

## A LINE ELEMENT ALGORITHM FOR CURVE FLOW PROBLEMS IN THE PLANE

STEPHEN ROBERTS<sup>1</sup>

(Received 11 July 1990; revised 18 February 1992)

### Abstract

In this paper we shall describe a numerical method for the solution of curve flow problems in which the normal velocity of the curve depends locally on the position, normal and curvature of the curve. The method involves approximating the curve by a number of line elements (segments) which are only allowed to move in a direction normal to the element. Hence the normal of each line element remains constant throughout the evolution. In regions of high curvature elements naturally tend to accumulate. The method easily deals with the formation of cusps as found in flame propagation problems and is computationally comparable to a naive marker particle method. As a test of the method we present a number of numerical experiments related to mean curvature flow and flows associated with flame propagation and bushfires.

### 1. Introduction

Suppose that at time  $t$  we have a region  $\Omega_t$  in the plane with boundary  $\partial\Omega_t$ . We are interested in the evolution in time of the region  $\Omega_t$  such that the velocity of the boundary  $\partial\Omega_t$ , at any point, is normal to the boundary and has a speed depending on its position in  $R^2$ , on the direction of the normal and on the curvature of the boundary at that point. Specifically, if we can parameterise the boundary so that  $\partial\Omega_t = \{\mathbf{x}(s, t) : s \in S\}$  then the evolution of the position vector  $\mathbf{x}(s, t)$  is given by

$$\frac{\partial}{\partial t} \mathbf{x}(s, t) = F(\mathbf{x}, \nu, H)\nu \quad (1)$$

<sup>1</sup>Mathematics Department, The Faculties, Australian National University.  
© Australian Mathematical Society, 1993, Serial-fee code 0334-2700/93

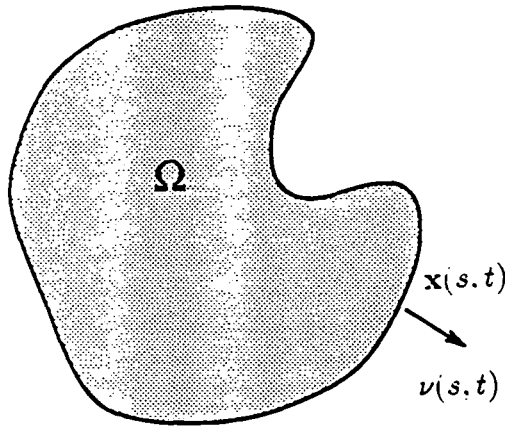


FIGURE 1. Interface  $\partial\Omega$ , with normal  $\nu(s, t)$  at point  $\mathbf{x}(s, t)$ .

where  $\nu(s, t)$  and  $H(s, t)$  are respectively the outward normal and the curvature of the interface at the point  $\mathbf{x}(s, t)$  (see Figure 1). In this paper we shall introduce a numerical method for modeling equations of this form, and use the method to study a number of specific applications.

Many interesting problems can be formulated using an equation of type (1). It is common to assume that the position of the interface between burnt and unburnt gases in a uniformly mixed medium satisfies (1) with the function  $F$  given by

$$F(\mathbf{x}, \nu, H) = c - \epsilon H$$

for  $c$  and  $\epsilon$  constant and  $\epsilon$  small and non-negative ([6]). In fact, in many physical situations the limiting case  $\epsilon = 0$  provides a very good approximation.

In Australia we are particularly interested in modeling the propagation of bushfires. Empirical studies indicate that some grass fires under the influence of a constant wind tend to form elliptical burn profiles. This corresponds to a normal speed given by

$$F(\mathbf{x}, \nu, H) = g(\nu \cdot \mathbf{n}_w) + \sqrt{h^2 + (f^2 - h^2)(\nu \cdot \mathbf{n}_w)^2}$$

where  $\mathbf{n}_w$  is the direction of the wind and  $f, g, h$  are positive experimental constants (see [2]). Essentially there is a constant outward term (speed  $h$ ) which is modified by the effect of the wind; retarded at the back of the fire (speed  $f - g$ ), forced at the head of the fire (speed  $f + g$ ). We note that in the case

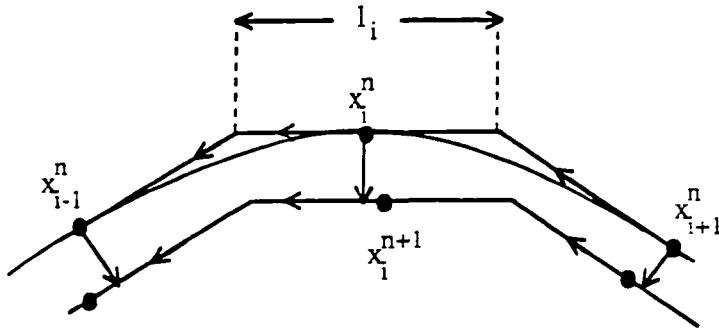


FIGURE 2. Approximation of part of the interface by line elements. The evolution operator is approximated by translating the elements in a normal direction.

of bushfires the standard models do not include a curvature term. This is due to the fact that curvature terms are usually only considered to be of importance on a scale much smaller than the overall size of a common bushfire (i.e., in the  $\epsilon \rightarrow 0$  limit above).

Pure mean curvature flow

$$F(\mathbf{x}, \nu, H) = -H$$

can be used to describe the evolution of the annealing front of a metal (see [9]). On the theoretical side, if an initial curve is embedded (non self intersecting), then the mean curvature flow will flow the curve to a point in a finite time, such that the enclosed area decreases linearly to zero. As the curve shrinks, the shape of the curve converges to a circle (see [5]).

