_{\nabla\psi}$ will project vectors onto M at x. This is very useful for putting vector fields onto surfaces. Note especially that $P_{\nabla\psi}\nabla u$, evaluated on M, is the projection of the gradient vector ∇u in $\mathbf{R}^3$ onto M. This turns out to be equivalent to the surface gradient of u on M. Similarly, $P_{\nabla\psi}\nabla \cdot X$, evaluated on M, is equivalent to the surface divergence of X on M. We now present a few useful properties of this operator.

Proposition 1 *Let v, w, z be vectors, X a vector field, and u a real valued function, all in $\mathbf{R}^3$. Also let e_i denote the ith vector of the standard orthonormal basis of $\mathbf{R}^3$. Then we have the following identities:*

(a) $P_w v \cdot z = v \cdot P_w z = P_w v \cdot P_w z$.

(b) $(P_X \nabla)_i u = \nabla u \cdot P_X e_i$.

(c) $P_{\nabla u}\nabla \cdot (P_{\nabla u} X) = \nabla \cdot (P_{\nabla u} X |\nabla u|)\frac{1}{|\nabla u|}$.

3.1 Projecting $\mathbf{R}^2$ Equations onto Surfaces

In the course of studying the motion of curves on surfaces, we will need to study partial differential equations on surfaces. Usually, from work already done using the original level set method, we already know the form of the partial differential equation corresponding to the same type of motion for curves in $\mathbf{R}^2$. Thus one easy way to get the correct evolution equation on the surface would be to change the equation for curves in $\mathbf{R}^2$ accordingly, i.e., projecting it onto the surface, hopefully preserving its important properties.

Given a point x on M, we will project the form of the equation onto the surface at this point. Let ν be the normal vector of M at x and let $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$ be an orthonormal basis of $\mathbf{R}^3$ with $e_3 = \nu$. Also let $\tilde{\partial}_i$ be the derivative corresponding to $\tilde{e}_i$, $i = 1, 2, 3$. We can then write the partial differential equation on M at x by treating the tangent plane at x as $\mathbf{R}^2$, where the form of the equation is known. This means we will put all quantities in the $\mathbf{R}^2$ equation onto the tangent plane at x. This just involves changing those quantities to fit the new frame $\tilde{e}_1$ and $\tilde{e}_2$. Note this will especially involve the surface gradient vector operator defined by

$$\nabla^S u = \tilde{\partial}_1 u \tilde{e}_1 + \tilde{\partial}_2 u \tilde{e}_2,$$

for u a function on M and

$$\nabla^S \cdot X = \langle \tilde{\partial}_1 X, e_1 \rangle + \langle \tilde{\partial}_2 X, e_2 \rangle,$$

for X a vector field on M. For example, on M and at x, the Laplacian of a function u takes the form $\tilde{\partial}_1\tilde{\partial}_1 u + \tilde{\partial}_2\tilde{\partial}_2 u$, which can be written as $\nabla^S \cdot \nabla^S u$.

So in this case, ∇ is simply replaced by ∇^S to get from the $\mathbf{R}^2$ Laplacian to the surface Laplacian. We will use this procedure to project other partial differential equations in $\mathbf{R}^2$ onto surfaces. Assuming that the important properties of the equations are preserved during this transition, this is a quick way to get the evolution equation we want on M.

The main replacement when projecting $\mathbf{R}^2$ partial differential equations onto surfaces is, as we have seen, changing ∇ to ∇^S. The connection between the surface gradient ∇^S and our previous operator $P_{\nabla\psi}\nabla$ is given by

Proposition 2 *We have the following properties:*

(a) For u a real valued function in $\mathbf{R}^3$,

$$\nabla^S u = P_{\nabla\psi}\nabla u,$$

on M, where $\nabla^S u$ means the surface gradient applied to the restriction of u on M.

(b) For X a vector field in $\mathbf{R}^3$ which is tangent to M on M,

$$\nabla^S \cdot X = P_{\nabla\psi}\nabla \cdot X,$$

on M, where $\nabla^S \cdot X$ means the divergence of the restriction of X on M with respect to the surface gradient.

So ∇^S and $P_{\nabla\psi}\nabla$ are equivalent on M. The difference is that $P_{\nabla\psi}\nabla$ is easier to deal with numerically. Because of this, we will write all our equations using this form. We also note the importance of replacing the integral over $\mathbf{R}^2$, $\int_{\mathbf{R}^2} dx$, with the surface integral, $\int_S dA$, which is equivalent to $\int_{\mathbf{R}^3} \delta(\psi)|\nabla\psi| dx$, when projecting $\mathbf{R}^2$ equations onto surfaces. Finally, the $\mathbf{R}^2$ equation should be invariant under a rotation of frames in $\mathbf{R}^2$, otherwise, projecting it onto the surface may not be a well defined process.

Thus using all our tools, we can easily write all the geometric quantities of curves on surfaces in terms of our representation, i.e., in terms of ψ and ϕ. We will see examples of this later on when dealing with various geometrically based motions. Examples can also be seen in [3] for the case of unconstrained curves moving in $\mathbf{R}^3$. For more on the geometry of curves and surfaces and partial differential equations, see [9] and [24].

4 General Numerics

The main advantage of our approach lies in the effective numerical schemes that can be used to solve the partial differential equations associated to the motions. In general, we lay down a uniform grid in $\mathbf{R}^3$. In reality, not all the points in this grid need to be used since we only have to solve the equation in a neighborhood of the curves. This is called a local level set method (see, e.g., [22]), which we will discuss later. The level set functions ψ and ϕ are either given or created on this grid initially. Usually we use analytical expressions for ψ but we may just as well only give values for ψ at the points in our grid. The partial differential equation for ϕ is then solved by using appropriate finite difference schemes, which the uniform grid lets us easily create and implement. Also, note under our representation, the curve will not leave the surface and so the constraint that the curve lies on the surface is always satisfied. In fact, the curve location does not need to be determined for our computations but only when the curve is to be plotted. The plotter we use is the one used in [3] which divides the space into tetrahedra and uses linear approximations of ψ and ϕ in each tetrahedron to solve for the intersection of the zero level sets. The level set method representation also automatically takes care of any merging that may occur. The partial differential equations for the evolution are just solved in the same way until the end time regardless of whether merging has occurred or not. Of course, this ease in handling merging is one of the main reasons for using a level set based method. However, the method is still attractive in general because of its simplicity in using uniform grids and finite difference schemes. Finally, note that very complicated surfaces are easily taken care of since ψ is given as a set of data on grid points.

The numerical algorithm can easily be extended to higher dimensions but because of the uniform grid, operating in very high dimensions can be overly expensive. Computing in $\mathbf{R}^4$ is viable but above this, the method may not be the most efficient.

5 Introduction to Flows

In the following sections, we will use our format to generate and solve evolution equations for curves on surfaces moving under constant normal flow, geodesic curvature flow, Wulff flow, and flow under fixed enclosed surface area. In the process, we will develop other uses for these flows such as obtaining signed distance functions, geodesics, Wulff minimal curves, and Wulff shapes. These flows and their applications all come from flows and applications for curves in $\mathbf{R}^2$ (see [19]). Finally, we extend our results to allow the surface to also move. We will mostly derive the evolution equations in multiple ways, by projecting an $\mathbf{R}^2$ equation onto the surface, by finding a velocity field under which to move the curves, and sometimes through modified gradient descent minimizing an energy. The first way is quick and easy but the other ways are more geometric and intuitive.

As notation, we will use the term "surface" to denote the types of surfaces generated by zero level sets of level set functions and "curve" to denote the types of curves generated by the intersections of zero level sets of two level set functions. Note thus a "curve" or "surface" may actually be a collection of curves or surfaces.

6 Flow Under A Given Velocity Field

We first consider the simple problem of moving a curve on a surface by a given and fixed velocity field tangent to the surface. This can later be used on more general motions by looking at more general forms of velocity fields. The first step is to extend all our quantities to $\mathbf{R}^3$, creating ψ from the surface, an initial ϕ from the curve, and v from the velocity field, unless these are already given initially. There are various numerical methods that can do this, e.g., see [5]. The evolution for ϕ then becomes

$$\phi_t + P_{\nabla\psi} v \cdot \nabla\phi = 0,$$

which means we are moving the level sets of ϕ in $\mathbf{R}^3$ under the velocity field $P_{\nabla\psi} v$. The projection matrix in front of v keeps each level set of ψ independent from the others so that the flow on one level set of ψ will not affect or be affected by the flow on the others. Note on the surface we are interested in, $P_{\nabla\psi} v = v$. So under this velocity field, for a given level set surface of ψ, the level sets of ϕ on that surface will move according to the velocity field projected onto that surface and, especially, the zero level set of ϕ on the zero level set surface of ψ, i.e. the curves on M, will move according to v on M. This means the evolution equation gives the flow on M under the velocity field v, which is what we want.

A more detailed way to see this is to look at the surface $\{\psi = C_2\}$ and the curve on that surface $\gamma(s,t)$ obtained from the intersection of $\{\phi = C_1\}$, taken at time t, with the surface. We study the flow of γ on the surface according to a vector field tangent to the surface, $P_{\nabla\psi} v$. Considering general C_1 and C_2 allows us to obtain an evolution equation valid in all of $\mathbf{R}^3$. From the definition of γ, we have $\phi(\gamma, t) = C_1$ for all s and t. Therefore, taking a derivative with respect to t gives

$$\nabla\phi(\gamma, t) \cdot \gamma_t + \phi_t(\gamma, t) = 0.$$

The curve moves under the vector field $P_{\nabla\psi}v$ implies that $\gamma_t = P_{\nabla\psi(\gamma)}v(\gamma)$. Therefore, the form of the equation becomes

$$\phi_t(\gamma, t) + P_{\nabla\psi(\gamma)}v(\gamma) \cdot \nabla\phi(\gamma, t) = 0.$$

So, on the curve, we have

$$\phi_t + P_{\nabla\psi}v \cdot \nabla\phi = 0.$$

Since C_1 and C_2 are arbitrary, we then imply that this equation is valid in all of $\mathbf{R}^3$, giving us back the same equation as before.

Our process of projecting $\mathbf{R}^2$ evolution equations onto surfaces gives the same evolution equation. For curves in $\mathbf{R}^2$ and using the original level set method, the evolution equation is

$$\phi_t + v \cdot \nabla\phi = 0,$$

where v is a velocity field given in $\mathbf{R}^2$. We want to look at the form of this partial differential equation on the surface M, i.e., to project the equation onto the surface. Given x on M, note $\nabla\psi$ is normal to M at x and let $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$ be an orthonormal basis in $\mathbf{R}^3$ with $\tilde{e}_3 = \nabla\psi$ at x. This frame then allows us to define the surface gradient operator ∇^S at x as before, and so the equation on the surface will take the form

$$\phi_t + v \cdot \nabla^S\phi = 0,$$

or, in detail,

$$\phi_t + \langle v, \tilde{e}_1\rangle\tilde{\partial}_1\phi + \langle v, \tilde{e}_2\rangle\tilde{\partial}_2\phi = 0.$$

This can be rewritten in the usual format,

$$\phi_t + v \cdot P_{\nabla\psi}\nabla\phi = 0,$$

which is equivalent to what we obtained previously. So projecting the $\mathbf{R}^2$ equation on the surface also gives the correct evolution equation.

Higher dimensions can also be considered with the same evolution equation simply by taking ψ and ϕ functions in $\mathbf{R}^n$ and v a velocity field in $\mathbf{R}^n$.

The derived evolution equation is a partial differential equation of Hamilton Jacobi form and can be numerically solved using Total Variation Diminishing Runge-Kutta (TVD-RK) of third order in time (see [23]) and Hamilton Jacobi Weighted Essentially Non-Oscillatory method (WENO) of fifth order in space using the Godunov scheme[12]. The associated Courant-Friedrichs-Lewy (CFL) condition says that Δt, the time step, must be less than a constant times Δx, the spatial step, with the constant depending on the magnitude of v. Also, the singularity arising from $|\nabla\psi| = 0$ needs to be regularized. This can be accomplished, for example, by replacing $|\nabla\psi|$ with $\sqrt{\psi_{x_1}^2 + \psi_{x_2}^2 + \psi_{x_3}^2 + \epsilon^2}$, where ϵ is positive and very small, when it appears in a denominator.

The above process can then be used to derive evolution equations for more general flows. First, a valid velocity field v, which now may depend on ϕ and its derivatives, must be derived. This will depend on the type of flow being considered. Then the evolution equation will take the same form as above,

$$\phi_t + P_{\nabla\psi} v \cdot \nabla\phi = 0.$$

This equation will move the level sets of ϕ in $\mathbf{R}^3$ under the desired velocity field and thus will move the zero level set of ϕ on M according to the flow being considered. It is also valid in more space dimensions, where ψ and ϕ are real valued functions on $\mathbf{R}^n$ and the projection matrix is an n by n matrix. Note we cannot use the above discretization anymore for general v. The valid discretization of the equation will depend on the form of v, for example, if $-P_{\nabla\psi} v \cdot \nabla\phi$ is elliptic, then we can use central difference schemes. We will constantly use this velocity field process to derive and validate the evolution equations for our flows.

7 Constant Normal Flow

A difficult but important flow involves moving a curve in the outward normal direction at a constant speed C on the surface. This means at time t, the curve we are looking for is the set of points of distance Ct, measured on the surface, away from γ_0 in the outward direction. Note that moving inward corresponds to C being negative. For curves in $\mathbf{R}^2$, this flow has been used to find flame fronts (see [18]) and is an integral part of many other applications (see [19]).

