Numer. Math. (2004) 99: 365–379
Digital Object Identifier (DOI) 10.1007/s00211-004-0566-1

Numerische Mathematik

A convergent monotone difference scheme for motion of level sets by mean curvature

Adam M. Oberman*

Department of Mathematics, Simon Fraser University, 8888 University Drive, Burnaby, BC, V5A 1S6, Canada. e-mail: aoberman@cs.sfu.ca

Received January 13, 2004 / Revised version received September 16, 2004 /
Published online: November 5, 2004 –

Dedicated to the memory of my father.

Summary. An explicit convergent finite difference scheme for motion of level sets by mean curvature is presented. The scheme is defined on a cartesian grid, using neighbors arranged approximately in a circle. The accuracy of the scheme, which depends on the radius of the circle, dx, and on the angular resolution, $d\theta$, is formally $O(dx^2 + d\theta)$. The scheme is explicit and nonlinear: the update involves computing the median of the values at the neighboring grid points. Numerical results suggest that despite the low accuracy, acceptable results are achieved for small stencil sizes. A numerical example is presented which shows that the centered difference scheme is non-convergent.

Mathematics Suject Classification (2000): 35K65, 35K55, 65M06, 65M12

1 Introduction

There has been a great deal of interest in motion by mean curvature, in terms of both theory and computation. The level set method has been successful both as a framework for the theoretical study and as a numerical method for the simulation of the motion. Applications of motion by mean curvature, and of related geometric motions, can be found in many areas, including differential geometry, fluid dynamics, combustion, front propagation [Set99, OF03] and image processing [OP03].

* The author would like to thank P.E. Souganidis for valuable discussions, and the University of Texas at Austin for its hospitality during the course of this work.

Despite the great deal of interest, no practical, provably convergent numerical scheme has been proposed for the equation for motion of level sets by mean curvature.

The equation for motion of level sets by mean curvature in $\mathbb{R}^n$ is given by

$$\text{(MC)} \qquad \begin{aligned} u_t &= \Delta_1 u \equiv |Du| \operatorname{div}\left(\frac{Du}{|Du|}\right) \\ &\equiv \sum_{i=1}^{n} u_{x_i x_i} - \frac{1}{|Du|^2} \sum_{i,j=1}^{n} u_{x_i x_j} u_{x_i} u_{x_j}. \end{aligned}$$

This equation arises from the well-known level set method, which gives the normal velocity, v_n, of the level set of the function $u(x)$ as $v_n = u_t/|Du|$. The unit normal of the level set is given by the direction of the gradient, $Du/|Du|$, and the mean curvature is the divergence of the unit normal, resulting in (MC).

In two dimensions, the spatial operator Δ_1 can be rewritten in a suggestive form. Writing $\mathbf{n} = (u_x, u_y)/|Du|$ and $\mathbf{t} = \mathbf{n}^{\perp}$ for the unit normal and unit tangent, respectively, gives

$$\Delta_1 u = \frac{1}{|Du|^2}\left(u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}\right) = \frac{\partial^2 u}{\partial \mathbf{t}^2}.$$

Generalizing this observation to higher dimensions, the spatial operator Δ_1 can be regarded as the $(n-1)$-dimensional Laplacian restricted to the tangent space of the graph of the level set of the function u.

Previous work A numerical method for the motion of level sets by mean curvature appeared in [OS88]. Each level set of the partial differential equation (MC) moves with velocity proportional to the mean curvature. Existence and uniqueness of viscosity solutions to (MC) was proven by [ES91] and [CGG91]. Motion (of one level set) by mean curvature also arises as the singular limit of a semilinear reaction diffusion equation. This approach leads to indirect numerical methods [BG95], [ESS92]. The Bence-Merriman-Osher scheme [MBO94], which can be viewed as the singular limit of the reaction diffusion equation, consists of solving the heat equation, followed by thresholding. Finite element schemes which solve (MC) directly have been built [Wal96], but the theory in this case does not ensure uniqueness of solutions. A class of difference schemes for quasilinear PDE, including motion by mean curvature, was presented in Crandall and Lions [CL96]. The schemes are not practical, since they require using a very large stencil.

1.1 Viscosity solutions

The theory of viscosity solutions is a powerful tool for proving existence and uniqueness results for a wide class of nonlinear scalar partial differential equations. While the theory has a historical connection to the method of vanishing viscosity, the two should not be confused.

