

A Simple, Fast and Stabilized Flowing Finite Volume Method for Solving General Curve Evolution Equations

Karol Mikula^{1,*}, Daniel Ševčovič² and Martin Balažovjeh¹

¹ Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava, Slovak Republic.

² Department of Applied Mathematics and Statistics, Comenius University, 842 48 Bratislava, Slovak Republic.

Received 25 October 2008; Accepted (in revised version) 29 May 2009

Communicated by Bo Li

Available online 29 July 2009

Abstract. A new simple Lagrangian method with favorable stability and efficiency properties for computing general plane curve evolutions is presented. The method is based on the flowing finite volume discretization of the intrinsic partial differential equation for updating the position vector of evolving family of plane curves. A curve can be evolved in the normal direction by a combination of fourth order terms related to the intrinsic Laplacian of the curvature, second order terms related to the curvature, first order terms related to anisotropy and by a given external velocity field. The evolution is numerically stabilized by an asymptotically uniform tangential redistribution of grid points yielding the first order intrinsic advective terms in the governing system of equations. By using a semi-implicit in time discretization it can be numerically approximated by a solution to linear penta-diagonal systems of equations (in presence of the fourth order terms) or tri-diagonal systems (in the case of the second order terms). Various numerical experiments of plane curve evolutions, including, in particular, nonlinear, anisotropic and regularized backward curvature flows, surface diffusion and Willmore flows, are presented and discussed.

AMS subject classifications: 35K65, 65N40, 53C80, 35K55, 53C44, 65M60

Key words: Geometric partial differential equations, evolving plane curves, mean curvature flow, anisotropy, Willmore flow, surface diffusion, finite volume method, semi-implicit scheme, tangential redistribution.

*Corresponding author. *Email addresses:* mikula@math.sk (K. Mikula), sevcovic@fmph.uniba.sk (D. Ševčovič), balazoviech@math.sk (M. Balažovjeh)

1 Introduction

The main purpose of this paper is to propose a simple, fast and stable Lagrangian method for computing evolution of closed smooth embedded plane curves driven by a normal velocity of the form $\beta(\partial_s^2 k, k, \nu, x)$ which may depend on the intrinsic Laplacian $\partial_s^2 k$ of the curvature k , on the curvature k itself, on the tangential angle ν and the curve position vector x . We shall restrict our attention to the following form of the normal velocity:

$$\beta = -\delta \partial_s^2 k + b(k, \nu) + F(x). \quad (1.1)$$

Here $\delta \geq 0$ is a constant and $b = b(k, \nu)$ is a smooth function satisfying $b(0, \cdot) = 0$. If $\delta > 0$ then there is no constraint on the monotonicity of b with respect to k . On the other hand, if $\delta = 0$, then we shall assume the function b is strictly increasing with respect to the curvature.

There are many interesting evolutionary models in various applied fields of science, technology and engineering that can be described by geometric Eq. (1.1). For example, putting $\delta = 0$ we obtain the normal velocity $\beta = b(k, \nu) + F(x)$ representing the well-known anisotropic mean curvature flow arising in the motion of material interfaces during solidification and in affine invariant shape analysis (see, e.g., [1, 9, 10, 15, 16, 22, 23, 32]). The term $F(x)$ represents an external driving force, like e.g., a given velocity field projected to the normal vector of a curve, or any other constant or scalar function depending on the current curve position. It drives the curve in the inner (if $F(x) > 0$) or outer (if $F(x) < 0$) normal direction. In the case $\delta = 1$ two well-known examples arise when studying a motion of the so-called elastic curves. It is the surface diffusion, in the case $b = 0$, and the Willmore flow where $b = -\frac{1}{2}k^3$. The surface diffusion is often used in computational fluid dynamics and material sciences, where the encompassing area of interface should be preserved. The case when $b = -\frac{1}{2}k^3$ arises from the model of the Euler-Bernoulli elastic rod — an important problem in structural mechanics [6, 11, 12]. The evolutionary models having the normal velocity of the form (1.1) are often adopted in image segmentation where elastic and geodesic curves are used in order to find image objects in an automatic way [7, 19, 20, 27, 29]. By our method we are able to handle a regularized backward mean curvature flow in which b is a decreasing function of the curvature k like e.g. $b(k, \nu) = -k$. We regularize the backward mean curvature flow by adding a small fourth order diffusion term $0 < \delta \ll 1$. To our knowledge, first experiments of this kind are presented in this paper.