We first use our approach for projecting $\mathbf{R}^2$ equations onto the surface to quickly generate the evolution equation. The corresponding evolution equation for curves in $\mathbf{R}^2$, using the original level set method, takes the form

$$\phi_t + C|\nabla\phi| = 0.$$

Once again, given x on M, let $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$ be an orthonormal basis in $\mathbf{R}^3$ with

$\tilde{e}_3 = \nabla\psi$ at x. This allows us to define ∇^S at x and so the evolution equation on the surface takes the form

$$\phi_t + C|\nabla^S \phi| = 0,$$

or, in detail,

$$\phi_t + C\sqrt{(\tilde{\partial}_1 \phi)^2 + (\tilde{\partial}_2 \phi)^2}.$$

This can then be rewritten as

$$\phi_t + C|P_{\nabla\psi}\nabla\phi| = 0,$$

which is the correct equation. We will, however, verify that it indeed moves a curve in the outward normal direction at speed C by rederiving it using the more intuitive velocity field approach.

In the velocity field approach, we want to calculate a velocity field v under which the level sets of ϕ, and especially the zero level set, will move in the correct manner. For fixed t, consider the surface $\{\psi = C_1\}$ and the curve generated by intersecting this surface with $\{\phi = C_2\}$, where C_1 and C_2 are constants. Note the case we are interested in is $C_1 = C_2 = 0$ but by considering arbitrary C_1 and C_2, we get a velocity field valid in all of $\mathbf{R}^3$ which can be used to evolve ϕ in $\mathbf{R}^3$. Now on this curve, v should be normal to the curve, have length C, and be tangent to the surface. Such a v will give the desired motion for the curve on the surface. From this, we deduce

$$v = C\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}.$$

Note, we could use vector cross products instead, since we are in $\mathbf{R}^3$, along with the identity

$$\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|} = \frac{\nabla\psi \times \nabla\phi}{|\nabla\psi \times \nabla\phi|},$$

to rewrite our expressions, but we will stick with the more general form. Also, if $C = 1$, note v is the outward normal of the curve on the surface. We will use this fact in many later computations.

Under such a velocity field, the evolution equation for ϕ takes the form

$$\phi_t + v \cdot \nabla\phi = 0,$$

since $P_{\nabla\psi}v = v$. Simplifying, we get

$$v \cdot \nabla\phi \quad = \quad C\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|} \cdot \nabla\phi$$

$$
\begin{aligned}
&= C\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}\cdot P_{\nabla\psi}\nabla\phi \\
&= C|P_{\nabla\psi}\nabla\phi|.
\end{aligned}
$$

So the evolution equation for moving curves on surfaces by constant normal flow is

$$\phi_t + C|P_{\nabla\psi}\nabla\phi| = 0,$$

or, using vector cross products,

$$\phi_t + C\frac{|\nabla\psi\times\nabla\phi|}{|\nabla\psi|} = 0.$$

This is the same equation as we obtained previously by projecting the $\mathbf{R}^2$ equation onto the surface.

Note if we have a partial differential equation of the above form, even with C depending on ϕ and its derivatives, then we say the curve will move by speed C in the normal direction. In fact, all evolution equations for flows can be written in this form. This is because given a velocity field v tangent to the level set surfaces of ψ, then at each point x, we can decompose v in terms of $\dfrac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}$ and the vectors perpendicular to it. Thus $v\cdot P_{\nabla\psi}\nabla\phi$ is equal to $C|P_{\nabla\psi}\nabla\phi|$, for some C, and so moving under the vector field v is the same as moving in the normal direction by speed C.

The partial differential equation we derived with C constant is of Hamilton-Jacobi form and so we discretize it using Hamilton-Jacobi WENO of fifth order along with Local Lax-Friedrichs (LLF) in space and TVD-RK of third order in time. To satisfy the CFL condition, Δt needs to be smaller than a constant times Δx. The term $|P_{\nabla\psi}\nabla\phi|$ is also regularized to remove the singularity arising from $|\nabla\psi| = 0$.

In Table 1, we see that the our discretization is second order accurate before merging occurs. This was checked for a circle moving on a sphere by unit normal flow, i.e., flow in the normal direction at unit speed. The whole algorithm, including the second order accurate plotter, is included in this test. By using a higher order plotter, we should be able to get higher order accuracy.

In Figure 1, we show a curve moving over two mountains by unit normal flow. The curve breaks into two pieces, with each piece moving up each mountain. In Figure 2, we show a curve moving on a volcano. The curve starts outside the volcano and goes up and into the core. In Figure 3, we

grid size	error	order
$32 \times 32 \times 32$	0.00561706	
$64 \times 64 \times 64$	0.001703	1.7217
$128 \times 128 \times 128$	0.000447336	1.9286
$256 \times 256 \times 256$	0.000115601	1.9522

Table 1: This is the order of accuracy analysis for unit normal flow. The example considered was a circle moving on a sphere. The results show second order accuracy. Note the accuracy of the whole algorithm, including the second order interpolation scheme used to plot the curve, is tested.

show a curve on a two holed torus. The curve moves across the two holed torus, breaking and merging multiple times. Thus we see that the motion of the curve by constant normal flow on complicated surfaces, even when merging occurs, is easily handled by our algorithm. Finally, we show in Figure 4 flow in the normal direction by a non-constant speed. For each point on the curve, this speed is equal to a function β evaluated at the outward normal vector of the curve. The function we chose is $\beta(x) = |x_1|+|x_2|+|x_3|$, which is related to crystal shapes. Note the squarish aspect of the growing curve.

We can also study the behavior of the flow in higher dimensions. The evolution equation

$$\phi_t + C|P_{\nabla\psi}\nabla\phi| = 0,$$

is still valid with ψ and ϕ real valued functions over the space $\mathbf{R}^n$. When we drop a dimension and flow points on curves, the evolution equation takes the form

$$\phi_t + C\frac{|\psi_y\phi_x - \psi_x\phi_y|}{|\nabla\psi|} = 0.$$

Note the numerator of the second term is the absolute value of the Jacobian of ψ and ϕ. Thus it is possible, for example, to do constant normal flow of a two dimensional surface constrained on a hypersurface in $\mathbf{R}^4$ or constant normal flow of points constrained on a curve.

8 Signed Distance Function

In an extension of constant normal flow, we wish to find the signed distance of each point on a surface M away from a curve γ confined to M. The signed distance on M of a point away from a curve is the minimal distance measured on the surface, with a negative sign if the point lies inside the curve, of that point to the points of the curve. Obtaining signed distance allows for construction of geodesics and can be used for path planning on manifolds. It can also reveal important information about a surface's geometry. We solve the problem by setting ψ to have M as its zero level set and trying to find a real valued function d in $\mathbf{R}^3$ such that given a point $x \in M$, $d(x)$ gives the signed distance of x away from γ. d is thus uniquely defined on M, though not in $\mathbf{R}^3$, and we call d a signed distance function of γ on M. Note since on M, $d = 0$ only at γ, we have, as before, that γ is the intersection between the zero level sets of ψ and d. Also note this problem is different from the ones we have previously studied because we are looking for a function defined on all of M rather than one defined near γ only.

For γ a curve in $\mathbf{R}^2$, finding the signed distance function using the original level set method is accomplished by introducing a time element and creating a partial differential equation whose steady state solution values give signed distance. Starting with a level set function ϕ initially having γ as its zero level set and negative inside γ, the equation

$$\phi_t + \operatorname{sgn}(\phi(x,0))(|\nabla\phi| - 1) = 0,$$

will give signed distance as its steady state viscosity solution. The signum function keeps $\phi = 0$ on γ for all time and the rest of the equation tries to force $|\nabla\phi| = 1$, making the steady state solution a signed distance function. We will derive the correct evolution equation on the surface in two ways, by looking at this equation written on the surface and by using the philosophy behind this equation to recreate it on the surface. In projecting the equation on the surface M, we fix x on M and $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$ an orthonormal basis of $\mathbf{R}^3$ with $\tilde{e}_3 = \nabla\psi$ at x. Then ∇^S is defined at x and the equation takes the form

$$\phi_t + \operatorname{sgn}(\phi(x,0))(|\nabla^S\phi| - 1) = 0,$$

or, in detail,

$$\phi_t + \operatorname{sgn}(\phi(x,0))\left(\sqrt{(\tilde{\partial}_1\phi)^2 + (\tilde{\partial}_2\phi)^2} - 1\right) = 0.$$

This can then be written as

$$\phi_t + \operatorname{sgn}(\phi(x,0))(|P_{\nabla\psi}\nabla\phi| - 1) = 0.$$

This is the correct equation but we will rederive it in the more detailed and intuitive way by following the basic philosophy behind the $\mathbf{R}^2$ equation.

To find d on M, we can imitate the method for curves in $\mathbf{R}^2$, i.e., introduce a time element and create a partial differential equation that has d as its steady state solution on M. Let ϕ initially be such that the intersection of its zero level set with M is γ, with ϕ negative inside γ. If ϕ is a signed distance function on the surface, then it must satisfy $|P_{\nabla\psi}\nabla\phi| = 1$, i.e., $|\nabla^S\phi| = 1$, on M. So we wish to create an evolution equation such that on M, the steady state solution satisfies this property while keeping the zero level set of ϕ fixed at its original position. One such candidate is

$$\phi_t + \mathrm{sgn}(\phi(x,0))(|P_{\nabla\psi}\nabla\phi| - 1) = 0,$$

which is the same as the equation we derived previously. The steady state viscosity solution on M of this evolution equation will be d. Note that the evolution equation is solved in all of space but steady state may sometimes only be achieved at M.

This equation is also valid in space dimensions other than three and, in fact, the equation for distance on curves in $\mathbf{R}^2$ takes the form,

$$\phi_t + \mathrm{sgn}(\phi(x,0))\left(\frac{\sqrt{|\nabla\psi|^2|\nabla\phi|^2 - (\nabla\psi\cdot\nabla\phi)^2}}{|\nabla\psi|} - 1\right) = 0.$$

In $\mathbf{R}^3$, with vector cross products, the equation can be written as

$$\phi_t + \mathrm{sgn}(\phi(x,0))\left(\frac{|\nabla\psi\times\nabla\phi|}{|\nabla\psi|} - 1\right) = 0.$$

The signed distance evolution equation is of Hamilton-Jacobi form and we solve it using Hamilton-Jacobi fifth order WENO-LLF in space and third order TVD-RK in time. We also replace the signum function by a smooth version (see [22]) and regularize to remove the singularity occurring at $|\nabla\psi| = 0$. To satisfy the CFL condition, Δt needs to be less than some constant multiple of Δx.

In Table 2, we see that the algorithm for finding signed distance functions is first order accurate. This is because the curve is moved slightly during iterations of the method. Theoretically this should not happen but because of the numerical signum function and because of the grid, we see a small shift. Table 3 shows that the method has a high order of accuracy when looking at the quantity $||P_{\nabla\psi}\nabla d| - 1|$. So altogether, this means the numerically computed signed distance function is a high order signed distance function for a slightly perturbed curve.

In Figure 5, we show a curve on a volcano along with the other contours of the distance function. Note the contours are well-spaced. In Figure 6,

grid size	error	order
$32 \times 32 \times 32$	0.020416	
$64 \times 64 \times 64$	0.0106933	0.9330
$128 \times 128 \times 128$	0.00526517	1.0222
$256 \times 256 \times 256$	0.00261509	1.0096

Table 2: This is the order of accuracy analysis for the signed distance function. The curve from which distance was measured was a circle and the surface was a sphere. The results show first order accuracy.

grid size	error	order
$32 \times 32 \times 32$	0.000527837	
$64 \times 64 \times 64$	$4.73339e-06$	6.8011
$128 \times 128 \times 128$	$4.75352e-08$	6.6377

Table 3: This is the order of accuracy analysis for the signed distance function, measuring $||P_{\nabla\psi}\nabla\phi| - 1|$. The curve and surface were the same as in Table 2. The results show high order accuracy.

we show a curve on a torus along with the other contours of the distance function. Once again, the contours are well-spaced. Thus we see that the signed distance function may be used to create grids on surfaces.

8.1 Keeping Level Set Functions Well Behaved During Flows

One important application for signed distance functions is the role it can play in keeping the level sets of a level set function well behaved on the surface during a flow. This helps reduce numerical inaccuracies that may appear from an overly steep or flat level set function. For curves in $\mathbf{R}^2$, this is accomplished by making the level set function into the signed distance function to its zero level set at each time step of the flow. We can do the same for level set functions on surfaces. Note since ψ can be chosen to be well behaved or made so by replacing it by the signed distance function in $\mathbf{R}^3$ to its zero level set surface, we will only study the effect that different ϕ have and assume ψ is already well behaved.

Certain types of flows may result in a bunching of level sets, where the function restricted on the surface is steep, or a spreading out of level sets, where the function is almost flat (see, e.g., [8]). Numerically, this is undesirable and may introduce large errors in the finite difference approximations. Further errors may also be introduced when interpolating to find the location of the curve, especially if the function on the surface is almost flat. Finally, flatness may cause singularities if we need to divide by the magnitude of the surface gradient, as is done in geodesic curvature flow. But if the level set function is constrained to be a signed distance function, then the surface gradient will have a magnitude of value 1 everywhere except at kinks. This makes the level set function well behaved, especially and most importantly near the curve. When we consider a particular flow, i.e., solve an evolution equation for ϕ, the signed distance constraint is enforced usually by iterating the corresponding partial differential equation a few times after every time step of the flow. We only need to iterate a few times since usually only the information around the curve affects its motion and so we only need to enforce signed distance in a neighborhood of the curve. Note the zero level set of ϕ theoretically remains fixed when iterating to a signed distance function and so this process should not affect the flow of the curve on the surface.

Another way computations may break down is when the level sets of ϕ become tangent to the surface. Note this has nothing to do with the level sets of ϕ on the surface, where the signed distance constraint makes ϕ well behaved, but with the behavior of the level sets of ϕ off the surface. For example, $\phi = x_2 - Cx_1$ is already a signed distance function on the surface $x_2 = 0$ for all $C > 0$ but as C tends to zero, the level sets of ϕ become tangent to the surface. Thus the surface gradient becomes zero, and especially numerically inaccurate, and also any small perturbation of ϕ may greatly shift the location of the curve or even introduce spurious curve parts. To prevent this from happening, we want to make the level set surfaces of ϕ perpendicular to M on M, especially near the curve. This can be accomplished by iterating a few steps of the partial differential equation

$$\phi_t + \mathrm{sgn}(\psi)\frac{\nabla\psi}{|\nabla\psi|} \cdot \nabla\phi = 0,$$

at each step of the flow after signed distance is enforced. Note this equation forces $\frac{\nabla\psi}{|\nabla\psi|} \cdot \nabla\phi = 0$ at steady state so that the level sets of ϕ will be perpendicular to the surface. It also keeps the level sets of ϕ fixed on the surface so that signed distance on the surface is preserved. The fast marching method[27] might also be used in place of the above partial differential equation.