If the initial curve is immersed and self intersecting, then it is important to understand the behaviour of the curve as it evolves to a singularity. In Section 3 we shall describe some preliminary numerical experiments aimed at this problem.

In this paper we introduce a simple method for the numerical solution of equations of the form (1). An obvious way to approximate an equation of this type is to place a number of marker points around the initial curve and then use finite differences to approximate the evolution equation. Unfortunately this leads to a number of problems (see Sethian [8]). In particular there is a need to refine accurately regions of high curvature and to deal with the formation of cusps. Standard explicit time-stepping schemes run into stability problems if neighbouring marker points come too close. Essentially the time step needs to

be of the same order as the square of the minimum distance between two neighbouring points. In many cases, the distances between marker points become very small, even though the solution at that point has very little curvature. While our method will still need a similar time restraint for curvature problems, it will only be in regions of high curvature that the distance between computational points will become small. For flame propagation problems with  $F = c$ , it is possible for cusps to form. This is a case in which the naive marker-point method is guaranteed to fail for reasonable choices of time step (see [8] Appendix A).

Our method involves approximating the curve with a number of line elements (segments) with a fixed angle between neighbouring elements. The accuracy of the method improves as the size of this angle is decreased. The idea is to evolve these line elements as lines and not as points. Each line element has a well-defined normal which remains constant throughout the evolution. Given a time step  $\Delta t$ , the positions of the elements are evolved in the direction normal to the element and a distance approximating  $\Delta t F(\mathbf{x}, \nu, H)$  (see Figure 2). Elements representing inflection points on the curve are deleted from the calculation if the approximating flow produces a situation in which the variation of the curve increases (see Figure 4). At each time step the lengths of elements are monitored, and if the length of an element becomes too long it is replaced by a collection of elements with the same tangent direction as the original element, separated by elements generating inflection points (see Figure 5). Together, this algorithm provides a stable method for the representation of the normal to a curve and allows for the accurate representation of regions of high curvature. A full description of the algorithm is given in the next section.

Recently Taylor [10] has used essentially the same method to study equilibrium and moving crystal surfaces. In her case, the angle is determined by the crystalline structure of the material being modelled, and the evolution of the crystalline surface is given by a steepest descent flow for an associated crystalline energy.

It is worth noting that, like any method that represents an interface as a parameterised curve, problems will arise if the interface is made up of a large number of disconnected regions. The method will then have to deal with the possible intersection of distinct components. Provided the number of components is not too large, it would be relatively easy to provide the appropriate bookkeeping to deal with such intersections. At the intersection points, elements could be added to coalesce two components into one.

However, for complicated situations it would probably be better to use one of the techniques in which the interface is resolved as the level set of a function

defined over a two dimensional region. One such method is the fractional volume method, which is based on a Huygens principle. A function is defined on a grid of cells covering the domain of interest. If the function value is one, the cell lies completely inside the region bounded by the interface; if the value is zero the cell is completely outside the region. Fractional values indicate that the interface intersects the cell. An approximation of the interface inside a cell is reconstructed using the fraction volume data for the cell together with neighbouring cells. The approximate interface is then moved in such a way as to model the evolution equation, which in turn updates the value of the fractional volume for each grid cell (see [3]). This method seems to be well adapted to problems that do not depend on curvature or on an accurate representation of the normal of the interface. In particular, they work well in the case  $F(\mathbf{x}, \nu, H) = c + \mathbf{w}(\mathbf{x}) \cdot \nu$  where we have a constant outward normal flow together with an advection term (wind). In particular the method has been used to study turbulent flow in a combustion tunnel [4]. Unfortunately there does not seem to be a reasonable way to accurately approximate a curvature term using fractional volumes (see [8]). Hence our method is superior for curvature-dependent problems.

Another approach considers the interface as the level set of a function which satisfies a Hamilton-Jacobi equation. If  $\psi(\mathbf{x}, t)$  is a function such that the level sets  $\Omega_t = \{\mathbf{x} : \psi(\mathbf{x}, t) = 0\}$  satisfy (1), then formally

$$\psi_t - F(\mathbf{x}, \nu, H) \sqrt{\psi_x^2 + \psi_y^2} = 0. \quad (2)$$

This formulation provides a very powerful and general framework to study equations of type (1) and allows for a general way to deal with cusps and the evolution of the equations after singularities have formed. In addition the method naturally generalises to higher dimensions. Numerical methods developed for the solution of conservation laws have been applied to (2) with great success (see [7]). On the other hand, two problems are evident. First, this formulation increases the dimension of the problem from one to two and so is computationally more intensive than a simple parameterised curve flow method. Secondly, the problem is solved on a grid and so regions of high curvature are smoothed onto the computational grid. To analyse the structure of singularities it would be necessary to refine the grid substantially or to implement an adaptive grid refinement algorithm. For the problems of interest to us, namely the singularity structure of curve flows dependent on curvature and simple flame propagation flows, our method is competitive.

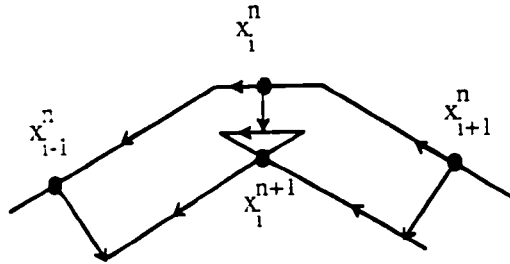


FIGURE 3. Stabilising algorithm: negatively oriented line elements are moved to the intersection point of the neighbouring elements.

