Sixth Mississippi State Conference on Differential Equations and Computational Simulations, *Electronic Journal of Differential Equations*, Conference 15 (2007), pp. 251–264.
ISSN: 1072-6691. URL: http://ejde.math.txstate.edu or http://ejde.math.unt.edu
ftp ejde.math.txstate.edu (login: ftp)

GAME-THEORETIC SCHEMES FOR GENERALIZED CURVATURE FLOWS IN THE PLANE

MATTHEW RUDD

Dedicated with gratitude and best wishes to Klaus Schmitt on his 65-th birthday

Abstract. Extending some recent work by Kohn and Serfaty, we discuss a class of two-player discrete-time games whose value functions approximate the solutions of related geometric partial differential equations. Of particular interest are the numerical implications of the resulting game-theoretic approximation schemes for geometric motions.

1. Introduction

If a hypersurface $\Gamma_0 \subset \mathbb{R}^N$ moves over time with the known normal velocity V, then its subsequent evolution yields a family $\{\Gamma_t : t \geq 0\}$ of subsets of $\mathbb{R}^N$. The subsets Γ_t may continue to have codimension one, or they may undergo topological changes by breaking, merging, or fattening. Among the various techniques used to analyze moving interfaces in the presence of such possible singularities, the level set approach has proven very effective and leads to the geometric equations studied below. For the sake of completeness, let us recall the level set description of propagating fronts: we look for a function $u : \mathbb{R}^N \times \mathbb{R}^+ \to \mathbb{R}$ such that

$$\Gamma_t = \{x \in \mathbb{R}^N : u(x,t) = 0\}\,, \quad \text{for } t > 0\,,$$

by observing (cf. the derivation in [28]) that u solves the initial-value problem

$$u_t + |Du|\, V = 0\,, \quad u(\cdot, 0) = u_0 \tag{1.1}$$

in an appropriate sense, where the initial datum $u_0 : \mathbb{R}^N \to \mathbb{R}$ is such that

$$\Gamma_0 = \{x \in \mathbb{R}^N : u_0(x) = 0\}\,.$$

Of course, the specific nature of the level-set equation in (1.1) depends on the form of the normal velocity function V. For the simple function $V = V(x,t)$, for example, we obtain the Hamilton-Jacobi problem

$$u_t + |Du|\, V(x,t) = 0\,, \quad u(\cdot, 0) = u_0\,, \tag{1.2}$$

2000 *Mathematics Subject Classification.* 35K10.
Key words and phrases. Geometric level set equations; curvature flows; dynamic programming.

Published February 28, 2007.

and more general Hamilton-Jacobi equations arise for velocity functions of the form $V = V(n, x, t)$, where

$$n = n(x,t) = \frac{Du}{|Du|} \tag{1.3}$$

is the unit normal to the front at the point x at time t. There are well-known connections between such first-order initial-value problems and deterministic games involving either one player (i.e., optimal control problems) or two players (see, e.g., [2, 14, 17, 29]).

When the normal velocity V also depends on the spatial derivative of the normal vector, i.e., the tensor

$$Dn = Dn(x,t) = \frac{1}{|Du|}\,(I - n \otimes n)\,D^2 u\,, \tag{1.4}$$

with n given by (1.3), equation (1.1) is second-order. We are particularly interested in normal velocities of the form

$$V = V(x,t,n,Dn) = -\operatorname{Tr}\left[A(n,x,t)Dn\right]$$

for continuous maps $A : \mathbb{R}^N \times \mathbb{R} \times \mathbb{R} \to \left(\mathcal{S}^N\right)^+$, where $\left(\mathcal{S}^N\right)^+$ is the cone of positive-definite, symmetric $N \times N$ matrices. Such a map A yields the initial-value problem

$$u_t - |Du| \operatorname{Tr}\left[A(n,x,t)Dn\right] = 0\,, \quad u(\cdot,0) = u_0\,. \tag{1.5}$$

The explicit expressions in (1.3) and (1.4) expose the degeneracies of this equation more clearly than the shorthand of equation (1.5), but we write the equation in this form for simplicity. For the special case in which $A \equiv I$, equation (1.5) is the mean curvature equation,

$$u_t - |Du|\,\kappa = 0\,, \quad \text{where } \kappa := \operatorname{div}\Big(\frac{Du}{|Du|}\Big) \tag{1.6}$$

is the mean curvature of the front. We therefore refer to the equation in (1.5) as the generalized curvature equation.

Beginning with the independent efforts of Evans and Spruck [18], who concentrated on the mean curvature equation, and Chen, Giga and Goto [11], who studied equations of the form (1.7), several authors have developed a thorough viscosity theory for geometric initial-value problems of the form

$$u_t + F(x,t,Du,D^2u) = 0\,, \quad u(\cdot,0) = u_0\,, \tag{1.7}$$

where *geometric* means that, for any vector p and any symmetric matrix $X \in \mathcal{S}^N$,

$$F(x,t,\lambda p, \lambda X + \mu p \otimes p) = \lambda F(x,t,p,X) \quad \text{for } \lambda > 0\,,\ \mu \in \mathbb{R}\,. \tag{1.8}$$

The theory for problem (1.7) also requires that F be *degenerate elliptic*, i.e., non-increasing in its matrix argument:

$$F(x,t,p,X) \geq F(x,t,p,Y) \quad \text{when } X \leq Y\,. \tag{1.9}$$

Our assumptions above about the map A guarantee that the generalized curvature equation (1.5), when written in the form (1.7), satisfies both (1.8) and (1.9). The basic punchline is that, for a uniformly continuous initial value $u_0 \in UC(\mathbb{R}^N)$, problem (1.7) has a unique uniformly continuous viscosity solution; we refer to [4], [6], [32] and the references therein for relevant definitions and more details. Henceforth, we will concentrate on problems in the plane $\mathbb{R}^2$.