The partial differential equation is of Hamilton Jacobi form and we solve it using fifth order WENO-Godunov in space and third order TVD-RK in time. The CFL condition says Δt needs to be less than a constant multiple of Δx.

8.2 Geodesics

The signed distance function can also be used to compute geodesics on surfaces from points to curves. This means given a curve γ on M, we want to find the shortest path on M from any point on M to γ. This can be accomplished using a signed distance function d of γ on M. In fact, the shortest path is simply the part of the integral curve of the vector field $-dP_{\nabla\psi}\nabla d$ drawn from the chosen point to γ. This simply means the shortest path starts at the chosen point and follows the steepest descent direction of d on M with speed d. The speed is thus zero at γ and so we follow the integral curve until convergence. The integral curves, $y(s)$, of the vector field are curves in $\mathbf{R}^3$ and can be computed according to the ordinary differential equation

$$\dot{y}(s) = -d(y(s))P_{\nabla\psi(y(s))}\nabla d(y(s)).$$

For a chosen point x on M, the geodesic from x to γ is thus found by solving the above ordinary differential equation with initial condition $y(0) = x$. This can be done numerically using a Runge-Kutta scheme.

When we want the geodesic between two points a and b, we can first get a $\tilde{d}$ which gives the signed distance function to a small curve surrounding the point a, i.e., approximating a. Then $d = \tilde{d} + \tilde{d}(a)$ is an approximate signed distance function to a on the surface which is exact when the small curve approximating a is at a uniform distance away from a. Using this d in the ordinary differential equation above along with the initial condition $x(0) = b$ allows us to calculate an approximate geodesic. Or, we can require that signed distance be given initially in a neighborhood of the point a and then solve for a signed distance function d to a on M by iterating the corresponding evolution equation but only outside the neighborhood of initial given values, the given values being fixed. We can then use this d along with $x(0) = b$ for our initial condition to calculate geodesics.

A drawback of this signed distance function method for geodesics is that when there are two or more geodesics, we have almost no control over which one or whether any will be chosen. Also note numerical approximations of the geodesics are not forced to lie on the surface, unlike what happens in our basic representation.

In Figure 7, we show a curve in the core of a volcano and the geodesics from certain points to that curve. The geodesics travel up the volcano and down into the core to reach the curve. In Figure 8, we show a curve wrapped around a torus and the geodesics from certain points to that curve. The geodesics travel across the torus and around the hole to reach the curve. Thus we see how the signed distance function can be used to find geodesics from points to curves on surfaces.

9 Geodesic Curvature Flow

One of the most geometrically important motions of curves on surfaces is geodesic curvature flow. This motion is important as curve shortening and can be used to get geodesic curves of surfaces and even to generate minimal surfaces on hypersurfaces of $\mathbf{R}^4$. Constructing the correct evolution equation can be accomplished in a few ways, all of which lead to the same equation. The first way is by projecting the corresponding evolution equation for curves in $\mathbf{R}^2$ onto the surface. The second way involves finding the curvature times normal vectors of the curve in $\mathbf{R}^3$ and projecting them onto the surface. This gives the velocity vectors with which to move the curve. The third way is studying modified gradient descent minimizing the length of the curve constrained on the surface. The fact that these are all equivalent means moving a curve by curvature is a minimization of the length of the curve.

First Method: Projecting $\mathbf{R}^2$ Equation Onto Surface.

We note that the corresponding evolution equation in $\mathbf{R}^2$ takes the form

$$\phi_t = \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right)|\nabla\phi|,$$

where $-\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right)$ is the mean curvature of the curve. Given x on M and an orthonormal basis $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$ in $\mathbf{R}^3$ with $\tilde{e}_3 = \nabla\psi$ at x, we can define ∇^S at x. So the partial differential equation put onto M at x takes the form

$$\phi_t = \nabla^S \cdot \left(\frac{\nabla^S\phi}{|\nabla^S\phi|}\right)|\nabla^S\phi|,$$

where, in fact, $-\nabla^S \cdot \left(\frac{\nabla^S\phi}{|\nabla^S\phi|}\right)$ is the geodesic curvature of the curve. In detail, the equation is

$$\phi_t = \left(\tilde{\partial}_1\left(\frac{\tilde{\partial}_1\phi}{\sqrt{(\tilde{\partial}_1\phi)^2+(\tilde{\partial}_2\phi)^2}}\right)+\tilde{\partial}_2\left(\frac{\tilde{\partial}_2\phi}{\sqrt{(\tilde{\partial}_1\phi)^2+(\tilde{\partial}_2\phi)^2}}\right)\right)\sqrt{(\tilde{\partial}_1\phi)^2+(\tilde{\partial}_2\phi)^2},$$

with

$$-\tilde{\partial}_1 \left(\frac{\tilde{\partial}_1 \phi}{\sqrt{(\tilde{\partial}_1 \phi)^2 + (\tilde{\partial}_2 \phi)^2}} \right) - \tilde{\partial}_2 \left(\frac{\tilde{\partial}_2 \phi}{\sqrt{(\tilde{\partial}_1 \phi)^2 + (\tilde{\partial}_2 \phi)^2}} \right),$$

the geodesic curvature. We then rewrite all this as

$$\phi_t = \nabla \cdot \left(\frac{P_{\nabla\psi} \nabla \phi}{|P_{\nabla\psi} \nabla \phi|} |\nabla \psi| \right) \frac{|P_{\nabla\psi} \nabla \phi|}{|\nabla \psi|},$$

with

$$-\nabla \cdot \left(\frac{P_{\nabla\psi} \nabla \phi}{|P_{\nabla\psi} \nabla \phi|} |\nabla \psi| \right) \frac{1}{|\nabla \psi|},$$

the geodesic curvature. This equation translates to moving a curve on M in the normal direction by geodesic curvature, which is what we want.

Second Method: Projection of Free Space Curvature Times Normal Vector.

Consider the surface $\{\psi = C_1\}$ and the curve generated by intersecting this surface with $\{\phi = C_2\}$, C_1 and C_2 constants. This means $\nabla\psi \times \nabla\phi$ taken on the curve is parallel to the tangent vector of the curve. So the tangent vector of the curve can be written as $T = \frac{\nabla\psi \times \nabla\phi}{|\nabla\psi \times \nabla\phi|}$. Now the curvature times normal vector of the curve in $\mathbf{R}^3$, κN, is the change in the tangent vector along the curve. Therefore, using directional derivatives, we get

$$\kappa N = (\nabla T_1 \cdot T, \nabla T_2 \cdot T, \nabla T_3 \cdot T),$$

where $T = (T_1, T_2, T_3)$ (see [3]). We now project this onto the surface to get $P_{\nabla\psi} \kappa N$. Using this as our velocity field leads to the evolution equation

$$\phi_t = -P_{\nabla\psi} \kappa N \cdot \nabla \phi.$$

This equation also gives geodesic curvature motion of curves on surfaces.

Third Method: Energy Minimization.

We consider the energy

$$E(\phi) = \int_{\mathbf{R}^3} \delta(\phi)\delta(\psi) |P_{\nabla\psi} \nabla \phi| |\nabla \psi| dx,$$

which gives the length of the curve represented by the intersection between the zero level sets of ϕ and ψ.

Proposition 3 *The Euler-Lagrange equation of this energy is*

$$0 = -\nabla \cdot \left(\frac{P_{\nabla\psi} \nabla \phi}{|P_{\nabla\psi} \nabla \phi|} |\nabla \psi| \right) \delta(\psi)\delta(\phi).$$

Replacing $\delta(\psi)\delta(\phi)$, which we treat as smoothed out delta functions, by $\frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|}$ in our gradient descent gives us the evolution equation

$$\phi_t = \nabla \cdot \left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|} |\nabla\psi| \right) \frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|},$$

which is exactly what we got using the first method. Note because everything is in $\mathbf{R}^3$, we can also write this equation as

$$\phi_t = \nabla \cdot \left(\frac{(\nabla\psi \times \nabla\phi) \times \nabla\psi}{|\nabla\psi \times \nabla\phi|} \right) \frac{|\nabla\psi \times \nabla\phi|}{|\nabla\psi|^2}.$$

We made the above replacement because it matches the equation derived using the first method. Also, according to standard level set practice[28], we see that $\delta(\psi)\delta(\phi)$ should be replaced by a quantity that yields a gradient descent algorithm for minimizing the enclosed surface area of γ with inward normal flow at unit speed. The enclosed surface area for our curve on M is given by $\int_{\mathbf{R}^3} H(-\phi)\delta(\psi)|\nabla\psi|dx$, where ψ is static and H is the one dimensional Heaviside function. So the Euler-Lagrange equation is

$$0 = -\delta(\phi)\delta(\psi)|\nabla\psi|,$$

and gradient descent with the above replacement gives

$$\phi_t - \frac{|\nabla\psi \times \nabla\phi|}{|\nabla\psi|} = 0,$$

which is inward normal flow at unit speed.

Equivalence.

We will now show that the evolution equations for the first and third methods are equivalent to the evolution equation for the second method. The main result is the following identity,

Proposition 4

$$\nabla \cdot (T \times \nabla\psi) = \kappa N \cdot (\nabla\psi \times T),$$

where

$$T = \frac{\nabla\psi \times \nabla\phi}{|\nabla\psi \times \nabla\phi|}.$$

Using this to expand the right hand side of the evolution equation in the second method, we get

$$\begin{aligned}
-P_{\nabla\psi}\kappa N\cdot\nabla\phi &= -P_{\nabla\psi}\nabla\phi\cdot\kappa N \\
&= -\kappa N\cdot\left(\frac{|\nabla\psi|^2\nabla\phi-(\nabla\phi\cdot\nabla\psi)\nabla\psi}{|\nabla\psi|^2}\right) \\
&= -\kappa N\cdot\left(\frac{(\nabla\psi\times\nabla\phi)\times\nabla\psi}{|\nabla\psi|^2}\right) \\
&= -\kappa N\cdot\left(\frac{(\nabla\psi\times\nabla\phi)\times\nabla\psi}{|\nabla\psi\times\nabla\phi|}\right)\frac{|\nabla\psi\times\nabla\phi|}{|\nabla\psi|^2} \\
&= -\kappa N\cdot(\nabla\psi\times T)\frac{|\nabla\psi\times\nabla\phi|}{|\nabla\psi|^2} \\
&= -\nabla\cdot(T\times\nabla\psi)\frac{|\nabla\psi\times\nabla\phi|}{|\nabla\psi|^2} \\
&= \nabla\cdot\left(\frac{(\nabla\psi\times\nabla\phi)\times\nabla\psi}{|\nabla\psi\times\nabla\phi|}\right)\frac{|\nabla\psi\times\nabla\phi|}{|\nabla\psi|^2},
\end{aligned}$$

which is the right hand side of the evolution equation in the third method. This means all the evolution equations are equivalent. We summarize this result in the following,

Proposition 5

$$P_{\nabla\psi}\kappa N\cdot\nabla\phi = -\nabla\cdot\left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}|\nabla\psi|\right)\frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|}.$$

The resulting evolution equation is valid in any dimension and thus it is possible to study, for example, minimal surfaces on hypersurfaces in $\mathbf{R}^4$.

We also have the property

Proposition 6 *The evolution equation is degenerate second order parabolic.*

Thus we use central differencing in space with third order TVD-RK in time to numerically solve the evolution equation. We also regularize the equation to remove the singularities arising at $|\nabla\psi| = 0$ and $|P_{\nabla\psi}\nabla\phi| = 0$. To satisfy the CFL condition, Δt needs to be less than a constant multiple of Δx^2.

In Table 4, we see that the method is second order accurate. This result was obtained by studying a circle moving by geodesic curvature flow on a sphere.

In Figure 10, we show a curve moving on two mountains. The curve needs to move over the mountains before it can shrink to a point and disappear.

grid size	error	order
$32 \times 32 \times 32$	0.00203138	
$64 \times 64 \times 64$	0.000540219	1.9108
$128 \times 128 \times 128$	0.00014037	1.9443

Table 4: This is the order of accuracy analysis for geodesic curvature flow of a circle on a sphere. The result shows second order accuracy.

In Figure 11, we show a curve moving on a bent plane. Note the surface has a kink in it. The curve navigates over this without any problems. In Figure 12, we show a curve on a cylinder. The curve evolves and wraps tightly around the cylinder, forming a circle. This is a geodesic curve for the surface.

10 Wulff Flow

We now consider the problem of evolving a curve by Wulff flow on a surface. This means minimizing the Wulff energy $\int_\gamma \beta(\nu)ds$, where $\beta\colon S^2 \to (0,\infty)$ and ν is the unit normal of γ lying on the surface M. We will only study the case of convex Wulff energies (see Proposition 8 below). Also, note when $\beta \equiv 1$, the Wulff energy is the length of the curve. Thus Wulff flow is a certain generalization of geodesic curvature flow which is related, for curves in $\mathbf{R}^2$ and surfaces in $\mathbf{R}^3$, to crystal shapes (see [25]). We make a homogeneous degree one extension of β to $\mathbf{R}^3$ and then rewrite the Wulff energy using our usual representation to get

$$E(\phi) = \int_{\mathbf{R}^3} \beta\left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}\right)\delta(\psi)\delta(\phi)|\nabla\psi \times \nabla\phi|dx.$$

Proposition 7 *The Euler-Lagrange equation of this energy is*

$$0 = -\nabla \cdot (P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi)|\nabla\psi|)\delta(\psi)\delta(\phi).$$

So the evolution equation, enacting the usual replacement to the delta functions, can be written as

$$\phi_t = \nabla \cdot (P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi)|\nabla\psi|)\frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|}.$$

This moves a curve by Wulff flow on a surface. The evolution equation also satisfies

Proposition 8 *The evolution equation is degenerate second order parabolic if*

$$\nabla^2 \beta \left(\frac{P_{\nabla\psi} \nabla\phi}{|P_{\nabla\psi} \nabla\phi|} \right)$$

is nonnegative definite.