### 2. The algorithm

In this section we shall define our method in more detail, but first we introduce some notation. (Reference to Figure 2 may be of help). In our method the curve is approximated by a number of line elements (line segments) with a fixed angle between neighbouring elements. To element  $i$  we assign a fixed angle  $\phi_i \in [0, 2\pi]$  such that the tangent  $\mathbf{t}_i$  to the element is given by  $\mathbf{t}_i = (\cos(\phi_i), \sin(\phi_i))$  and the normal is given by  $\mathbf{v}_i = (-\sin(\phi_i), \cos(\phi_i))$ . To specify the line  $L_i^n$ , at time  $n\Delta t$ , which contains the  $i^{th}$  element as a segment, we need to specify a point  $\mathbf{p}_i^n$  on that line. With  $\mathbf{p}_i^n$  specified,

$$L_i^n = \{ \mathbf{y} = \mathbf{p}_i^n + \alpha \mathbf{t}_i : \alpha \in R \}$$

where we note that the direction of the line is already given by  $\mathbf{t}_i$ .

We use the notation  $\mathbf{a}_i^n$  for the intersection of the lines  $L_{i-1}^n$  and  $L_i^n$  and  $\mathbf{b}_i^n$  for the intersection of the lines  $L_i^n$  and  $L_{i+1}^n$ . The  $i^{th}$  line element at time  $n\Delta t$  is given by the line segment from  $\mathbf{a}_i^n$  to  $\mathbf{b}_i^n$ . The midpoint of the  $i^{th}$  element is given by  $\mathbf{x}_i^n = \frac{1}{2}(\mathbf{a}_i^n + \mathbf{b}_i^n)$  and the length of this line segment is denoted  $\ell_i$ . If the vector  $\mathbf{b}_i^n - \mathbf{a}_i^n$  points in the same direction as the tangent  $\mathbf{t}_i$ , then we say that the element is positively oriented.

The evolution of (1) is approximated by using a simple first order Euler time stepping scheme. Let the size of the time step be denoted  $\Delta t$ . Then a reference point  $\mathbf{p}_i^{n+1}$  at time  $(n + 1)\Delta t$  is given by

$$\mathbf{p}_i^{n+1} = \mathbf{p}_i^n + \Delta t F(\mathbf{x}_i^n, \mathbf{v}_i, H_i^n) \mathbf{v}_i . \tag{3}$$

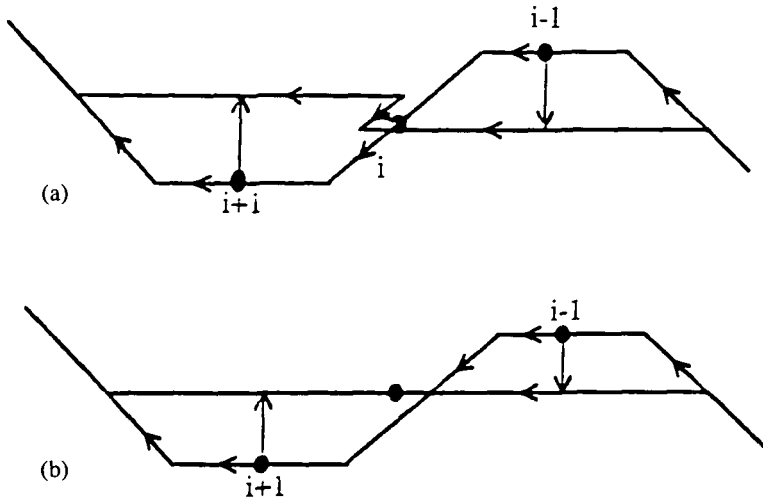


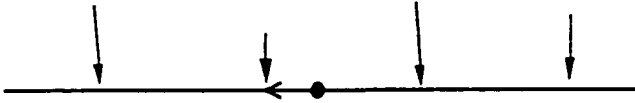
FIGURE 4. Stabilising inflection points: (a) Evolution of inflection points without stabilising. (b) With stabilising, elements representing inflection points with negative orientation are removed and neighbouring elements are coalesced.

Given the new reference points the intersection points  $\mathbf{a}_i^{n+1}$  and  $\mathbf{b}_i^{n+1}$  can then be easily determined. In (3) above,  $H_i^n$  is some approximation to the mean curvature of the curve at the midpoint of the  $i^{\text{th}}$  element. In Section 3.3 we describe and test two methods for discretising mean curvature.

If all the elements are initially positively oriented, then by choosing  $\Delta t$  small enough we can ensure that the elements remain positively oriented. Essentially the condition on the time step so that the elements remain positively oriented is equivalent to a stability condition. Figure 3 shows a situation in which the chosen step size is too large. Instead of choosing a smaller time step it is possible to add a stabilising procedure in which elements are displaced so that they become positively oriented. The elements are moved to the intersection point of the previous and next elements.

Inflection points are handled somewhat differently. An element is considered to be an inflection point if both neighbours have the same angle  $\phi$ . Figure 4a shows the typical situation that arises for an inflection point if the time step is chosen too large. In this case the stabilising procedure coalesces the two neighbours into one element at an average position of the neighbouring elements (so that enclosed area is conserved) and deletes the element corresponding to the

Different velocity vectors influencing a long line element



Long element broken into smaller elements



FIGURE 5. Adding elements: If elements become too long they are broken into parallel elements separated by inflection points.

inflection point (Figure 4b). We see that the stabilising procedure reduces the total variation of the curve and conserves signed arc-length and enclosed area. (See Sethian [8] for a discussion of variation in the flame propagation case).

Finally there are flows in which the variation of the curve can increase with time. It is then necessary to add extra elements in regions in which the length of the elements become too long. We have two parameters  $\ell_{max}$  and  $\ell_{new}$  corresponding to the maximum length allowed for any element and the maximum length allowed for any new element. If an element is detected which is longer than  $\ell_{max}$ , then the element is broken into a number of elements lying on the original element and with the same angle  $\phi$ , separated by inflection point elements (see Figure 5). The orientation of the inflection points are chosen so that if the time-step is sufficiently small the evolution equation will flow the elements so as to have positive length.