2. Approximating curvature flows with discrete-time games

We begin by describing the basic game studied by Kohn and Serfaty in [23]; Spencer introduced this game in [33] as a tool for combinatorial problems. To set up the game, let the final maturity time $T > 0$ and objective function $u_0 \in UC(\mathbb{R}^2)$ be given. Suppose that the game begins at time $t < T$ at the position $x \in \mathbb{R}^2$, and define the game's time step $\varepsilon > 0$ by

$$\varepsilon := \frac{T-t}{m}, \quad \text{for some integer } m \in \mathbb{N}. \tag{2.1}$$

The two players, Paul and Carol, now alternate their play as follows:

Mean curvature game.

(i) Paul chooses a unit vector $v \in S^1$.
(i) Carol accepts or reverses the direction v chosen by Paul.
(i) Paul moves from x to $x+\sqrt{2\varepsilon}bv$, where $b = \pm 1$ according to Carol's previous play.

Paul and Carol play the next round of the game at time $t + \varepsilon$ at the new position $x + \sqrt{2\varepsilon}\, bv$. By (2.1), the game runs for precisely m rounds, after which their final position is $x_T \in \mathbb{R}^2$ and Paul pays Carol the amount $u_0(x_T)$. Paul and Carol have the opposing goals of minimizing and maximizing this payoff, respectively. Paul's value function is therefore

$$u^\varepsilon(x,t) = \min\{u_0(x_T)\}, \tag{2.2}$$

where the minimum is taken over all possible final states x_T that can be reached in m plays when starting at x at time t.

Kohn and Serfaty have shown in [23] that

$$\lim_{\varepsilon \to 0} u^\varepsilon(x,t) = u(x,t)\,, \tag{2.3}$$

where u solves the terminal-value problem

$$u_t + |Du| \operatorname{Tr}[Dn] = 0\,, \quad u(\cdot, T) = u_0\,. \tag{2.4}$$

By reversing time, we see that u solves the initial-value problem (1.5) in the special case in which $A \equiv I$, i.e., the mean curvature equation. Going backward in time, the level sets of u therefore evolve by mean curvature and can be approximated by the level sets of u^ε. We can exploit this fact to obtain a very simple algorithm for mean curvature flow. For example, suppose that the simple closed curve Γ_0 and $u_0 \in UC(\mathbb{R}^2)$ satisfy

$$\Gamma_0 = \{x \in \mathbb{R}^2 : u_0(x) = 0\}\,. \tag{2.5}$$

Γ_0's evolution by mean curvature yields the sets

$$\Gamma_t = \{x \in \mathbb{R}^2 : u(x, T-t) = 0\}\,, \quad \text{for} \quad t > 0\,,$$

which we can approximate with the sets

$$\Gamma_t^\varepsilon = \{x \in \mathbb{R}^2 : u^\varepsilon(x, T-t) = 0\}\,, \quad \text{for } t > 0\,.$$

In particular, $u^\varepsilon(x, T-\varepsilon)$ will be 0 if and only if Paul can reach Γ_0 in exactly one round of the game; this can only occur if x is the midpoint of a segment of length $2\sqrt{2\varepsilon}$ whose endpoints lie on Γ_0. Similarly, $u^\varepsilon(x, T-2\varepsilon)$ can only be zero if x is the midpoint of a segment of length $2\sqrt{2\varepsilon}$ with endpoints on $\Gamma_\varepsilon^\varepsilon$, and so on. This provides a simple, mesh-free algorithm for mean curvature flow which will be discussed in greater detail below.

To handle the more general geometric initial-value problem (1.5), we must modify the basic game just described. Paul and Carol must now play as follows, where S^1 denotes the unit circle in $\mathbb{R}^2$ with its standard metric and $v^\perp$ denotes a vector perpendicular to v relative to the standard Euclidean inner product, i.e., if $v = (v_1, v_2)$, then $v^\perp := (-v_2, v_1)$.

Generalized curvature game.

(i) Paul chooses a vector $w \in S^1$, thereby defining his displacement vector

$$v := A(w, x, T-t)w^\perp \tag{2.6}$$

as well as the pairing at (x, t):

$$\langle y, z\rangle_A := z^T \left(A(w, x, T-t)^T\right)^{-1} y\,, \quad \text{for } y, z \in \mathbb{R}^2\,. \tag{2.7}$$

(ii) Carol accepts or reverses the vector v defined by (2.6).

(iii) Paul moves from x to $x + \sqrt{2\varepsilon}\, bv$, where $b = \pm 1$ according to Carol's previous play.

Paul and Carol play the next round of the game at time $t + \varepsilon$ at the new position $x + \sqrt{2\varepsilon}\, bv$, and the underlying pairing for the next turn will come from $A(\tilde{w}, x + \sqrt{2\varepsilon} bv, T - t - \varepsilon)$, where $\tilde{w} \in S^1$ is Paul's next choice. As before, the game runs for precisely m rounds, after which their final position is $x_{_T} \in \mathbb{R}^2$ and Paul must pay Carol the amount $u_0(x_{_T})$. Since Paul and Carol again have the opposing goals of minimizing and maximizing this payoff, respectively, Paul's value function is still

$$u^\varepsilon(x, t) = \min\{u_0(x_{_T})\}, \tag{2.8}$$

where the minimum is taken over all possible final states $x_{_T}$ that can be reached in m plays when starting at x at time t.