To see that the equation we derived is the same as when projecting the $\mathbf{R}^2$ evolution equation on the surface, we note that for curves in $\mathbf{R}^2$, Wulff flow is given by

$$\phi_t = \nabla \cdot \nabla\beta(\nabla\phi)|\nabla\phi|.$$

So given x on M and $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$ an orthonormal basis for $\mathbf{R}^3$ with $\tilde{e}_3 = \nabla\psi$ at x, we can define ∇^S at x and, thus, the equation on the surface takes the form

$$\phi_t = \nabla^S \cdot \nabla^S \beta(\nabla^S \phi)|\nabla^S \phi|,$$

or, in detail,

$$\phi_t = (\tilde{\partial}_1(\tilde{\partial}_1\beta(\tilde{\partial}_1\phi\tilde{e}_1 + \tilde{\partial}_2\phi\tilde{e}_2)) + \tilde{\partial}_2(\tilde{\partial}_2\beta(\tilde{\partial}_1\phi\tilde{e}_1 + \tilde{\partial}_2\phi\tilde{e}_2)))\sqrt{(\tilde{\partial}_1\phi)^2 + (\tilde{\partial}_2\phi)^2}.$$

This can then be written as

$$\phi_t = P_{\nabla\psi}\nabla \cdot (P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi))|P_{\nabla\psi}\nabla\phi|,$$

which is equivalent to the equation derived using energy minimization. Note higher dimensions can also be considered by using the same equation.

We numerically solve the evolution equation using second order central differencing on all spatial derivatives. The time derivative is discretized using TVD-RK of third order. The equation is also regularized at the singularities that occur at $|\nabla\psi| = 0$ and $|P_{\nabla\psi}\nabla\phi| = 0$.

In Figure 13, we show a curve moving on the bottom of a paraboloid. The Wulff energy we used was with a smoothed out version of $\beta(x) = |x_1| + |x_2| + |x_3|$. Its exact form is

$$\beta(x) = \sqrt{x_1^2 + \epsilon^2} + \sqrt{x_2^2 + \epsilon^2} + \sqrt{x_3^2 + \epsilon^2},$$

with $\epsilon = 0.1$. Thus the curve develops a squarish shape while shrinking. In Figure 14, we show a curve moving on a bent plane. The curve once again develops a squarish shape and we see that computing over kinks in the surface, which cause kinks in the curve, is not a problem for our algorithm.

10.1 Wulff Minimal Curves

The evolution equation for Wulff flow can be slightly altered to give a method for finding Wulff minimal curves on surfaces. Given a set of points on M, we want to find the curve on the surface that passes through these points with the minimum Wulff energy. We will call the given points boundary points. Thus Wulff minimal curves are the one dimension version of Wulff minimal surfaces (see [6]). This problem may be useful in the study of properties of crystals on surfaces.

For $\beta \equiv 1$, we are searching for the curve on the surface of minimal length that passes through the boundary points. When M is a plane, this curve is the boundary of the convex hull of those points. For general surfaces in $\mathbf{R}^3$, the curve will be piecewise geodesics. We find the solution to this problem by solving to steady state the zero level set of ϕ on M in the evolution equation

$$\phi_t = -\mu P_{\nabla\psi}\kappa N \cdot \nabla\phi,$$

where μ is smooth with $\mu(x) = 0$ if x is a boundary point and $\mu(x) = 1$ outside a small neighborhood of the boundary points. The initial curve, γ_0, is chosen to pass through the boundary points. This approach is an extension of the one used in [6].

In the case where the boundary points consist of just two points, a and b, and γ_0 is chosen carefully, we get the geodesic between a and b. However, if the initial curve γ_0 is not chosen to be topologically equivalent to the answer, parts of it may merge at later time and not evolve into what we want.

For general β, the evolution equation we are interested in is

$$\phi_t = \mu\nabla \cdot (P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi)|\nabla\psi|)\frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|}.$$

We note that when M is a plane, the solution is usually still the boundary of the convex hull of the boundary points. Numerically, the evolution equation is solved using the same finite difference schemes as in the Wulff flow case. The μ is just treated as a coefficient in front of the rest of the equation. For higher dimensions, the same evolution equation holds since it is already in its general form. Thus it is possible to study Wulff minimal surfaces constrained on hypersurfaces in $\mathbf{R}^4$.

Creating a γ_0, or the corresponding initial level set function, that passes through the boundary points may not be easy but sometimes we can simply take as γ_0 any curve on the surface that encompasses all the boundary points. Thus the initial ϕ is easy to construct. When we run the evolution equation in time, the curve will shrink and sometimes end up going through all the boundary points. Other more robust interpolating methods can also be used (see [29]).

11 Fixed Enclosed Surface Area

We now consider the problem of evolving under a certain motion a curve γ on a surface M with the constraint that the surface area of the part of the surface enclosed by γ is fixed in time. For curves in $\mathbf{R}^2$, this can be used to study bubbles or other fluids that conserve enclosed area or volume (see [13] and [30]). We will mainly look at geodesic curvature flow and will occasionally comment on more general motions. In this case, the energy involving the length of the curve coupled with the constraint gives us the energy we are interested in. The constraint can be translated as the condition that $\int_{\mathbf{R}^3} H(-\phi)\delta(\psi)|\nabla\psi|dx$ remains constant throughout time. Note this means if γ is a collection of curves, then the total enclosed area is fixed, not the area enclosed by each curve in the collection. So the new energy to consider is

$$E(\phi) = \int_{\mathbf{R}^3} \delta(\psi)\delta(\phi)|P_{\nabla\psi}\nabla\phi|dx - \lambda \int_{\mathbf{R}^3} H(-\phi)\delta(\psi)|\nabla\psi|dx,$$

where λ is a Lagrange multiplier.

For other flows, we can replace the first integral with the energy corresponding to the type of flow. This just means we are coupling a different energy with the constraint. For example, if we want Wulff flow then we use the Wulff energy. Details of this will be given when we discuss Wulff shapes on surfaces.

The Euler-Lagrange equation then becomes

$$0 = -\nabla \cdot \left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}|\nabla\psi| \right) \delta(\psi)\delta(\phi) + \lambda|\nabla\psi|\delta(\psi)\delta(\phi).$$

Under our usual replacement for $\delta(\psi)\delta(\phi)$ and previous results, we get the evolution equation

$$\phi_t + \lambda|P_{\nabla\psi}\nabla\phi| = \nabla \cdot \left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}||\nabla\psi| \right) \frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|}.$$

We can find the value of λ by enforcing the constraint,

$$0 = \frac{d}{dt}\int_{\mathbf{R}^3} H(-\phi)\delta(\psi)|\nabla\psi|dx$$

$$
\begin{aligned}
&= \int_{\mathbf{R}^3} \phi_t \delta(\phi)\delta(\psi)|\nabla\psi|dx \\
&= \int_{\mathbf{R}^3} \left(\nabla \cdot \left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}|\nabla\psi| \right) - \lambda |P_{\nabla\psi}\nabla\phi| \right) \delta(\phi)\delta(\psi)|\nabla\psi|dx.
\end{aligned}
$$

Solving for λ in this equation gives

$$
\lambda = -\frac{\displaystyle\int_{\mathbf{R}^3} \nabla \cdot \left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}|\nabla\psi| \right) \delta(\phi)\delta(\psi)|\nabla\psi|dx}{\displaystyle\int_{\mathbf{R}^3} |P_{\nabla\psi}\nabla\phi|\delta(\phi)\delta(\psi)|\nabla\psi|dx}.
$$

All this together defines the evolution equation for ϕ that moves a curve by geodesic curvature flow while keeping the enclosed surface area fixed. This equation is also valid and can be used in higher dimensions. For more on the process of fixing enclosed area or volume, see [13] and [30].

Numerically, the right hand side of the evolution equation is handled in the manner corresponding to the flow. The left hand side is in Hamilton Jacobi form and we solve it as in the constant normal flow section, i.e., using third order TVD-RK in time and Hamilton Jacobi fifth order WENO-LLF in space. At each Runge-Kutta step, we solve for λ by using a second order approximation for the integrals, which are only taken near the front because of the delta functions, and using ϕ from the previous step in the integrands.

When deriving the evolution equation from the corresponding $\mathbf{R}^2$ equation,

$$
\phi_t + \lambda|\nabla\phi| = \nabla \cdot \nabla\beta(\nabla\phi)|\nabla\phi|,
$$

with

$$
\lambda = -\frac{\displaystyle\int_{\mathbf{R}^2} \nabla \cdot \nabla\beta(\nabla\phi)|\nabla\phi|\delta(\phi)dx}{\displaystyle\int_{\mathbf{R}^2} |\nabla\phi|\delta(\phi)dx},
$$

we note that all the terms have been considered previously except the λ term. For λ, we would like to take the integrals over the surface instead of over $\mathbf{R}^2$. This means we change the integral from $\int_{\mathbf{R}^2} dx$ to $\int_{\mathbf{R}^3} \delta(\psi)|\nabla\psi|dx$. Using this, the rest of the terms carry over as before and so projecting the $\mathbf{R}^2$ equation on the surface gives the same evolution equation derived above.

We can also consider flows that are not minimizations of energies. If we want the curve to move according to the equation

$$
\phi_t = -v \cdot \nabla\phi,
$$

where v can depend on ψ, ϕ, and their derivatives, then the constrained motion can be given by

$$\phi_t + \lambda |P_{\nabla\psi} \nabla\phi| = -v \cdot \nabla\phi,$$

where

$$\lambda = -\frac{\int_{\mathbf{R}^3} (v \cdot \nabla\phi)\delta(\phi)\delta(\psi)|\nabla\psi| dx}{\int_{\mathbf{R}^3} |P_{\nabla\psi}\nabla\phi|\delta(\phi)\delta(\psi)|\nabla\psi| dx}.$$

This will move a curve according to v while keeping the enclosed surface area fixed. Note that this makes no mention of the evolution equation coming from minimizing an energy. However, when there is an energy for the flow, such as in geodesic curvature flow or Wulff flow, the evolution equation makes more sense. Also, note λ is now not exactly a Lagrange multiplier. Higher dimensional motions preserving enclosed area can be considered using the same evolution equations.

In Figure 15, we show a curve moving by geodesic curvature flow with a fixed enclosed surface area constraint on a paraboloid. The steady state curve is a circle symmetrically wrapped around the paraboloid. In Figure 16, we show a curve moving by the same flow on a sphere. The initial curve is elliptical in nature. The steady state curve is a circle on the sphere.

11.1 Wulff Shapes

We can get further interesting shapes on surfaces by running the evolution equation for Wulff flow with fixed enclosed surface area and looking at the steady state of the zero level set of ϕ on M. For M a plane, i.e., for curves in $\mathbf{R}^2$, this is a Wulff shape, which is the shape certain crystals form (see [25],[20]). These shapes are also of interest as surfaces in $\mathbf{R}^3$. Following the steps for deriving the evolution equation for the enclosed surface area preserving motion on surfaces, we start with the energy

$$E(\phi) = \int_{\mathbf{R}^3} \beta \left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|} \right) |P_{\nabla\psi}\nabla\phi||\nabla\psi|\delta(\psi)\delta(\phi) dx - \lambda \int_{\mathbf{R}^3} H(-\phi)\delta(\psi)|\nabla\psi| dx,$$

where λ is a Lagrange multiplier, and get the evolution equation

$$\phi_t + \lambda |P_{\nabla\psi}\nabla\phi| = \nabla \cdot (P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi)|\nabla\phi|)\frac{|P_{\nabla\psi}\nabla\phi|}{|\nabla\psi|},$$

where

$$\lambda = \frac{\int_{\mathbf{R}^3} \nabla \cdot (P_{\nabla\psi} \nabla \beta(P_{\nabla\psi} \nabla\phi) |\nabla\phi|) \delta(\phi) \delta(\psi) |\nabla\psi| dx}{\int_{\mathbf{R}^3} |P_{\nabla\psi} \nabla\phi| \delta(\phi) \delta(\psi) |\nabla\psi| dx}.$$

The steady state curve on M of this evolution equation gives a Wulff shape on the surface. Wulff shapes in higher dimensions can be found using the same equations.

In Figure 17, we show a curve moving under Wulff flow while fixing the enclosed surface area. The steady state shape is a Wulff shape on the surface and is squarish in nature since $\beta(x)$ used was a smoothed out version of $|x_1| + |x_2| + |x_3|$. More complicated curves and surfaces with topological changes in the curves can also be considered using our algorithm.

12 Moving Curves on Moving Surfaces

We now extend our results to include moving curves on moving surfaces. Since the surface is moving, ψ will now depend on time with the zero level set of ψ at any time giving the surface at that time. Also, the curve on the surface at any time is given by the intersection of the zero level sets of ψ and ϕ at that time. To follow the surface and curve, we need only follow ψ and ϕ or, more accurately, the zero level set of ψ and the intersection of this with the zero level set of ϕ. The initial surface and curve are given and represented by an initial ψ and ϕ.

Suppose the motion we want for the curve satisfies on fixed surfaces, i.e., ψ fixed in time,

$$\phi_t + v \cdot \nabla\phi = 0,$$

for some velocity field v tangent to the level set surfaces of ψ that can depend on ψ, ϕ, and their derivatives. Suppose the motion of the surface itself satisfies

$$\psi_t + w \cdot \nabla\psi = 0,$$

for some velocity field w that can depend on ψ and its derivatives but not on ϕ in any way. The fact that w does not need to be parallel to the normal vector of the surface means the surface is allowed to twist within itself without changing its shape. Thus we can get the velocity field under which to move the curve by adding the two velocity fields v and w. This means the evolution equation is

$$\phi_t + (v + w) \cdot \nabla\phi = 0.$$

Note the curve and also the surface may undergo merging during the evolution process.