Without going into needless technical detail, we highlight parts of the theory which provide relevant background material to this work. For further details, we refer the interested reader to the following references. An introduction to viscosity solutions for Hamilton-Jacobi equations and optimal control can be found in Chapter 10 of [Eva98]. A more extensive introductory work is [Cra97], which contains exercises for the reader. The standard reference is [CIL92], which includes a survey of the range of equations covered by the theory (pages 4-9).

Viscosity solutions apply to nonlinear elliptic equations of the form $F(x, u, Du, D^2u) = 0$, or nonlinear parabolic equations of the form $u_t = F(x, u, Du, D^2u)$. The entire class of equations covered by theory, which includes (MC), is called *degenerate elliptic*, a term which includes both the elliptic and parabolic case.

The theory allows for *non-smooth solutions*. In practice we often can work with continuous functions, while treating them as if they were smooth. This is a consequence of the definition of viscosity solutions, which involves touching candidate solutions from above or below by a smooth test function, and using the derivatives of the test function in the equation.

An important property which holds for solutions of degenerate elliptic equations is the *comparison principle*. A generalization of the maximum principle, the comparison principle says that if u and v are solutions of a degenerate elliptic equation, and if $u \leq v$ on the boundary of the domain, then $u \leq v$ in the entire domain. In particular for parabolic equations: if $u \leq v$ initially and on the boundary, then $u \leq v$ for all time.

Another important property is the *stability* of solutions under perturbations. Roughly speaking, this property says that solutions of perturbed equations converge pointwise to solutions of the unperturbed equation, provided that each of the equations under consideration is degenerate elliptic. The example which motivated the theory, and from which the name viscosity solutions is derived, is to add a vanishing amount of the Laplacian. The notion of perturbation is very broad, and includes finite difference approximations [BS91].

While the theory of viscosity solutions is a powerful tool for proving existence and uniqueness results, *regularity* of solutions must be addressed by other means. Regularity results for the underlying equation impinge on the possible accuracy of solution methods. For difference methods which typically rely on Taylor series expansions for consistency, the accuracy of

the method is limited by the smoothness of solutions. Consequently, while formal accuracy estimates may obtained by routine methods for numerical schemes, proving these estimates is generally more difficult, and may require a recapitulation of the regularity proof.

1.2 Convergence of difference schemes

The approximation theory set out in [BS91] applied to a broad class of difference schemes. In this section we sacrifice generality for the sake of clarity and describe those parts of the convergence theory which apply to difference schemes.

The properties needed for convergence are the following: monotonicity, consistency, and stability. It bears mentioning that there is no accuracy requirement for convergence. For difference schemes, simple conditions for monotonicity can be derived [Obe05], and in a restricted context, the stability requirement, which is quite weak, follows.

Monotonicity Monotonicity for difference schemes is a discrete version of the comparison principle. Monotonicity is the requirement that if u, v are solutions of a scheme with boundary data g, h, respectively, then $g \leq h$ (on the boundary) implies $u \leq v$ (on the entire domain). For one-step parabolic schemes the conditions simplifies to the following.

Definition 1 (Monotonicity) *The explicit finite difference scheme is monotone provided that the value of the solution at the next time step is a nondecreasing function of the values at the previous step.*

This condition is simple to check at each grid point. For example, the explicit Euler method for the heat equation gives

$$u_j^{n+1} = (1 - \lambda)u_j^n + \lambda(u_{j-1}^n + u_{j+1}^n)/2,$$

where $\lambda = 2\,dt/dx^2$. The value u_j^{n+1} is given explicitly in terms of values at the previous time step. It is a non-decreasing function of these values provided $0 \leq \lambda \leq 1$. In this case the condition for monotonicity coincides with the usual Courant-Freidrichs-Lewy (CFL) condition. The reader may verify that the implicit Euler discretization is monotone for any positive value of λ.

Consistency The substantial part of the consistency requirement for difference schemes is a verification of consistency by the usual method of Taylor expansions. While solutions need not be smooth, the theory requires that we check consistency only for smooth functions. However care must be taken in the definition of consistency at points where the PDE is singular, as when

$Du = 0$ in (MC). Nevertheless, at singular points, the requirements are weaker, so these points will not usually pose additional difficulties.

We define consistency in the interior of the domain as follows.