The main idea of our approach is based on accompanying the geometric equation (1.1) by a stabilizing tangential velocity and in rewriting it into a form of intrinsic an partial differential equation (PDE) for the curve position vector. The resulting PDE contains fourth, second and first order spatial differential terms that are approximated by means of the flowing finite volume method [25]. For time discretization we follow semi-implicit approach leading to a solution to a linear system of equations at each time level. That can be done efficiently, and, due to tangential stabilization, we hope that this direct Lagrangian

method can be considered as an efficient counterpart to the well-known level-set based methods for description of the curve evolution discussed in [13, 14, 17, 30, 31, 33].

The paper is organized as follows: in the next section we derive the intrinsic PDE for description of a family of plane curves. Then we present our numerical approximation scheme. Finally, we discuss various numerical experiments showing applicability of our approach.

2 Governing equations

An immersed regular plane curve Γ can be parameterized by a smooth function $x: S^1 \rightarrow \mathbb{R}^2$, i.e., $\Gamma = \{x(u), u \in S^1\}$ having a strictly positive local length term $g = |\partial_u x| > 0$. Taking into account periodic boundary conditions at $u = 0, 1$ we can identify the circle S^1 with the interval $[0, 1]$. The unit arc-length parameterization is denoted by s . Clearly, $ds = g du$. For the tangent vector we have $\vec{T} = \partial_s x$ and we can choose the unit inward normal vector \vec{N} such that $\det(\vec{T}, \vec{N}) = 1$. The tangential angle $\nu = \arg(\vec{T})$, $\vec{T} = (\cos \nu, \sin \nu)^\top$ and $\vec{N} = (-\sin \nu, \cos \nu)^\top$.

Let a regular smooth initial curve $\Gamma^0 = \text{Image}(x^0) = \{x^0(u), u \in [0, 1]\}$ be given. We shall represent a family of plane curves $\Gamma^t = \text{Image}(x(\cdot, t)) = \{x(u, t), u \in [0, 1]\}$ that evolves according to the geometric Eq. (1.1) by its position vector x satisfying the following equation:

$$\partial_t x = \beta \vec{N} + \alpha \vec{T}. \quad (2.1)$$

It is well-known that the presence of any tangential velocity functional α in (2.1) has no impact on the shape of evolving curves. However, it may help to redistribute points along the curve. As a consequence, it can significantly stabilize numerical computations. The reader is referred to papers [2, 3, 5, 8, 18, 21, 24–28, 34, 35] for detailed discussion on how a suitable tangential stabilization can prevent a numerical solution from forming various undesired singularities. We will specify our choice of a tangential velocity α later. Since

$$\begin{aligned} \partial_s^4 x &= \partial_s^3 \vec{T} = \partial_s^2 (k \vec{N}) = \partial_s^2 k \vec{N} + 2 \partial_s k \partial_s \vec{N} + k \partial_s^2 \vec{N} \\ &= \partial_s^2 k \vec{N} - 2(\partial_s k) k \vec{T} - k \partial_s (k \vec{T}) \\ &= \partial_s^2 k \vec{N} - 3k(\partial_s k) \vec{T} - k^2 \partial_s \vec{T} \\ &= \partial_s^2 k \vec{N} - \frac{3}{2} \partial_s (k^2) \partial_s x - k^2 \partial_s^2 x, \end{aligned}$$

we have

$$(-\partial_s^2 k) \vec{N} = -\partial_s^4 x - k^2 \partial_s^2 x - \frac{3}{2} \partial_s (k^2) \partial_s x. \quad (2.2)$$