As an example, suppose we want the curve to move outward in its normal direction at unit speed. Then we get $v \cdot \nabla\phi = |P_{\nabla\psi}\nabla\phi|$. Suppose we also want the surface to move outward in its normal direction at unit speed. The equation for this is

$$\psi_t + |\nabla\psi| = 0,$$

with $w = \dfrac{\nabla\psi}{|\nabla\psi|}$. Therefore, $w \cdot \nabla\phi = \dfrac{\nabla\psi \cdot \nabla\phi}{|\nabla\psi|}$. So the sum of the two velocity fields gives the evolution equation for the desired motion of the curve,

$$\phi_t + P_{\nabla\psi}\nabla\phi + \frac{\nabla\psi \cdot \nabla\phi}{|\nabla\psi|} = 0.$$

Another example is where the curve itself does not move but the surface moves under the velocity field w. Thus the motion of the curve in $\mathbf{R}^3$ is due only to the motion of the surface. This specific problem, called region tracking, was first solved in [2] using the same representation we use here. In this case,

$$\psi_t + w \cdot \nabla\psi = 0,$$

is the equation for the motion of the surface, and thus,

$$\phi_t + w \cdot \nabla\phi = 0,$$

is the equation for the motion of the curve.

All this can also be done for other previously described motions except for the case of fixed enclosed surface area. In this case, we need to clarify what we really want since the surface may shrink until its total surface area is smaller than the enclosed surface area to be fixed. Higher dimensions are also covered by the above evolution equations.

A drawback of this method is that spurious curves may appear when surfaces merge. This happens when a part of the surface with negative values of ϕ touches a part of the surface with positive values. At the place of contact, a zero level set of ϕ is created in between the positive and negative values and so gives rise to a spurious curve on the surface. If the problem we are considering is curves on surfaces then this is a wrong answer. But if we look at a different problem, then the spurious curve actually makes sense. Let us think of the curve as the boundary of the set of negative values of ϕ on the surface and the movement of the curve as being due to the expansion or contraction of that set. The negative values may denote one substance and the positive values a different substance, as in two phase flow. When negative and positive values touch, a new boundary for the set of negative values needs to be created and, hence, we should get a curve appearing there to separate the positive and negative values. This way of thinking is not only convenient here but may also be useful in physical applications.

In Figure 18, we show initially a circle on a plane. The circle is moving by constant normal flow on the plane and the plane is moving by constant normal flow in $\mathbf{R}^3$. The final picture shows at the final time a dilated circle on a translated plane. More complicated curves and surfaces can also be handled by the algorithm.

13 Local Level Set Method

A curve is a one dimensional object, so to solve an evolution equation in all of $\mathbf{R}^3$ is overly expensive. In most cases, we only need to solve the equation in a neighborhood of the curve. Exceptions, however, include getting signed distance functions to curves, where the ϕ is needed over the whole surface, or cases where curves can appear out of nowhere, for example, in the active contour method of Chan and Vese[4]. But in most of the motions we have studied here, the evolution equation is only needed in a neighborhood of the curve.

We have succeeded in localizing near the surface, i.e., retaining only the grid points that are near the surface. This is optimal for problems that need ϕ defined on the whole surface, e.g., getting signed distance. We create a data structure to hold only the grid points close to the surface. The structure only needs to be created once, at the beginning, since the surface is static. This immediately cuts down on our memory storage. Also, we solve our partial differential equations only at the retained grid points in this structure, thus greatly speeding up the method. To determine which grid points are near the surface and thus should be in the structure, we look at the distances in $\mathbf{R}^3$ of those points away from the surface. Only points under a certain value, a constant times Δx, are retained, which makes this method optimal when ϕ is needed over the whole surface. We use the fast marching method once at the beginning to create the distance values at the grid points.

In actuality, we will only solve our partial differential equation in a smaller neighborhood of the surface than the neighborhood of retained points. This is done so that the stencils of the finite difference schemes we use will not exit the neighborhood of retained points. Fortunately, the fast marching method we used to obtain distance to the surface, as a byproduct, also gives an ordering of the points with respect to their distance values, from least to greatest. We can then use this to enforce Neumann boundary conditions on the boundary of the smaller neighborhood by extending the values there, following the normal vectors of the boundary, to the larger neighborhood. Note the normal vectors of the boundary are in the same direction as the gradients of the distance values and thus following the ordering given by the fast marching method correctly propagates the values. So even though the partial differential equation is only solved in the smaller neighborhood, the finite difference schemes will use values in the whole neighborhood. The method, under these operations, still retains the same accuracy.

We may also want to make sure that the numerical boundary conditions will not adversely affect the behavior of ϕ on the surface. For this, we can make the level set surfaces of ϕ perpandicular to the surface while fixing the values of ϕ on the surface. This is accomplished using the evolution equation described previously in Section 8 for this purpose. The fast marching method can also be used instead. The process of making the level sets of ϕ perpendicular to the surface, however, may reduce the accuracy of the method.

So far, we have constructed a local level set method that is optimal for solving partial differential equations over the whole surface but not for solving just in a neighborhood of a curve. For this, we currently have a method that has the potential to be optimal in both speed and memory but has not yet been programmed in such a way. It similarly involves retaining only the grid points that are near the zero level sets of ψ and ϕ, solving in a smaller neighborhood of these retained points, and making the level sets of ψ and ϕ well behaved as described previously in Section 8. This ensures that the boundaries of the neighborhood do not affect the motion of the curve. However, the data structure is no longer static since the curve is moving, taking the neighborhood along with it. This aspect slightly complicates the problem and especially the programming issues.

Table 5 shows the local level set method applied to constant normal flow. The evolution equations are solved only in a neighborhood of the surface. Also, the level sets of ϕ are not enforced to be perpandicular to the surface. The test includes all elements of our method, including the plotter. It was generated by looking at a circle moving on a sphere and the results show roughly second order accuracy (compare to Table 1). Running the algorithm to make the level sets of ϕ perpandicular to the surface will slightly move the contours on the surface and reduce the accuracy to first order. We remark again that much finer grids can be used than for the global method

grid size	error	order
$8 \times 8 \times 8$	0.053125	
$16 \times 16 \times 16$	0.016274	1.7068
$32 \times 32 \times 32$	0.00561706	1.5347
$64 \times 64 \times 64$	0.00173415	1.6956
$128 \times 128 \times 128$	0.000433395	2.0005
$256 \times 256 \times 256$	0.000169642	1.3532
$320 \times 320 \times 320$	0.000107546	2.0425

Table 5: This is the order of accuracy analysis for the local level set method for constant normal flow. The grid size represents the equivalent sized grid if all grid points were used. The example considered was a circle moving on a sphere (as in Table 1). Because of the behavior of the error for the $256 \times 256 \times 256$ case, we only say the method is roughly second order accurate. Note we can run the program on a grid equivalent to $320 \times 320 \times 320$ with this algorithm.

that solves in all of $\mathbf{R}^3$. Table 6 shows the local level set method applied to finding signed distance functions. Once again, the level sets of ϕ are not enforced to be perpandicular to the surface. The result is first order accuracy, as in the global case. The table was generated by looking at a circle on a sphere. Note, much finer grids can be used than for the global method. All in all, the local level set method is faster and needs less memory than the global method while still being able to preserve the accuracy of the method.

14 Higher Dimensions and Codimensions

We can further extend our method to higher dimensions and codimensions (see [3]) by using more functions $\phi_1, \ldots, \phi_k$ and $\psi_1, \ldots, \psi_m$ in $\mathbf{R}^n$, for $k+m \leq n$. The intersection of the zero level sets of $\psi_1, \ldots, \psi_m$ gives the constraint, and the intersection of this with the intersection of the zero level sets of $\phi_1, \ldots, \phi_k$ gives the object to be moved under the constraint. This means the constraint surface has dimension $n-m$ and on this, we move an object with dimension $n-m-k$. The actual motions will be carried out under a system of evolution equations for $\phi_1, \ldots, \phi_k$. Note, however, the fact that our methods are grid based, usually using uniform grids, means the size of computer memory needed to run simulations in very high dimensions may

grid size	error	order
$10 \times 10 \times 10$	0.04269	
$20 \times 20 \times 20$	0.0300296	0.5075
$40 \times 40 \times 40$	0.0170282	0.8185
$80 \times 80 \times 80$	0.00864061	0.9787
$160 \times 160 \times 160$	0.00435087	0.9898
$320 \times 320 \times 320$	0.00213815	1.0249

Table 6: This is the order a accuracy analysis for the local level set method for distance functions. The grid size represents the equivalent sized grid if all grid points were used. The curve considered was a circle on a sphere (as in Table 2). The method is first order accurate. Note we can run the program on a grid equivalent to $320 \times 320 \times 320$ with this algorithm.

be restrictive, even with a local level set method.

15 Conclusion

We have devised a level set based method for moving curves constrained on surfaces. This method can accurately handle a wide variety of curves and surfaces and motions. It can also extend all the results of the original $\mathbf{R}^2$ level set method and thus conceivably has a wide range of applications. Basic applications already allow us to create signed distance functions, geodesics, and various interesting crystal shapes on surfaces. The limitations of our method are just the limitations of any level set based approach. Finally, the method is easy to implement because complex surface topologies and procedures, such as merging or breaking or keeping the curve on the surface, are all handled automatically.

16 Acknowledgments

The authors would like to thank Shiu-Yuen Cheng for his help on the geometry aspects of the method, especially the proofs of the propositions.

17 Proofs of Propositions

Proof of Proposition 1

We will prove that these identities hold in $\mathbf{R}^n$ for arbitrary n.

(a) This follows from the fact that P_w is a symmetric matrix and $P_w^2 = P_w$.

(b) This follow from the fact that

$$(P_X\nabla)_i u = P_X \nabla u \cdot e_i = \nabla u \cdot P_X e_i.$$

(c) We prove this property by brute force calculations and, for simplification, summing over repeated indices. Let e_i be the vector with 1 for its ith component and 0 for the rest. This means for the jth component, $(e_i)_j = \delta_{ij}$. So we have,

$$\begin{aligned}
\nabla^{\nabla u} \cdot (P_{\nabla u} X) &= \nabla_i^{\nabla u} (P_{\nabla u} X)_i \\
&= \nabla((P_{\nabla u} X)_i) \cdot P_{\nabla u} e_i \\
&= \left[\left(\delta_{ij} - \frac{u_{x_i} u_{x_j}}{|\nabla u|^2}\right) X_j\right]_{x_k} \left(\delta_{ki} - \frac{u_{x_k} u_{x_i}}{|\nabla u|^2}\right) \\
&= \left[\left(\delta_{ij} - \frac{u_{x_i} u_{x_j}}{|\nabla u|^2}\right) X_j\right]_{x_i} - \left[\left(\delta_{ij} - \frac{u_{x_i} u_{x_j}}{|\nabla u|^2}\right) X_j\right]_{x_k} \frac{u_{x_k} u_{x_i}}{|\nabla u|^2}.
\end{aligned}$$

Calling the first term I and the second term J, we have

$$I = \left[\left(\delta_{ij} - \frac{u_{x_i} u_{x_j}}{|\nabla u|^2}\right) X_j\right]_{x_i} = \nabla \cdot (P_{\nabla u} X),$$

and

$$\begin{aligned}
J &= -\left[\left(\delta_{ij} - \frac{u_{x_i} u_{x_j}}{|\nabla u|^2}\right) X_j\right]_{x_k} \frac{u_{x_k} u_{x_i}}{|\nabla u|^2} \\
&= -\frac{u_{x_k} u_{x_i} (X_i)_{x_k}}{|\nabla u|^2} + \left(\frac{u_{x_i} u_{x_j} X_j}{|\nabla u|^2}\right)_{x_k} \frac{u_{x_k} u_{x_i}}{|\nabla u|^2} \\
&= -\frac{u_{x_k} u_{x_i} (X_i)_{x_k}}{|\nabla u|^2} + \frac{u_{x_i} u_{x_j} (X_j)_{x_k} u_{x_k} u_{x_i}}{|\nabla u|^4} + \left(\frac{u_{x_i} u_{x_j}}{|\nabla u|^2}\right)_{x_k} \frac{X_j u_{x_k} u_{x_i}}{|\nabla u|^2} \\
&= \left(\frac{u_{x_j} u_{x_i}}{|\nabla u|^2}\right)_{x_k} \frac{X_j u_{x_k} u_{x_i}}{|\nabla u|^2} \\
&= \left(\frac{u_{x_i} u_{x_j x_k}}{|\nabla u|^2} + \frac{u_{x_j} u_{x_i x_k}}{|\nabla u|^2} - \frac{2 u_{x_j} u_{x_i} u_{x_m} u_{x_m x_k}}{|\nabla u|^4}\right) \frac{X_j u_{x_k} u_{x_i}}{|\nabla u|^2} \\
&= \frac{u_{x_k} u_{x_j x_k} X_j}{|\nabla u|^2} - \frac{u_{x_i} u_{x_j} u_{x_k} u_{x_i x_k} X_j}{|\nabla u|^4} \\
&= \left(X_i - \frac{u_{x_i} u_{x_j} X_j}{|\nabla u|^2}\right) \frac{u_{x_k} u_{x_i x_k}}{|\nabla u|^2} \\
&= P_{\nabla u} X \cdot \frac{\nabla |\nabla u|}{|\nabla u|}.
\end{aligned}$$

So altogether,

$$\nabla^{\nabla u} \cdot (P_{\nabla u} X) = \nabla \cdot (P_{\nabla u} X |\nabla u|) \frac{1}{|\nabla u|},$$

which completes the proof.

Proof of Proposition 2

We will prove the results in general for $\mathbf{R}^n$. We sum over repeated indices for convenience.