Definition 2 (Consistency) *The numerical scheme* $\mathcal{F}^{dx,d\theta}$ *is* consistent *with* (MC) *if for every smooth function* ϕ*, and for every* $x, t > 0$

$$\lim_{dx,d\theta \to 0} \mathcal{F}^{dx,d\theta}(\phi) = \phi_t - \Delta_1 \phi \tag{1}$$

at (x, t) *if* $D\phi(x, t) \neq 0$*, and*

$$\phi_t + \lambda \leq \liminf_{dx,d\theta \to 0} \mathcal{F}^{dx,d\theta}(\phi) \leq \limsup_{dx,d\theta \to 0} \mathcal{F}^{dx,d\theta}(\phi) \leq \phi_t + \Lambda \tag{2}$$

at (x, t)*, where* λ, Λ *are the least and greatest eigenvalues, respectively, of* $-D^2\phi(x, t)$*, otherwise.*

Stability The stability requirement is satisfied if the numerical solutions are bounded uniformly in time. This requirement follows from the (discrete) comparison principle: numerical solutions are bounded by the maximum and minimum of the initial values.

1.3 Non-convergence of the centered difference scheme

An obvious approach to building a difference scheme for (MC) is to simply combine centered finite differences for each of the terms in the equation. We present a numerical experiment which demonstrates that this approach is not convergent.

Discretization The example uses a nine point stencil on a uniform grid, with centered differences for the u_x, u_y, u_{xx}, and u_{yy} terms, and the symmetric discretization (alternating signs at the stencil points $(\pm dx, \pm dx)$) for the u_{xy} term.

Boundary conditions The function $\sin(2\pi(x - y))$, which is a steady solution of (MC), is used as initial data on the unit square with periodic boundary conditions.

Results The monotone scheme was exact, leaving the solution unchanged to within numerical precision. Using the centered difference scheme, the solution contracted rapidly, tending towards a constant. On a 32×32 grid after a time of 0.05 the oscillation of the function decreased by a factor of approximately 65. On a 128×128 grid after the same time the factor was approximately 400.

Remark 1 While one particular choice of discretization was made, we claim that the behavior is universal amongst these types of schemes for the following reason. Nine point schemes will generically use information from each of the neighbors, while the geometry of the equation dictates that only information from grid points along a particular direction (the tangent direction) should be used.

Remark 2 Another scheme for (MC) is to compute the gradient using nearby data, and then compute the second derivative in a direction determined by the gradient. This method is not monotone: increasing the values at the grid points used to compute the gradient will change the resulting direction, and the resulting value of the second derivative may decrease.

1.4 Outline of the monotone scheme

The monotone scheme uses as neighbors lattice points chosen to be close to a uniform arrangement about a circle. Computing the spatial operator is simple: $\Delta_1 u = 2(u(x) - u_*)/dx^2 + O(dx^2 + d\theta)$, where dx is the radius of the circle, $d\theta$ is the maximum angle in radians between points on the circle, $u(x)$ is the value at the center of circle, and u_* is the median of the values of the neighbors. The time discretization is simply explicit Euler, with time step dt restricted by the nonlinear CFL condition $dt \leq dx^2/2$.

In order for the scheme to converge, both dx, and $d\theta$ must go to zero, although they may do so independently. Call the discretization parameter, $d\theta$, the *directional resolution*. Increasing the directional resolution is accomplished by enlarging the number of neighboring grid points used in the calculation. Define n_θ to be the width of the stencil and n_S to be the number of neighbors used.

Remark 3 The need for a larger stencil is supported by [MW53], which shows that for a fixed grid, it is possible to find linear elliptic PDEs for which no monotone, second order accurate discretization exists. Although the proof is non-constructive, a simple example with this property is the second derivative in a direction with an irrational slope [CL96].

However this negative result can be misleading, since it places too much emphasis on the accuracy condition. Addressing the example, the line with irrational slope comes arbitrarily close to a lattice point, and using a nearby lattice point gives a valid (but less accurate) representation of the operator. For this reason, we introduce another parameter, $d\theta$, to represent the directional resolution of a choice of neighbors. In fact, with sufficient resolution in dx, $d\theta$, monotone discretizations of linear elliptic operators can be found, accurate to $O(dx^2 + d\theta)$.

1.5 The scheme in its simplest form

In this section we present the simplest version of the scheme. Despite its limited accuracy, this version of the scheme may be of practical use. A possiblility is in applications to image processing, where the data itself has a natural discrete structure.

This version of scheme is presented with an arbitrary spatial resolution, dx, but with a nine point stencil. This fixes the directional resolution, $d\theta$, to be $2\pi/8$. Setting the time step $dt = dx^2/2$, (the largest possible for stability) gives the following scheme.