As with the mean curvature game analyzed in [23], we want to understand the behavior of the value functions u^ε, as $\varepsilon \to 0$, for this generalized game. To analyze their limit, we rely on the fact that each value function u^ε satisfies the dynamic programming equation

$$u^\varepsilon(x, t) = \min_{w_1 \in S^1} \max_{b_1 = \pm 1} \left\{ u^\varepsilon(x + \sqrt{2\varepsilon}\, b_1 v_1, t + \varepsilon) \right\}, \tag{2.9}$$

where the vectors v_1 and w_1 are related by (2.6); the subscripts here emphasize the fact that b_1, w_1 and v_1 correspond to Paul's first play. A similar equation holds for $u^\varepsilon(x + \sqrt{2\varepsilon}\, b_1 v_1, t + \varepsilon)$, thereby advancing from time $t + \varepsilon$ to time $t + 2\varepsilon$:

$$u^\varepsilon(x, t) = \min_{w_1 \in S^1} \max_{b_1 = \pm 1} \min_{w_2 \in S^1} \max_{b_2 = \pm 1} \left\{ u^\varepsilon(x + \sqrt{2\varepsilon}\, b_1 v_1 + \sqrt{2\varepsilon} b_2 v_2, t + 2\varepsilon) \right\}.$$

Continuing in this manner m times yields

$$u^\varepsilon(x, t) = \min_{w_1 \in S^1} \max_{b_1 = \pm 1} \cdots \min_{w_m \in S^1} \max_{b_m = \pm 1} \left\{ u^\varepsilon(x + \sqrt{2\varepsilon} \sum_{i=1}^m b_i v_i, T) \right\}. \tag{2.10}$$

After m rounds, however, we reach the end of the game, where $u^\varepsilon(\cdot, T)$ equals the known terminal value u_0. We may therefore rewrite (2.10) in the form

$$u^\varepsilon(x, t) = \min_{w_1 \in S^1} \max_{b_1 = \pm 1} \cdots \min_{w_m \in S^1} \max_{b_m = \pm 1} \left\{ u_0(x + \sqrt{2\varepsilon} \sum_{i=1}^m b_i v_i) \right\}, \tag{2.11}$$

obtaining a representation formula for u^ε as a function of u_0. In terms of the operator

$$S(\varepsilon) : UC(\mathbb{R}^2) \to UC(\mathbb{R}^2)$$

defined by

$$(S(\varepsilon)u)\,(x) := \min_{w\in S^1}\max_{b=\pm 1}\left\{u(x+\sqrt{2\varepsilon}\,bv)\right\}, \tag{2.12}$$

formula (2.11) becomes

$$u^\varepsilon = (S(\varepsilon))^m\,u_0\,,$$

and we want to determine

$$\lim_{m\to\infty}(S(\varepsilon))^m\,u_0 = \lim_{\varepsilon\to 0}(S(\varepsilon))^m\,u_0\,. \tag{2.13}$$

We will see that this limit is the unique solution u of the terminal-value problem

$$u_t + |Du|\,\mathrm{Tr}\,[\,A(n,x,T-t)Dn\,] = 0\,,\quad u(\cdot,T) = u_0\,, \tag{2.14}$$

where $A : \mathbb{R}^N\times\mathbb{R}\times\mathbb{R} \to \left(\mathcal{S}^N\right)^+$ is continuous and $\left(\mathcal{S}^N\right)^+$ denotes the cone of positive-definite, symmetric $N\times N$ matrices. Before proving this fact, let us briefly describe its numerical implications. Defining the set Γ_0 by (2.5) as before, the sets Γ_t correspond to the generalized curvature flow of Γ_0, and the sets Γ_t^ε are their approximations via the value function u^ε. The point x will belong to $\Gamma_\varepsilon^\varepsilon$ if and only if Paul can reach Γ_0 in one play, i.e., if and only if there exists $w\in S^1$ such that

$$x \pm \sqrt{2\varepsilon}A(w,x,T-\varepsilon)w^\perp \in \Gamma_0\,.$$

This leads to a conceptually simple algorithm for generalized curvature flow, but the nonlinear dependence on w certainly complicates its implementation. We will come back to numerical issues below in Section 3.

The following theorem analyzes the limit (2.13).

Theorem 2.1. *Let $u_0\in UC(\mathbb{R}^2)$ be given, let u be the unique viscosity solution of (2.14), and define $\varepsilon := \frac{T-t}{m}$ for $m\in\mathbb{N}$ and $t<T$. Then*

$$u(\cdot,t) = \lim_{m\to\infty}(S(\varepsilon))^m\,u_0 = \lim_{\varepsilon\to 0}(S(\varepsilon))^m\,u_0\,. \tag{2.15}$$

Proof. According to the general convergence results of Barles and Souganidis ([5]; see also [30],[31]), modified as in [3] to handle the degeneracy in equation (2.14), we must show that our approximation scheme is stable, monotone and consistent. In particular, we must verify that $S(\cdot)$ has the following four properties:

(i) $S(0) = I$,
(ii) $S(\varepsilon)\,(u+c) = S(\varepsilon)u + c$, for any constant c,
(iii) $S(\varepsilon)u \le S(\varepsilon)\tilde{u}$ whenever $u\le\tilde{u}$, and
(iv) for any smooth ϕ and any $x\in\mathbb{R}^2$,

$$\lim_{\varepsilon\to 0}\Big(\frac{(S(\varepsilon)\phi)\,(x)-\phi(x)}{\varepsilon}\Big) = |D\phi(x)|\,\mathrm{Tr}\,[A(n,x,T-t)Dn] \tag{2.16}$$

if $D\phi(x)\neq 0$, where

$$n = \frac{D\phi(x)}{|D\phi(x)|}\quad\text{and}\quad Dn = \frac{1}{|D\phi(x)|}\,(I-n\otimes n)\,D^2\phi(x)\,, \tag{2.17}$$

while

$$\lambda \le \liminf_{\varepsilon\to 0}\Big(\frac{(S(\varepsilon)\phi)\,(x)-\phi(x)}{\varepsilon}\Big) \le \limsup_{\varepsilon\to 0}\Big(\frac{(S(\varepsilon)\phi)\,(x)-\phi(x)}{\varepsilon}\Big) \le \Lambda \tag{2.18}$$

if $D\phi(x) = 0$, where λ and Λ are the extreme eigenvalues of $D^2\phi(x)$.

