CONTOUR PARAMETRIZATION VIA ANISOTROPIC MEAN CURVATURE FLOWS

P. SUÁREZ-SERRATO AND E.I. VELÁZQUEZ RICHARDS

ABSTRACT. We present a new implementation of anisotropic mean curvature flow for contour recognition. Our procedure couples the mean curvature flow of planar closed smooth curves, with an external field from a potential of point-wise charges. This coupling constrains the motion when the curve matches a picture placed as background. We include a stability criteria for our numerical approximation.

CONTENTS

Introduction 1
Outline 4
1. Short-time Existence and Uniqueness of Solutions 6
   Step 1 8
   Step 2 9
   Step 3 12
2. Anisotropic MCF Proposition 12
3. Numerical anisotropic MCF 16
4. Implementation of our numerical anisotropic MCF to Contour Parametrization 18
   Execution times 20
5. Conditioning and Stability of our Numerical scheme 21
6. Conclusions 28
References 29

INTRODUCTION

Many applied and theoretical problems have been studied through the analysis of manifold deformations [5, 6, 10]. The description of these deformations by an evolution equation imposed via a geometric quantity are referred to as geometric flows. Applications include, for example, the growth of crystals, the modeling of fluids, and digital image recognition. Since their conception there has been continued interest in the development of numerical approximations to these flows [21].

The difficulty in analyzing these flows numerically depends on the geometric quantity in evolution (e.g. curvature, metric tensor, the manifold itself). In particular, the mean curvature flow (MCF) deforms a hypersurface in the normal direction $\hat{n}$ with a speed proportional to its mean curvature $H$. This flow has an associated quasilinear parabolic equation in terms of an immersion $X$ of the hypersurface into the ambient manifold:

$$\frac{\partial X}{\partial t} = -H \hat{n}$$

---

Date: February 19, 2022.
Support for this project came from DGAPA-UNAM PAPIIT IN102716 and UC-MexUS CN-16-43 grants, and the PASPA program from DGAPA-UNAM.
The numerical methods employed to solve this equation are classified as Eulerian or Lagrangian methods, depending on the discrete representation of the surface or curve in evolution. In Eulerian methods the evolution is tracked by values at fixed positions on a gridded ambient. Conversely, in Lagrangian methods, the object in evolution is tracked explicitly through the position of its points. In consequence, Lagrangian methods have two advantages: they require smaller data storage than Eulerian methods, and the solution is computed explicitly; although generally their error estimates are difficult to estimate precisely. Eulerian methods are now a powerful and frequently used technique for mean curvature flow applications since the development of the Level Set Method [21, 24]. To our knowledge, there are very few examples of Lagrangian methods that approach this problem [25, 13], our work adds to this list.

The disadvantage of Lagrangian methods are discussed in many works. Two problems arise commonly when straightforward discretization of equation (1) is performed: (i) a numerical instability as shown in Figure 1 and (ii) loop formation which contradicts the comparison principle (see [7, 18]), as shown in Figure 2 for a cycloid (solid line) and its first iteration result (polygonal curve) compared with a circumference (dashed curve).

Here we present a Lagrangian method for contour parametrization. To accomplish this application, we assume that a planar closed and differentiable curve is drawn on a 2D digital image. Then, we evolve the curve by mean curvature flow, but constraining the motion of the curve by the objects in the image. If only one object is initially inside the curve, as the curve shrinks, it will match parts of the boundary of the object. The main problem using MCF to recognize images is to couple the restriction and the flow, because it implies that the curve is not evolving uniformly. That is, not all points in the curve will move, even when the curvature at those points is different from zero. This kind of flow is called anisotropic. To avoid certain numerical difficulties, our scheme considers a curve motion along tangential and normal directions, as in [13]. The results of [13] correspond to an unconstrained flow, so the stability and convergence results found there can not be compared with ours. Since the numerical procedure is based on certain MCF properties, we provide the necessary technical details about existence and uniqueness of solutions for MCF in Theorems 4 and 9.

Kimura developed a Lagrangian method which numerically reproduces the mean curvature flow for curves, based on a redistribution of points by reparametrization by arc length [13]. That method represents an initial simply closed smooth curve \( \varphi(s) \) by an ordered set of \( N \)
Figure 2. Loop formation in numerical discretization by standard Euler method of equation (1) for a cycloid (left), and detail (right).

points. The position of each point is updated to represent the curve after a time $\Delta t$. This method assumes a $\delta$–tubular neighborhood in which, for a small time interval $\Delta t$, the whole evolution is inside, and therefore may be parameterized as

$$X(s, r; t) = \varphi(s) + r \mathbf{n}(s, t).$$

In this tubular neighborhood the map $X$, given by

$$(s, r) \mapsto \{x \in \mathbb{R}^2 \mid \text{dist}(x, \varphi(t)) < \delta\}$$

is invertible. A correction term is then added to Euler’s formula (see [13]). Figures (3-5) show numerical examples with Kimura’s method, the evolution in time is represented by the $z$ axis.

Our main theoretical result is the postulation of an anisotropic MCF, for which short time existence of solutions is shown in Theorem (10). Furthermore, we detail a numerical scheme to approximate solutions and apply it to the task of contour parametrization.

Many varieties of an anisotropic MCF can be proposed, Figure (7) compares the MCF with two versions of anisotropic MCF. Our evolution scheme is an adaptation for contour recognition of Kimura’s planar curve evolution. We approach the stability of our scheme through von Neumann’s analysis. In our analysis, the parameters in error propagators (see Equation (38)) are time dependent. Then, the stability condition cannot be determined for all time but only for the next time step. We establish our main stability criterion in Proposition 13.

In contrast to previous approaches, our method is a Lagrangian scheme whose main features are: (I) Estimation of curvature bounds are not required. (II) It is formulated as a Poisson problem with a boundary condition given implicitly. This condition links Poisson’s problem with the MCF. Our proposed Poisson problem couples MCF and the field due to a point charge distribution. (III) Finding a solution requires to solve a $2N$ linear system. (IV) When it is implemented to a contour parametrization problem, only the pixel value at each point constrains the motion. (V) These constraints are handled through the source function and the boundary condition in Poisson’s equation. Our numerical experiments reveal that this Poisson formulation can match
Figure 3. Numerical MCF of cycloid. (a) An evolving cycloid (outer curve) evolves by MCF, as time increase, it converges to a circumference. (b) 3D projection of (a), the time parameter $t$ is represented by the vertical axis.

Figure 4. Numerical MCF of non-convex curve (I). (a) The outer curve is the initial non-convex curve, it deforms to a circumference and shrinks as time elapses. (b) 3D projection of (a), the time parameter $t$ is represented by the vertical axis.

some non-convex shapes perfectly. In general, matching non-convex shapes successfully depends on the charge distribution. Our scheme takes the numerical approximation for curvature and an equidistant distribution of points from [13], and through our proposed Poisson formulation we can handle these constraints. In Figure 6 we present a contour matching with an initial circumference as the evolving curve. The color gradient advances with time from red to green. A high contrast between red and green represents a fast matching, and a smooth color gradient indicates regions which require more iterations to reach the shape boundary in black.

Outline

This work is presented as follows: §1 the proof of existence and uniqueness of solutions for MCF and some of its properties, included for completeness sake; §2 the presentation of
Figure 5. Numerical MCF of non-convex curve (II). (a) The outer curve is the initial non-convex curve, it shrinks as time elapses. (b) 3D projection of (a), the time parameter $t$ is represented by the vertical axis.

Figure 6. A non-convex contour parametrization example of our technique. An initial red circumference is evolving by our AMCF, the enclosed black region constrains its motion (see section §4). The color gradient from red to green represents the initial and final curve position. A high contrast between red and green represents a fast matching.

our conditioned or anisotropic MCF and formulas for its solutions; §3 the numerical implementation of our anisotropic flow; §4 the implementation of the anisotropic MCF for contour parametrization; §5 details the stability properties of our numerical scheme, and §6 contains our conclusions and suggestions for future directions.
Figure 7. A non-convex red curve evolves by anisotropic MCF (left and center) and MCF (right). In the figure on the left the evolution is obtained by replacing in the MCF equation the curvature $\kappa$ of the curve with $\min(\kappa, 0)$. In the figure in the center, we replace $\kappa$ by $\max(\kappa, 0)$.

1. Short-time Existence and Uniqueness of Solutions

The main result of this section proves existence and uniqueness of solutions to an isotropic version of MCF. The proof includes new technical details, in particular bounds on the displacement of the immersed curve (equation (4) in Lemma 3), which are essential for the computations in the associated numerical problem in section 3.

This proof requires to rewrite equation (1) in terms of a reparametrization of the evolving manifold. Recall the following two theorems.

Theorem 1. (Xi-Ping Zhu [30]) The mean curvature flow equation for a hypersurface $M \subset \mathbb{R}$ with metric tensor $g$ is equivalent to

\[
\frac{\partial X}{\partial t} = \Delta_g X
\]

For the reader’s convenience we include a proof here.

Proof. Let $A$ be the second fundamental form of a manifold $M$ and $\nabla$ the induced connection by the ambient manifold connection $\bar{\nabla}$. In local coordinates the right side of mean curvature flow equation expands to

\[
H \mathbf{n} = \sum_{ij} g^{ij} A_{ij} \mathbf{n} = \sum_{ij} g^{ij} \left( \nabla_{\partial X/\partial x^i} \frac{\partial X}{\partial x^j} + \nabla_{\partial X/\partial x^j} \frac{\partial X}{\partial x^i} \right) \mathbf{n} = \sum_{ij} g^{ij} \left( \nabla_{\partial X/\partial x^i} \frac{\partial X}{\partial x^j} - \left( \nabla_{\partial X/\partial x^j} \frac{\partial X}{\partial x^i} \right)^\top \right)
\]
We focus on the individual components and develop further:

\[ H \hat{n}^\alpha = \sum_{i,j} g^{ij} \left( \frac{\partial^2 X}{\partial x^i \partial x^j} - \sum_{k,l,\beta} g^{kl} \frac{\partial X^\alpha}{\partial x^k} \frac{\partial X^\beta}{\partial x^l} \frac{\partial^2 X}{\partial x^i \partial x^j} \right) \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 X}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \frac{\partial^2 X}{\partial x^i \partial x^j} \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \frac{\partial^2 X}{\partial x^i \partial x^j} \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \Delta g \Phi^\alpha \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \Delta g \Phi^\alpha \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \Delta g \Phi^\alpha \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \Delta g \Phi^\alpha \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \Delta g \Phi^\alpha \]

\[ = \sum_{i,j} g^{ij} \frac{\partial^2 \phi}{\partial x^i \partial x^j} - \sum_{i,j} g^{ij} \sum_{k,l,\beta} g^{kl} \frac{\partial \phi^\alpha}{\partial x^k} \frac{\partial \phi^\beta}{\partial x^l} \Delta g \Phi^\alpha \]

\[ Theorem 2. \textit{(Mantegazza [18])} \text{ Let } M^n \text{ be a compact submanifold of } \mathbb{R}^{n+1} \text{ with induced metric } g \text{ and } \tilde{X} : M \times [0, t) \rightarrow \mathbb{R}^{n+1} \text{ be an immersion which satisfies at every point } p \text{ in } M \text{ and every time } t \text{ in } [0, \tau) \]

\[ \frac{\partial \tilde{X}}{\partial t}(p, t) = \Delta_g \tilde{X}(p, t) + F(p, t) \]

\[ \tilde{X}(\cdot, 0) = \tilde{X}_0(\cdot) \]

\[ \text{here } F(p, t) \text{ is in } d\tilde{X}_1|_{(p, t)}(T_p M). \text{ Then, there exist a family of diffeomorphisms } \varphi \text{ such that } \tilde{X} \circ \varphi \text{ satisfies equation (7). Conversely, given an immersion } X : M \times [0, \tau) \rightarrow \mathbb{R}^{n+1} \text{ and a reparametrization } \varphi \text{ such that } X \circ \varphi \text{ is a mean curvature flow, then there is a field } F \text{ in } dX_1|_{(p, t)}(T_p M) \text{ such that satisfies equation (3).} \]

In the following, we restrict the study to plane curves, compare with [10] and [18]. Set a regular smooth closed plane curve \( \gamma_0 \) with unitary tangent vector \( T_0 \), normal vector \( \hat{n}_0 \) and curvature \( \kappa_0 \).