Hence we have a method that deals with regions of high curvature and cusps in a stable and efficient manner.

### 3. Numerical results

In this section we describe some numerical experiments, in which we test our method on a number of representative flows corresponding to different choices of the velocity function  $F(\mathbf{x}, \nu, H)$ .



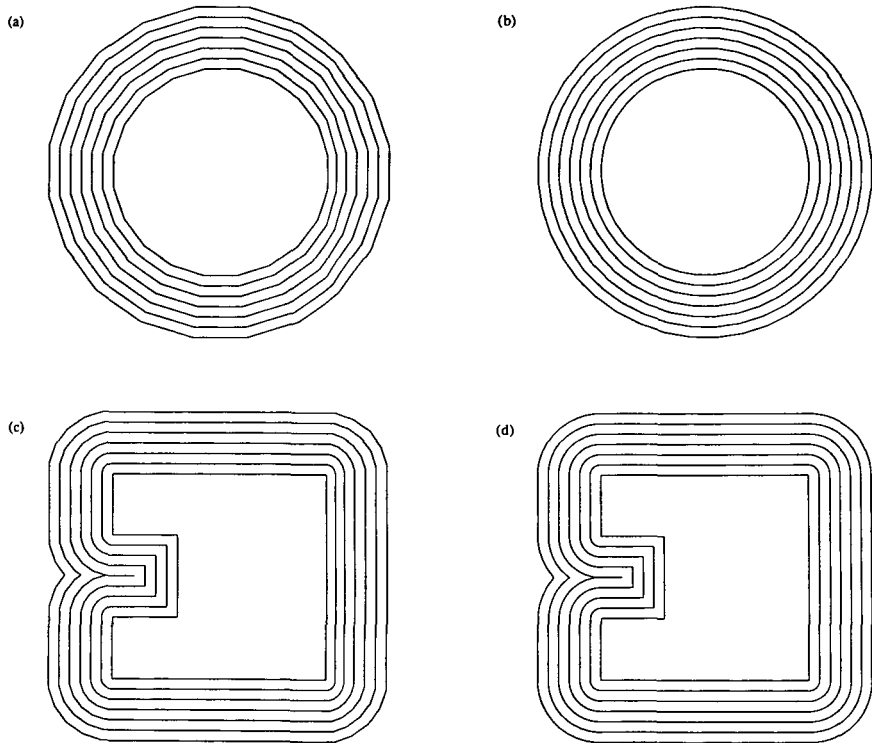


FIGURE 6. Constant outward flow ( $F(\mathbf{x}, \nu, H)$ ).

- (a) Initial circle with angle between elements  $\alpha = \pi/10$  (20 elements).
- (b) Initial circle with  $\alpha = \pi/20$  (40 elements).
- (c) Initial non-convex region with  $\alpha = \pi/10$  (40 elements).
- (d) Initial non-convex region with  $\alpha = \pi/20$  (80 elements).

**3.1. Constant outward flow** As a first test of our method we evolved a number of curves using a constant outward velocity  $F(\mathbf{x}, \nu, H) = 1$ . The main problems associated with this evolution are the formation of cusps in a finite time and the evolution of the curve once a cusp has formed. Both of these problems are easily dealt with using line elements. First a cusp can be easily detected if a number of consecutive elements intersect at a point and have zero length. The stabilising procedure automatically produces cusps to ensure that no elements are negatively oriented. The evolution of the cusp proceeds by the elements generating the cusp gradually being able to evolve to strictly positively oriented elements without the need to stabilise. For instance, a rectangle can immediately

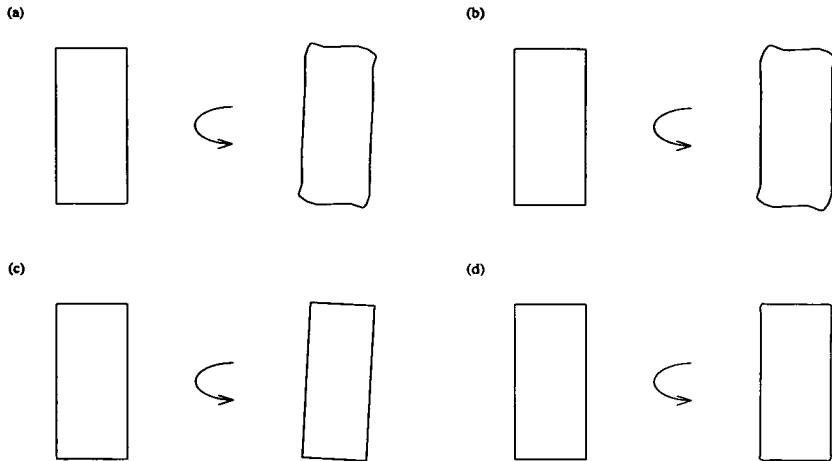


FIGURE 7. Constant angular rotation: Rectangle at times  $t = 0$  and 3.14. In all cases we consider angle  $\alpha = \pi/10$ .

(a) Time-step  $\Delta t = 0.01$ ,  $\ell_{max} = 0.2$  (b) Time-step  $\Delta t = 0.005$ ,  $\ell_{max} = 0.2$   
 (c) Time-step  $\Delta t = 0.01$ ,  $\ell_{max} = 0.05$  (d) Time-step  $\Delta t = 0.005$ ,  $\ell_{max} = 0.05$   
 Note that accuracy is improved as time-step and  $\ell_{max}$  decreases.

evolve to a curve without cusps or corners. The elements initially generating the corners immediately spread out to form a curve approximating an arc of a circle. The error produced by this approximation is of order  $\alpha^2$ , where  $\alpha$  is the angle between consecutive elements. See Figure 6 for the evolution of a circle and a non-convex region with the angle between elements being  $\alpha = \pi/10$  and  $\pi/20$ . In these problems the production of extra elements as the lengths of the elements become large does not increase the accuracy of the method. It is only the time step  $\Delta t$  and angle difference  $\alpha$  which matters.