The first two of these properties are trivial. The third follows easily from the definition of $S(\varepsilon)$: for $u, \tilde{u} \in UC(\mathbb{R}^2)$ with $u \le \tilde{u}$, let $w \in S^1$ be optimal for $\tilde{u}$ with $v = A(w, x, T-t)w^\perp$. Then

$$(S(\varepsilon)u)(x) \le \max_{b=\pm 1} u(x + \sqrt{2\varepsilon} bv) \le \max_{b=\pm 1} \tilde{u}(x + \sqrt{2\varepsilon} bv) = (S(\varepsilon)\tilde{u})(x),$$

verifying the monotonicity of the scheme.

To check the scheme's consistency (property (iv)), pick any smooth ϕ and a point $x \in \mathbb{R}^2$, and suppose, first of all, that $D\phi(x) \neq 0$. Recall the Riemannian structure on $\mathbb{R}^2$ defined by (2.7) and Taylor expand ϕ to obtain

$$\begin{aligned}(S(\varepsilon)\phi)(x) &= \min_{w\in S^1} \max_{b=\pm 1} \left\{ \phi(x + \sqrt{2\varepsilon} bv) \right\} \\ &= \min_{w\in S^1} \max_{b=\pm 1} \left\{ \phi(x) + \sqrt{2\varepsilon} b \langle D\phi(x), v\rangle_A + \varepsilon \langle D^2\phi(x)v, v\rangle_A + o(\varepsilon^{3/2}) \right\},\end{aligned}$$

from which see that

$$\frac{(S(\varepsilon)\phi)(x) - \phi(x)}{\varepsilon} = \min_{w\in S^1} \max_{b=\pm 1} \left\{ \frac{\sqrt{2}b}{\sqrt{\varepsilon}} \langle D\phi(x), v\rangle_A + \langle D^2\phi(x)v, v\rangle_A + o(\varepsilon^{1/2}) \right\}.$$

Choosing

$$w = \frac{D\phi(x)}{|D\phi(x)|}, \tag{2.19}$$

the corresponding displacement vector v is

$$v = \frac{1}{|D\phi(x)|} A(w, x, T-t)\, (D\phi(x))^\perp \tag{2.20}$$

and $\langle D\phi(x), v\rangle_A = 0$; any choice of w for which the latter does not hold will clearly not be optimal for small ε. By the definition of the underlying pairing (2.7), we have

$$\langle D^2\phi(x)v, v\rangle_A = v^T \left(A(w, x, T-t)^{-1} \right)^T D^2\phi(x) v,$$

and a direct calculation using (2.19) and (2.20) reveals that

$$\langle D^2\phi(x)v, v\rangle_A = |D\phi(x)| \operatorname{Tr}\left[A(n, x, T-t) Dn \right],$$

with n and Dn given by (2.17). Property (iv) follows directly in this case. If, instead, $D\phi(x) = 0$, consistency is even simpler, as the first-order term in the Taylor expansion vanishes and the remaining second-order term satisfies (2.18). $\square$

The Taylor expansion used in this proof is essentially the heuristic presented in [23], but the machinery developed by Barles and Souganidis [5] renders this argument perfectly rigorous. In addition to being short and simple, the resulting proof seems more natural from the point of view of standard viscosity techniques than the proof offered in [23].

3. Algorithms and examples

There are by now numerous papers devoted to algorithms for curvature-dependent flows. Before discussing our game-theoretic algorithm in detail, we will briefly summarize some of the extant approaches, postponing a thorough comparison of these methods for a future publication.

The first schemes for mean curvature and related motions relied on finite difference methods on uniform grids ([1],[9],[13]), and subsequent work focused on developing faster, more efficient algorithms. In [24], Bence, Merriman and Osher introduced a very successful algorithm, diffusion-generated motion by mean curvature, which Evans [15] and Barles and Georgelin [3] subsequently analyzed. Ishii, Pires and Souganidis [22] then consolidated the work of Bence, Merriman and Osher with that of Gravner and Griffeath on models of cellular automaton dynamics [20], resulting in the more general framework of threshold dynamics type approximation schemes. Ruuth and Merriman provide an interesting discussion of some of these developments in [27]; they point out, for example, that diffusion-generated motion by mean curvature amounts to tracking the center of a circle as it traverses the curve in such a way that precisely half of its area is always inside the curve. Described this way, we see that the Bence-Merriman-Osher scheme has a similar geometric flavor to our game-theoretic scheme. It seems much simpler, however, to track the midpoint of a segment than to track the center of a circle while maintaining this area condition. A further simplifying feature of our scheme is that it does not require a grid for computations.

Motivated by developments in mathematical morphology and image processing, Catté et al. [10] developed a different approach which leads to the same min-max operator used here. Their point of view is quite different from ours and from that in [23], however, and, unlike ours, their implementation is grid-based. Following the earlier work by Evans [15], Catté et al. prove their analogue of Theorem 2.1 with nonlinear semigroup techniques; in semigroup parlance, we may interpret equation (2.15) as an exponential formula of Crandall-Liggett type ([12],[26]), with the scheme $S(\varepsilon)$ playing the role of the resolvent operator. In addition to their differing philosophy and implementation, we note that Catté et al. do not consider the general geometric equation (2.14) to which the game-theoretic approach applies.

There are at least two other noteworthy algorithms for curvature-dependent flows which have appeared recently. Oberman [25] has proposed and analyzed a grid-based scheme for mean curvature motion which, in the language of image processing, is a simple and efficient implementation of a median filter. Cao and Moisan [8], on the other hand, have developed an interesting algorithm which relies on the geometry of the curve and does not involve a computational grid; we refer to Cao's book [7] for more on this algorithm and the role of curve evolution in image processing.