Let us define the following auxiliary functions:

$$\phi(k, \nu) = -\delta k^2 + c(k, \nu), \quad c(k, \nu) = b(k, \nu)/k, \quad (2.3)$$

$$v(k, \nu) = \frac{3}{2} \delta \partial_s (k^2) + \partial_s \phi(k, \nu) - \alpha. \quad (2.4)$$

Since the function b is assumed to be smooth and $b(0, \nu) = 0$ the function $c(k, \nu) = b(k, \nu) / k$ is smooth as well. Using the Frenet formulae we obtain

$$b(k, \nu) \vec{N} = c(k, \nu) \partial_s^2 x, \quad \phi(k, \nu) \partial_s^2 x = \partial_s(\phi(k, \nu) \partial_s x) - \partial_s \phi(k, \nu) \partial_s x.$$

Hence, for the normal velocity β of the form (1.1) we end up with the following higher-order intrinsic PDE for the position vector $x = x(s, t)$:

$$\partial_t x + v(k, \nu) \partial_s x = \delta(-\partial_s^4 x) + \partial_s(\phi(k, \nu) \partial_s x) + F(x) \vec{N}(v). \quad (2.5)$$

Numerical approximation of a solution x to the above PDE forms the basis of our direct Lagrangian approach.

It is known (see, e.g., [25]) that a family of plane curves $\Gamma^t = \text{Image}(x(\cdot, t))$, $t \in [0, T)$, that evolves according to (2.1) can be also represented by a solution to the following system of intrinsic parabolic-ordinary differential equations:

$$\partial_t k = \partial_s^2 \beta + \alpha \partial_s k + k^2 \beta, \quad (2.6)$$

$$\partial_t v = \partial_s \beta + \alpha k, \quad (2.7)$$

$$\partial_t g = -gk\beta + g\partial_s \alpha. \quad (2.8)$$

In [26–28] the system (2.6)-(2.8) was solved numerically for the curvature k , tangent angle v and the local length g . Knowing these quantities one can reconstruct the curve evolution. Asymptotically uniform redistributions for the second order flows with driving force (see [26]) and for the fourth order flows (see [28]) were also proposed.

In this paper we follow a different approach when compared to [26, 28]. It is much faster and simpler from computational point of view. We do not solve the system (2.6)-(2.8), but the position vector Eq. (2.5) is directly numerically discretized. On the other hand, from the analytical point of view, the system (2.6)-(2.8) describes evolution of useful geometric quantities that can be utilized in designing proper tangential velocities for stabilization of numerical computations. For example, in the case of convex curves, it is sufficient to solve just Eq. (2.6) on the fixed parameter interval given by the range of v . Then no grid point redistribution is necessary [22, 23]. Eq. (2.7) was used recently in designing the tangential velocities corresponding to the so-called crystalline curvature flow, where $\partial_t v = 0$ (cf. [34, 35]). In [26] the authors proposed and analyzed a new type of tangential redistribution referred to as the *asymptotically uniform grid points redistribution*. Here we follow a similar idea but adopt it in a rather different way. More precisely, in order to compute the corresponding tangential velocity α (depending on the curvature, tangential angle and local length) we use information from a solution to (2.5) only, i.e. we do not solve the system (2.6)-(2.8).

We will see that such a simple and straightforward approach yields a stable, fast and precise solution to our problem. The corresponding numerical scheme is efficiently stabilized by an appropriate choice of the tangential velocity term α in (2.4). Let us note that other known tangential velocities, like the one preserving relative local length [18, 21, 25],

locally diffusive redistribution [8, 27, 28], crystalline curvature redistribution [35] or curvature adjusted redistribution [4, 34] can be also incorporated straightforwardly to the numerical scheme by corresponding change of the term α entering Eq. (2.4).