The proof of short-time existence and uniqueness of solutions for the MCF is divided in three steps. We include these details here because they will be relevant to our main result in the analysis of solutions to anisotropic MCF in Theorem [10] below.
Step 1. First, we reparametrize the deformation of the curve in the normal direction for a small time interval.

Lemma 3. Let \( \epsilon > 0 \) be small enough such that for \( t \) in \([0, \epsilon]\) the evolution by mean curvature of \( \gamma_0 \) is within a tubular neighborhood. If the deformations are parameterized by

\[
\gamma(s, t) = \gamma_0(s) + f(s, t)\hat{n}_0
\]

Then, the evolution equation for (4) is given by

\[
\frac{\partial f}{\partial t} = \frac{1 - \kappa_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}} \kappa(t)
\]

Proof. Using equation (4) we compute the unitary tangent vector \( T \), the normal vector \( \hat{n} \) and the curvature \( \kappa \) of \( \gamma(s, t) \)

\[
T = \frac{\gamma'}{|\gamma'|} = \frac{\gamma_0' + f'\hat{n}_0 - \kappa_0 f T_0}{||\gamma_0' + f'\hat{n}_0 - \kappa_0 f T_0||} = \frac{(1 - \kappa_0 f) T_0 + f'\hat{n}_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}}
\]

\[
\hat{n} = \frac{\hat{n}_0 - \frac{f'}{|\gamma'|} T}{||\hat{n}_0 - \frac{f'}{|\gamma'|} T||} = \frac{1 - \frac{(f')^2}{|\gamma'|^2}}{||\hat{n}_0 - f' T||} \hat{n}_0 - \frac{f'(1 - \kappa_0 f)}{|\gamma'|^2} T_0 = \frac{(1 - \kappa_0 f) \hat{n}_0 + f' T_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}}
\]

\[
\kappa(t) = \frac{|T' \wedge T''|}{|\gamma'|^3} = \frac{|\{ (1 - \kappa_0 f) T_0 + f' \hat{n}_0 \} \wedge \{ (-\kappa_0 - 2\kappa_0 f') T_0 + (\kappa_0 (1 - \kappa_0 f) + f'') \hat{n}_0 \}|}{|\gamma'|^3} = \frac{(1 - \kappa_0 f) f'' + 2 \kappa_0 (f')^2 + \kappa_0 f f' - 2 \kappa_0^2 f + \kappa_0^3 f^2}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2)^{3/2}}}
\]

As \( \gamma(s, t) \) solves (1), we differentiate equation (4) and take the component along \( \hat{n}_0 \):

\[
\frac{\partial f}{\partial t} = \kappa(t) \langle \hat{n}, \hat{n}_0 \rangle.
\]

Substituting and solving for \( \partial f / \partial t \) we obtain

\[
\frac{\partial f}{\partial t} = \frac{1 - \kappa_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}} \kappa(t)
\]

\( \square \)
Step 2. Denote by $W^k_2$ the set of $L^2$ functions over $\gamma_0$ whose $k$ derivatives are also in $L^2$. Let $u$ and $v$ be functions in $W^k_2$, consider the inner product
\[ \langle u, v \rangle_{W^k_2} = \int_{\gamma_0} \sum_{i,j \leq k} \frac{\partial^i u}{\partial x^i} \frac{\partial^j v}{\partial x^j} \, dx, \]
and denote by $W^{\alpha,\beta}_2$ the set of $L^2$ functions over $\gamma_0 \times (0, \infty)$ whose $\alpha$ spatial derivatives and $\beta$ time derivatives are also in $L^2$. We now state the linearized problem:

**Theorem 4.** Let $u_0$ be a $W^{\alpha+1}_2$ function over $\gamma_0$, $h \in W^{\alpha,\beta}_2$, and $L_0$ be a linear parabolic operator. The problem
\[
(LP) \begin{cases}
L_0(u) = h \\
u(\cdot, 0) = u_0
\end{cases}
\]
has a solution.

Before proceeding to the proof of theorem 4, we will need the following:

**Lemma 5.** (Garding’s inequality [29]) Let $k > 0$ an integer, suppose that the application of a linear differential operator $A$ over a $2k$-times differentiable function $f$ over $M^n \times [0,T)$ is given by
\[ A(f(x)) := \sum_{\alpha,\beta=1}^n a_{\alpha\beta} \partial_\alpha (a_{\beta\alpha} \partial_\beta f) + \sum_{\alpha=1}^n b_\alpha (b_\alpha f) + cf. \]
Let $\xi$ be in $\mathbb{R}^k$ and $\lambda, \mu$ be positive constants. Suppose that the coefficients $a_{\alpha\beta}$ are bounded and satisfy
\[ \lambda \| \xi \|^2 \leq \sum_{\alpha,\beta=1}^n a_{\alpha\beta} \xi^\alpha \xi^\beta. \]
Then, if $u(x)$ is in $W^k_2(M)$, there are constants $c_1 > 0$ and $c_2$ such that
\[ \int_M \left( \sum_{\alpha,\beta=1}^n a_{\alpha\beta} \partial_\alpha u \partial_\beta u - \sum_{\alpha=1}^n b_\alpha u \partial_\alpha u + cu^2 \right) \, dx \geq c_1 \| u \|^2_{W^k_2(M)} - c_2 \| u \|^2_{L^2(M)} \]

**Lemma 6.** (Modified Lax-Milgram theorem [27]) Let $H$ be a Hilbert space, $V \subset H$ a dense subspace and $a(u, v)$ a bilinear form in $H \times V$ such that
(1) Let $u \in H$, $v \in V$ and some $C > 0$
\[ |a(u, v)| \leq C \| u \|_H \| v \|_V. \]
(2) For every $v \in V$ and some $\delta > 0$
\[ |a(v, v)| \geq \delta \| v \|_H^2. \]
Then for every bounded linear operator $F$ in $H$ and $v$ in $V$, there exists $u$ in $H$ such that $F(v) = a(u, v)$.

Let $f, g$ in $C^\infty_0(\gamma_0 \times [0, \infty))$, we recall the following definitions from [10]:
\[ \langle f, g \rangle_{LL_m} = \int_0^\infty e^{-mt} f g \, dt \]
\[ \langle f, g \rangle_{LW^k_m} = \int_0^\infty e^{-mt} \langle f, g \rangle_{W^k_2} \, dt. \]
Here $LL$ and $LW$ are the Hilbert spaces resulting from the completion of the space of $C_0^\infty(\gamma_0 \times [0, \infty))$ functions with compact support using the norms above, in that order.

Let $V$ be the space of $C^\infty$ functions over $(\gamma_0 \times [0, \infty))$ such that $V(\cdot, t) = 0$ for small and large values of $t$, and let $WW^k$ be the completion of $V$ with the norm associated to the following inner product

$$\langle f, g \rangle_{WW^k} = \langle f, g \rangle_{LW^2_m} + \langle f_t, g_t \rangle_{LL_m}.$$  

Let

$$P^s = \{ f : \gamma_0 \times [0, \infty) \to \mathbb{R} : \left| \partial^i_t f \right|_{LW^2_m} < \infty, \forall i \leq s \}$$

endowed with

$$\langle f, g \rangle_{P^s} = \sum_{i \leq s} \left( \partial^i_t f, \partial^i_t g \right)_{LW^2(m-s)}.$$  

From the theory of parabolic linear partial differential equations theory we invoke [8, p. 351-352].

**Definition 7.** Let $Q := \gamma_0 \times (0, \epsilon)$ and $\varphi$ be a $W^{1,1}_2$ function over $\gamma_0 \times [0, \infty)$ which vanishes at $t = 0$. We say that $u(x, t)$ in $WW$ is a weak solution of the linearized problem (LP) if

$$\int_Q (u_t \varphi + a(x, t)u_x \varphi_x + b(x, t)u_x \varphi_t + c(x, t)u \varphi_t) \, dx \, dt = \int_Q h(x, t)\varphi \, dx \, dt$$  

The following lemma will be useful to construct a bounded bilinear operator, and then we will be able to apply Lemma 6.

**Lemma 8.** (See [8, Sec. 7.1.2] and [27, Sec. 3.5.1]) Let $m$ be a positive constant. An $u(x, t)$ in $WW$ is a weak solution of (LP) if and only if

$$\int_Q (u_t \varphi_t + a(x, t)u_x \varphi_{tx} + b(x, t)u_x \varphi_{tt} + c(x, t)u \varphi_t) e^{-mt} \, dx \, dt = \int_Q h(x, t)\varphi_t e^{-mt} \, dx \, dt.$$  

We can now begin the proof of theorem 4.