3.1. **Mean curvature flow.** Having briefly surveyed these other techniques, we now describe the algorithms inspired by the two-player games above, beginning with the simpler case of mean curvature motion. To approximate the mean curvature flow of a planar curve Γ_0, we proceed as follows, retaining the notation Γ_t and Γ_t^ε used earlier.

Idealized mean curvature algorithm.

(i) Choose the time step $\varepsilon > 0$.

(ii) Having computed the approximation $\Gamma_{j\varepsilon}^\varepsilon$ for some positive integer j, find all pairs $(x_1, x_2) \in \Gamma_{j\varepsilon}^\varepsilon$ such that

$$\operatorname{dist}(x_1, x_2) = 2\sqrt{2\varepsilon}\,, \tag{3.1}$$

and, for each pair, add the point

$$\bar{x} := \frac{x_1 + x_2}{2}$$

to the new set $\Gamma^{\varepsilon}_{(j+1)\varepsilon}$.

(iii) Stop after k rounds when there is no pair $(x_1, x_2) \in \Gamma^{\varepsilon}_{k\varepsilon}$ satisfying (3.1).

This is an idealized scheme for several reasons. First, we must somehow discretize the original curve Γ_0 in order to perform computations, and one cannot expect pairs of points from the resulting discrete set to satisfy (3.1) exactly. Second, if Γ_0's evolution by mean curvature results in fattening, it is not clear how to find all pairs satisfying (3.1) efficiently. Since a specific example of the latter situation may be useful, suppose for the moment that Γ_0 is the left-hand set in Figure 1. It

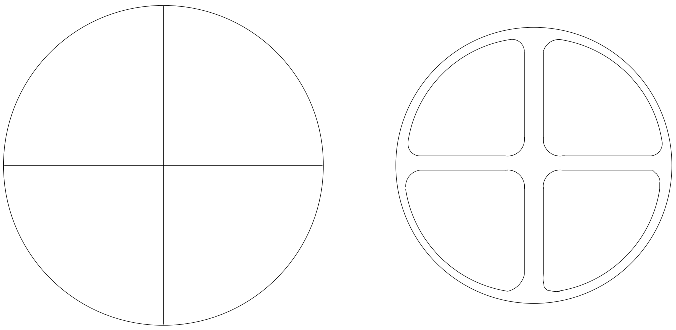

FIGURE 1. The set on the left fattens as it evolves by its mean curvature, yielding sets with interior like the one on the right. The set on the right is solid, like a steering wheel with four holes.

is well-known that the set on the left will fatten as it evolves by its mean curvature (it also appears, for instance, in [16]); one way to see this, in fact, is to apply the idealized algorithm above by hand on a piece of scrap paper. The set on the right is typical of the resulting evolution. From the point of view of the mean curvature game, fattening occurs when the trajectory of the traversing segment's midpoint is not unique. Accommodating this possibility complicates the implementation of the algorithm, so we keep things simple in the present paper by assuming that Γ_0 is a simple closed curve. Future work will address a more robust implementation.

Instead of discretizing with a grid of some sort, we will approximate the curves $\Gamma^{\varepsilon}_{j\varepsilon}$ directly with discrete sets, initializing computations with a discrete set Δ^{ε}_0 of points belonging to the initial curve Γ_0. This leads to the following straightforward implementation of the scheme described above.

Practical mean curvature algorithm.

(i) Choose the time step $\varepsilon > 0$.

(ii) Suppose that, for some integer $j \geq 0$, we have a discrete set $\Delta^{\varepsilon}_{j\varepsilon}$ which approximates the curve $\Gamma^{\varepsilon}_{j\varepsilon}$. We assume that the points in $\Delta^{\varepsilon}_{j\varepsilon}$ are sorted, corresponding to an oriented traversal of $\Gamma^{\varepsilon}_{j\varepsilon}$.

(iii) Begin at the first point $x_0 \in \Delta^{\varepsilon}_{j\varepsilon}$. Let x_1 and x_2 be the first consecutive pair of points in $\Delta^{\varepsilon}_{j\varepsilon}$ such that

$$\operatorname{dist}(x_1, x_0) < 2\sqrt{2\varepsilon} \quad \text{and} \quad \operatorname{dist}(x_2, x_0) > 2\sqrt{2\varepsilon}\,, \tag{3.2}$$

and define

$$\bar{x} := \frac{2x_0 + x_1 + x_2}{4}\,.$$

The point $\bar{x}$ is the midpoint of the segment whose endpoints are the midpoints of the segments connecting x_0 and x_1 and x_0 and x_2, respectively. Add $\bar{x}$ to the discrete approximation $\Delta^{\varepsilon}_{(j+1)\varepsilon}$ of the curve $\Gamma^{\varepsilon}_{(j+1)\varepsilon}$.

(iv) Advance to the second point in $\Delta^{\varepsilon}_{j\varepsilon}$ and repeat the preceding procedure, adding a second point to $\Delta^{\varepsilon}_{(j+1)\varepsilon}$. Continue in this way until all of $\Delta^{\varepsilon}_{j\varepsilon}$ has been exhausted, thereby completing $\Delta^{\varepsilon}_{(j+1)\varepsilon}$. (When dealing with points at the end of $\Delta^{\varepsilon}_{j\varepsilon}$, the points x_1 and x_2 will come from the beginning of $\Delta^{\varepsilon}_{j\varepsilon}$. This can be handled easily with a simple array data structure.)

(v) Stop after k rounds when (3.2) cannot be satisfied by points in $\Delta^{\varepsilon}_{k\varepsilon}$.