Let us briefly outline the basic idea behind the asymptotically uniform grid points redistribution. Let us denote by

$$L_t = \int_{\Gamma^t} ds = \int_0^1 g(u, t) du$$

the length of an evolving curve Γ^t . Integrating Eq. (2.8) along the curve and taking into account periodicity of α at $u = 0, 1$ we obtain

$$\frac{dL}{dt} + \langle k\beta \rangle_{\Gamma} L = 0, \quad (2.9)$$

where

$$\langle k\beta \rangle_{\Gamma} = \frac{1}{L} \int_{\Gamma} k\beta ds$$

denotes the curve average of the quantity $k\beta$ over a curve Γ .

When designing the tangential velocity α , it is worth to study the time evolution of the quantity g/L . From a numerical point of view it corresponds to the local grid point distances divided by the averaged local distance L/n , where n is the number of grid points. We refer the reader to content of the next section for details. In that sense, it represents a deviation of local grid point distances from being uniformly distributed. Moreover, if we define the quantity $\theta = \ln(g/L)$ and we take into account Eqs. (2.8) and (2.9) we conclude

$$\partial_t \theta + k\beta - \langle k\beta \rangle_{\Gamma} = \partial_s \alpha. \quad (2.10)$$

From (2.10) we can observe that, by an appropriate choice of $\partial_s \alpha$, we can control behavior of θ and, subsequently, of the ratio g/L also. Our choice of α is based on the following particular setup (see [26])

$$\partial_s \alpha = k\beta - \langle k\beta \rangle_{\Gamma} + (e^{-\theta} - 1)\omega(t), \quad (2.11)$$

where $\omega \in L^1_{loc}([0, T_{\max}))$ and T_{\max} is the maximal existence time of evolving curve.

The most simple choice $\omega(t) \equiv 0$ yields $\partial_t \theta = 0$. Hence

$$\frac{g(u, t)}{L_t} = \frac{g(u, 0)}{L_0} \quad \text{for any } u \in S^1, t \in [0, T_{\max}),$$

and we obtain the so-called tangential redistribution preserving the relative local length (cf. [18, 21, 25]). On the other hand, assuming

$$\int_0^{T_{\max}} \omega(\tau) d\tau = +\infty, \quad (2.12)$$

then, by inserting (2.11) into (2.10) and solving the corresponding ordinary differential equation $\partial_t \theta = (e^{-\theta} - 1) \omega(t)$, we obtain $\theta(u, t) \rightarrow 0$ as $t \rightarrow T_{\max}$ and hence

$$\frac{g(u, t)}{L_t} \rightarrow 1 \quad \text{as } t \rightarrow T_{\max} \quad \text{uniformly w.r. to } u \in [0, 1].$$

It means that redistribution of grid points along a curve becomes *asymptotically uniform* as t approaches the maximal time of existence T_{\max} . In the case when the family $\{\Gamma^t, t \in [0, T)\}$ shrinks to a point as $t \rightarrow T_{\max}$, in order to fulfill (2.12), one can choose

$$\omega(t) = \kappa_2 \langle k\beta \rangle_{\Gamma^t},$$

where $\kappa_2 > 0$ is a positive constant. By (2.9) we have

$$\int_0^t \omega(\tau) d\tau = -\kappa_2 \int_0^t \ln L^\tau d\tau = \kappa_2 (\ln L^0 - \ln L_t) \rightarrow +\infty \quad \text{as } t \rightarrow T_{\max}$$

because $\lim_{t \rightarrow T_{\max}} L_t = 0$. On the other hand, if the length L_t is always away from zero and $T_{\max} = +\infty$ one can take $\omega(t) = \kappa_1$, where $\kappa_1 > 0$ is a positive constant. Summarizing, a suitable choice of the tangential velocity functional α that helps to redistribute grid points uniformly along the evolved curve (in any case of shrinking, expanding or reaching an equilibrium curve shape) is given by a solution to the equation

$$\partial_s \alpha = k\beta - \langle k\beta \rangle_\Gamma + (L/g - 1)\omega, \quad \omega = \kappa_1 + \kappa_2 \langle k\beta \rangle_\Gamma, \quad \alpha(0, t) = 0, \quad (2.13)$$

where $\kappa_1, \kappa_2 \geq 0, \kappa_1 + \kappa_2 > 0$, are given constants. In all computations to follow in Section 4, we choose $\kappa_2 = 0$ which is sufficient when the computations end up before an eventual extinction of a curve. By the boundary condition imposed on $\alpha(0, t)$ we prescribe the motion of the point $x(0, t)$ in the normal direction only. Therefore, the tangential velocity term α is determined uniquely.