**Proof of Theorem 4.** First we will prove the existence of a weak solution.

Let $A : WW^2 \times V \to \mathbb{R}$ be a bilinear form given by

$$A(u, v) = \int_0^\infty \int e^{-mt} (u_t v_t + a(x, t)u_x v_{tx} + b(x, t)u_x v_t + c(x, t)uv_t) \, dx \, dt$$  

here $m$ is a constant, $u \in WW^2$, $v \in V$, and $a, b, c$ are the coefficients of the linearized problem (LP). Let $F(v)$ be a linear operator over $V$ given by

$$F(v) := \int_0^\infty \int h \, v \, e^{-mt} \, dx \, dt.$$  

We will prove that $A$ satisfies the hypotheses of lemma 6:

- Condition (1). The coefficients $a, b, c$ and $h$ are bounded because they were found by linearization of a bounded function in a tubular neighborhood, see Equation 5 in Lemma 3. Let $C$ be the maximum of the bounds for $a, b$ and $c$. Then,

$$A(u, v) \leq \int_0^\infty \int e^{-mt} (u_t v_t + C(u_x v_{tx} + u_x v_t + uv_t)) \, dx \, dt$$

$$\leq \max(1, C) \int_0^\infty \int e^{-mt} (u_t v_t + u_x v_{tx} + u_x v_t + uv_t) \, dx \, dt.$$
Noticing that this double integral corresponds to the inner product of \( u \) and \( v \) in \( P^1_2 \). Then,
\[
|A(u, v)| \leq \max(1, C) \| u \| \| v \|_{P^1_2} \leq \max(1, C) \| u \|_{W^1} \| v \|.
\]

- To verify that condition (2) also holds observe that:
\[
A(v, v) = \int_0^\infty \int (v^2_i + auxvt + bvxv_i + cvv_i) e^{-mt} dx \, dt
\]
\[
= \int_0^\infty \int v^2_i e^{-mt} dx \, dt + \int_0^\infty \int (auxv + bvx + cvv_i) e^{-mt} dx \, dt
\]
\[
= \|v_i\|^2_{L^2_m} + \int_0^\infty \int (avxv + bvxv + cvv_i) e^{-mt} dx \, dt
\]
Integrate by parts to obtain:
\[
A(v, v) = \|v_i\|^2_{L^2_m} + \frac{m}{2} \int_0^\infty \int e^{-mt} (av^2_x + bvx + cv^2) \, dx \, dt
\]
\[
- \frac{1}{2} \int_0^\infty \int e^{-mt} (avxv^2 + bvxv + cv^2) \, dx \, dt - \frac{1}{2} \int av^2_x|_{t=0} dx
\]
Let \( S \) be an upper bound for \( a_t, b_t, c_t \). Using lemma 5 there exists constants \( C_1 \) and \( C_2 \) such that:
\[
A(v, v) \geq \|v_i\|^2_{L^2_m} + \frac{m}{2} \int_0^\infty e^{-mt} C_1 \|v\|^2_{W^1_2} \, dt - \frac{1}{2} \int_0^\infty e^{-mt} S \|v\|^2_{W^1_2} \, dt
\]
\[
- C_2 \frac{m}{2} \int_0^\infty e^{-mt} \|v\|^2_{L^2} \, dt - \frac{C}{2} \|v_x(0)\|^2_{L^2}
\]
\[
= \|v_i\|^2_{L^2_m} + \left( \frac{mC_1 - S}{2} \right) \|v\|_{L^2}^2 - \frac{C_2}{2} \|v\|_{L^2}^2 - \frac{C}{2} \|v_x(0)\|^2_{L^2}
\]
\[
\geq \|v\|^2_{W^1_2}
\]
The last inequality follows from having properly chosen \( m \).

From the above arguments we obtain that for all \( v \in V \) there exists \( u_L \) such that \( F(v) = A(u_L, v) \). Since Equation (11) can be obtained from (12) and (13), \( u_L \) is therefore a weak solution. \( \square \)

We proved the existence of weak solutions. However, the regularity of weak solutions for linear parabolic problems depends on the regularity of the data \( h \) and \( u_0 \). Since the existence of solutions for linear parabolic problems is an auxiliary result in our work, we only refer to [8 Sec. 7.1.3] for the regularity extension.

The uniqueness of weak solutions for linear parabolic operators is also stated in [5] by considering the difference of two weak solutions \( w := u_1 - u_2 \). Since \( w \) satisfies the problem (LP) with \( h = w(\cdot, 0) = 0 \), from lemma 5 and substituting \( w \) into (10) we obtain
\[
\frac{d}{ds} |w(\cdot, s)|^2_{L^2} \leq C |w(\cdot, s)|^2_{L^2}
\]
Substituting the last inequality into the derivative of $|w|^2_{L^2} e^{-Cs}$, one obtains
\[
\frac{d}{ds} \left( |w(s)|^2_{L^2} e^{-Cs} \right) \leq 0
\]
Since $|w(s)|^2_{L^2} e^{-Cs}$ is non-negative, $|w(t)|^2_{L^2} e^{-Ct} - |w(0)|^2_{L^2} \leq 0$. Thus $w = 0$ implies $u_1 = u_2$.
A similar procedure can be used to prove the uniqueness of solutions for quasilinear parabolic problems. The main hypothesis is an estimate similar to (14), in particular for MCF see [4].

**Step 3.** We now proceed to extend the linearized solution of normal deformation (Theorem 4) to the nonlinear problem

**Theorem 9.** Given a quasilinear parabolic problem
\[
\begin{align*}
\partial_t u - a(u, x, t)u_{xx} - b(u, x, t)u_x - c(u, x, t)u &= h(u, x, t) \\
u(\cdot, 0) &= u_0.
\end{align*}
\]
in $\gamma_0 \times [0, \epsilon)$. Then, there exist $\epsilon > 0$ such that this problem has a solution.

**Proof.** Define the map $L : P^2 \rightarrow W^{2,1}_2 \times P^1$ by
\[
u \mapsto (u_0, L_1(u)).
\]
Notice that a function $u$ such that $L(u) = 0$ is a solution for the quasilinear problem. To this end, let $L_{u_0}$ be the linearization of $L_1$ around $u_0$, by Theorem 4 the problem
\[
(15)
\]
has a solution. Let $w^t := w(\cdot, t)$ and $\{w^t\}$ be a sequence converging to $u_0$ as $t$ tends to 0 from above. Suppose that $w$ is a solution of $(15)$, and let $a(u_0, x, t)$, $b(u_0, x, t)$, $c(u_0, x, t)$ and $h(u_0, x, t)$ be the coefficients of $L_{u_0}$. Additionally, consider the linearization of $L_1$ around $w$, let $a(w, x, t)$, $b(w, x, t)$, $c(w; x, t)$ and $h(w; x, t)$ be its linearized coefficients. From $(15)$ we have
\[
L_1(w) = \partial_t w - P(x, t, w, D_x w)
\]
\[
= h(u_0, x, t) + a(u_0, x, t)w_{xx} + b(u_0, x, t)w_x + c(u_0, x, t)w - P(x, t, w, w_x, w)w_x
\]
\[
= h(u_0)u_{xx} - h(w)w_{xx} + a(u_0)u_x - a(w)w_x + b(u_0)u - b(w)w + c(u_0)u - c(w)w.
\]
here $P(x, t, w, D_x w) = a(w, x, t)u_{xx} + b(w, x, t)u_x + c(w, x, t)u + h(w, x, t)$. Notice that $w \rightarrow u_0$ as $t$ tends to 0 from above, implies $L_1(w) \rightarrow 0$ as $t \rightarrow 0^+$.

We can take $\epsilon$ small enough such that $w(x, t)$ be in a neighborhood of $u_0$ for all $t$ in $[0, \epsilon)$. Solving each linear problem, we end with a sequence $w^t$ such that $L(w^t) = (u_0, L_1(w^t))$. Consider $L(u) = (u_0, 0)$ by continuity. Invoking the Inverse Function Theorem for Banach spaces [28, Sec. 4.13], $L$ is a local diffeomorphism. Therefore, the map $L(u) = (u_0, 0)$ is locally invertible. □

2. **Anisotropic MCF Proposition**

This section is focused on Lagrangian methods. These methods track the evolving curve by explicitly computing the update of coordinates for each point in the curve.

A curve $\gamma$ evolving by MCF will stop its evolution on some point, if the curvature $\kappa$ at that point equals zero. If we need to evolve the curve further even though $\kappa = 0$, we need to modify
the Equation [1] using another curvature dependent normal flow. The equation that will describe this new flow is the following:

\[ \frac{\partial \gamma}{\partial t} = v \hat{n}, \]

here the speed \( v \) is assumed to depend on the curvature.

Consider the system:

Let \( \rho \) be a given distribution, and \( \gamma_0 \) be an initial smooth simple closed curve. We are interested in the evolution of \( \gamma_0 \) by

\[
\begin{align*}
(P1) \quad & \frac{\partial \gamma}{\partial t} = \frac{\partial u}{\partial \hat{n}} \hat{n} \\
& \gamma(\cdot, 0) = \gamma_0
\end{align*}
\]

(P2) \quad \Delta u = \rho \quad \text{in } \Omega \\
\quad u(x, 0) = g(x) \quad \text{in } \partial \Omega

Notice that in (P1) we have a normal field which drives the evolution, and in (P2), a Poisson equation which defines a field in terms of its potential \( u \) with Dirichlet’s boundary condition. To link these problems with the MCF, we impose that if \( \rho \) equals zero, then the MCF is recovered i.e. \( \partial u/\partial n = \kappa \). Next, we need to prove the existence and uniqueness of solutions for (P1) and (P2) before we introduce our numerical solution.

The following theorem is our main theoretical result in this paper:

**Theorem 10.** Let \( \gamma_0 \) be the initial curve, denote by \( \Omega \) its interior. Let \( r > 0 \) and \( p \) be a point in \( \Omega \) such that \( |x - p| > 0 \) for all points \( x \) in a \( r \)-tubular neighborhood of \( \gamma_0 \). Let \( g \) be a \( C^2(\partial \Omega) \) function, \( \rho(y) = \delta(y - p) \) be the Dirac’s delta distribution, and \( \gamma_0 \) be an initial smooth simple closed curve. Then, the system (P1)-(P2) has a unique solution.

**Proof.** The problem (P2) is a Poisson equation with a Dirichlet boundary condition. Therefore, the existence and uniqueness of its solution is well known when \( g \) is a \( C^2 \) function with compact support (see, for example, [8]).