Figures 2 and 3 illustrate the application of this mean curvature algorithm to two sets, a unit circle and a nonconvex curve built out of four half-circles of radius 1. Both computations used a segment of length 0.1, corresponding to the time step $\varepsilon = 1/800$, and each snapshot indicates the corresponding number of rounds of the game. The initial discretization Δ^{ε}_0 of the circle consisted of 48 uniformly spaced points, and it took 0.48 seconds to compute all 338 rounds of its resulting evolution. For the nonconvex curve, Δ^{ε}_0 consisted of 96 points, its evolution to extinction lasted for 1163 rounds, and it took 5.79 seconds to compute its full evolution.

It is well-known that a simple closed curve will convexify, become circular, and shrink to a point as it evolves by mean curvature ([19],[21]), a fact borne out by these examples. (This provides a quick but crude check that the algorithm works as it should.) Although we have avoided the possibility of fattening, it is worth emphasizing that this algorithm does not require any convexity.

3.2. **Generalized curvature flow.** To approximate the generalized curvature flow of Γ_0 according to equation (2.14), we cannot simply track the midpoint of a segment of length $2\sqrt{2\varepsilon}$ as it traverses the curves $\Gamma^{\varepsilon}_{j\varepsilon}$. Instead, we must modify the mean curvature algorithm as follows.

Idealized generalized curvature algorithm.

(i) Choose the time step $\varepsilon > 0$.

(ii) Having computed the approximation $\Gamma^{\varepsilon}_{j\varepsilon}$ for some positive integer j, find all pairs $(x_1, x_2) \in \Gamma^{\varepsilon}_{j\varepsilon}$ such that the problem

$$\sqrt{2\varepsilon}\, A(w, \bar{x}, T - (j+1)\varepsilon)w = v \tag{3.3}$$

has a solution $w \in S^1$, where

$$v := x_2 - x_1 \quad \text{and} \quad \bar{x} := \frac{x_1 + x_2}{2}\,. \tag{3.4}$$

For each such pair (x_1, x_2), add the point $\bar{x}$ to the new set $\Gamma^{\varepsilon}_{(j+1)\varepsilon}$.

(iii) Stop after k rounds when (3.3) has no solution $w \in S^1$ for any pair $(x_1, x_2) \in \Gamma^{\varepsilon}_{k\varepsilon}$.

This is an idealized scheme for the same reasons given above in the case of mean curvature flow. A practical implementation in this case is much more challenging, however, since we must deal with the nonlinear problem (3.3). At this preliminary stage, we only present results for the trivial generalized curvature flow problem in which

$$A(n, x, t) \equiv A$$

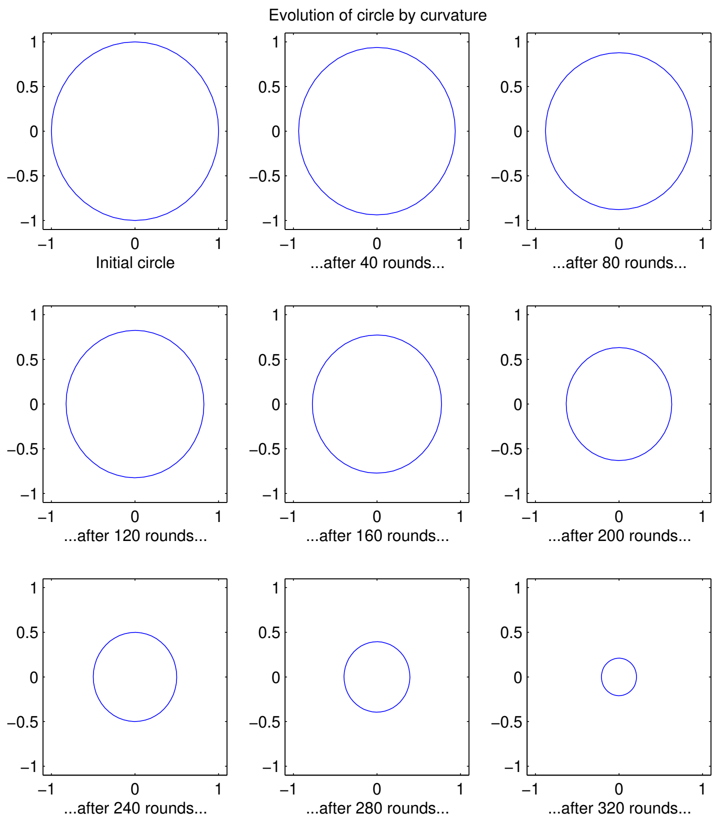

FIGURE 2. Snapshots of the motion of a circle by its mean curvature.

for some constant nonsingular matrix A. In this very special case, we simply change step (ii) of the practical mean curvature algorithm as follows: starting with the first point $x_0 \in \Delta^{\varepsilon}_{j\varepsilon}$, find the first consecutive points x_1 and x_2 in $\Delta^{\varepsilon}_{j\varepsilon}$ such that

$$\|A^{-1}(x_1 - x_0)\| < 2\sqrt{2\varepsilon} \quad \text{and} \quad \|A^{-1}(x_2 - x_0)\| > 2\sqrt{2\varepsilon}\,, \tag{3.5}$$

where $\|\cdot\|$ is the standard Euclidean norm. Add the point

$$\bar{x} := \frac{2x_0 + x_1 + x_2}{4}$$

to $\Delta^{\varepsilon}_{(j+1)\varepsilon}$ and, continuing in this manner, complete the set $\Delta^{\varepsilon}_{(j+1)\varepsilon}$.