1. Sort the values of the eight nearest neighbors, $u_1 \leq \cdots \leq u_8$.
2. Compute the median, $u_* = (u_4 + u_5)/2$.
3. Replace the current value by the median, $u(x, y, t + dt) = u_*(x, y, t)$.
4. Iterate.

The scheme is *monotone*: increasing any of the neighboring values will lead to an increase (or no change) in the value of the solution.

Accuracy: given data from a twice differentiable function, the scheme computes two derivatives (with accuracy $O(dx^2)$) in the grid direction closest to the direction of the tangent of the level set. For the current scheme, the directions vectors are in the axial and the diagonal directions. The nine-point scheme computes one of u_{xx}, u_{yy}, and $u_{xx} \pm 2u_{xy} + u_{yy}$.

2 Presentation and convergence proof of the scheme

In this section we present the scheme. The scheme can be implemented in general (structured, unstructured) geometries, in two or more dimensions. For concreteness, we first address the scheme on a uniform grid in two dimensions. In this context, a detailed convergence proof is given. The requirements for the choice of neighbors specified so that the scheme may easily be adapted to an unstructured grid. The extension to higher dimensions is then addressed.

2.1 The scheme in two dimensions

We begin with the spatial operator and suppress the time dependence. The scheme is defined in the interior of the domain. The uniform grid depends on two parameters: dx, and n_θ. The spatial resolution is dx, which should not be confused with the lattice spacing, defined to be dx/n_θ.

Define the scheme at each point in terms of the reference point (x_0, y_0), and the n_S neighbors at $(x_1, y_1), \ldots, (x_{n_S}, y_{n_S})$. The values of the solution at point (x_i, y_i) are denoted u_i. The *direction vectors* at (x_0, y_0) are the unit vectors, $\mathbf{v}_i$, in the direction of $(x_i, y_i) - (x_0, y_0)$, $i = 1, \ldots, n_S$.

We begin by specifying the neighbors, which are distributed on lattice points near the circle of radius dx. This is followed by defining the spatial discretization in terms of the values at the neighbors. Then the explicit Euler method is used for the time step.

Finally monotonicity and consistency of the resulting scheme is established.

A choice of lattice points near the circle of radius dx. The scheme requires that the neighbors be located close to a uniform distribution about a circle of radius dx. For the purposes of the convergence proof, we specify a choice of points for arbitrarily small $d\theta$. In practice, relatively few neighbors may be used, and they may be chosen by hand. Particular choices of neighbors are specified in the final section.

The number of neighbors to be chosen in the stencil is $n_S = 4\lfloor \pi n_\theta/4 \rfloor$. (Here the "floor" operator $\lfloor x \rfloor$ denotes the greatest integer less than x.) Let $d\theta = 2\pi/n_S$. Choose the neighbors in the first quadrant as follows,

$$(x_i, y_i) = dx\left(\frac{1}{n_\theta}\lfloor n_\theta \cos(i\, d\theta)\rfloor, \frac{1}{n_\theta}\lfloor n_\theta \sin(i\, d\theta)\rfloor\right), \tag{3}$$

for $i = 0, \ldots, \left(\frac{n_S}{4} - 1\right)$ then rotate the points by $\pi/2$, π, and $3\pi/2$ to determine the neighbors in the remaining three quadrants. (The points are distinct.) Thus for $i = 0, \ldots n_S$,

$$(x_i, y_i) = dx(\cos(i\, d\theta), \sin(i\, d\theta)) + (e_i, f_i), \tag{4}$$

where $0 \le e_i, f_i \le dx/n_\theta$.

The scheme Sort the values $\{u_1, \ldots u_{n_S}\}$ in increasing order, and let u_* be the *median*, the average of the middle two sorted values. (There are an even number of neighbors). The spatial operator $\Delta_1 u$ is discretized as $2(u_0 - u_*)/dx^2$. The explicit Euler discretization is then used to get the explicit solution map,

$$u(\cdot, t + dt) = \left(1 - 2\frac{dt}{dx^2}\right) u(\cdot, t) + 2\frac{dt}{dx^2} u_*(\cdot, t), \tag{5}$$

where the time step is bounded by the nonlinear CFL condition,

$$dt \le \frac{dx^2}{2}. \tag{CFL}$$

Accuracy As we show below, the (formal) accuracy of the spatial scheme is

$$\frac{u_0 - u_*}{dx^2} - \Delta_1 u = O(dx^2 + d\theta). \tag{6}$$

The time accuracy is $O(dt) = O(dx^2)$.