The problem (P1) represents a curve being deformed in the normal direction with speed \( \partial u/\partial \hat{n} \). Therefore, we can consider a tubular neighborhood about \( \gamma_0 \), and verify the existence and uniqueness of solutions of (P1) for small time. Similarly to Lemma [3] let \( \hat{n}_0 \) be the unitary normal at time \( t = 0 \), we reparametrize the deformations inside the \( r \)-tubular neighborhood by:

\[
\gamma(s, t) = \gamma_0(s) + f(s, t, u)\hat{n}_0.
\]

Taking the time derivative of the last equation, and recalling (P1), we obtain

\[
\frac{\partial \gamma}{\partial t} = \left( \frac{\partial f}{\partial t} \right) \hat{n}_0 = \frac{\partial u}{\partial \hat{n}} \hat{n}.
\]

As \( \hat{n}_0 \) is a unitary vector, we can state the partial equation for \( f \):

\[
\frac{\partial f}{\partial t} = \frac{\partial u}{\partial \hat{n}} \langle \hat{n}, \hat{n}_0 \rangle.
\]

The dot product in equation (16) is already computed in Lemma [3], and \( u \) is the solution of the Poisson problem (P2). Then, invoking the general solution for Poisson’s problem and the dot product in Lemma [3], we rewrite equation (16):

\[
\frac{\partial f}{\partial t} = \frac{\partial}{\partial \hat{n}} \left( \int_{\Omega} G(x - y) \rho(y) \, d\sigma_y + \int_{\partial \Omega} \frac{\partial G}{\partial n_y}(x - y) g(y) \, dy \right) \frac{1 - \kappa_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}}
\]
The integrands in previous equation are Green’s function and the Poisson kernel, whose existence is known [14, Sec. 8.2]. Comparing (17) and (5) we notice that the second derivative of \( f \) does not appear explicitly. Consequently, some assumptions are required in order to link (P1) and (P2) with the mean curvature flow: We claim that

\[
\frac{\partial}{\partial n} \int_{\partial \Omega} \frac{\partial G}{\partial n}(x - y) g(y) \, dy = \kappa(x, t)
\]

holds, we will carry out the demonstration at the end of this proof. Substituting Equations (8) and (18) into (17), we obtain

\[
\frac{\partial f}{\partial t} = \left( \frac{\partial}{\partial n} \int_{\Omega} G(x - y) \rho(y) \, d\sigma_y \right) \frac{1 - \kappa_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}}
+ \kappa(t) \frac{1 - \kappa_0}{\sqrt{(1 - \kappa_0 f)^2 + (f')^2}}.
\]

The term involving \( \kappa(t) \) carries the evolution by mean curvature (compare with lemma 3). The additional term depends on the distribution function \( \rho \), and does not include second order derivatives of \( f \). Thus, the parabolicity is not affected. The integral in this term equals

\[
\int_{\Omega} G(x - y) \rho(y) \, d\sigma_y = \int_{\Omega} G(x - y) \delta(y - p) \, d\sigma_y = G(x - p).
\]

Recall the following properties of Green’s function [8]:

- \( G(x - y) \) equals zero at the boundary \( \partial \Omega \).
- Green’s function satisfies \(-\Delta G(x - y) = \delta(x)\).
- In general, Green’s function takes the form \( G(x - y) = \Phi(|x - y|) + \phi(x, y) \). Here, \( \Phi(|x - y|) \) is the fundamental solution of Laplace’s equation, \( \phi(x, y) \) is known as the corrector function and satisfies
  \[
  \Delta_x \phi(x, y) = 0 \quad \text{in} \quad \Omega
  \]
  \[
  \phi(x, y) = \Phi(|x - y|) \quad \text{in} \quad \partial \Omega.
  \]
- For a 2-D problem,
  \[
  \Phi(|x - y|) = -\frac{1}{2\pi} \log(|x - y|).
  \]

Then, the derivatives of \( G(x - y) \) exist because \( p \) is outside the tubular neighborhood, and it is linearizable in this neighborhood. Consequently, equation (19) can be linearized as a parabolic partial differential equation. The solution for the linearized problem exists by Theorem (4), and it can be extended to a solution of (19) by Theorem (9).

Finally, we state the proof of Equation (18), requiring that \( \partial u / \partial n = \kappa \) if \( \rho = 0 \):

When \( \rho = 0 \) in (P2), we want to recover the MCF. Then, from problem (P1) and \( \rho = 0 \) in (P2), we can state the Laplace’s problem with Neumann’s boundary condition

\[
(P3) \quad \begin{cases}
  \Delta u = 0 \quad \text{in} \quad \Omega \\
  \frac{\partial u}{\partial n} = \kappa \quad \text{in} \quad \partial \Omega.
\end{cases}
\]
Recalling problem (P2), let \( x \) be a point in the boundary \( \partial \Omega \). By Green’s function relation to the Laplacian as recalled above, we write

\[
g(x) = u(x) = \int_{\partial \Omega} \Delta y G(x - y) u(y) \, dy
\]

\[
= \int_{\Omega} G(x - y) \Delta u(y) \, dy + \int_{\partial \Omega} \left( u(y) \frac{\partial G}{\partial n} + G(x - y) \frac{\partial u}{\partial n} \right) \, dy.
\]

The last equality follows from Green’s identity. Substituting problem (P3) yields:

\[
g(x) = u|_{x \in \partial \Omega} = \int_{\partial \Omega} \left( u(y) \frac{\partial G}{\partial n} + G(x - y) \kappa(y) \right) \, dy
\]

Since \( G(x - y) \) is null on the boundary, the second term cancels. Taking the normal derivative

\[
\frac{\partial g}{\partial n} = \frac{\partial u}{\partial n} = \frac{\partial}{\partial n} \int_{\partial \Omega} u(y) \frac{\partial G}{\partial n_y} \, dy.
\]

Using von Neumann’s boundary condition in (P3), one obtains (18).

The uniqueness of solutions for Poisson problems determines the uniqueness of solutions for P2 and P3. Given \( \gamma_1(t) = (x_1(t), y_1(t)) \) and \( \gamma_2(t) = (x_2(t), y_2(t)) \) two solutions of P1, consider \( \gamma = \gamma_1 - \gamma_2 \). Since both solutions satisfy the initial data, then \( \gamma(\cdot, 0) = (x(0), y(0)) = (0, 0) \) and \( \partial u/\partial \eta = 0 \). This implies that \( \gamma \) is stationary and \( x_1(t) = x_2(t) \) and \( y_1(t) = y_2(t) \).

**Remark.** A curve evolution by (P1-P2) subject to Equation (18) when \( \rho(y) = 0 \) states a coupled system defined by problems (P1, P2 and P3).

The next section deals with the numerical solution for our curve evolution. To achieve the computations, we use the integral representation for problem (P2) and (P3). Recalling the theory of harmonic functions and Poisson type problems [8], every solution of a Dirichlet problem

\[
\begin{cases}
\Delta u = h \quad \text{in} \quad \Omega \\
u(x) = g(x) \quad \text{in} \quad \partial \Omega
\end{cases}
\]

has a unique solution, and it must moreover satisfy Poisson’s Representation Formula:

**Lemma 11.** Let \( x \) be in \( \Omega \) and let

\[
\Phi(x) := \frac{1}{2\pi} \log \left( \frac{1}{|x|} \right)
\]

be the fundamental solution of Laplace’s equation and \( u \) be a solution of Poisson’s equation, then the following formula holds:

\[
(20) \quad \int_{\Omega} \Phi(x - y) \Delta u(y) \, dy = \int_{\partial \Omega} \Phi(x - y) \frac{\partial u}{\partial n} \, dy - \int_{\partial \Omega} u \frac{\partial \Phi}{\partial n} \, dy - u(x)
\]

See [8] for a proof.

In the next section, we will use a similar formula for \( x \) in \( \partial \Omega \). These results will be crucial for our numerical approximation.

**Proposition 12.** Let \( x \) be in \( \partial \Omega \), and \( u \) be a solution of Poisson’s equation, then the following holds:

\[
(21) \quad \int_{\Omega} \Phi(x - y) \Delta u(y) \, dy = \int_{\partial \Omega} \left( \Phi(x - y) \frac{\partial u}{\partial n} - u \frac{\partial \Phi}{\partial n}(x - y) \right) \, dy - \frac{1}{2} u(x)
\]
Proof. Notice that in Equation (20) \( \Phi \) becomes singular when \( x \) tends to the boundary \( \partial \Omega \). Consider a neighborhood \( \ell(\delta, x) \) over \( \partial \Omega \) and then take the limit \( \delta \to 0 \). For the integrals on the right hand side over \( \partial \Omega \) in Equation (20) we use Green’s second identity:

\[
\int_{\partial\Omega} \Phi(x - y) \frac{\partial u}{\partial n} \, dy - \int_{\partial\Omega} u \frac{\partial \Phi}{\partial n} \, dy - u(x) = \int_{\partial\Omega - \ell(\delta, x)} \left( \Phi(x - y) \frac{\partial u}{\partial n} - u \frac{\partial \Phi}{\partial n}(x - y) \right) \, dy + \int_{\ell(\delta, x)} u \Delta \Phi(x - y) \, dy + \int_{\ell(\delta, x)} \left( \Phi(x - y) \frac{\partial u}{\partial n} - u \frac{\partial \Phi}{\partial n}(x - y) \right) \, dy.
\]

Then,

\[
\int_{\partial\Omega} \Phi(x - y) \frac{\partial u}{\partial n} \, dy - \int_{\partial\Omega} u \frac{\partial \Phi}{\partial n} \, dy - u(x) = \int_{\partial\Omega - \ell(\delta, x)} \left( \Phi(x - y) \frac{\partial u}{\partial n} - u \frac{\partial \Phi}{\partial n}(x - y) \right) \, dy + \int_{\ell(\delta, x)} \Phi(x - y) \frac{\partial u}{\partial n} \, dy - \int_{\ell(\delta, x)} u \frac{\partial \Phi}{\partial n}(x - y) \, dy.
\]

For the first integral over \( \ell(\delta, x) \) in the last equation, the following limit holds:

\[
\lim_{\delta \to 0} \int_{\ell(\delta, x)} \Phi(x - y) \frac{\partial u}{\partial n} \, dy = 0
\]

For the second integral over \( \ell(\delta, x) \):

\[
- \int_{\ell(\delta, x)} u(y) \frac{\partial \Phi}{\partial n}(x - y) \, dy = - \frac{1}{2\pi \delta} \int_{\ell(\delta, x)} u(y) \, dy
\]

As \( \delta \) tends to 0, the last expression equals \(-\frac{1}{2} u(x)\) in the limit. Finally, we rewrite equation (22):

\[
\int_{\Omega} \Phi(x - y) \Delta u(y) \, dy = \int_{\partial\Omega} \left( \Phi(x - y) \frac{\partial u}{\partial n} - u \frac{\partial \Phi}{\partial n}(x - y) \right) \, dy - \frac{1}{2} u(x).
\]

\[\square\]

3. Numerical anisotropic MCF

In this section we develop the procedure for the numerical solution of the system (P1-P2-P3).

