4

Motion Involving Mean Curvature

4.1 Equation of Motion

In the last chapter we discussed the motion of an interface in an externally generated velocity field $\vec{V}(\vec{x}, t)$. In this chapter we discuss interface motion for a self-generated velocity field $\vec{V}$ that depends directly on the level set function ϕ. As an example, we consider motion by mean curvature where the interface moves in the normal direction with a velocity proportional to its curvature; i.e., $\vec{V} = -b\kappa\vec{N}$, where $b > 0$ is a constant and κ is the curvature. When $b > 0$, the interface moves in the direction of concavity, so that circles (in two dimensions) shrink to a single point and disappear. When $b < 0$, the interface moves in the direction of convexity, so that circles grow instead of shrink. This growing-circle effect leads to the growth of small perturbations in the front including those due to round-off errors. Because $b < 0$ allows small erroneous perturbations to incorrectly grow into $O(1)$ features, the $b < 0$ case is *ill-posed*, and we do not consider it here. Figure 4.1 shows the motion of a wound spiral in a curvature-driven flow. The high-curvature ends of the spiral move significantly faster than the relatively low curvature elongated body section. Figure 4.2 shows the evolution of a star-shaped interface in a curvature-driven flow. The tips of the star move inward, while the gaps in between the tips move outward.

The velocity field for motion by mean curvature contains a component in the normal direction only, i.e., the tangential component is identically zero. In general, one does not need to specify tangential components when devising a velocity field. Since $\vec{N}$ and $\nabla\phi$ point in the same direction,

S. Osher et al., *Level Set Methods and Dynamic Implicit Surfaces*

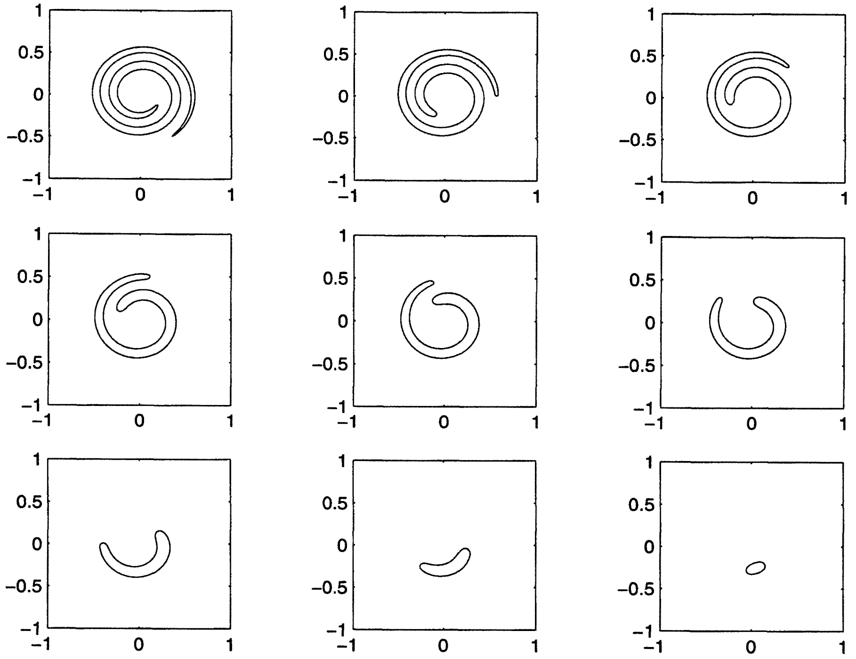

Figure 4.1. Evolution of a wound spiral in a curvature-driven flow. The high-curvature ends of the spiral move significantly faster than the elongated body section.