It is worth to note that the speed of relaxation of the quantity $\theta = \ln(g/L)$ is controlled by the constant $\kappa_1 > 0$. We typically take $\kappa_1 = \mathcal{O}(1)$ in all our computations. More precisely, in experiments presented in section 4 we take $\kappa_1 \in [3, 10]$. This is due to the fact that the speed of the tangential velocity α defined as in (2.13) is increasing with respect to the parameter κ_1 . Therefore choosing considerably larger values of κ_1 would make the governing Eq. (2.5) strongly convective dominant which may yield a necessity of small time steps in our numerical scheme.

3 Numerical approximation scheme

A family of plane curves that evolves according to (1.1) is numerically represented by a family of "flowing" discrete plane points x_i^j where the index $i = 1, \dots, n$, denotes space discretization and the index $j = 0, \dots, m$, stands for a discrete time stepping. Assuming

a uniform division of the time interval $[0, T]$ with a time step $\tau = \frac{T}{m}$ and a uniform division of the fixed parameterization interval $[0, 1]$ with a step $h = 1/n$, a discrete point x_i^j corresponds to $x(ih, j\tau)$. Due to periodic boundary conditions and smoothness of the evolved curve we have used the additional points defined by $x_{-1}^j = x_{n-1}^j, x_0^j = x_n^j, x_{n+1}^j = x_1^j, x_{n+2}^j = x_2^j$. The tangential velocity of a flowing node x_i^j is denoted by α_i^j .

The system of difference equations corresponding to (2.13) and (2.5) will be constructed at each discrete time step $j\tau$ by using the flowing finite volume method proposed in [25]. First we solve (2.13) for the tangential velocities α_i^j and then Eq. (2.5) for the position vectors $x_i^j, i = -1, \dots, n+2$. Remaining quantities involved in (2.13) and (2.5), as the curvature, tangential angle, local and total length of the curve are computed from the curve position vector x^{j-1} from the previous time step $j-1$.

In order to build our numerical scheme we construct the so-called *flowing finite volume* $[x_{i-1}^j, x_i^j]$ and also corresponding flowing dual volumes $[\tilde{x}_i^j, \tilde{x}_{i+1}^j]$ where $\tilde{x}_i^j = (x_{i-1}^j + x_i^j)/2$. Then the approximate local lengths of flowing finite volumes r_i^j , curvatures k_i^j and tangential angles v_i^j are given by piecewise constant values in the flowing finite volumes. Similarly, the other quantities α_i^j, x_i^j and approximate lengths of dual volumes q_i^j are considered as piecewise constant in flowing dual volumes.

Let a discrete representation of the evolving curve be given at time level $j-1$ by discrete points $x_i^{j-1}, i = -1, \dots, n+2$. At every new time level j we compute