Figures 4 and 5 illustrate the application of this generalized curvature algorithm to the circle and nonconvex curve treated earlier, again using a segment of length 0.1 in both examples. The matrix A used in both cases is

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix},$$

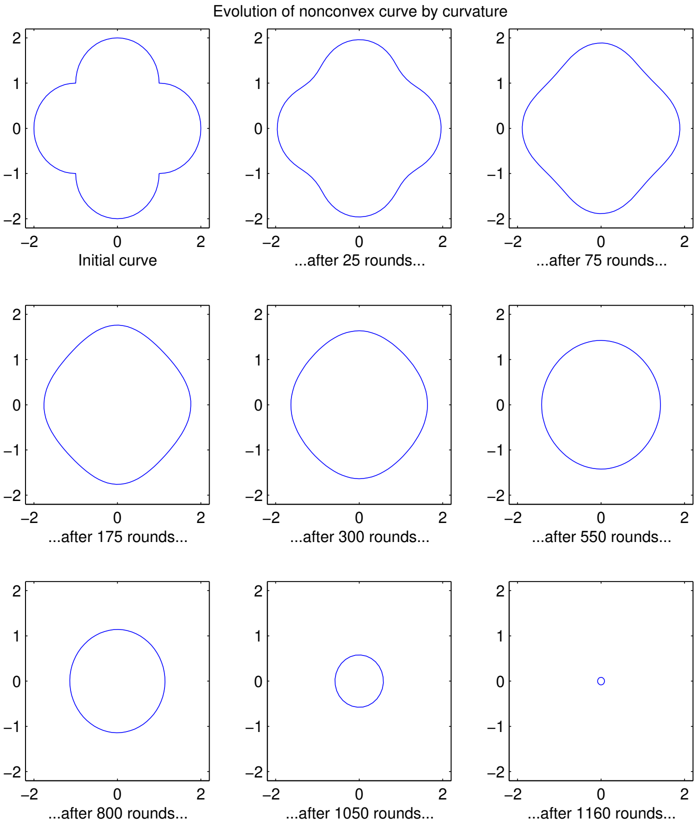

FIGURE 3. Snapshots of the motion of a nonconvex curve by its mean curvature.

and one sees clearly that both curves approach "circles" relative to the underlying pairing (2.7) defined by A.

Future work will confront the more difficult problem (3.3), as well as numerical analysis questions which we have ignored here (e.g., convergence rates and the interaction of the time step ε with the resolution of the initial discretization Δ_0^ε). The application of these ideas to other problems, such as the motion of curves in $\mathbb{R}^3$ and the active contour model in [9], will also be discussed elsewhere.

Acknowledgments. I discussed the main ideas presented here in an invited talk at the 6th UAB-MSU Conference on Differential Equations and Computational Simulations, held at Mississippi State University in May 2005. It is a pleasure to thank the conference organizers for all of their hospitality and hard work, and it was a tremendous honor to participate in the special session honoring Klaus Schmitt, a wonderful advisor and friend. Happy birthday, Klaus!

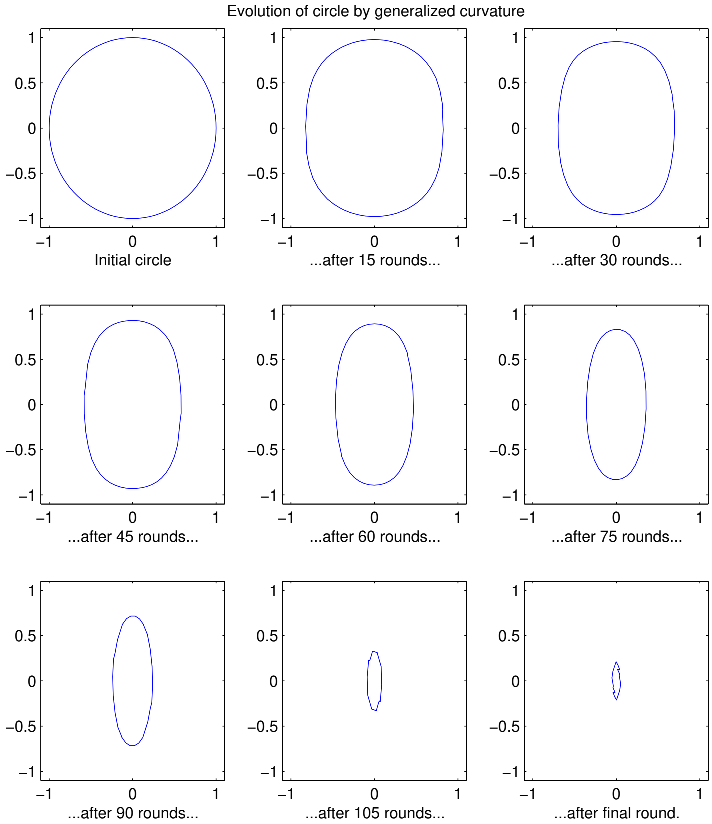

FIGURE 4. Snapshots of the motion of a circle by generalized curvature.

REFERENCES

[1] L. Alvarez, P.-L. Lions, and J.-M. Morel, *Image selective smoothing and edge detection by nonlinear diffusion. II*, SIAM J. Numer. Anal., 29 (1992), pp. 845–866.

[2] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhäuser Boston Inc., Boston, MA, 1997.

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

[4] G. Barles, H. M. Soner, and P. E. Souganidis, *Front propagation and phase field theory*, SIAM J. Control Optim., 31 (1993), pp. 439–469.

[5] G. Barles and P. E. Souganidis, *Convergence of approximation schemes for fully nonlinear second order equations*, Asymptotic Analysis, 4 (1991), pp. 271–283.

[6] G. Barles and P. E. Souganidis, *A new approach to front propagation problems: theory and applications*, Arch. Rational Mech. Anal., 141 (1998), pp. 237–296.

[7] F. Cao, *Geometric Curve Evolution and Image Processing*, vol. 1805 of Lecture Notes in Mathematics, Springer-Verlag, Berlin, 2003.

[8] F. Cao and L. Moisan, *Geometric computation of curvature driven plane curve evolutions*, SIAM J. Numer. Anal., 39 (2001), pp. 624–646 (electronic).