Suppose that a curve \( \varphi \) evolves by an equation of the form

\[
\frac{\partial \varphi}{\partial t} = v n.
\]

This flow may be discretely approximated as follows: Let \( \Delta t \) be the step in time and \( k \) be a positive constant such that \( k\Delta t \) represents the elapsed time. The evolving curve is represented
by a set of $N$ points $\{\varphi^k_j\}$, here $1 \leq j \leq N$. For an illustrative diagram see Figure 8. The tangent vectors at $\varphi^k_j$ can be approximated using a high order finite difference formula, for example

$$\frac{d\varphi^k_j}{ds}\bigg|_{s_j} \approx -\varphi^k_{j+2} + 8\varphi^k_{j+1} - 8\varphi^k_{j-1} + \varphi^k_{j-2}$$

$$= -\frac{\varphi^k_{j+2} + \varphi^k_{j} + 8\varphi^k_{j+1} - 8\varphi^k_{j-1} + 8\varphi^k_{j} - \varphi^k_{j-2}}{12|\varphi^k_{j+1} - \varphi^k_{j}|}.$$  \hspace{1cm} (24)

Using Equation (24) and the assumption that for all points the quantity $|\varphi^k_{j+1} - \varphi^k_{j}|$ can be approximated by a constant when fixing $k$, in [13] a set of formulas is presented, and they also approximate the tangent vectors $T_j$ and curvature $\kappa_j$:

$$d_j := |\varphi^k_{j+1} - \varphi^k_{j}|$$

$$\tau_i := \frac{i}{|i|} \frac{\varphi^k_{j+i} - \varphi^k_{j}}{d_i}, \text{ for } i = -2, -1, 1, 2$$

$$T_j = -\tau_2 + 4\tau_1 + 4\tau_{-1} - \tau_{-2}$$

$$\kappa_j = \frac{6}{d_1 - d_{-1}} \frac{2(\tau_1 - \tau_{-1})}{d_1 - d_{-1}} + \frac{(1 - \mu)2(\tau_2 - \tau_{-2})}{d_2 - d_{-2}} + \frac{\mu \nu_1}{d_1 - d_{-1}}$$  \hspace{1cm} (25)

Equation (23) does not have a tangential term, but an approximation for tangential vectors is needed when we reparametrize the evolution in a tubular neighborhood. By Theorem (2), we know that any reparametrization will give rise to a tangential term in the evolution equation. Then, the formula for $T_j$ provides the direction of this tangent field, we only need to know the coefficients $a^k_j$ of this field at each $\varphi_j$. Notice that $|\varphi^k_{j+1} - \varphi^k_{j}|$ is a first order approximation for tangential speed. Then, rescaling $T_j$ formula to get $|\varphi^k_{j+1} - \varphi^k_{j}| = 1$, it represents a numerical arc-length parametrization. In [13], the reparametrization is such that $|\varphi^k_{j+1} - \varphi^k_{j}|$ is constant for every $j$ and fixed $k$, and the $a^k_j$ are given by

$$a^k_{j+1} - a^k_j = \frac{\nu_k/\Delta t - d^k_j}{\Delta t}$$

$$\sum_{j=1}^{n} a^k_j = 0.$$  \hspace{1cm} (26)

When $\nu = \kappa$ in Equation (23), that is isotropic mean curvature flow, an approximation for the term along the normal direction was given in [13]. Here we are concerned with finding a
numerical solution to the system (P1-P2), and we seek an approximation of the form
\begin{equation}
\varphi_{j}^{k+1} = \varphi_{j}^{k} + \Delta t \left( a_{j}^{k} T_{j}^{k} + v_{j}^{k} n_{j}^{k} \right).
\end{equation}

Equation (26) can be used to compute the tangential component of Equation (27) for the problem (P1-P2). The normal component requires in addition the use of Proposition (12). Substituting the data of problem (P2) into equation (21), we obtain
\begin{equation}
- \int_{\partial\Omega} \Phi(x - y) \frac{\partial u(y)}{\partial n} dy = - \int_{\Omega} \Phi(x - y) \rho(y) d\sigma y + \int_{\partial\Omega} u(y) \frac{\partial \Phi}{\partial n}(x - y) dy + \frac{1}{2} u(x).
\end{equation}

The last two terms in the right side of (28) depend on the values of \(u\) on the boundary \(\partial\Omega\). According to Theorem 10, we define \(u\) implicitly through (P3). Then, in addition to (28), we need to consider Poisson’s representation formula for (P3):
\begin{equation}
- \int_{\partial\Omega} \Phi(x - y) \kappa(y) dy = \int_{\partial\Omega} u(y) \frac{\partial \Phi}{\partial n}(x - y) dy + \frac{1}{2} u(x).
\end{equation}

Discretizing (28) and (29) we find two systems of \(N\) linear equations. Notice that, in this case, we can solve (29) independently because there is no \(\partial u/\partial n\) dependence.

In order to solve equations (28) and (29) numerically, we need to approximate the integrands on certain points. To approximate \(\kappa\) at each \(\varphi_{j}^{k}\), we use (25) from Kimura’s numerical scheme. In general, from discrete values at \(\varphi_{j}^{k}\) and \(\varphi_{j+1}^{k}\), we can determine the values for points in between by linear interpolation. This procedure is required to compute \(\kappa\), \(u\) and the normal vector for points along each arc \(\varphi_{j}, \varphi_{j+1}\). In contrast, Laplace’s fundamental solution \(\Phi\) and its derivatives can be substituted explicitly.

One can split the integrals in (28) and (29) into \(N\) integrals over the arcs \(\varphi_{j}, \varphi_{j+1}\). If each arc is reparametrized by an interpolating function, all the integrals can be reduced to a linear system of \(N\) equations, which can be solved numerically. Then, the procedure for (28) and (29) is to solve (29) for \(u\), and use these values to interpolate \(u\) at the integrands in (28).

4. IMPLEMENTATION OF OUR NUMERICAL ANISOTROPIC MCF TO CONTOUR PARAMETRIZATION

For a contour parametrization problem, let \(\varphi\) be a closed planar and smooth curve drawn on a picture. In addition, suppose that the curve is enclosing a unique object in the image. We perform a curve evolution that moves the curve towards the boundary of the object. We now apply the numerical procedure in the last section to perform this evolution. First, we will describe the procedure and some assumptions we will need, and finally we will state the algorithm.

To achieve all the numerical computations we need to define a suitable function \(\rho\) for problem (P2). As we pointed out previously, \(\rho\) encodes the normal field which allows the curve to evolve further when \(\kappa = 0\). In analogy with many physical problems, suppose a point \(p\) is inside \(\partial\Omega\), and let \(\delta(x - p)\) be the Dirac delta function. Then, let \(\rho(y) = -\delta(y - p)\) be the distribution in (P2) and in equation (28), the integral \(\int_{\Omega} \Phi(x - y) \rho(y) dy\) is easily computed and curve will shrink toward \(p\). At each time \(k\Delta t\) we need to recompute the coefficients \(a_{j}^{k}\) and \(v_{j}^{k}\). Thus, there is no restriction in choosing a non fixed \(p\), as long as it remains inside the curve \(\varphi\) because the problem (P2) requires a \(\rho\) defined in the area \(\Omega\) enclosed by \(\varphi\). Although a non fixed \(p\) may match the curve and the object faster, a stability result in such a context would be
more complicated to state. The stability of our scheme will be analyzed in the next section.

As the curve evolves by equation (27), at each \( \varphi_j^k \) we can pick a pixel value. Suppose, for illustration purposes, that the picture on which the curve is evolving is in high contrast. That is, the pixels which belong to the object have a very different value from those belonging to the background. Then, we can distinguish if a point reaches the object by its pixel value.

Recalling (19), the the curve evolves by a combination of the contributions of \( \rho \) and a standard MCF. When a point \( \varphi_j^k \) meets the object, we need to constrain its motion vanishing \( \rho \) and its curvature \( \kappa \). For our examples we considered black pixel objects in a white background. Let \( \text{Pix}(y) \) be the pixel value function, we can make \( \rho \) and \( \kappa \) vanish through:

\[
\rho^*(y) = -\delta(y - p) \frac{\text{Pix}(y)}{255}, \quad \kappa^*(y) = \kappa(y) \frac{\text{Pix}(y)}{255}.
\]

Division by 255 in the previous system of equations normalizes the Pix function, because the color intensities are usually represented by values in \([0, 255]\). Additionally, we can consider other pixel value functions depending on the number of color channels (RGB or CMYK).

To include these constraints, we summarize the previous ideas as follows:

I Let \( k = 0, \Delta t \) be positive, consider a set of \( N \) points \( \{ \varphi_j^k \} \) representing a closed planar smooth curve, a high contrast picture with a single object enclosed by the set of points, and an initial set of \( \{ (c_i, p_i) \} \) required to define \( \rho^*(y) = \sum c_i \delta(y - p_i) \) according to (30).

II Check if the set of points \( y_i \) are in the enclosed area of \( \{ \varphi_j^k \} \). If it is not, finish procedure (or optionally, compute/ask for a new set \( \{ (c_i, y_i) \} \)).

III Compute the tangents, normals, curvatures and tangential coefficients \( a_j^k \) using (25) and (26).

IV Compute the normal \( \partial u_j^k / \partial n_j \) coefficients solving a \( 2N \) linear system:

IV.a Discretize equation (28) to obtain \( N \) linear equations with \( 2N \) unknown variables \( u_j \) and \( \partial u_j / \partial n_j \). These equations cannot be modified because they represent \( (P1) \) and \( (P2) \). Our task is to determine the normal velocities \( \partial u_j / \partial n_j \).

IV.b If we use (30) to determine where \( \rho \) is zero, problem \( (P3) \) takes the form:

\[
\Delta u = \rho^* \text{ on } \Omega, \quad \frac{\partial u}{\partial n} = \kappa^* \text{ on } \partial \Omega.
\]

This system switches off \( \rho \) and \( \kappa \) only for those points which have zero pixel value. After discretization by (28), this problem will lead to \( N \) linear equations which can be solved numerically along with IV.a.

V Compute the new positions \( \{ \varphi_j^{k+1} \} \) substituting \( a_j^k \) and \( v_j^k := \partial u_j / \partial n_j \) into (27).

VI Check how many points have reached the object using its pixel value, if most of them have reached the object (according to the set threshold), then stop.

VII Return to II.

The stopping point in VI will depend on a pre-assigned threshold. We used values above 90%—explained below—for our experiments. Alternatively, one could to fix a maximum number of iterations to be carried out. This is a feature of our approach that can not be avoided, as otherwise the process will continue indefinitely.

The previous steps are distilled into the following algorithm. It has a fixed maximum possible value of \( C \geq 90\% \) of contour reached, as an example.
Algorithm 1 Contour parametrization process

1: procedure
2: Initialize:
3: \{\varphi_j\}^{k=0} \leftarrow \text{points over initial curve}
4: \{(c_i, y_i)\} \leftarrow \text{data to define } \rho
5: \Delta t \leftarrow \text{fixed step size}
6: loop over \(k\):
7: if there is \((c_i, y_i)\) not inside the curve then
8: STOP
9: end if
10: # for all \(\varphi^k_j\) compute:
11: \text{Pix}^k_j \leftarrow \text{pixel value}
   \quad \text{T}^k_j \leftarrow \text{tangent vector}
   \quad \text{n}^k_j \leftarrow \text{normal vector}
12: \text{K}^k_j \leftarrow \text{curvature vector}
   \quad \text{a}^k_j \leftarrow \text{tangent coefficient}
13: \text{u}^k_j, (\partial u_j / n_j)^k \leftarrow \text{normal coefficient using (28), (29) and (30)}
14: \{\varphi^{k+1}_j\} \leftarrow \text{update using (27)}
15: \text{C} \leftarrow \text{number of points matching the object}
16: if \text{C} \geq 90\% then
17: STOP
18: end if
19: end loop
20: end procedure

We present some examples obtained with this procedure in Figure 9 with their detailed parameters in Table 1. To estimate the accuracy in the contour detection, we compare the black pixels area and the area inside the last curve in evolution. The source code for these examples is available at https://github.com/V3du4rd0/AMCF.