(a) Fix a point x on M. Let ν be the outward unit normal vector to M at x. Now given two orthonormal bases in $\mathbf{R}^n$, e_i and $\tilde{e}_i$, $i = 1, \ldots, n$, let δ_i, let δ_i and $\tilde{\delta}_i$ be $(P_\nu \nabla)_i$ under the frames e_i and $\tilde{e}_i$, respectively. This means

$$\begin{aligned} \delta_i &= \left(\delta_{ij} - \frac{\langle \nu, e_i \rangle \langle \nu, e_j \rangle}{|\nu|^2} \right) \partial_j \\ \tilde{\delta}_i &= \left(\delta_{ij} - \frac{\langle \nu, \tilde{e}_i \rangle \langle \nu, \tilde{e}_j \rangle}{|\nu|^2} \right) \tilde{\partial}_j, \end{aligned}$$

where ∂_i and $\tilde{\partial}_i$ correspond to the frames e_i and $\tilde{e}_i$, respectively. Because of orthonormality, we have that $\tilde{e}_i = a_{ij} e_j$ with the a_{ij} forming an orthogonal matrix, i.e., $a_{ij} a_{ik} = a_{ji} a_{ki} = \delta_{jk}$. Thus, we have $\tilde{\delta}_i = a_{ij} \delta_j$. Therefore,

$$\tilde{e}_i \tilde{\delta}_i = e_i \delta_i.$$

Now taking e_i, $i = 1, \ldots, n$, to be the standard orthonormal basis in $\mathbf{R}^n$ and $\tilde{e}_i$, $i = 1, \ldots, n$, to be an orthonormal basis of $\mathbf{R}^n$ with $\tilde{e}_n = \nu$, we get

$$\begin{aligned} \tilde{e}_i \tilde{\delta}_i &= \nabla^S \\ e_i \delta_i &= P_\nu \nabla. \end{aligned}$$

So

$$\nabla^S = P_\nu \nabla,$$

and, especially, if u is a real valued function in $\mathbf{R}^n$, then

$$\nabla^S u = P_\nu \nabla u.$$

(b) Continuing with the above computations, given X a vector field in $\mathbf{R}^n$, we have

$$\langle \tilde{\delta}_i X, \tilde{e}_i \rangle = \langle \delta_i X, e_i \rangle.$$

Therefore,

$$\nabla^S \cdot X = P_\nu \nabla X.$$

Proof of Proposition 3

Apply Proposition 7 with $\beta(p) = |p|$, $p \in \mathbf{R}^3$.

Proof of Proposition 4

See proof of Lemma 1 in [3].

Proof of Proposition 5

See main body of section for proof.

Proof of Proposition 6

With $\beta(p) = |p|$, we get $\nabla\beta(p) = \frac{p}{|p|}$ and $\nabla^2\beta(p) = \frac{1}{|p|}P_p$. So

$$\nabla^2\beta\left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}\right) = P_{P_{\nabla\psi}\nabla\phi}.$$

Therefore, using Proposition 8, we find that the principle matrix for the right hand side of the evolution equation is $P_{\nabla\psi}P_{P_{\nabla\psi}\nabla\phi}P_{\nabla\psi}$ and also since $P_{P_{\nabla\psi}\nabla\phi}$ is nonnegative definite, with one zero eigenvalue and the rest being equal to one, the evolution equation is thus degenerate second order parabolic.

To actually find the eigenvalues of the principle matrix, we note that $\nabla\psi$ and $P_{\nabla\psi}\nabla\phi$ are eigenvectors corresponding to the zero eigenvalue, since $P_{\nabla\psi}\nabla\psi = 0$ and $P_{P_{\nabla\psi}\nabla\phi}P_{\nabla\psi}P_{\nabla\psi}\nabla\phi = P_{P_{\nabla\psi}\nabla\phi}P_{\nabla\psi}\nabla\phi = 0$, respectively. Also, given any other vector, $v \in \mathbf{R}^3$, perpendicular to these two eigenvectors, we have

$$P_{\nabla\psi}P_{P_{\nabla\psi}\nabla\phi}P_{\nabla\psi}v = P_{\nabla\psi}P_{P_{\nabla\psi}\nabla\phi}v = P_{\nabla\psi}v = v.$$

Therefore, we can conclude that the principle matrix has two zero eigenvalues with all the rest being equal to one.

Proof of Proposition 7

See proof of Lemma 2 in [3].

Proof of Proposition 8

Let $F(q,x) = P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}q)|\nabla\psi|$. Then we have

$$(F_i)_{q_j}(q,x) = P_{\nabla\psi}\nabla^2\beta(P_{\nabla\psi}q)P_{\nabla\psi}|\nabla\psi|,$$

where $\nabla^2\beta$ is the Hessian matrix for β.

Therefore, the principle matrix for $\nabla\cdot(P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi)|\nabla\psi|)\frac{|P_{\nabla\psi}|}{|\nabla\psi|}$ is

$$P_{\nabla\psi}\nabla^2\beta(P_{\nabla\psi}\nabla\phi)P_{\nabla\psi}|P_{\nabla\psi}\nabla\phi|,$$

which can be rewritten as

$$P_{\nabla\psi}\nabla^2\beta\left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}\right)P_{\nabla\psi},$$

since $\nabla^2\beta$ is homogeneous of degree -1.

Therefore, if the matrix $N = \nabla^2\beta\left(\frac{P_{\nabla\psi}\nabla\phi}{|P_{\nabla\psi}\nabla\phi|}\right)$ is nonnegative definite, then $x^T N x \geq 0$ for all $x \in \mathbf{R}^3$. This implies for any $y \in \mathbf{R}^3$, taking $x = P_{\nabla\psi}y$ gives that $y^T P_{\nabla\psi} N P_{\nabla\psi} y \geq 0$ and so $P_{\nabla\psi} N P_{\nabla\psi}$ is also nonnegative definite. Note $\nabla\psi$ is an eigenvector corresponding to the 0 eigenvalue.

So we have shown if N is nonnegative definite, then the evolution equation

$$\phi_t = \nabla \cdot (P_{\nabla\psi}\nabla\beta(P_{\nabla\psi}\nabla\phi)|\nabla\psi|)\frac{|P_{\nabla\psi}|}{|\nabla\psi|}$$

is degenerate second order parabolic.

References

[1] D. Adalsteinsson and J.A. Sethian *A Fast Level Set Method for Propagating Interfaces.* J. Comput. Phys., 118:269–277, 1995.

[2] M. Bertalmio, G. Sapiro, and G. Randall. *Region Tracking on Surfaces Deforming via Level-Sets Methods.* Scale-Space, 330–338, 1999.

[3] P. Burchard, L.T. Cheng, B. Merriman, and S. Osher. *Motion of Curves in Three Spatial Dimensions Using a Level Set Approach.* UCLA CAM Report, 00-29, 2000, submitted to J. Comput. Phys.

[4] T. Chan and L. Vese. *An Active Contour Model Without Edges.* Lecture Notes in Comp. Sci., 1687:141–151, 1999.

[5] S. Chen, B. Merriman, S. Osher, and P. Smereka. *A Simple Level Set Method for Solving Stefan Problems.* J. Comput. Phys., 135:8, 1995.

[6] L.T. Cheng. *The Level Set Method Applied to Geometrically Based Motion, Materials Science, and Image Processing.* Ph.D. Thesis, UCLA, 2000.

[7] D.L. Chopp. *Computing minimal surfaces via level set curvature flow.* J. Comput. Phys., 106:77–91, 1993.

[8] D.L. Chopp. *Flow Under Geodesic Curvature.* UCLA CAM Report, 92-23, 1992.

[9] M. do Carmo. *Differential geometry of curves and surfaces.* Translated from the Portuguese. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1976.

[10] Ph. Hoch and M. Rascle. *Hamilton-Jacobi Equations on a Manifold and Applications to Grid Generation and Refinement.* preprint, 1999

[11] T.Y. Hou, J.S. Lowengrub, and M.J. Shelley. *Removing the Stiffness from Interfacial Flows with Surface Tension.* J. Comput. Phys., 114:312, 1994.

[12] G.S. Jiang and D. Peng. *Weighted ENO Schemes for Hamilton Jacobi Equations.* SIAM J. Scient. Comput., 21(6):2126–2143, 2000.

[13] M. Kang, B. Merriman, S. Osher, and P. Smereka. *Level Set Approach for the Motion of Soap Bubbles with Curvature Dependent Velocity or Acceleration.* UCLA CAM Report, 96-19, 1996.

[14] R. Kimmel. *Intrinsic Scale Space for Images on Surfaces: The Geodesic Curvature Flow.* Graphical Models and Image Processing, 59(5):365–372, 1997.

[15] R. Kimmel, A. Amir, and A.M. Bruckstein. *Finding Shortest Paths on Surfaces Using Level Sets Propagation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(6):635–640, 1995.

[16] R. Kimmel and N. Kiryati. *Finding the Shortest Paths on Surfaces by Fast Global Approximation and Precise Local Refinement.* International Journal of Pattern Recognition and Artificial Intelligence, 10(6):643–656, 1996.

[17] R. Kimmel and J.A. Sethian. *Computing geodesic paths on manifolds.* Proc. Natl. Acad. Sci. USA, 95:8431–8435, 1998.

[18] G.H. Markstein. *Nonsteady Flame Propagation.* Pergamon Press, Oxford 1964.

[19] S. Osher and R. Fedkiw. *Level Set Methods.* UCLA CAM Report, 00-08, 2000, submitted to J. Comput. Phys.

[20] S. Osher and B. Merriman. *The Wulff Shape as the Asymptotic Limit of a Growing Crystalline Interface.* Asian J. Math., 1(3):560–571, 1997.

[21] S. Osher and J.A. Sethian. *Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations.* J. Comput. Phys., 79(1):12–49, 1988.

[22] D. Peng, B. Merriman, S. Osher, H.K. Zhao, M. Kang. *A PDE-based fast local level set method.* J. Comput. Phys., 155(2):410–438, 1999.

[23] C.W. Shu and S. Osher *Efficient implementation of essentially nonoscillatory shock-capturing schemes.* J. Comput. Phys., 77(2):439–471, 1988.

[24] L. Simon. *Seminar on Geometric Measure Theory.* Birkhauser Verlag, 1986.

[25] H.M. Soner. *Motion of a Set by the Curvature of Its Boundary.* J. Diff. Eqn., 101:313–372, 1993.

[26] M. Sussman, P. Smereka, and S. Osher. *A level set method for computing solutions to incompressible two-phase flow.* J. Comput. Phys., 114:146–159, 1994.

[27] J.N. Tsitsiklis. *Efficient Algorithms for Globally Optimal Trajectories.* IEEE Transactions on Automatic Control, 40:1528–1538, 1995.

[28] H.K. Zhao, T.F. Chan, B. Merriman, and S. Osher. *A variational level set approach to multiphase motion.* J. Comput. Phys., 127:179–195, 1996.

[29] H.K. Zhao, S. Osher, B. Merriman, and M. Kang. *Geometric interpolation of curves and surfaces.* UCLA CAM Report, 98-7, 1998, to appear in Comp. Vision and Image Understanding.

[30] 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., 143:495–518, 1998.

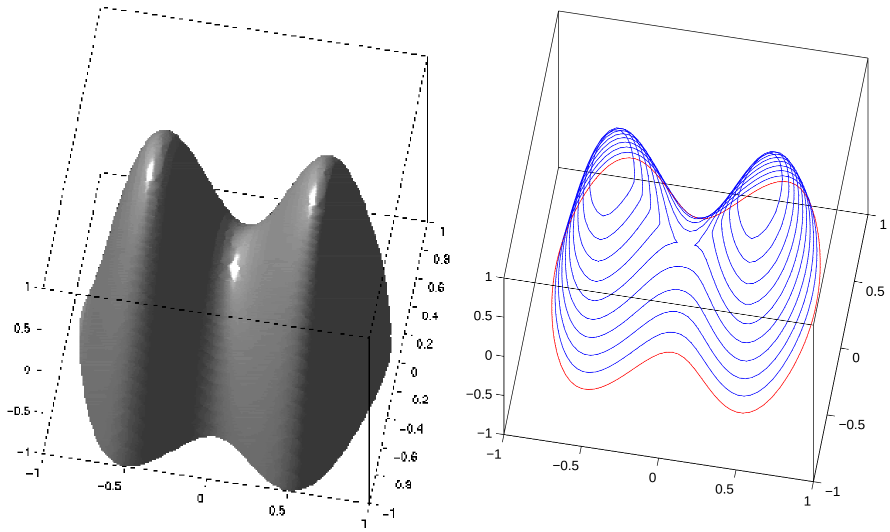

Figure 1: The surface, two mountains, is shown on the left and the evolution of a curve is shown separately on the right. The curve is moving inward by unit normal flow and breaks into two smaller curves, one on each mountain, during the flow.

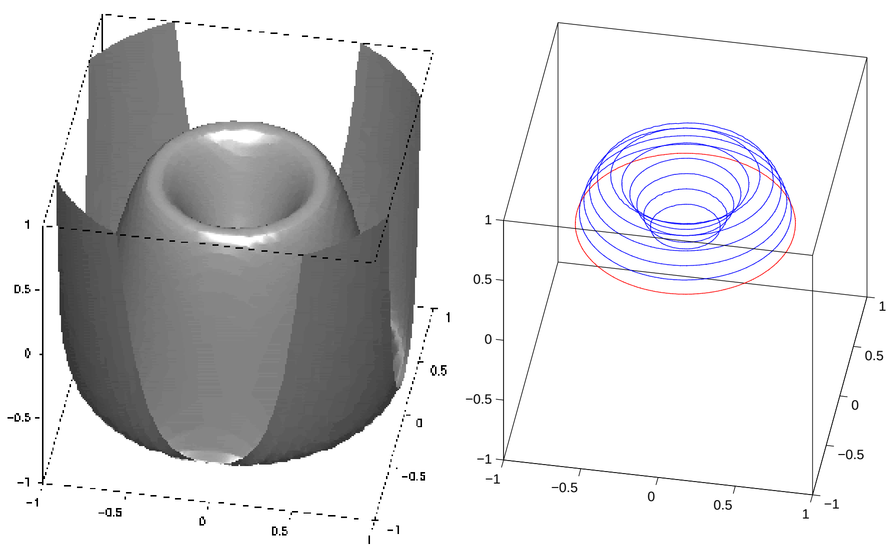

Figure 2: The surface, a volcano, is shown on the left and the evolution of a curve is shown separately on the right. The curve is moving inward by unit normal flow and flow up the side of the volcano, then down into the core.