**3.2. Constant angular rotation** Another important test of our method is a flow which simply rotates the object. In fact this is a very severe test of our method as it involves moving elements appreciably in the tangential direction. Recall that our method is based on the assumption that the evolution is dominated by the normal flow. In this rotating case the creation of elements is crucial for the approximation of the flow. We used an initial curve given by a rectangle and flowed with a constant angular velocity  $F(\mathbf{x}, \nu, H) = \mathbf{x}^\perp \cdot \nu$  (here  $\mathbf{x}^\perp$  is obtained by rotating  $\mathbf{x}$  by an angle  $\pi/2$  in counter-clockwise direction). The sides of the

rectangle rotate and so must be generated by a continuously changing collection of elements separated by inflection points. At the corners, elements become negatively oriented and need to be stabilised. Figure 7 shows the evolution of the rectangle using an angle  $\alpha = \pi/10$ , time steps of  $\Delta t = 0.01$  and  $0.005$ , and  $\ell_{max} = 0.2$  and  $0.05$ . We observe that it is the time step and the maximum length of the elements which controls the accuracy. It should be noted that the total number of elements generating the curve is of the order of 100 for  $\ell_{max} = 0.2$  and 400 for  $\ell_{max} = 0.05$ . In all cases we used  $\ell_{new} = \ell_{max}/3$ . Observe that the sides of the rectangle are generally made up of a number of elements separated by inflection points.

**3.3. Mean curvature flow** Next we tested our method on pure mean-curvature flow. The mean curvature of an element can be calculated in a number of ways. We concentrated on two methods. First we calculated the curvature of an element by calculating the radius of a circle tangent to the element and its two neighbours. This method produces very large curvatures if the length of an element is small. To ensure stability of the method we must ensure that the time step be smaller than a constant times the square of the smallest length of any element (parabolic stability requirement). With this choice of time step it is not necessary to use the stabilising algorithm as only inflection point elements can evolve into a negatively oriented state. Figure 8 shows the results of using this method to study the formation of a singularity for an embedded curve. The singularity has been magnified so that the curvature of the point on the axis is equal to one. We have been able to continue this evolution to stages in which the spatial resolution of the method extends over at least 7 orders of magnitude.

Secondly we approximated the mean curvature by

$$H_i^n = \frac{1}{2}(\phi_+ - \phi_-)/d_i ,$$

where  $\phi_+$  is the angle of the first element after element  $i$  with positive length,  $\phi_-$  is the angle of the last element before element  $i$  with positive length and  $d_i$  is the distance between the midpoints of these two associated elements. This second method is better for dealing with corners and allows for larger time steps in these situations. The parabolic time step restraint is still necessary, but now the time step needs to be less than a constant times the square of the smallest length of any element with strictly positive length, i.e. we allow elements to have zero length. Figure 9 shows the result of using the second method for the evolution of a number of initial curves. It should be noted that in all these cases the time-step is bounded by the square of the minimum length of elements with