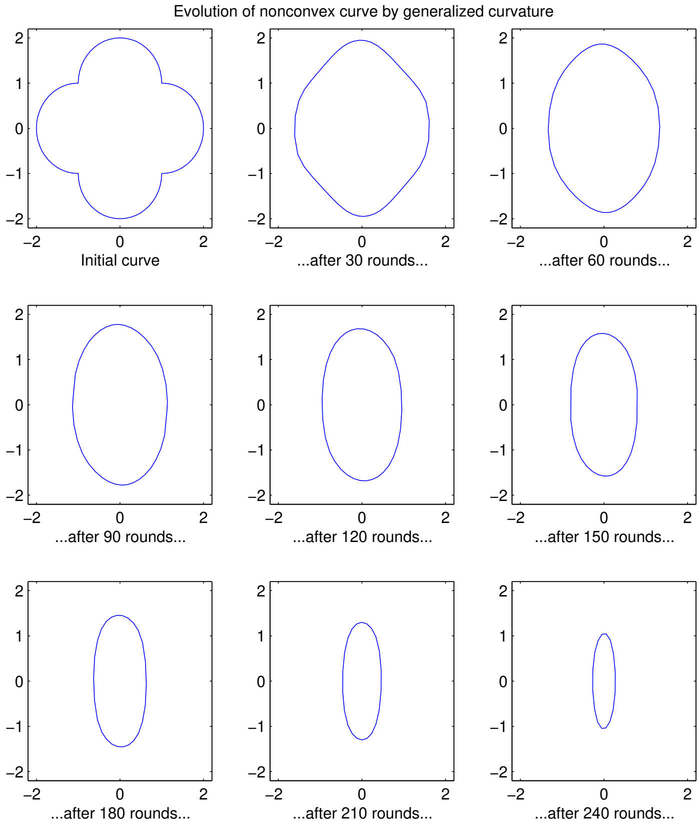

FIGURE 5. Snapshots of the motion of a nonconvex curve by generalized curvature.

[9] V. Caselles, F. Catté, T. Coll, and F. Dibos, *A geometric model for active contours in image processing*, Numer. Math., 66 (1993), pp. 1–31.

[10] F. Catté, F. Dibos, and G. Koepfler, *A morphological scheme for mean curvature motion and applications to anisotropic diffusion and motion of level sets*, SIAM J. Numer. Anal., 32 (1995), pp. 1895–1909.

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

[12] M. G. Crandall and T. M. Liggett, *Generation of semigroups of nonlinear transformations on general Banach spaces*, Amer. J. Math., 93 (1971), pp. 265–298.

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

[14] R. J. Elliott, *Viscosity Solutions and Optimal Control*, vol. 165 of Pitman Research Notes in Mathematics Series, Longman Scientific & Technical, Harlow, 1987.

[15] L. C. Evans, *Convergence of an algorithm for mean curvature motion*, Indiana Univ. Math. J., 42 (1993), pp. 533–557.

[16] L. C. Evans, *Regularity for fully nonlinear elliptic equations and motion by mean curvature*, in Viscosity Solutions and Applications (Montecatini Terme, 1995), vol. 1660 of Lecture Notes in Math., Springer, Berlin, 1997, pp. 98–133.

[17] L. C. Evans and P. E. Souganidis, *Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations*, Indiana Univ. Math. J., 33 (1984), pp. 773–797.

[18] L. C. Evans and J. Spruck, *Motion of level sets by mean curvature. I*, J. Differential Geom., 33 (1991), pp. 635–681.

[19] M. Gage and R. Hamilton, *The heat equation shrinking convex plane curves*, J. Diff. Geom., 23 (1986), pp. 69–96.

[20] J. Gravner and D. Griffeath, *Threshold growth dynamics*, Trans. AMS, 340 (1993), pp. 837–870.

[21] M. Grayson, *The heat equation shrinks embedded plane curves to round points*, J. Diff. Geom., 26 (1987), pp. 285–314.

[22] H. Ishii, G. E. Pires, and P. E. Souganidis, *Threshold dynamics type approximation schemes for propagating fronts*, J. Math. Soc. Japan, 51 (1999), pp. 267–308.

[23] R. V. Kohn and S. Serfaty, *A deterministic–control–based approach to motion by curvature*, To appear, Comm. Pure Appl. Math.

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

[25] A. M. Oberman, *A convergent monotone difference scheme for motion of level sets by mean curvature*, Numer. Math., 99 (2004), pp. 365–379.

[26] M. Rudd, *Weak and strong solvability of parabolic variational inequalities in Banach spaces*, J. Evol. Equ., 4 (2004), pp. 497–517.

[27] S. J. Ruuth and B. Merriman, *Convolution–generated motion and generalized Huygens' principles for interface motion*, SIAM J. Appl. Math., 60 (2000), pp. 868–890 (electronic).

[28] J. A. Sethian, *Level Set Methods and Fast Marching Methods*, vol. 3 of Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, 1999.

[29] P. Soravia, *Generalized motion of a front propagating along its normal direction: a differential games approach*, Nonlinear Anal., 22 (1994), pp. 1247–1262.

[30] P. E. Souganidis, *Approximation schemes for viscosity solutions of Hamilton-Jacobi equations*, J. Differential Equations, 59 (1985), pp. 1–43.

[31] P. E. Souganidis, *Max-min representations and product formulas for the viscosity solutions of Hamilton-Jacobi equations with applications to differential games*, Nonlinear Anal., 9 (1985), pp. 217–257.

[32] P. E. Souganidis, *Front propagation: theory and applications*, in Viscosity Solutions and Applications (Montecatini Terme, 1995), vol. 1660 of Lecture Notes in Math., Springer, Berlin, 1997, pp. 186–242.

[33] J. Spencer, *Balancing games*, J. Combinatorial Theory Ser. B, 23 (1977), pp. 68–74.

Matthew Rudd

Department of Mathematics, University of Texas at Austin, Austin, TX 78712, USA

E-mail address: rudd@math.utexas.edu