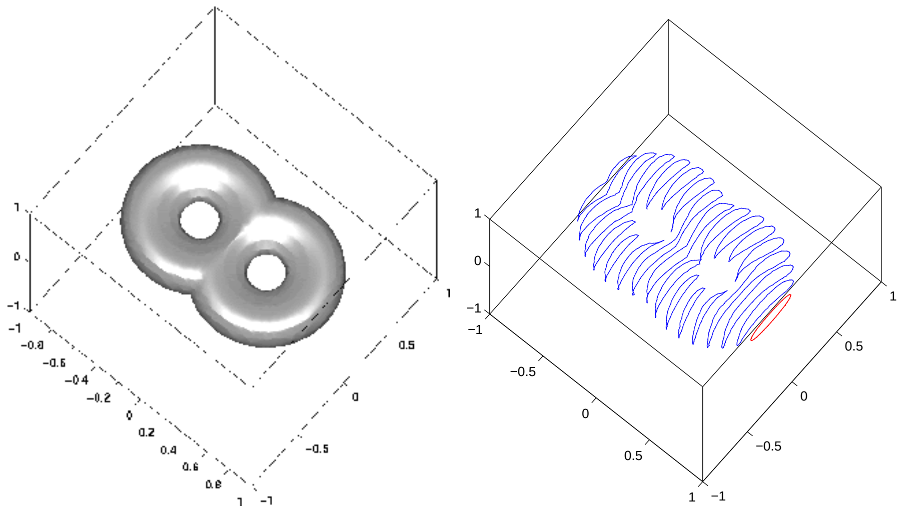

Figure 3: The surface, a two holed torus, is shown on the left and the evolution of a curve is shown separately on the right. The curve is moving inward by unit normal flow, translating to the left on the two holed torus while breaking and merging multiple times.

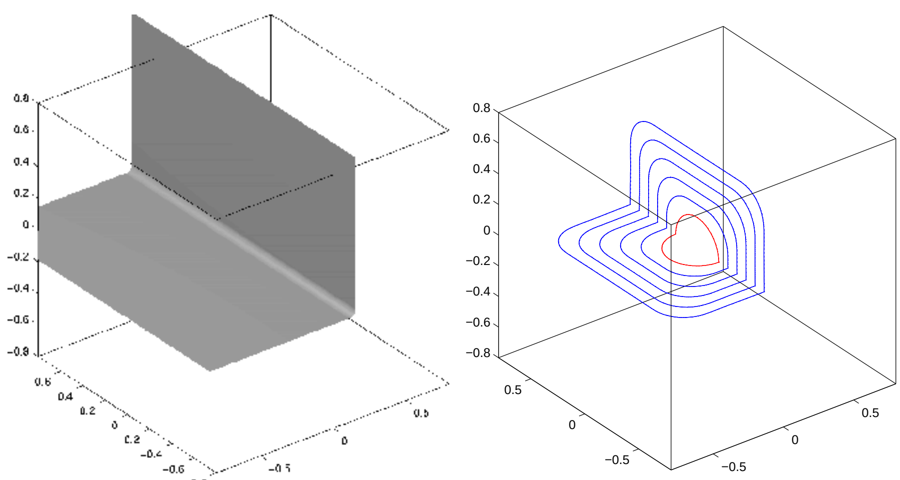

Figure 4: The surface, a folded plane, is shown on the left and the evolution of a curve is shown separately on the right. The curve is moving outward in the normal direction by a non-constant speed. The chosen speed, related to crystal growth, causes the curve to develop a squarish aspect as it expands.

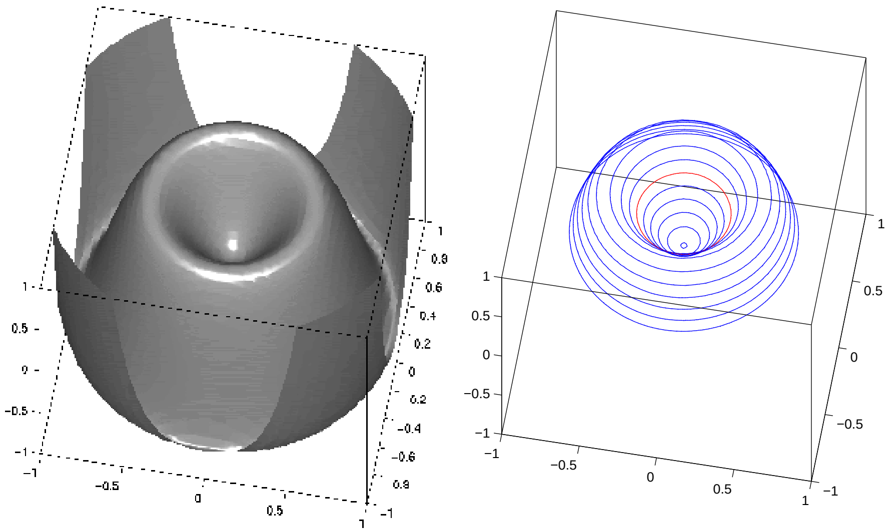

Figure 5: The surface, a volcano, is shown on the left and the contours of the signed distance function are shown on the right. The picture is similar to that of constant normal flow on a volcano. Note the contours are well spaced.

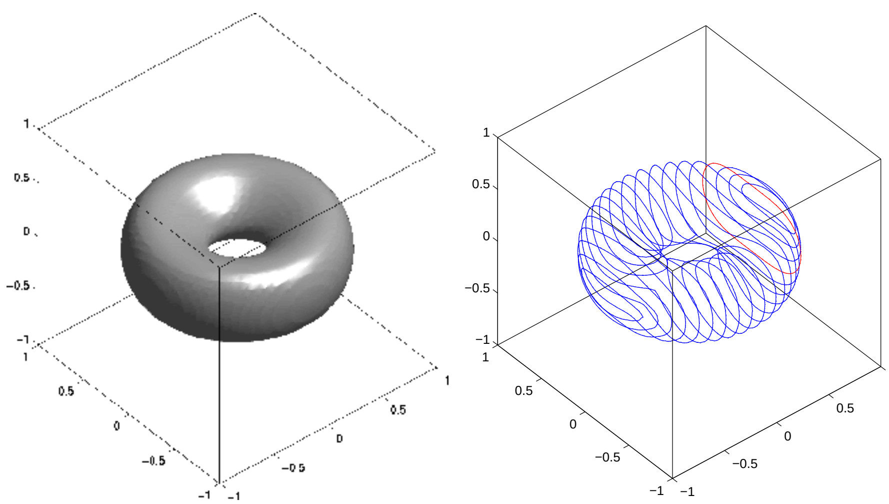

Figure 6: The surface, a torus, is shown on the left and the contours of the signed distance function are shown on the right. Note the contours are well spaced on the torus.

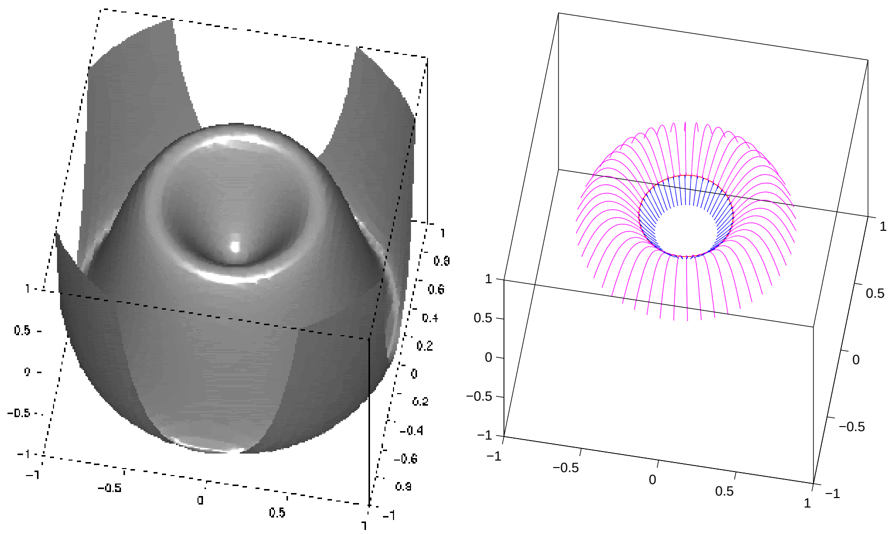

Figure 7: The volcano surface is shown on the left and geodesics from various points to a curve in the volcano core are shown on the right. The geodesics travel up the volcano and down into its core.

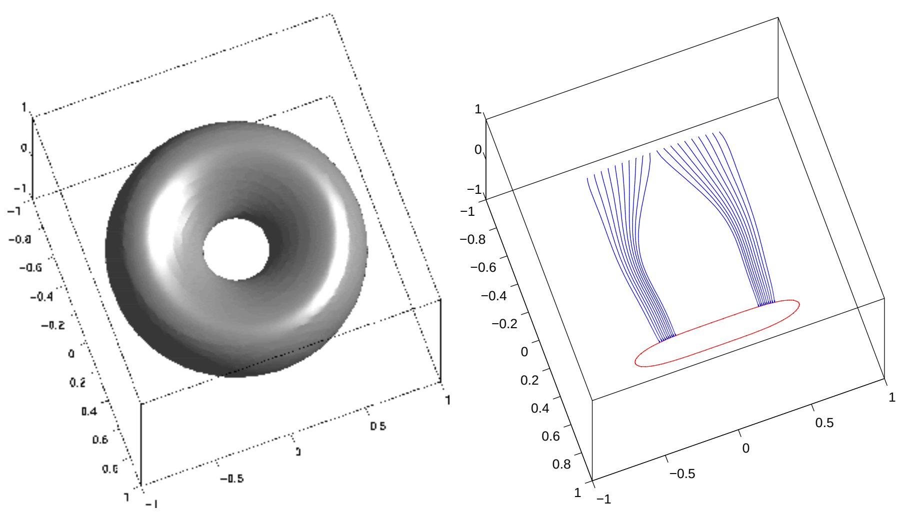

Figure 8: A torus is shown on the left and geodesics from various points to a curve on the torus are shown on the right. The geodesics travel across the torus, around the hole in the middle, to reach the curve.

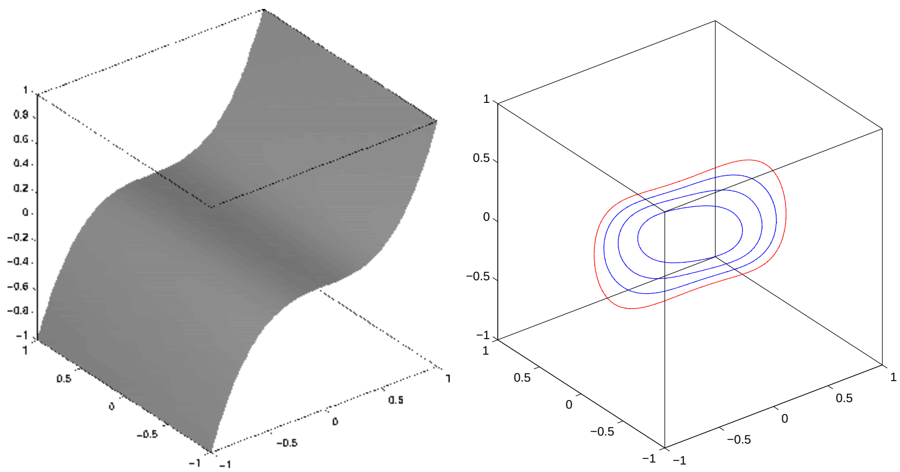

Figure 9: A simple surface is shown on the left and the evolution of a curve under geodesic curvature flow is shown on the right. The curve shrinks on the surface, minimizing its length, until it disappears.

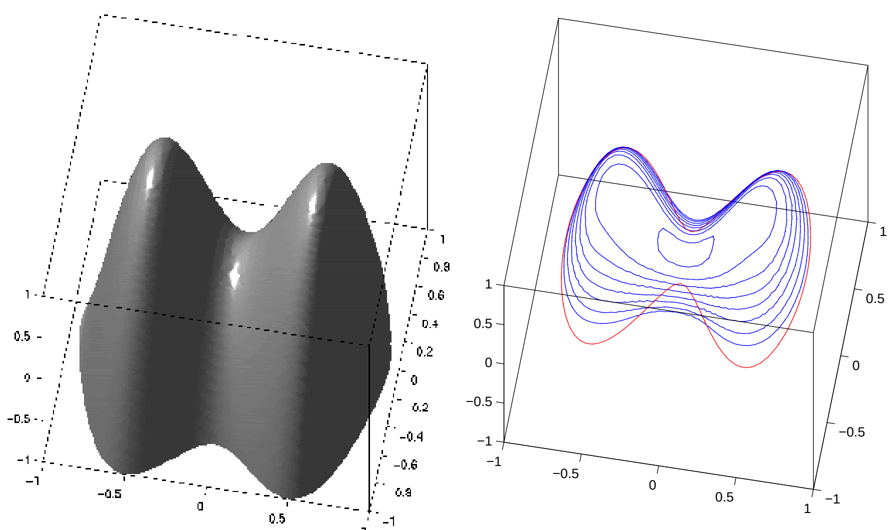

Figure 10: The surface, two mountains, is shown on the left and the evolution of a curve under geodesic curvature flow is shown on the right. The curve is shrinking but needs to move over the mountains before it can disappear.