Execution times

We used a small number of points to represent the evolving curve. Nevertheless, our numerical examples demand a considerable number of iterations. These iterations increase the computation time in a single processor computer. In order to reduce the execution time, we parallelized step 11, 12 and 13 in Algorithm 1. This will reduce execution times if a long number of points are needed. All the numerical experiments were performed using double precision float point arithmetic. We used Nvidia’s Jetson computer model TK1, with 2GB RAM memory, ARM A15 processor and Nvidia’s Tegra K1 graphic card with 192 CUDA cores.

To provide a picture of how fast our process can be performed, we implement the Algorithm (1) varying the number of points \(N\) and setting the time step \(\Delta t = 1/N^2\), we plot the time versus the number of points. This (execution time) plot is presented in Figure 10. Here, the dots represent the mean time computed by taking ten samples, and the bars are the deviation of these values. The plot was obtained from a circumference as initial curve, enclosing a black circle picture with the same center. The Algorithm (1) was initialized with only one \(p\) also
Figure 9. Non-convex and convex contour parametrization examples using our AMCF. A radius 8 circumference evolves according Algorithm 1 to match different pictures. In these examples, the pictures are $1800 \times 1800$ pixels, scaled to 1 pixel = 0.001, only one charge was placed at $p = (0, 0)$. We display few curves for demonstrative purposes. The number of iterations and additional parameters are shown in Table 1 below.

Table 1. Parameters used for pictures in Figure 9. Symbology: Number of points $N$, time step $\Delta t$, Parameter for curvature approximation $\mu$ according to [13], number of iterations $\# \text{it}$, and the accuracy $\text{Acc.}(\%)$.

| N   | $\Delta t \times 10^{-3}$ | $\mu$ | $\# \text{it}$ | $\text{Acc.}(\%)$ |
|-----|--------------------------|-------|-----------------|-------------------|
| Fig. 9 a | 30 | 1.11 | 0.15 | 26140 | 97 |
| plotted each 1000 it. | 100% of points matched |
| Fig. 9 b | 30 | 1.11 | 0.15 | 30820 | 98 |
| plotted each 2000 it. | 100% of points matched |

localized in the center of the curve. The evolution was stopped when at least 90% of the points had reached the object. These results were achieved in parallel to speed up the process.

5. Conditioning and Stability of our Numerical scheme

In this section we state our main results on the numerical analysis of the evolution scheme. Let $A$ be the resulting $2N \times 2N$ matrix associated to the linear system when Equations (28) and (29) are discretized.
In order to understand the effect of the source $p$, observe that equations (28) and (29) depend on $p$ through $\Phi(|x-p|)$. As Laplace’s fundamental solution $\Phi$ is a logarithmic function, when $p$ approaches the boundary, larger values of $\Phi$ and its derivatives will appear for the curve points near $p$. Consequently, these values may increase some entries in the matrix in the $2N$ linear system. Therefore, the matrix norm $\| \cdot \|_\infty$ may change drastically when $|x-p|$ is lower than one because

$$
\|A\|_\infty = \max_{0 \leq i \leq n-1} \sum_{j=0}^{n-1} |a_{ij}|.
$$

The condition number $C$ of a matrix $A$ with the norm $\| \cdot \|_\infty$ is thereby affected:

$$
C(A) = \|A\|_\infty \|A^{-1}\|_\infty.
$$

Let $\{p_i\}_{i=0}^n$ be a set of source coordinates. Denote by $C_{i,k}$ the condition number of $A$ at time $k\Delta t$ when $p = p_i$. We now repeat example a in Figure 9 for different charge positions $p_i$, from $p_0 = (0,0)$ to $p_6 = (-3.5, -5)$. In Figure 11 we present how the condition number $C(A)$ increases as $p$ approaches the curve, the condition number when $p$ is at $p_0$ is used as reference ($C_{i,k}/C_{0,k}$). Notice in Figure 11 that the highest values of $C_{i,k}$ are obtained for $i = 6$, which corresponds to the point $(p_6)$ which is closest to boundary.
Figure 11. The example in figure (a) is repeated for different coordinates $p_i$ from $p_0 = (0,0)$ to $p_6 = (-3.5,-5)$. The condition number $C(A)$ of the matrix increases as $p_i$ approaches the curve points (a). The $x$ axis is the time when the curve matches the black pixels nearest to $(-3.5,-5)$. In (b) we display schematically the chosen coordinates for $p_i$.

Remark. In order to maximize the distance between $p$ and the curve in a convex contour parameterization problem, a suitable choice of $p$ at time $k\Delta t$ is the centroid coordinates of the set $\{\varphi_j^k\}_{j=0}^{N-1}$. Further investigating other possible choices of placement for the charge in the general case, or increasing the number of charges, constitutes an interesting avenue for future research, and lies outside the scope of our work here.

We now focus on the stability of the updating scheme

$$\varphi_j^{k+1} = \varphi_j^k + \Delta t \left( a_j^k T_j^k + v_j^k n_j^k \right).$$

Let $\varphi(s,t)$ be the curve at time $t = k\Delta t$ represented by a set of $N$ points $\{\varphi_j^k\}_{j=0}^{N-1}$, its length $L$ is approximated by $\ell = \sum_{j=1}^{N} |\varphi_{j+1} - \varphi_j|$. We will rewrite the tangential and normal vectors in (31) and thus obtain an expression only in terms of the set $\{\varphi_j^k\}_{j=0}^{N-1}$. Recall the definitions of $T_j^k$ and $n_j^k$:

\begin{align}
T_j^k &:= \frac{-\tau_2 + 4\tau_1 + 4\tau_{-1} - \tau_{-2}}{6} \\
\tau_i &:= \text{sign}(i)(\varphi_{j+i}^k - \varphi_j^k) \\
d_i &:= |\varphi_{j+i}^k - \varphi_j^k| \\
n_j^k &:= (T_j^k)^\perp = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} T_j^k
\end{align}

Therefore

$$T_j^k = \frac{1}{6} \left( -\frac{(\varphi_{j+2} - \varphi_j)}{d_2} + \frac{4(\varphi_{j+1} - \varphi_j)}{d_1} - \frac{4(\varphi_{j-1} - \varphi_j)}{d_{-1}} + \frac{(\varphi_{j-2} - \varphi_j)}{d_{-2}} \right).$$
To determine the numerical error propagation in (31), let \( \varphi_{s,s}^{N_1-1} \) be an exact solution of system (P1), (P2) and (P3), then all points \( \varphi_{s,s} \) must satisfy (31). If another initial representation of the same curve evolves, the final positions in both sets may represent different curves, because the approximation errors in each set propagate differently. The core in the following analysis is to compare the error amplification when the relative distances \(|\varphi_{s+1}^k - \varphi_s^k|\) change, and this is crucial for the stability as the evolving curves shrink along the flow.

Let \( \{\varphi_{*,s}\}_{s=0}^{N_1-1} \) be the exact solution, and let \( \{\varphi_s\}_{s=0}^{N_2-1} \) be another discrete representation of the same evolving curve with uniformly distributed points. Suppose that both sets have one point \( \varphi_j \) in common. The difference between these sets is the number of points, this condition prevents us to compare the coordinates point-wise, except for the common point \( \varphi_j \) and when the same time has elapsed.

As \( \varphi_s \) is an exact solution of the system (P1), (P2) and (P3), then it must satisfy (31). At the time \((k+1)\Delta t\), the coordinates for the common point \( \varphi_j \) will differ by a small error \( \varepsilon_j^k \)

\[
\varphi_j^{k+1} = \varphi_{s,j}^{k+1} + \varepsilon_j^{k+1},
\]

denoting \( \varepsilon(\varphi_j^k) \) by \( \varepsilon_j^k \), we point out that \( \varepsilon_j^k \neq 0 \) except for \( s = j \). We will get an equation for \( \varepsilon_j^{k+1} \) in terms of the error in the additional parameters:

\[
a_j^k = a_{s,j}^k + a_{\varepsilon,j}^k \quad \quad v_j^k = v_{s,j}^k + v_{\varepsilon,j}^k \quad \quad n_j^k = n_{s,j}^k + n_{\varepsilon,j}^k
\]

and so on for \( \tau_i \) and \( T_i^k \) in \( \varphi_j^k \). Notice that, even when \( \varphi_j^k = \varphi_{s,j}^k \), and therefore \( \varepsilon_j^k = 0 \), the previous parameters depend non linearly on other points. Thus, the coefficients which drive the evolution \( (a_j^k \text{ and } v_j^k) \) are different for \( \{\varphi_s\}_{s=0}^{N_2-1} \) and \( \{\varphi_{*,s}\}_{s=0}^{N_1-1} \).

Since both sets of points are uniformly distributed at time \( k\Delta t \), then \( a_j^k = a_{s,j}^k = 0 \). We can now rewrite Equation (31)

\[
\varphi_{*,j}^{k+1} + \varepsilon_{*,j}^{k+1} = \varphi_{*,j}^k + \varepsilon_j^k + \Delta t \left( (a_{*,j}^k + a_{\varepsilon,j}^k)(T_{*,j}^k + T_{\varepsilon,j}^k) + (v_{*,j}^k + v_{\varepsilon,j}^k)(n_{*,j}^k + n_{\varepsilon,j}^k) \right).
\]

**Remark.** We cancel the tangential terms because the points are uniformly distributed at time \( k \), but in further iterations, for example \( \varphi_j^{k+2} \), this is not possible in general. Consequently, this analysis determines whether choosing a different number of points is appropriate to reduce the error amplification only for the next time step. If we do not assume this uniform tangential distribution, the resulting equation will be complicated to analyze, even when (31) is linearized, because it would then depend on the projection of \( \varepsilon_s \) along the segments \( \varphi_{*,s}; \varphi_{*,s+1} \) for all \( s = 0,\cdots, N-1 \).