$$\begin{aligned}
 r_i^j &= |R_i|, \quad R_i = (R_{i_1}, R_{i_2}) = x_{i-1}^{j-1} - x_{i-1}^j, \quad i = 0, \dots, n+2, \quad q_i^j = \frac{1}{2} (r_i^j + r_{i+1}^j), \\
 k_i^j &= \frac{\text{sgn}(\det(R_{i-1}, R_{i+1}))}{2r_i^j} \arccos\left(\frac{R_{i+1} \cdot R_{i-1}}{r_{i+1}^j r_{i-1}^j}\right), \quad i = 1, \dots, n, \quad k_0^j = k_n^j, \quad k_{n+1}^j = k_1^j, \\
 v_0^j &= \arccos(R_{0_1}/r_0^j), \quad \text{if } R_{0_2} \geq 0, \quad v_0^j = 2\pi - \arccos(R_{0_1}/r_0^j), \quad \text{if } R_{0_2} < 0, \\
 v_i^j &= v_{i-1}^j + r_i^j k_i^j, \quad i = 1, \dots, n, \quad v_{n+1}^j = v_1^j + 2\pi, \\
 \beta_i^j &= \frac{\delta}{r_i^j} \left(\frac{k_i^j - k_{i-1}^j}{q_{i-1}^j} - \frac{k_{i+1}^j - k_i^j}{q_i^j} \right) + b(k_i^j, v_i^j) + \frac{F(x_i^{j-1}) + F(x_{i-1}^{j-1})}{2}, \quad i = 1, \dots, n, \\
 L^j &= \sum_{l=1}^n r_l^j, \quad B^j = \frac{1}{L^j} \sum_{l=1}^n r_l^j k_l^j \beta_l^j.
 \end{aligned}$$

In order to compute the tangential velocity α we integrate (2.13) over the time varying flowing finite volume $[x_{i-1}, x_i]$,

$$\int_{x_{i-1}}^{x_i} \partial_s \alpha \, ds = \int_{x_{i-1}}^{x_i} k\beta - \langle k\beta \rangle_\Gamma + (L/g - 1)\omega \, ds.$$

Hereafter we use the notation $\int_{x_{i-1}}^{x_i} \psi \, ds$ for integral of the quantity ψ over the curve arc

$\widehat{\tilde{x}_{i-1}, \tilde{x}_i}$. Hence at any time level t we have the relation for approximation of the difference $\alpha_i - \alpha_{i-1}$

$$\alpha_i - \alpha_{i-1} \approx r_i(k_i\beta_i - \langle k\beta \rangle_\Gamma) + (hL - r_i)\omega.$$

Taking into account discrete time stepping in the previous relation we obtain the following expression for up-dated values of the tangential velocity at the j -th time level:

$$\alpha_i^j = \alpha_{i-1}^j + r_i^j(k_i^j\beta_i^j - B^j) + (hL^j - r_i^j)\omega, \quad i = 1, \dots, n, \quad \alpha_0^j = 0. \quad (3.1)$$

After computing tangential velocities we update diffusion and advection terms using

$$\begin{aligned} \phi_i^j &= -\delta(k_i^j)^2 + c(k_i^j, v_i^j), & i &= 1, \dots, n+1, \\ v_i^j &= \frac{3\delta(k_{i+1}^j)^2 - (k_i^j)^2}{2q_i^j} + \frac{\phi_{i+1}^j - \phi_i^j}{q_i^j} - \alpha_i^j, & i &= 1, \dots, n. \end{aligned}$$

In order to compute the curve position vector x^j at a new time level we integrate (2.5) over the time varying flowing dual volume $[\tilde{x}_i, \tilde{x}_{i+1}]$,

$$\int_{\tilde{x}_i}^{\tilde{x}_{i+1}} \partial_t x + v(k, \nu) \partial_s x \, ds = \int_{\tilde{x}_i}^{\tilde{x}_{i+1}} -\delta \partial_s^4 x + \partial_s(\phi(k, \nu) \partial_s x) + F(x) \vec{N}(\nu) \, ds.$$

Notice that the advective term $v(k, \nu)$ contains the tangential velocity α and partial derivatives of functions of k and ν . They can be approximated by constant difference quotients in flowing dual volumes. Therefore we can write

$$q_i \frac{dx_i}{dt} + v_i(\tilde{x}_{i+1} - \tilde{x}_i) = [-\delta \partial_s^3 x + \phi(k, \nu) \partial_s x]_{\tilde{x}_i}^{\tilde{x}_{i+1}} + q_i^j F(x_i) \vec{N}((v_i + v_{i+1})/2). \quad (3.2)$$