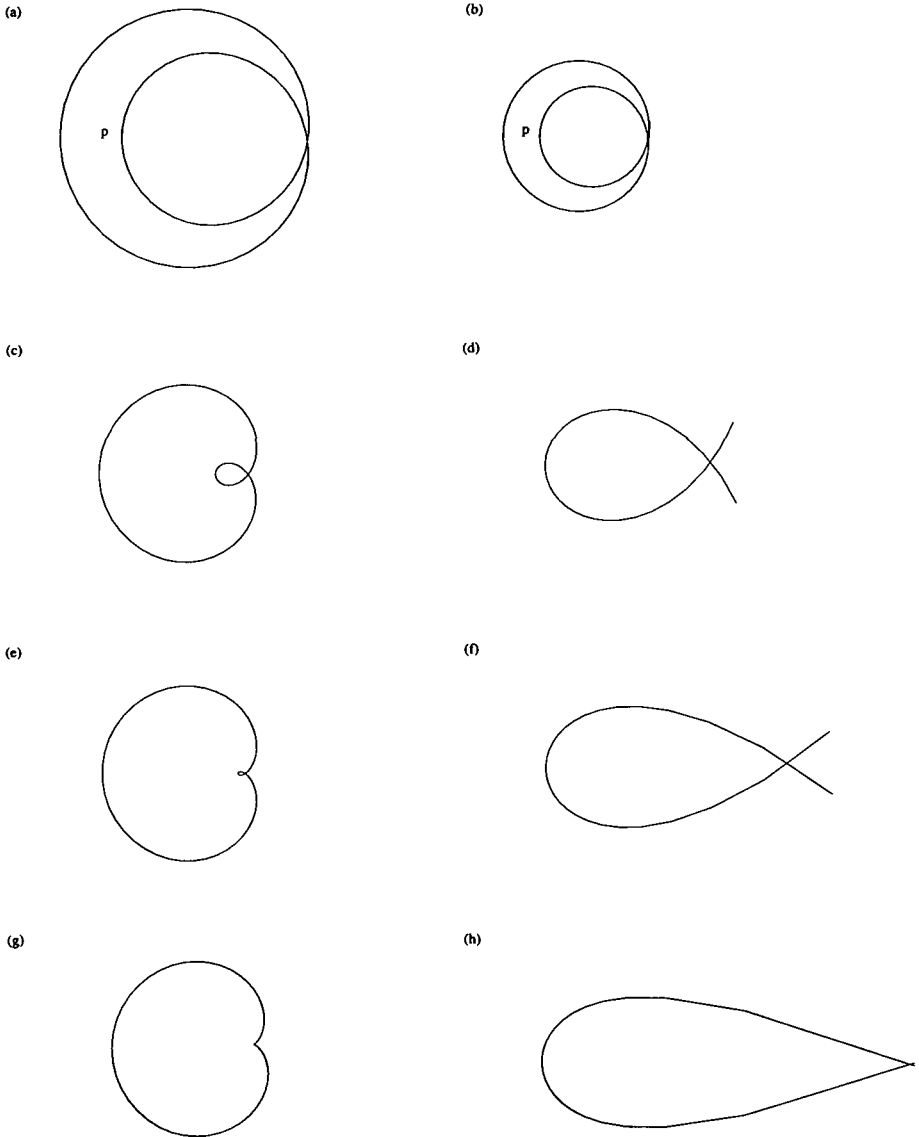


FIGURE 8. Mean curvature flow  $F(x, \nu, H) = -H$  using inscribed circle method. Formation of singularity from an initially self intersecting curve. Figures a, c, e, g show immersed curve at times 0.0, 0.10928, 0.112067, 0.112178 respectively. Figures b, d, f, h scale the singularity region so that curvature at point  $p$  is one.

Singularity Times		
Curves	$T_e$	$T_c$
Fig. 9a	0.5000	0.5016
Fig. 9b	0.5602	0.5561
Fig. 9c	0.1108	0.1104

TABLE 1. Exact singularity times  $T_e$  and computed singularity times  $T_c$  for the mean curvature flows depicted in Figure 9.

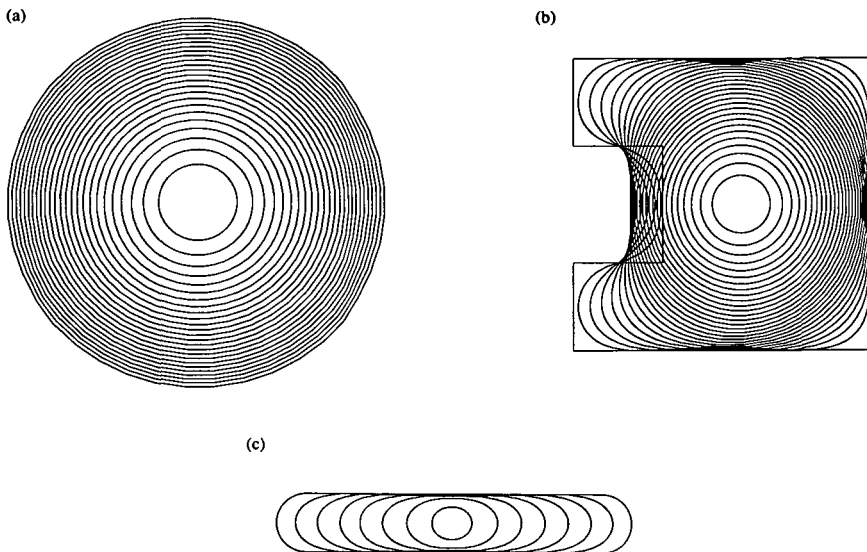


FIGURE 9. Mean curvature flow  $F(x, \nu, H) = -H$  using finite difference method. (a) Initially circular region,  $\alpha = \pi/20$ . (b) Initially non-convex region,  $\alpha = \pi/20$ . (c) Initially cigar shaped region,  $\alpha = \pi/20$ .

positive length. Notice that all the curves flow to point in a finite time, and the shape converges to a circle. Table 1 shows the computed singularity times together with the theoretical singularity time given by *Initial area*/ $2\pi$ .

**3.4. Combined flows** Finally we combined the constant outward flow with mean curvature flow. We tested our method on flows of the form  $F(x, \nu, H) = 1 - \epsilon H$ . Starting from an initially non-convex curve we tested the flow with  $\epsilon = 0.0, 0.1, 0.5, 1.5$ . As can be seen in Figures 10a-d, for larger values of  $\epsilon$  the mean curvature flow dominates and forces the flow to converge to a point in a finite time. For small  $\epsilon$ , the outward flow dominates. It is only in the regions