Monotonicity We begin with the observation that the median of n_S numbers is a nondecreasing function of those values. Therefore, the map defined by (5), which, provided (CFL) holds, gives $u_0(t + dt)$, as a convex combination of $u_0(t)$ and the median of the neighboring values, $u_1(t), \ldots, u_{n_S}(t)$, is nondecreasing.

Consistency To satisfy the consistency requirement, we need to check that (1) and (2) hold. This involves a direct argument involving Taylor expansions.

1. Given a smooth function u, and the grid points (x_i, y_i), $i = 1, \ldots, n_S$, defined according to the prescription (3), let u_* be the median of the values $u_i = u(x_i, y_i)$ for $i = 1, \ldots, n_S$.

2. First we show

$$u_* = \frac{1}{2}(u(x_i, y_i) + u(-x_i, -y_i)) + O(dx\, d\theta + dx^2) \tag{7}$$

where the angle $i\, d\theta$ is the angle closest (amongst the grid angles $i\, d\theta$) to the direction of the tangent vector $(-u_y, u_x)$. From the Taylor expansion,

$$u_i = u_0 + Du \cdot (x_i, y_i) + O(dx^2), \quad i = 1, \ldots, n_S,$$

write $Du = |Du|(\cos(\psi), \sin(\psi))$, for some angle ψ, and use (4) and $d\theta = O(1/n_\theta)$, to obtain

$$u_i = u_0 + |Du| \cos(\psi - i\, d\theta) + O(dx\, d\theta + dx^2), \quad i = 1, \ldots, n_S.$$

Next sort the values, allowing for an error in the ordering of $O(dx\, d\theta + dx^2)$. Measuring distance on the circle of radius dx, let i_1 be the index corresponding to an angle closest to ψ, and let i_2 be the next closest, and so on, until the last index, i_{n_S} is found. For distances less than π, the cosine is a decreasing function of distance from the origin, so

$$\cos(\psi - i_1\, d\theta) \geq \cos(\psi - i_2\, d\theta) \geq \cdots \geq \cos(\psi - i_{n_S}\, d\theta).$$

Thus

$$u_{i_1} \geq u_{i_2} \geq \cdots \geq u_{i_{n_S}}, \quad \text{up to } O(dx\, d\theta + dx^2).$$

Further, by the symmetric choice of points, we may assume that the middle two points point in opposite directions. So (7) is obtained.

3. Next we establish (6) from (7). Then (1) will follow from the fact that the explicit Euler method is consistent with the time derivative. Write $\mathbf{v}$ for unit vector in the direction (x_i, y_i). Using Taylor series expansions in the directions $\mathbf{v}$, $-\mathbf{v}$, we obtain

$$2(u_* - u_0) = (x_i^2 + y_i^2)\mathbf{v}^T D^2 u \mathbf{v} + O(dx^4).$$

Since $\mathbf{v} = \mathbf{t} + O(d\theta)$,

$$2(u_* - u_0) = (x_i^2 + y_i^2)\mathbf{t}^T D^2 u \mathbf{t} + O(dx^4 + dx^2 d\theta).$$

Dividing by dx^2 gives (6). Putting in forward Euler discretization in time gives (1).
4. When the gradient is zero, we need only show the weaker consistency condition (2). But this condition is satisfied by the average of the values at any two neighbors, so it certainly holds for the median.

Convergence Since the scheme is consistent and monotone, solutions converge, in the limit $dx, d\theta \to 0$ to the unique viscosity solution of (MC).

Boundary conditions Since (at least formally) solutions of (MC) have the property that level sets move with normal velocity equal to the mean curvature, the boundary conditions are irrelevant to level sets which do not contact the boundary. The same should be true (at least approximately) of numerical solutions. The implementation of boundary conditions should not affect the monotonicity property of the scheme. This is indeed the case for Dirichlet boundary conditions.

Rotation invariance The scheme has a discrete *rotation invariance*. For each fixed $d\theta$, the scheme is invariant under rotations by multiples of $d\theta$. This is an approximate statement: the exact angles are those given by the direction vectors, which are distributed at angles close to $d\theta$, as specified by (3).

2.2 Extension to higher dimensions

In this section we outline the extension of the scheme to higher dimensions. From the point of view of implementation, the scheme is quite similar: the spatial discretization is still given by computing the median of the neighbors.

The neighbors should be chosen to be distributed approximately uniformly on the sphere of radius dx. The $d\theta$ error is measured by the maximum aperture of a cone centered at the reference point which does not intersect the neighbors. The mean curvature operator is again discretized by computing the quotient of the difference between the reference point and the median of the neighbors with dx^2. The three dimensional version of the nine point scheme is the 27 point nearest neighbor scheme.