Denote by $\partial_s^k x_i$ and $\partial_s^k \tilde{x}_i$ the approximation of the k -th arc-length derivative $\partial_s^k x$ at points x_i and \tilde{x}_i , respectively. The differences at points $\tilde{x}_i, \tilde{x}_{i+1}$ in the above boundary integral terms are then naturally approximated by

$$\phi(k, \nu) \partial_s \tilde{x}_{i+1} - \phi(k, \nu) \partial_s \tilde{x}_i \approx \phi_{i+1}^j \frac{x_{i+1}^j - x_i^j}{r_{i+1}^j} - \phi_i^j \frac{x_i^j - x_{i-1}^j}{r_i^j}.$$

The third order terms appearing in the boundary integral in (3.2) can be approximated as

follows:

$$\begin{aligned} \partial_s^3 \tilde{x}_{i+1} - \partial_s^3 \tilde{x}_i &\approx \frac{\partial_s^2 x_{i+1} - \partial_s^2 x_i}{r_{i+1}} - \frac{\partial_s^2 x_i - \partial_s^2 x_{i-1}}{r_i} \\ &\approx \frac{1}{r_{i+1}} \left(\frac{\partial_s \tilde{x}_{i+2} - \partial_s \tilde{x}_{i+1}}{q_{i+1}} - \frac{\partial_s \tilde{x}_{i+1} - \partial_s \tilde{x}_i}{q_i} \right) - \frac{1}{r_i} \left(\frac{\partial_s \tilde{x}_{i+1} - \partial_s \tilde{x}_i}{q_i} - \frac{\partial_s \tilde{x}_i - \partial_s \tilde{x}_{i-1}}{q_{i-1}} \right) \\ &\approx \left(\frac{x_{i+2}^j - x_{i+1}^j}{r_{i+1}^j q_{i+1}^j r_{i+2}^j} - \frac{x_{i+1}^j - x_i^j}{r_{i+1}^j q_{i+1}^j r_{i+1}^j} \right) - \left(\frac{x_{i+1}^j - x_i^j}{r_{i+1}^j q_i^j r_{i+1}^j} - \frac{x_i^j - x_{i-1}^j}{r_{i+1}^j q_i^j r_i^j} \right) \\ &\quad - \left(\frac{x_{i+1}^j - x_i^j}{r_i^j q_i^j r_{i+1}^j} - \frac{x_i^j - x_{i-1}^j}{r_i^j q_i^j r_i^j} \right) + \left(\frac{x_i^j - x_{i-1}^j}{r_i^j q_{i-1}^j r_i^j} - \frac{x_{i-1}^j - x_{i-2}^j}{r_i^j q_{i-1}^j r_{i-1}^j} \right). \end{aligned}$$

The left hand side of (3.2) is approximated by means of backward time differences and by a central finite difference approximation of the advective term (up-wind technique can be also easily incorporated). We obtain

$$q_i \frac{dx_i}{dt} + v_i(\tilde{x}_{i+1} - \tilde{x}_i) \approx q_i^j \frac{x_i^j - x_i^{j-1}}{\tau} + v_i^j \frac{x_{i+1}^j - x_{i-1}^j}{2}.$$

Taking into account information from the previous time step $j-1$ in the driving term $F\vec{N}$, multiplying the third order terms approximation by $-\delta$ and putting all unknowns x_l^j , $l = i-2, \dots, i+2$ with their coefficients to the left hand side, we obtain a linear system of equations for updating the discrete position vector x^j of the evolved curve $\Gamma^{j\tau}$:

$$\mathcal{A}_i^j x_{i-2}^j + \mathcal{B}_i^j x_{i-1}^j + \mathcal{C}_i^j x_i^j + \mathcal{D}_i^j x_{i+1}^j + \mathcal{E}_i^j x_{i+2}^j = \mathcal{F}_i^j$$