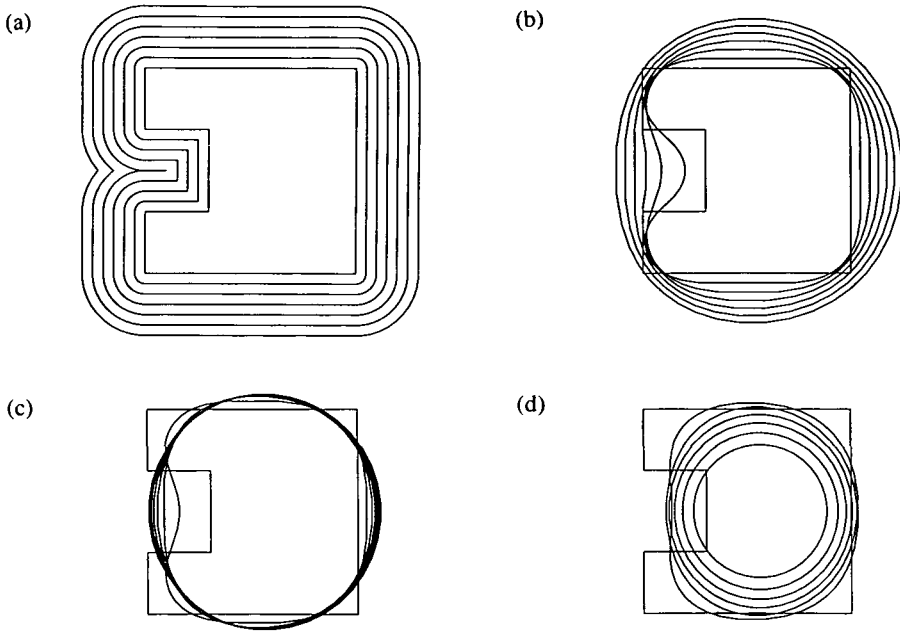


FIGURE 10. Combined flow  $F(\mathbf{x}, \mathbf{v}, H) = 1 - \epsilon H$ . (a)  $\epsilon = 0$ ,  $\alpha = \pi/10$ . (b)  $\epsilon = 0.5$ ,  $\alpha = \pi/10$ . (c)  $\epsilon = 1.0$ ,  $\alpha = \pi/10$ . (d)  $\epsilon = 1.5$ ,  $\alpha = \pi/10$ .

of high curvature (i.e., the corners and cusps) that the mean curvature term competes with the outward flow so as to smooth out these regions. Finally for intermediate values of  $\epsilon$  the outward flow and the mean curvature flow compete more evenly to produce the flows displayed in Figure 10.

#### 4. An application to bushfires

In this section we will apply our method to curve flows associated with the evolution of bushfires. It has been observed that many bushfires tend to form elliptical fronts (see [1] and [2]). It is a simple matter to see that the normal flow which corresponds to an elliptical front is given by

$$F(\mathbf{x}, \mathbf{v}, H) = g(\mathbf{v} \cdot \mathbf{n}_w) + \sqrt{h^2 + (f^2 - h^2)(\mathbf{v} \cdot \mathbf{n}_w)^2} \quad (4)$$

where  $\mathbf{n}_w$  is the direction of the wind and  $f, g, h$  are experimental constants (see [2]). The velocity of the head of the fire is given by  $f + g$ , the velocity of the tail

by  $f - g$ , and the side of the fire by  $h$ . We compared the evolution of a bushfire under the influence of wind using the normal flow given by (4) together with

$$F(\mathbf{x}, \nu, H) = g(\nu \cdot \mathbf{n}_w) + h + (f - h)(\nu \cdot \mathbf{n}_w)^2 \quad (5)$$

and

$$F(\mathbf{x}, \nu, H) = g(\nu \cdot \mathbf{n}_w) + h + (f - h)|\nu \cdot \mathbf{n}_w| \quad (6)$$

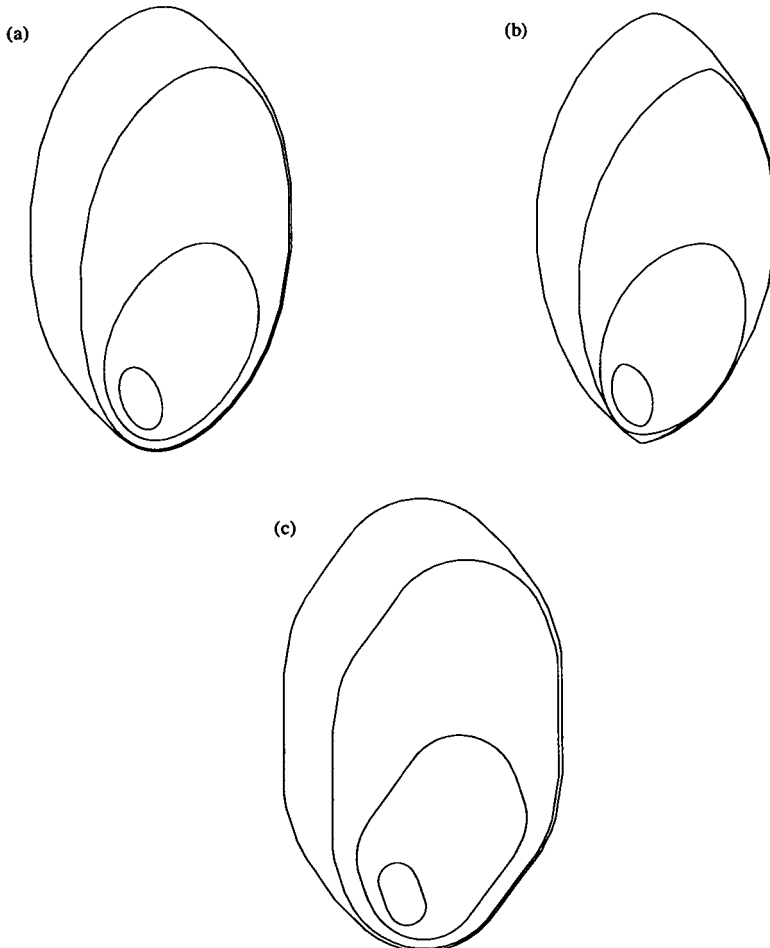


FIGURE 11. Bushfire fronts at 6, 14, 22, and 24 minutes using experimental data  $(\beta, f, g, h) = (0^\circ, 0.2, 0.133, 0.1)$  for  $t \in [0, 6]$ ,  $(-55^\circ, 0.437, 0.387, 0.212)$  for  $t \in (6, 14]$ ,  $(-22^\circ, 0.525, 0.475, 0.137)$  for  $t \in (14, 22]$  and  $(20^\circ, 0.95, 0.9, 0.1)$  for  $t \in (22, 24]$  ( $\beta$  is the angle of wind with respect to the initial wind direction).

(a) Normal flow given by (3). (b) Normal flow given by (4). (c) Normal flow given by (5).