Convergence in higher dimensions Convergence of the scheme is established by verifying monotonicity and consistency. Monotonicity follows as before. Demonstrating consistency requires additional work, outlined below.
1. To $O(dx^2)$, the mean curvature operator is equal to the Laplacian restricted to the tangent space of the level set. So it is enough to show that the median of the sampled values is (approximately) equal to the mean of the values distributed on a sphere of radius dx on the tangent space of the level set, since this gives a consistent discretization of the Laplacian.

2. Partition the sampled values into three sets according to the direction vector of their location as follows. Those with a (non-negligible) positive component in the direction of the normal, those with a (non-negligible) negative component in the direction of the normal, and the remainder, which lie in (or near) the tangent space to the level set. Within accuracy bounds, there are an equal number of values in the first and second sets; the largest values will be in the first set, the smallest will be in the second set. Thus the median of the sampled values will be the median of the values in the third set.
3. At this point it is enough to consider the problem of sampled values from a quadratic function on the intersection of a sphere with the tangent plane to the level set. Thus it is enough to show that for a quadratic function with no linear part, the mean of the sampled values is equal to the median. This is an elementary argument, which we give below for the three dimensional case.
4. Dropping the constant term, and choosing coordinates on the tangent plane, we sample values of a function of the form $f(x, y) = ax^2 + 2bxy + cy^2$ on the circle. We claim the mean, (range), and median are all $\mu = (a + c)/2$. The first two assertions are elementary. The third is true because we can pair off values of f symmetrically about the mean. Write f in polar coordinates

$$f(\theta) = \frac{a+c}{2} + b\sin(2\theta) + \frac{a-c}{2}\cos(2\theta).$$

Splitting the interval $[0, \pi]$ into four equal segments, there is a bijection from the first and third segments, and from the second and fourth segments between values of the form $\mu + v$ and values of the form $\mu - v$. Likewise a bijection holds for the interval $[\pi, 2\pi]$. As a result of this bijection, the median is equal to the mean.

3 Validation and numerics

We begin with a numerical consistency check. This is followed by testing the accuracy of the numerical solution against an exact solution, using different stencils and grid sizes.

3.1 Consistency

The accuracy in $d\theta$ depends on how the direction of the tangent lines up with the direction vectors of the neighboring grid points. The error is greatest when the tangent line lies midway between two direction vectors, and negligible when the tangent lines up with a direction vector.

Taking a quadratic polynomial constructed so that the tangent direction does not line up with a grid direction vector, we verify (6) using points uniformly distributed on the circle of radius dx, with $d\theta = 2\pi/n_S$. Next, we

Table 1. The discretization error with polynomials, with n_θ points about the circle of radius dx, and with n_θ levels, taking points on boundary of the square for polynomials $p1 = 6x + 1.25y + 4.5x^2/2 + 2.8xy - 5.2y^2/2$, $p2 = x - .01y + .5x^2/2 + 3.1xy - 2y^2/2$

Circle: n_θ	16	32	64	200	1000	10^5
error p1	−.1553	−.0123	−.0123	0.016	0.0024	.00008

Square: n_θ	2	4	8	16	32	64	128	256
error p1	−.11	−.11	−.11	.11	.05	−.027	.013	−.0068
error p2	.06	.06	.06	.06	.06	.03	.01	.01

show that for lattice points on the *boundary* of the square of radius dx, with n_θ levels, taking the median still gives a consistent scheme. This demonstrates the robustness of the discretization, since the distance of the points varies from dx to $\sqrt{2}/2\,dx$. The results are presented in Table 1. The improvements in accuracy occur in jumps as new direction vectors come closer to the direction of the tangent.

3.2 Implementation and validation

The implementation was performed in MATLAB. The code was under a hundred lines long, and ran (for the 80×80 grid) in a few seconds on a laptop.

Five sets of stencils were used, corresponding to different values of $d\theta$. The schemes are laid out in Table 2. The table lists the neighbors in the first quadrant, which are rotated to get the full set. The schemes are illustrated in Figure 1.