Recall that \( \varphi_{*,j}^k \) is an exact solution, so the last equation implies:

\[
\varepsilon_{*,j}^{k+1} = \Delta t \left( v_{*,j}^k n_{*,j}^k + v_{*,j}^k n_{*,j}^k + v_{*,j}^k n_{*,j}^k \right).
\]
To obtain an explicit expression for (35), we need the following computations:

\[
T_{\varepsilon j}^{k} = T_{j}^{k} - T_{*j}^{k}
\]

\[
= \frac{1}{6} \left( -\frac{(\varphi_{*j+2} - \varphi_{*j})}{d_{2}} + \frac{4(\varphi_{*j+1} - \varphi_{*j})}{d_{1}} - \frac{4(\varphi_{*j-1} - \varphi_{*j})}{d_{-1}} \right)
\]

\[
+ \frac{1}{6} \left( \frac{(\varphi_{*j+2} - \varphi_{*j})}{d_{2}} - \frac{\varepsilon_{j+2} + 4\varepsilon_{j+1} - 4\varepsilon_{j-1} + \varepsilon_{-2}}{d_{1}} \right)
\]

\[
- \frac{1}{6} \left( \frac{(\varphi_{*j+2} - \varphi_{*j})}{d_{2}} + \frac{4(\varphi_{*j+1} - \varphi_{*j})}{d_{1}} - \frac{4(\varphi_{*j-1} - \varphi_{*j})}{d_{-1}} + \frac{(\varphi_{*j-2} - \varphi_{*j})}{d_{-2}} \right).
\]

Assuming that the difference between \(N_{1}\) and \(N_{2}\) is small, we linearize \(1/d_{i}\). Then,

\[
T_{\varepsilon j}^{k} = \frac{1}{6} \left( \frac{(\varphi_{*j+2} - \varphi_{*j})}{d_{2}} \cdot \varepsilon_{j+2} + \frac{(\varphi_{*j+1} - \varphi_{*j})}{d_{1}} \cdot \varepsilon_{j+1} - \frac{(\varphi_{*j-1} - \varphi_{*j})}{d_{-1}} \cdot \varepsilon_{j-1} + \frac{(\varphi_{*j-2} - \varphi_{*j})}{d_{-2}} \cdot \varepsilon_{-2} \right)
\]

To simplify a future computation, we write the vectors \(\varphi_{*j+i} - \varphi_{*j}\) in terms of \(\varepsilon_{j+i}\) and \(\varepsilon_{j+i}^{\perp}\):

\[
T_{\varepsilon j}^{k} = C_{2} \varepsilon_{2} + C_{1} \varepsilon_{1} + C_{-1} \varepsilon_{-1} + C_{-2} \varepsilon_{-2}
\]

\[
+ D_{2} \varepsilon_{2} - D_{1} \varepsilon_{1} - D_{-1} \varepsilon_{-1}^{\perp} + D_{-2} \varepsilon_{-2}^{\perp},
\]

here

\[
C_{\pm 2} = \frac{\text{sign}(\pm 2)}{6} \left( \frac{(\varphi_{*j+2} - \varphi_{*j}, \varepsilon_{j+2})^{2}}{|\varphi_{*j+2} - \varphi_{*j}|^{3}|\varepsilon_{j+2}|^{2}} - \frac{1}{|\varphi_{j+2} - \varphi_{j}|} \right)
\]

\[
C_{\pm 1} = \frac{\text{sign}(\pm 1)}{6} \left( \frac{(\varphi_{*j+1} - \varphi_{*j}, \varepsilon_{j+1})^{2}}{|\varphi_{*j+1} - \varphi_{*j}|^{3}|\varepsilon_{j+1}|^{2}} + \frac{4}{|\varphi_{j+1} - \varphi_{j}|} \right)
\]

\[
D_{\pm i} = \frac{\text{sign}(\pm i)}{6} \left( \frac{(\varphi_{*j+i} - \varphi_{*j}, \varepsilon_{j+i})}{|\varphi_{*j+i} - \varphi_{*j}|^{3}|\varepsilon_{j+i}|^{2}} \right).
\]

Additionally, we write \(n_{\varepsilon j}^{k}\) in terms of \(\varepsilon_{j+i}\) and \(\varepsilon_{j+i}^{\perp}\):

\[
n_{\varepsilon j}^{k} = A_{2} \varepsilon_{j+2} + 4A_{1} \varepsilon_{j+1} + 4A_{-1} \varepsilon_{j-1} + A_{-2} \varepsilon_{j-2}
\]

\[
+ B_{2} \varepsilon_{j+2} + 4B_{1} \varepsilon_{j+1} + 4B_{-1} \varepsilon_{j-1} + B_{-2} \varepsilon_{j-2}.
\]

\[
A_{\pm i} = \frac{\text{sign}(\pm i)}{6} \left( \frac{(\varphi_{*j+i} - \varphi_{*j}, \varepsilon_{j+i})}{|\varphi_{*j+i} - \varphi_{*j}|^{3}|\varepsilon_{j+i}|^{2}} \right)
\]

\[
B_{\pm i} = \frac{\text{sign}(\pm i)}{6} \left( \frac{(\varphi_{*j+i} - \varphi_{*j}, \varepsilon_{j+i}^{\perp})}{|\varphi_{*j+i} - \varphi_{*j}|^{3}|\varepsilon_{j+i}^{\perp}|^{2}} \right)
\]

Later on, we will bound these constants. Let \(J = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}\) and \(I\) be the identity matrix, define

\[
\alpha_{\pm 2} = v_{\varepsilon j} A_{\pm 2} + (v_{*j} + v_{\varepsilon j}) D_{\pm 2}
\]

\[
\beta_{\pm 2} = v_{\varepsilon j} B_{\pm 2} + (v_{*j} + v_{\varepsilon j}) C_{\pm 2}
\]

\[
\alpha_{\pm 1} = 4A_{\pm 1} v_{\varepsilon j} - \text{sign}(\pm 1) (v_{*j} + v_{\varepsilon j}) D_{\pm 1}
\]

\[
\beta_{\pm 1} = 4B_{\pm 1} v_{\varepsilon j} + (v_{*j} + v_{\varepsilon j}) C_{\pm 1},
\]

we can now rewrite Equation 35

\[
\varepsilon_{j+1} = \Delta t \left( v_{\varepsilon j}^{k} n_{\varepsilon j}^{k} + v_{*j}^{k} n_{*j}^{k} + v_{*j}^{k} n_{*j}^{k} \right)
\]

\[
(36) = \Delta t \left( (\alpha_{2} I + \beta_{2} J) \varepsilon_{j+2} + (\alpha_{1} I + \beta_{1} J) \varepsilon_{j+1} + (\alpha_{-1} I + \beta_{-1} J) \varepsilon_{j-1} + (\alpha_{-2} I + \beta_{-2} J) \varepsilon_{j-2} \right).
\]
As the curve is closed, we can assume that \( \varepsilon \) is periodic of period \( L \) over the curve. Von Neumann’s theory implements Fourier analysis to determine whether the error \( \varepsilon_j \) remains bounded as time elapses. The discrete representation of the curve requires us to consider the discrete Fourier transformation of \( \varepsilon \) in the spatial coordinate and with periodic conditions. From Fourier analysis, \( \varepsilon \) can be written using the inverse Fourier transformation as

\[
\varepsilon(\varphi^k_j) = \frac{1}{N_2} \sum_{m=0}^{N_2-1} e^{i \frac{2\pi}{N_2} jm} b_m(k),
\]

\( b_m(k) = \sum_{m=0}^{N_2-1} e^{-i \frac{2\pi}{N_2} jm} \varepsilon(\varphi^k_j). \)

Thus

\[
\varepsilon_j^{k+1} = \frac{1}{N_2} \sum_{m=0}^{N_2-1} e^{i \frac{2\pi}{N_2} jm} \varepsilon_j(k+1) = \frac{1}{N_2} \sum_{m=0}^{N_2-1} \left( \begin{array}{c} \zeta \\ \eta \end{array} \right) e^{i \frac{2\pi}{N_2} jm} b_m(k).
\]

After substituting into (36) we find

\[
\frac{1}{N_2} \sum_{m=0}^{N_2-1} e^{i \frac{2\pi}{N_2} jm} b_m(k + 1) \quad = \quad \frac{1}{N_2} \sum_{m=0}^{N_2-1} \left( \begin{array}{c} \zeta \\ \eta \end{array} \right) e^{i \theta} \frac{2\pi}{N_2} jm b_m(k).
\]

Here

\[
\zeta = \Delta t \left( e^{2i \frac{2\pi}{N_2} m \alpha_2} + e^{i \frac{2\pi}{N_2} m \alpha_1} + e^{-i \frac{2\pi}{N_2} m \alpha_{-1}} + e^{-2i \frac{2\pi}{N_2} m \alpha_{-2}} \right)
\]

\[
\eta = \Delta t \left( e^{2i \frac{2\pi}{N_2} m \beta_2} + e^{i \frac{2\pi}{N_2} m \beta_1} + e^{-i \frac{2\pi}{N_2} m \beta_{-1}} + e^{-2i \frac{2\pi}{N_2} m \beta_{-2}} \right).
\]

Equation (38) defines a formula from which we deduce \( b_m(k) \) in terms of \( b_m(k - 1) \). However, this formula is not recursive because the matrix entries depend on time through \( \alpha_i \) and \( \beta_i \). Let \( M \) be given by

\[
M := \left( \begin{array}{cc} \zeta & -\eta \\ \eta & \zeta \end{array} \right)
\]

\[
b_m(k) = M b_m(k - 1).
\]

Then, using von Neumann’s analysis we can only determine how much does the error is being amplified at the next time step. To proceed with this analysis, we need for \( \varepsilon \) to remain bounded in the consecutive step. Therefore, we will compute the eigenvalues \( \lambda_1 \) and \( \lambda_2 \) of \( M \). Letting \( w_1 \) and \( w_2 \) be the associated eigenvectors, then \( \varepsilon \) can be written in terms of \( w_1 \) and \( w_2 \). Consequently, we will guarantee that \( \varepsilon \) does not grow by finding the constraints on \( N_2 \), such that the eigenvalues are bounded absolutely by one.

Let \( \theta := \frac{2\pi}{N_2} m \), and let \( \lambda_1 \), \( \lambda_2 \) be the eigenvalues of \( M \). Then, the eigenvalue moduli are

\[
|\lambda_1|^2 = \Delta t^2 \sum_{i=-2}^{2} \left( \cos(i \theta) \alpha_i - \sin(i \theta) \beta_i \right)^2 + \Delta t^2 \sum_{i=-2}^{2} \left( \sin(i \theta) \alpha_i + \cos(i \theta) \beta_i \right)^2
\]

\[
|\lambda_2|^2 = \Delta t^2 \sum_{i=-2}^{2} \left( \cos(i \theta) \alpha_i + \sin(i \theta) \beta_i \right)^2 + \Delta t^2 \sum_{i=-2}^{2} \left( \sin(i \theta) \alpha_i - \cos(i \theta) \beta_i \right)^2
\]

From now, we will only consider the amplification of the worse possible error. This error amplification can be obtained by considering the maximum of \( |\varepsilon_i| \) and replacing the values of
sine and cosine above by 1 or -1, in that order.