$\vec{T} \cdot \nabla\phi = 0$ for any tangent vector $\vec{T}$, implying that the tangential velocity components vanish when plugged into the level set equation. For example, in two spatial dimensions with $\vec{V} = V_n\vec{N} + V_t\vec{T}$, the level set equation

$$\phi_t + \left(V_n\vec{N} + V_t\vec{T}\right) \cdot \nabla\phi = 0 \tag{4.1}$$

is equivalent to

$$\phi_t + V_n\vec{N} \cdot \nabla\phi = 0, \tag{4.2}$$

since $\vec{T} \cdot \nabla\phi = 0$. Furthermore, since

$$\vec{N} \cdot \nabla\phi = \frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla\phi = \frac{|\nabla\phi|^2}{|\nabla\phi|} = |\nabla\phi|, \tag{4.3}$$

we can rewrite equation (4.2) as

$$\phi_t + V_n|\nabla\phi| = 0 \tag{4.4}$$

where V_n is the component of velocity in the normal direction, otherwise known as the *normal velocity*. Thus, motion by mean curvature is characterized by $V_n = -b\kappa$.

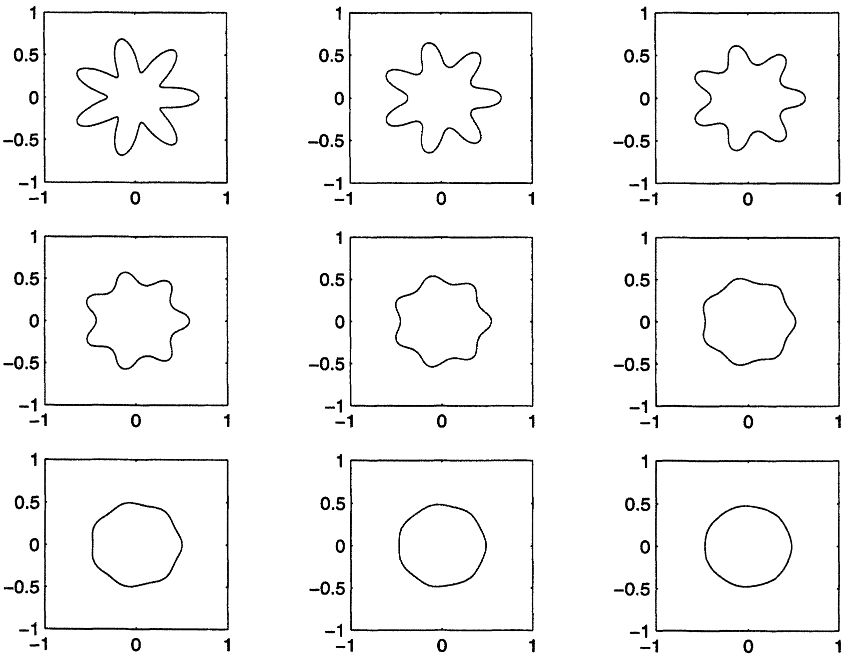

Figure 4.2. Evolution of a star-shaped interface in a curvature-driven flow. The tips of the star move inward, while the gaps in between the tips move outward.

Equation (4.4) is also known as the equation of the level set equation. Like equation (3.2), equation (3.2) is used for externally generated velocity fields, while equation (4.4) is used for (internally) self-generated velocity fields. As we shall see shortly, this is more than a notational difference. In fact, slightly more complicated numerical methods are needed for equation (4.4) than were proposed in the last chapter for equation (3.2).

Plugging $V_n = -b\kappa$ into the level set equation (4.4) gives

$$\phi_t = b\kappa|\nabla\phi|, \tag{4.5}$$

where we have moved the spatial term to the right-hand side. We note that $b\kappa|\nabla\phi|$ is a *parabolic* term that cannot be discretized with an upwind approach. When ϕ is a signed distance function, equation (4.5) becomes the heat equation

$$\phi_t = b\triangle\phi, \tag{4.6}$$

where ϕ is the temperature and b is the thermal conductivity. The heat equation is the most basic equation of the parabolic model.