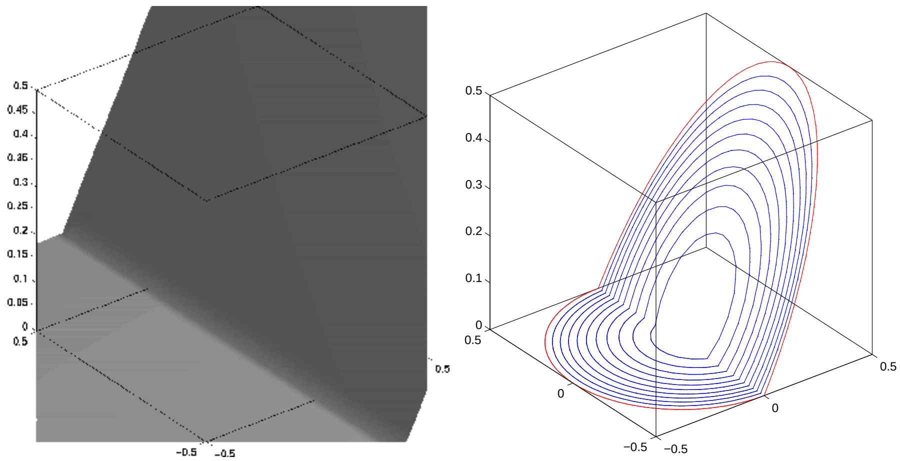

Figure 11: The surface, a bent plane, is shown on the left and the evolution of a curve under geodesic curvature flow is shown on the right. Note the surface has a kink in it and the curve shrinks over this kink without any problems.

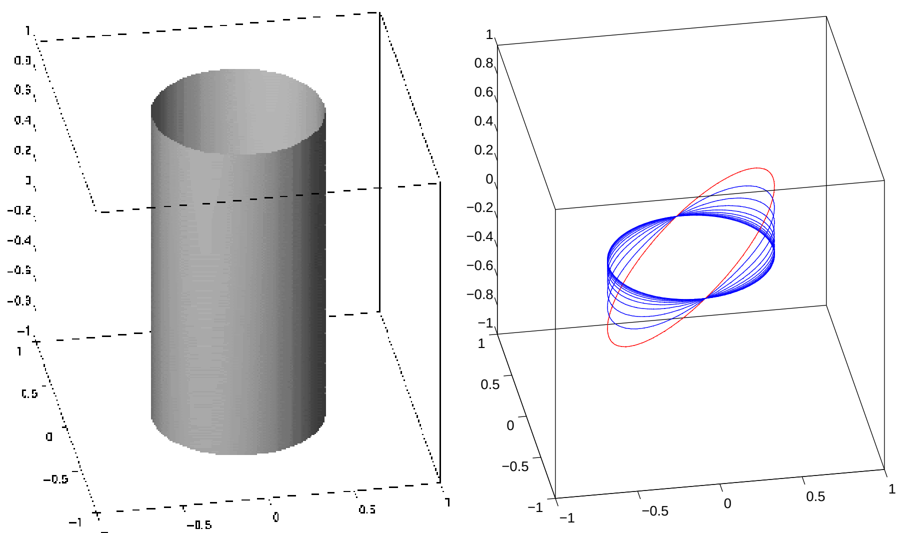

Figure 12: The surface, a cylinder, is shown on the left and the evolution of a curve under geodesic curvature flow is shown on the right. The curve ends up wrapping tightly around the cylinder, forming a geodesic curve, in this case a circle, on the surface.

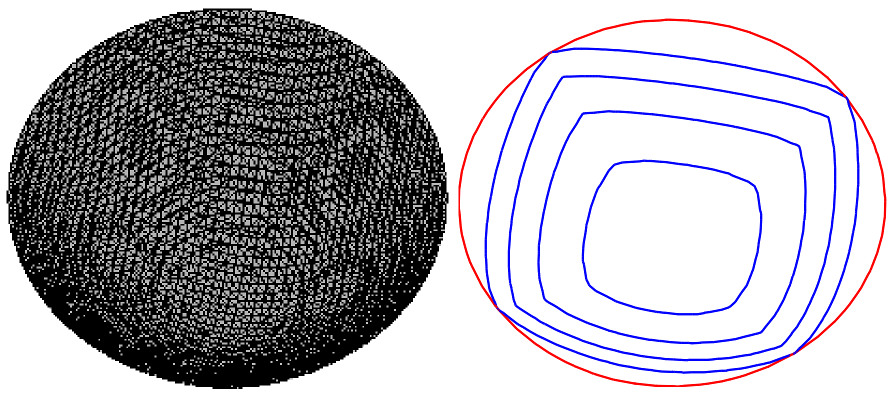

Figure 13: The surface, the bottom of a paraboloid, is shown on the left and the evolution of a curve under Wulff flow, with $\beta(x)$ a smoothed out form of $|x_1| + |x_2| + |x_3|$, is shown on the right. The curve shrinks, developing a squarish shape on the surface before disappearing.

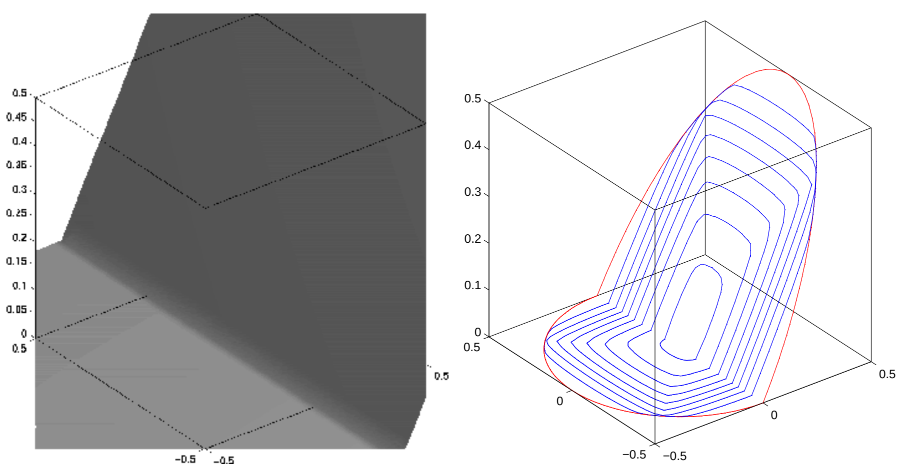

Figure 14: The surface, a bent plane, is shown on the left and the evolution of a curve under Wulff flow, with $\beta(x)$ a smoothed out form of $|x_1|+|x_2|+|x_3|$, is shown on the right. The curve shrinks, developing a squarish shape on the surface before disappearing. Note the kink in the surface does not present any problems.

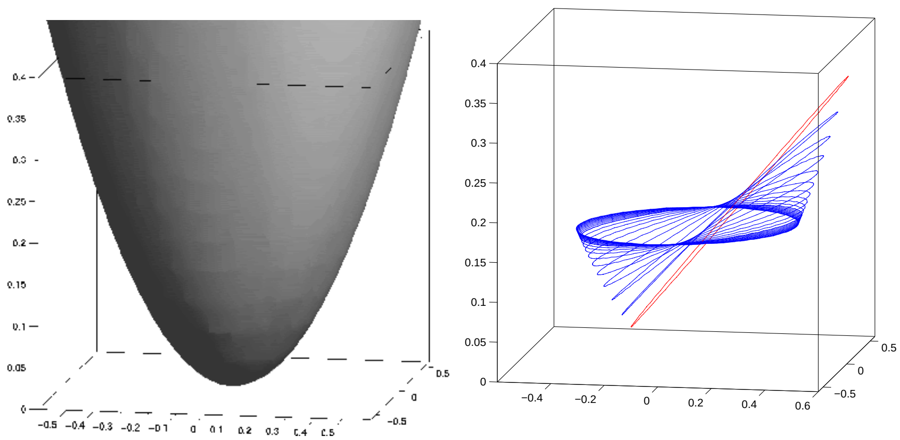

Figure 15: The surface, a paraboloid, is shown on the left and the evolution of a curve under geodesic curvature flow with fixed enclosed surface area is shown on the right. The initial curve evolves to a steady state curve, a circle symmetrically wrapped around the paraboloid.

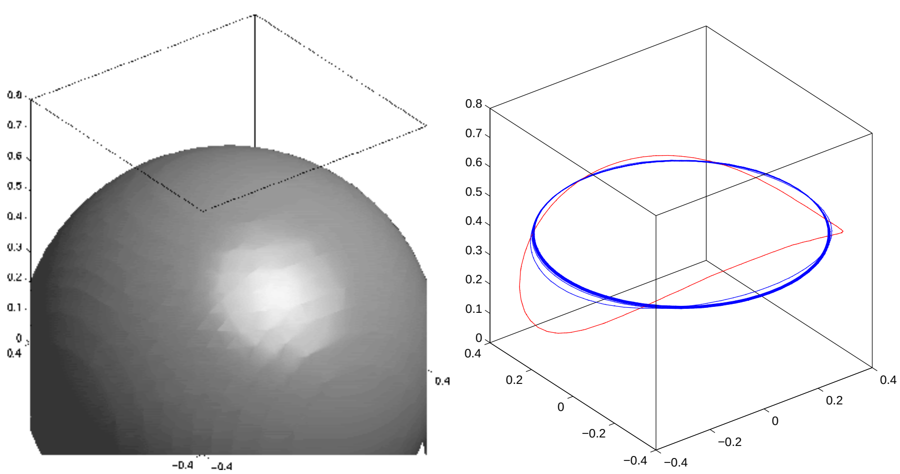

Figure 16: The surface, a sphere, is shown on the left and the evolution of a curve under geodesic curvature flow with fixed enclosed surface area is shown on the right. The initial curve evolves to a Wulff shape, a circle on the sphere.

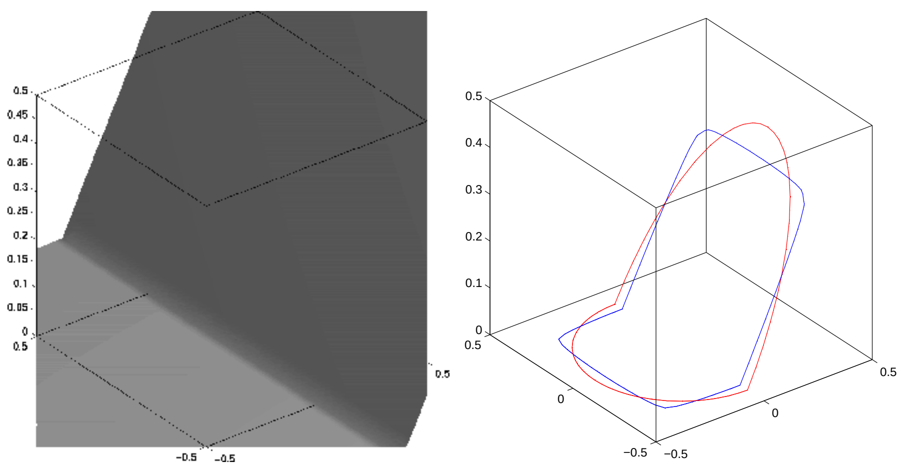

Figure 17: The surface, a bent plane, is shown on the left and the evolution of a curve under Wulff flow with fixed enclosed surface area is shown on the right. The initial curve evolves to a steady state curve, a smoothed out squarish shape on the surface.

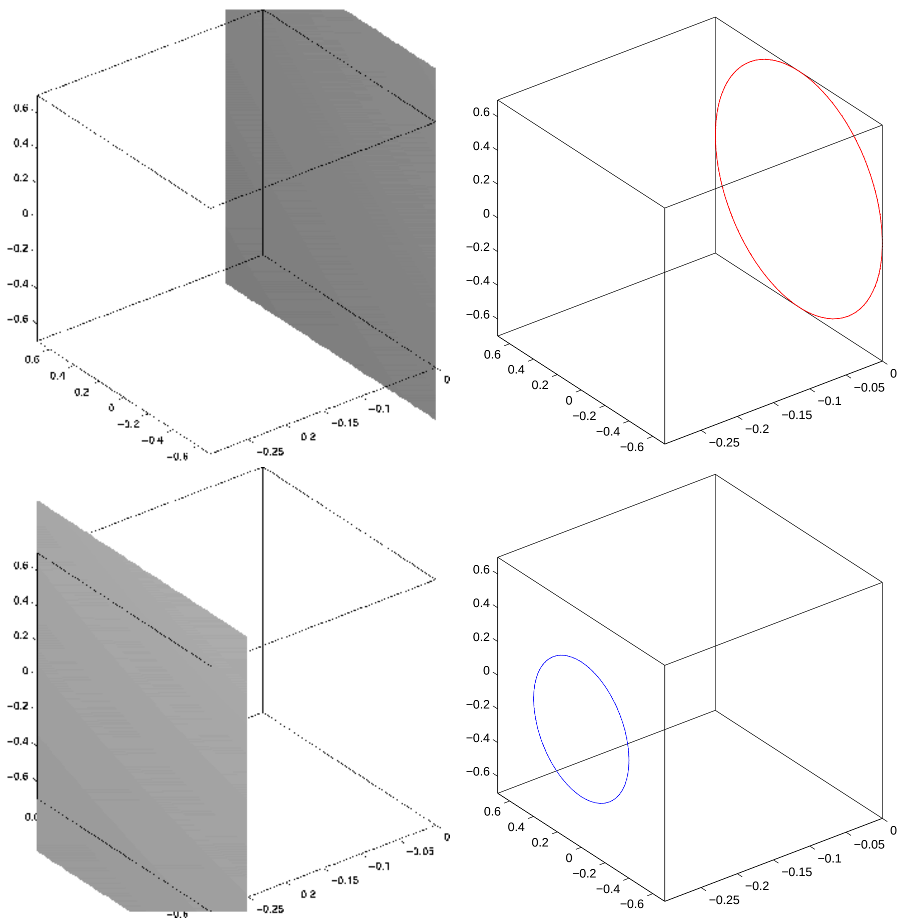

Figure 18: This is a moving curve on moving surface computation. The original surface and curve are shown in the two plots on the top. The final surface and curve are shown in the two plots below. The surface and curve are both moving by constant normal flow. The surface translates to the left while the curve shrinks.