As the points lie equidistantly, and do not overlap we find:

\[
|\varphi_{j+1} - \varphi_j| = |\varphi_{j-1} - \varphi_j| = \frac{L}{N_1} \quad \quad \quad |\varphi_{j+2} - \varphi_j| = |\varphi_{j-2} - \varphi_j| = \frac{L}{N_2}
\]

In addition,

\[
L \left| \frac{1}{N_1} - \frac{1}{N_2} \right| = \left| \varphi_{j+1} - \varphi_j \right| - \left| \varphi_{j+2} - \varphi_j \right| \leq |\varphi_{j+1} - \varphi_j + \varphi_{j+2} - \varphi_j| = |\varepsilon_{j+1} - \varepsilon_{j+2}|
\]

We consider the following bounds using \( \langle \frac{\varphi_{j+1} - \varphi_j}{|\varphi_{j+1} - \varphi_j|}, \varepsilon_{j+1} \rangle \leq |\varepsilon_{j+1}| \) and \( \langle \frac{\varphi_{j+2} - \varphi_j}{|\varphi_{j+2} - \varphi_j|}, \varepsilon_{j+2} \rangle \leq |\varepsilon_{j+2}| \)

\[
|A_i| \leq \frac{1}{6 |\varepsilon_{j+i}|} \quad \quad \quad |B_i| \leq \frac{1}{6 |\varepsilon_{j+i}|} \quad \quad \quad |D_i| \leq \frac{1}{6 |\varphi_{j+i} - \varphi_j|}
\]

Recalling (39) and (40), we now focus on a bound for \( \sum_{i=-2}^{2} \alpha_i + \beta_i \).

\[
\alpha_1 + \alpha_{-1} + \beta_1 + \beta_{-1} = 4v_{\varepsilon_{1}}(A_1 + A_{-1} + B_1 + B_{-1}) + (v_{j+1} + v_{j-1})(C_1 + C_{-1} + D_1 - D_{-1}) 
\leq \frac{8 |v_{\varepsilon_{1}}|}{3 \min(|\varepsilon_{j+i}|)} + \frac{|v_{j+1} + v_{j-1}|}{3L} (\max\{N_1, 4N_2\} + N_2)
\]

\[
\alpha_2 + \alpha_{-2} + \beta_2 + \beta_{-2} = v_{\varepsilon_{2}}(A_2 + A_{-2} + B_2 + B_{-2}) + (v_{j+2} + v_{j-2})(C_2 + C_{-2} + D_2 - D_{-2}) 
\leq \frac{|v_{\varepsilon_{2}}|}{3 \min(|\varepsilon_{j+i}|)} + \frac{|v_{j+2} + v_{j-2}|}{3L} (\max\{N_1, N_2\} + 2N_2)
\]

Assuming that the size of errors \(|\varepsilon_{j+i}|\) is almost the same for \( i = -2, -1, 1, 2 \). Then, the norm of the eigenvalues in (39) and (40) can be bounded by

\[
|\lambda|^2 \leq 2 \Delta t^2 \left( \frac{9 v_{\varepsilon_{1}}}{3 \min(|\varepsilon_{j+i}|)} + \frac{|v_{j+1} + v_{j-1}|}{3L} (2 \max\{N_1, 4N_2\} + 3N_2) \right)^2.
\]

**Remark.** The number of points that represents the curve determines the integral discretization, because we split the integral over \( \partial \Omega \) into \( N \) integrals over the arcs \( \varphi_s, \varphi_{s+1} \). However,
all these integrals are computed by an approximation formula, and therefore this error propagates to $v_j$, we called this error $v_\varepsilon$. Some well-known accurate formulas can be used to bound this integration error, in all numerical experiments we used the 3-points Gaussian Quadrature Formula. The accuracy of these formulas increase with the number of interpolated points. To proceed with the analysis, we will disregard $v_\varepsilon j$ in (41).

**Proposition 13.** Let $\{\varphi_{s}^*\}_{s=0}^{N_1-1}$ and $\{\varphi_{s}\}_{s=0}^{N_2-1}$ be two discrete representations of the same curve evolving by $P_1$, $P_2$ and $P_3$. Then the representation with the least number of points will produce lower error amplification.

**Proof.** Using Equation 41 and dealing with $v_\varepsilon j$ according to the last remark:

$$|\lambda| \leq 2\Delta t^2 \left( \frac{|v_{s,j}|}{3L} (2 \max\{N_1, 4N_2\} + 3N_2) \right)^2.$$ 

We are using Kimura’s uniform tangential redistribution scheme, so we are constrained to set $\Delta t = 1/N_2^2$. Therefore,

$$|\lambda| \leq 2 \frac{1}{N_1^4} \left( \frac{|v_{s,j}|}{3L} (2 \max\{N_1, 4N_2\} + 3N_2) \right)^2$$

$$= \frac{2}{9L^2} \left( \frac{4 \max\{N_1, 4N_2\} N_1^4}{N_1^4} + 6 \frac{N_2 \max\{N_1, 4N_2\}}{N_1^4} + 3N_2^2 \right).$$

Noticing that this norm will be minimized if $N_2 < N_1$, we conclude the proof. □

**Remark.** Proposition 13 only assumes two different representations of the same curve. Nevertheless, the use of the bound found in (41) assumes that $N_1$ and $N_2$ do not differ overly because we linearized the inverse of relative distances to get the bounds. Additionally, we suppose that both representations are suitable in the sense that the sum over the polygonal distances approximates the curve length, because the numerical integration depends on this hypothesis in order to handle $v_\varepsilon$.

6. CONCLUSIONS

We presented a novel anisotropic mean curvature geometric flow together with an implementation of it through a numerical scheme, applied to contour recognition tasks. Our proposal has the following features: (1) The curve evolution is given by an explicit scheme. Consequently, the required resolution for the contour recognition can used to choose the size of the stored data array. (2) The stability of our scheme can only be checked at each iteration using the explicit values of $v_{s,j} L$, $N_1$ and $N_2$, according to Proposition 13. (3) We provided specific criteria to improve the conditioning and to verify the stability of this method at each time step. Moreover, the implementation was optimized to run in parallel, and the code has been made publicly available.

Future improvements for this method may include increasing of the number of point charges that drive the curve evolution, and inferring optimal distributions and values for the potentials. Due to the unavailability of implementations—for example accessible in code repositories—of previous methods, we defer comprehensive comparisons with other schemes of numerical mean curvature flows [24, 25, 1].
REFERENCES

1. Martin Balazovjech and Karol Mikula. A higher order scheme for a tangentially stabilized plane curve shortening flow with a driving force. *SIAM Journal on Scientific Computing*, 33(5):2277–2294, 2011.
2. Kenneth A. Brakke. *The Motion of a Surface by Its Mean Curvature. (MN-20)*. Princeton University Press, 1978.
3. Martin Burger, Yanina Landa, Nicolay M. Tanushev, and Richard Tsai. *Algorithmic Foundation of Robotics VIII: Selected Contributions of the Eight International Workshop on the Algorithmic Foundations of Robotics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
4. Bing-Long Chen. Uniqueness and pseudolocality theorems of the mean curvature flow. *Communications in Analysis and Geometry*, 15(3):435–490, 2007.
5. David L. Chopp and James A. Sethian. Flow under curvature: singularity formation, minimal surfaces, and geodesics. *Experiment. Math.*, 2(4):235–255, 1993.
6. B. Chow, S.C. Chu, D. Glickenstein, C. Guenther, J. Isenberg, T. Ivey, D. Knopf, P. Lu, F. Luo, and L. Ni. *The Ricci Flow: Techniques and Applications*. Mathematical Surveys and Monographs. American Mathematical Society, 2015.
7. K. Ecker. *Regularity Theory for Mean Curvature Flow*. Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser Boston, 2012.
8. L.C. Evans. *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society, 2010.
9. G.B. Folland. *Introduction to Partial Differential Equations*. Princeton University Press, 1995.
10. Huisken Gerhard and Polden Alexander. *Calculus of Variations and Geometric Evolution Problems: Lectures given at the 2nd Session of the Centro Internazionale Matematico Estivo (C.I.M.E.) held in Cetraro, Italy, June 15–22, 1996*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
11. P. Hájek and M. Johanis. *Smooth analysis in Banach spaces*. De Gruyter Series in Nonlinear Analysis and Applications. De Gruyter, 2014.
12. Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.
13. Masato Kimura. Numerical analysis of moving boundary problems using the boundary tracking method. *Japan Journal of Industrial and Applied Mathematics*, 14(3):373–398.
14. S.G. Krantz. *Geometric Function Theory: Explorations in Complex Analysis*. Cornerstones. Birkhäuser Boston, 2007.
15. R.J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. Society for Industrial and Applied Mathematics, 2007.
16. Leevan Ling, Y. C. Hon, and M. Yamamoto. Inverse source identification for poisson equation. *Inverse Problems in Science and Engineering*, 13(4):433–447, 2005.
17. R. Malladi and J.A. Sethian. Image processing: Flows under min/max curvature and mean curvature. *Graphical Models and Image Processing*, 58(2):127 – 141, 1996.
18. C. Mantegazza. *Lecture Notes on Mean Curvature Flow*. Progress in Mathematics. Springer Basel, 2011.
19. W. W. Mullins. Two-dimensional motion of idealized grain boundaries. *Journal of Applied Physics*, 27(8):900–904, 1956.
20. Takashi Ohe and Kohzaburo Ohnaka. Boundary element approach for an inverse source problem of the poisson equation with a one-point-mass like source. *Applied Mathematical Modelling*, 18(4):216 – 223, 1994.
21. S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Applied Mathematical Sciences. Springer, 2003.
22. R.H. Pletcher, J.C. Tannehill, and D. Anderson. *Computational Fluid Mechanics and Heat Transfer, Third Edition*. Series in Computational and Physical Processes in Mechanics and Thermal Sciences. CRC Press, 2016.
23. S.A. Sauter and C. Schwab. *Boundary Element Methods*. Springer Series in Computational Mathematics. Springer Berlin Heidelberg, 2010.
24. J.A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 1999.
25. Daniel Ševčovič and Shigetoshi Yazaki. Evolution of plane curves with a curvature adjusted tangential velocity. *Japan Journal of Industrial and Applied Mathematics*, 28(3):413, 2011.
26. E. Pedersen T. Colding, W. Minicozzi. Mean curvature flow. *Bull. Amer. Math. Soc.*, 52:297 – 333, 2015.
27. Z. Wu, J. Yin, and C. Wang. *Elliptic & Parabolic Equations*. World Scientific, 2006.
28. E. Zeidler. *Nonlinear Functional Analysis and Its Applications: Fixed point theorems*. Nonlinear Functional Analysis and Its Applications. Springer-Verlag, 1985.

29. E. Zeidler and L.F. Boron. *Nonlinear Functional Analysis and Its Applications: II/ A: Linear Monotone Operators*. Monotone operators / transl. by the author and by Leo F. Boron. Springer New York, 1989.

30. X.P. Zhu. *Lectures on Mean Curvature Flows*. AMS/IP studies in advanced mathematics. American Mathematical Soc.

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF CALIFORNIA, SANTA BARBARA, ON LEAVE FROM, INSTITUTO DE MATEMÁTICAS, INSTITUTO DE MATEMÁTICAS, UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO, MEXICO CITY

E-mail address: pablo@im.unam.mx

INSTITUTO DE MATEMÁTICAS, UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO, MEXICO CITY