The numerical error was computed using the exact viscosity solution

$$u(x, y, t) = \min\left\{\frac{x^2 + y^2 - 1}{2} + t, 0\right\},$$

whose level sets are circles (or they are flat). While the solution is radial (with respect to the origin), it is not *locally* radial, so this solution is a valid

Table 2. The schemes used for a given stencil width, n_θ

n_θ	n_S	neighbors in the first quadrant
1	4	(1, 0)
1	8	(1, 0), (1, 1)
2	12	(2, 0), (2, 1), (1, 2)
3	16	(3, 0), (3, 1), (2, 2), (1, 3)
4	32	(4, 0), (4, 1), (4, 2), (3, 2), (3, 3), (2, 3), (2, 4), (1, 4)

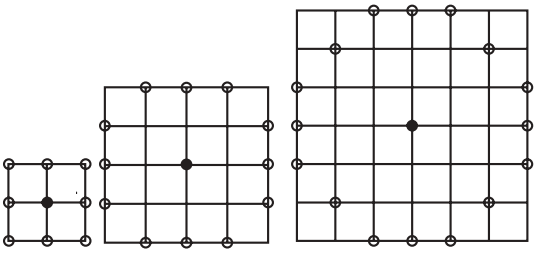

Fig. 1. Illustration of the schemes used for $n_S = 8, 12, 16$

Table 3. Error in the maximum norm for different schemes, as a function of the number of grid points used and the stencil size

Grid	$n_S = 4$	$n_S = 8$	$n_S = 12$	$n_S = 16$	$n_S = 32$
20×20	.110	.080	.035	.020	.024
40×40	.115	.080	.035	.024	.022
80×80	.119	.080	.035	.027	.013
160×160	.118	.080	.035	.027	.010
240×240	*	.080	.035	.027	.010
360×360	*	*	.035	.027	.010

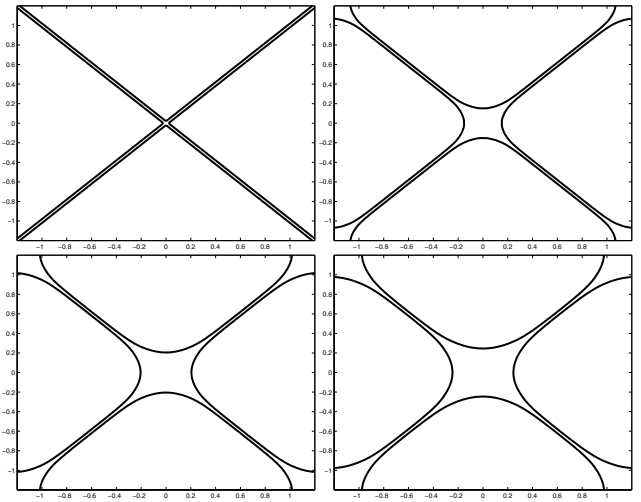

Fig. 2. Contour plots of the $-.02$, and .02 contours at times 0, .015, .03, .045

test of the $d\theta$ error. Taking the minimum with zero is convenient as it allows homogeneous Neumann boundary conditions to be used. The numerical error in the maximum norm, after solving for $t = .2$ is presented in Table 3.

Finally, we present an example which demonstrated the fattening phenomena [ES91]. Taking as initial data $|x| - |y|$, we compute the solution using