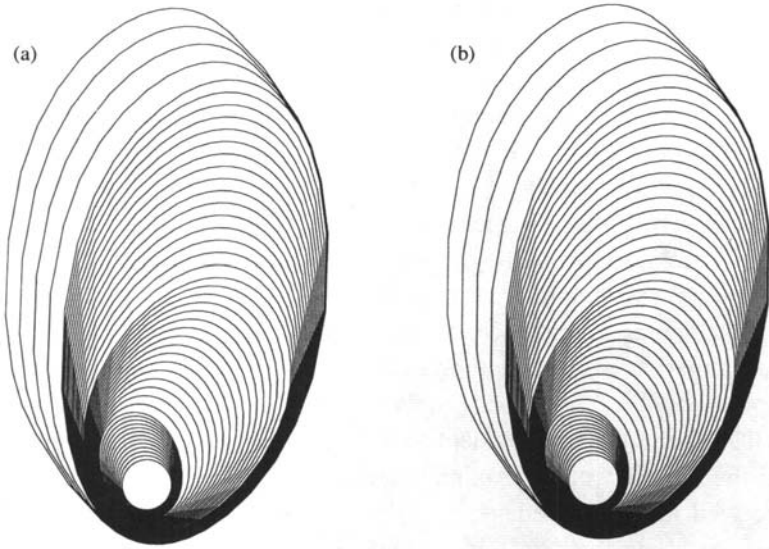


FIGURE 12. Bushfire fronts at 2 minute intervals using the same experimental data as for figure 11. (a) Normal flow using (5). (b) Normal flow using (7).

where  $f, g, h$  are the same experimentally obtained parameters which depend on wind speed.

In Figure 11 we display the evolution using the three normal flows above, using a set of values  $g, f, h$  and wind direction from an actual fire (see [1]). The curves show the computed position of the curve just as the wind direction and speed is about to change. We note that all three evolutions give qualitatively similar results.

In addition we evolved our bushfire using a flow of the form

$$F(\mathbf{x}, v, H) = g(v \cdot \mathbf{n}_w) + h + (f - h)(v \cdot \mathbf{n}_w)^2 - 0.05H. \quad (7)$$

Note that a curvature term has been added. Figure 12 shows the bushfire front at two minute intervals for flow with and without the added curvature term. We see that the added term tends to smooth out the area at the head and tail of the front. To deal with the extra second-order term in the equation it is necessary for the time step to depend on the square of the distance between line elements to maintain stability in the computation of the length of the elements. (Note that the distance between elements can only become small in regions of high



curvature.)

We draw two conclusions from these simple experiments. First, it is quite possible that an ideal fire acted upon by a constant wind will form a shape which only approximates the shape of an ellipse. As can be seen from the simulations, subtle differences in the shape of the fire front are associated with different local velocity functions. To provide an accurate simulation of a bushfire it is necessary to have an accurate form for this normal velocity of the evolution. This information can either be in the form of a statement about the fire evolving via a Huygens principle generated by small regions of a specific shape, or via a formula for normal velocity. Both are essentially equivalent. From this information an appropriate numerical method can be chosen, whether it be a Huygens principle, a standard marker particle method or the line element method described in this paper.

On the other hand, it is evident that provided the speed of the head, tail and side of the fire are well approximated, then the basic shape and size of the fire will be well approximated. Hence for instances in which only very general information about the direction and size of a fire is known and the region of interest is fairly homogeneous, reasonable simulations can still be obtained using any of the flows described above and perhaps even cruder assumptions.

## 5. Conclusion

In this paper we have described a numerical method which allows for the approximation of curve flow problems in which the normal velocity of the curve depends on a function of position, the normal and the curvature of the curve. The method naturally allows for a stable representation of the normal of a curve and automatically concentrates computational effort in regions of high curvature. With appropriate bookkeeping, the method could deal with the situation of a number of curves flowing in the plane with intersections occurring. The method is computationally comparable with a naive marker particle method, but can naturally deal with the cusp formation found in flame propagation problems. We feel that the method provides a stable and computationally efficient method for the study of curve flow problems in the plane.

### Acknowledgement

This work was supported by the Australian Research Council through grant A68830148, “A Theoretical and Numerical Study of Models for the Evolution of Interfaces”

### References

- [1] D. H. Anderson, E. A. Catchpole, N. J. de Mestre and T. Parkes, “Modelling the spread of grass fires”, *J. Austral. Math. Soc. (Series B)* **23** (1982) 451–466.
- [2] E. A. Catchpole, N. J. de Mestre and A. M. Gill, “Intensity of fire at its perimeter”, *Aust. For. Res.* **12** (1982) 47–54.
- [3] A. J. Chorin, “Flame advection and propagation algorithms”, *J. Comp. Phys.* **35** (1980) 1–11.
- [4] A. F. Ghoniem, A. J. Chorin and A. K. Oppenheim, “Numerical modelling of turbulent flow in a combustion tunnel”, *Philos. Trans. Roy. Soc. London Ser. A* **304** (1982) 303–325.
- [5] M. Grayson, “The heat equation shrinks embedded plane curves to points”, *J. Diff. Geom.* **26** (1987) 285–314.
- [6] G. H. Markstein, “Experimental and theoretical studies of flame front stability”, *J. Aero Sci.* **18** (1951) 199–209.
- [7] S. Osher and J. A. Sethian, “Fronts propagating with curvature dependent speed: Algorithms based on hamiltonian-jacobi formulations”, *J. Comput. Phys.* **79** (1988) 12–49.
- [8] J. A. Sethian, “Curvature and the evolution of fronts”, *Commun. Math. Phys.* **101** (1985) 487–499.
- [9] C. S. Smith, “Grain shape and other metallurgical applications of topology”, in *Metal Interfaces*, (American Society for Metal, Cleveland, 1952).
- [10] J. E. Taylor. “Crystals, in equilibrium and otherwise”, 1990, Videotape of 1989 AMS-MMA lecture, published by AMS.