When ϕ is a signed distance function, $b\kappa|\nabla\phi|$ and $b\triangle\phi$ are identical, and either of these can be used to calculate the right-hand side of equation (4.5). However, once this right-hand side is combined with a forward Euler time step (or a forward Euler substep in the case of RK), the new value of ϕ is not a signed distance function, and equations (4.5) and (4.6) can no longer be interchanged. If this new value of ϕ is reinitialized to a signed distance function (methods for doing this are outlined in Chapter 7), then $b\triangle\phi$ can be used in place of $b\kappa|\nabla\phi|$ in the next time step as well. In summary, equations (4.5) and (4.6) have the *same* effect on the interface location as long as one keeps ϕ equal to the signed distance function off the interface. Note that keeping ϕ equal to signed distance off the interface does not change the interface location. It only changes the implicit embedding function used to identify the interface location.

4.2 Numerical Discretization

Parabolic equations such as the heat equation need to be discretized using central differencing since the domain of dependence includes information from all spatial directions, as opposed to hyperbolic equations like equation (3.2), where information flows in the direction of characteristics only. Thus, the $\triangle\phi$ term in equation (4.6) is discretized using the second-order accurate formula in equation (1.9) in each spatial dimension (see equation (2.7)). A similar approach should therefore be taken in discretizing equation (4.5). The curvature κ is discretized using second-order accurate central differencing as outlined in equation (1.8) and the discussion following that equation. Likewise, the $\nabla\phi$ term is discretized using the second order accurate central differencing in equation (1.5) applied independently in each spatial dimension. While these discretizations are only second-order accurate in space, the dissipative nature of the equations usually makes this second-order accuracy sufficient.

Central differencing of $\triangle\phi$ in equation (4.6) combined with a forward Euler time discretization requires a time-step restriction of

$$\triangle t\left(\frac{2b}{(\triangle x)^2}+\frac{2b}{(\triangle y)^2}+\frac{2b}{(\triangle z)^2}\right)<1 \qquad (4.7)$$

to maintain stability of the numerical algorithm. Here $\triangle t$ is $O((\triangle x)^2)$, which is significantly more stringent than in the hyperbolic case, where $\triangle t$ is only $O(\triangle x)$. Enforcing $\triangle t = O((\triangle x)^2)$ gives an overall $O((\triangle x)^2)$ accurate discretization, even though forward Euler is used for the time differencing (i.e., since the first-order accurate $O(\triangle t)$ time discretization is $O((\triangle x)^2)$). Equation (4.5) can be discretized using forward Euler time stepping with the CFL condition in equation (4.7) as well.

The stringent $O((\triangle x)^2)$ time-step restriction resulting from the forward Euler time discretization can be alleviated by using an ODE solver with a larger stability region, e.g., an implicit method. For example, first-order accurate *backward Euler* time stepping applied to equation (4.6) gives

$$\frac{\phi^{n+1} - \phi^n}{\triangle t} = b\triangle\phi^{n+1}, \tag{4.8}$$

which has no time step stability restriction on the size of $\triangle t$. This means that $\triangle t$ can be chosen for accuracy reasons alone, and one typically sets $\triangle t = O(\triangle x)$. Note that setting $\triangle t = O(\triangle x)$ as opposed to $\triangle t = O((\triangle x)^2)$ lowers the overall accuracy to $O(\triangle x)$. This can be improved upon using the *trapezoidal rule*

$$\frac{\phi^{n+1} - \phi^n}{\triangle t} = b\left(\frac{\triangle\phi^n + \triangle\phi^{n+1}}{2}\right), \tag{4.9}$$

which is $O((\triangle t)^2)$ in time and thus $O((\triangle x)^2)$ overall even when $\triangle t = O(\triangle x)$. This combination of the trapezoidal rule with central differencing of a parabolic spatial operator is generally referred to as the *Crank-Nicolson* scheme.

The price we pay for the larger time step achieved using either equation (4.8) or equation (4.9) is that a linear system of equations must be solved at each time step to obtain ϕ^{n+1}. Luckily, this is not difficult given the simple linear structure of $\triangle\phi^{n+1}$. Unfortunately, an implicit discretization of equation (4.5) requires consideration of the more complicated nonlinear $\kappa^{n+1}|\nabla\phi^{n+1}|$ term.