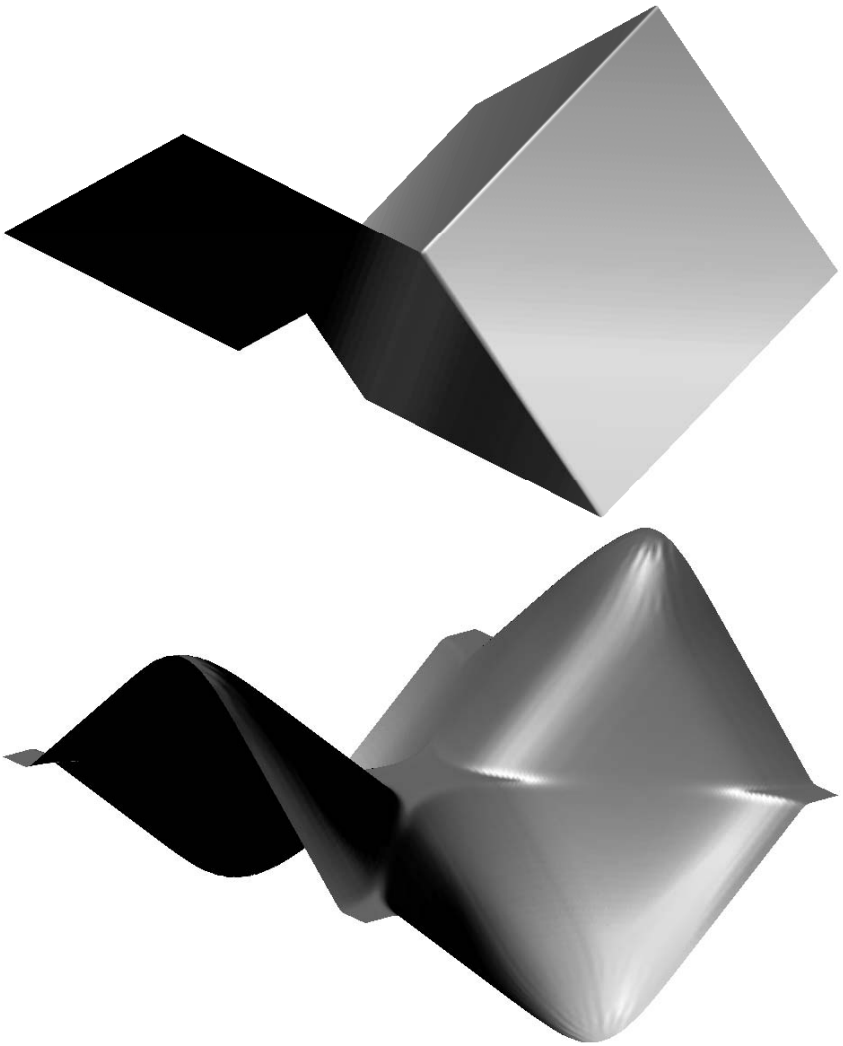

Fig. 3. Surface plot: initial data, and solution at time .03

the $n_\theta = 4$ scheme on a 200^2 grid. The solution is displayed in Figures 2 and 3.

References

[BG95] Barles, G., Georgelin, C.: A simple proof of convergence for an approximation scheme for computing motions by mean curvature. SIAM J. Numer. Anal. **32**(2), 484–500 (1995)

[BS91] Barles, G., Souganidis, P.E.: Convergence of approximation schemes for fully nonlinear second order equations. Asymptotic Anal. **4**(3), 271–283 (1991)

[CGG91] Chen, Y.G., Giga, Y., Goto, S.: Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations. J. Differential Geom. **33**(3), 749–786 (1991)

[CIL92] Crandall, M.G., Ishii, H., Lions, P.-L.: User's guide to viscosity solutions of second order partial differential equations. Bull. Amer. Math. Soc. (N.S.) **27**(1), 1–67 (1992)

[CL96] Crandall, M.G. Lions, P.-L.: Convergent difference schemes for nonlinear parabolic equations and mean curvature motion. Numer. Math. **75**(1), 17–41 (1996)

[Cra97] Michael G. Crandall. Viscosity solutions: a primer. In: Viscosity solutions and applications (Montecatini Terme, 1995), Springer, Berlin, 1997, pp. 1–43

[ES91] Evans, L.C., Spruck, J.: Motion of level sets by mean curvature. I. J. Diff. Geom. **33**(3), 635–681 (1991)

[ESS92] Evans, L.C., Soner, H.M., Souganidis, P.E.: Phase transitions and generalized motion by mean curvature. Comm. Pure Appl. Math. **45**(9), 1097–1123 (1992)

[Eva98] Evans, L.C.: Partial differential equations. American Mathematical Society, Providence, RI, 1998

[MBO94] Merriman, B., Bence, J.K., Osher, S.J.: Motion of multiple functions: a level set approach. J. Comput. Phys. **112**(2), 334–363 (1994)

[MW53] Motzkin, T.S., Wasow, W.: On the approximation of linear elliptic differential equations by difference equations with positive coefficients. J. Math. Physics **31**, 253–259 (1953)

[Obe05] Oberman, A.M.: Convergent difference schemes for degenerate elliptic and parabolic equations: Hamilton-Jacobi equations and free boundary problems. SINUM, To appear 2005

[OF03] Osher, S., Fedkiw, R.: Level set methods and dynamic implicit surfaces. volume 153 of Appl. Math. Sci. Springer-Verlag, New York, 2003

[OP03] Osher, S., Paragios, N.: (eds.), Geometric level set methods in imaging, vision, and graphics. Springer-Verlag, New York, 2003

[OS88] Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. J. Comput. Phys. **79**(1), 12–49 (1988)

[Set99] Sethian, J.A.: Level set methods and fast marching methods. volume 3 of Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, second edition, 1999. Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science.

[Wal96] Walkington, N.J.: Algorithms for computing motion by mean curvature. SIAM J. Numer. Anal. **33**(6), 2215–2238 (1996)