We caution the reader that one cannot substitute equation (4.6) for equation (4.5) when using an implicit time discretization. Even if ϕ^n is initially a signed distance function, ϕ^{n+1} will generally not be a signed distance function after the linear system has been solved. This means that $\triangle\phi^{n+1}$ is not a good approximation to $\kappa^{n+1}|\nabla\phi^{n+1}|$ even though $\triangle\phi^n$ may be exactly equal to $\kappa^n|\nabla\phi^n|$. Although we stress (throughout the book) the conceptual simplifications and computational savings that can be obtained when ϕ is a signed distance function, e.g., replacing $\vec{N}$ with $\nabla\phi$, κ with $\triangle\phi$, etc., we caution the reader that there is a significant and important difference between the two in the case where ϕ is not a signed distance function.

4.3 Convection-Diffusion Equations

The *convection-diffusion equation*

$$\phi_t + \vec{V}\cdot\nabla\phi = b\triangle\phi \tag{4.10}$$

includes both the effects of an external velocity field and a diffusive term. The level set version of this is

$$\phi_t + \vec{V}\cdot\nabla\phi = b\kappa|\nabla\phi|, \tag{4.11}$$

and the two can be used interchangeably if one maintains a signed distance approximation for ϕ off the interface. These equations can be solved using the upwind methods from the last chapter on the $\vec{V} \cdot \nabla\phi$ term and central differencing on the parabolic $b\triangle\phi$ or $b\kappa|\nabla\phi|$ term. A TVD RK time discretization can be used with a time-step restriction of

$$\Delta t \left(\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + \frac{|w|}{\Delta z} + \frac{2b}{(\Delta x)^2} + \frac{2b}{(\Delta y)^2} + \frac{2b}{(\Delta z)^2} \right) < 1 \tag{4.12}$$

satisfied at every grid point.

Suppose the $O(1)$ size b term is replaced with an $O(\triangle x)$ size ϵ term that vanishes as the mesh is refined with $\triangle x \to 0$. Then equation (4.10) becomes

$$\phi_t + \vec{V} \cdot \nabla\phi = \epsilon\triangle\phi, \tag{4.13}$$

which asymptotically approaches equation (3.2) as $\epsilon \to 0$. The addition of an artificial $\epsilon\triangle\phi$ term to the right-hand side of equation (3.2) is called the *artificial viscosity* method. Artificial viscosity is used by many authors to stabilize a central differencing approximation to the convective $\nabla\phi$ term in equation (3.2). This arises in computational fluid dynamics, where terms of the form $\epsilon\triangle\phi$ are added to the right-hand side of convective equations to pick out *vanishing viscosity solutions* valid in the limit as $\epsilon \to 0$. This vanishing viscosity picks out the physically correct *weak solution* when no classical solution exists, for example in the case of a discontinuous shock wave. It is interesting to note that the upwind discretizations discussed in the last chapter have numerical truncation errors that serve the same purpose as the $\epsilon\triangle\phi$ term. First-order accurate upwinding has an intrinsic $O(\triangle x)$ artificial viscosity, and the higher-order accurate upwind methods have intrinsic artificial viscosities with magnitude $O((\triangle x)^r)$, where r is the order of accuracy of the method.

In [146], Sethian suggested an entropy condition that required curves to flow into corners, and he provided numerical evidence to show that this entropy condition produced the correct weak solution for self-interesting curves. Sethian's entropy condition indicates that $\epsilon\kappa|\nabla\phi|$ is a better form for the vanishing viscosity than $\epsilon\triangle\phi$ for dealing with the evolution of lower-dimensional interfaces. This concept was rigorized by Osher and Sethian in [126], where they pointed out that

$$\phi_t + \vec{V} \cdot \nabla\phi = \epsilon\kappa|\nabla\phi| \tag{4.14}$$

is a more natural choice than equation (4.13) for dealing with level set methods, although these two equations are interchangeable when ϕ is a signed distance function.