for $i=1, \dots, n$, subject to periodic boundary conditions $x_{-1}^j = x_{n-1}^j$, $x_0^j = x_n^j$, $x_{n+1}^j = x_1^j$, $x_{n+2}^j = x_2^j$, where

$$\begin{aligned} \mathcal{A}_i^j &= \frac{\delta}{r_i^j q_{i-1}^j r_{i-1}^j}, \quad \mathcal{E}_i^j = \frac{\delta}{r_{i+1}^j q_{i+1}^j r_{i+2}^j}, \\ \mathcal{B}_i^j &= -\delta \left(\frac{1}{r_i^j q_{i-1}^j r_{i-1}^j} + \frac{1}{(r_i^j)^2 q_{i-1}^j} + \frac{1}{(r_i^j)^2 q_i^j} + \frac{1}{r_i^j q_i^j r_{i+1}^j} \right) - \frac{\phi_i^j}{r_i^j} - \frac{v_i^j}{2}, \\ \mathcal{D}_i^j &= -\delta \left(\frac{1}{r_i^j q_i^j r_{i+1}^j} + \frac{1}{(r_{i+1}^j)^2 q_i^j} + \frac{1}{(r_{i+1}^j)^2 q_{i+1}^j} + \frac{1}{r_{i+1}^j q_{i+1}^j r_{i+2}^j} \right) - \frac{\phi_{i+1}^j}{r_{i+1}^j} + \frac{v_i^j}{2}, \\ \mathcal{C}_i^j &= \frac{q_i^j}{\tau} - (\mathcal{A}_i^j + \mathcal{B}_i^j + \mathcal{D}_i^j + \mathcal{E}_i^j), \\ \mathcal{F}_i^j &= \frac{q_i^j}{\tau} x_i^{j-1} + q_i^j F(x_i^{j-1}) \vec{N} \left(\frac{v_i^j + v_{i+1}^j}{2} \right). \end{aligned}$$

The previous system is penta-diagonal if $\delta > 0$ and tri-diagonal in the case $\delta = 0$. In the latter case when $\delta = 0$, the monotonicity assumption on the function b guarantees the strict diagonal dominance of the tri-diagonal system matrix. In both cases, it is solved efficiently by means of the Gauss-Seidel iterative method or by its well-known successive over relaxation version SOR. We start the iterates from the previous time step vector x^{j-1} and, in practice, when using our asymptotically uniform tangential redistribution (AUTR), there are just few number of SOR iterations needed in order to achieve the solver accuracy goal. The iterative process is stopped when a difference of subsequent iterates in the maximum norm is less than the prescribed tolerance, e.g., $\text{TOL} = 10^{-10}$. Based on our practical experience, it should be also noted that the number of SOR iterations (corresponding to well-conditioning of an iteration matrix) is strongly lowered by such a proper choice of the tangential velocity. Consequently, it speeds up computations significantly making thus our numerical scheme fast and efficient. Computational times are reported in next section. In some examples of evolution of curves with high variation in the curvature or in checking the experimental order of convergence (see, e.g., Fig. 7 or Tables 1 and 2) we typically use small time steps $\tau \approx h^2$ or even $\tau \approx h^4$ (in some nontrivial cases of the fourth order flows). Nevertheless, taking larger time steps τ leads to satisfactory numerical results as it can be seen from other experiments presented in this paper. We observed neither occurrences of accumulation of grid points nor spurious numerical oscillations.

4 Discussion on numerical experiments

In the first numerical experiments we show a stabilizing effect of our scheme due to asymptotically uniform tangential redistribution (AUTR) for the case of selfsimilar affine invariant shrinking evolution of an initial ellipse with half-axes ratio 3:1 (Fig. 1a). When the grid points are moving only in the normal direction, the numerical computation collapses soon because of merging of grid points and spurious swallow-tails creation in the left and right end of the ellipse (Fig. 1b). In Fig. 2 we present an anisotropic curve shortening evolution of an ellipse computed again with help of the asymptotically uniform tangential redistribution. In all numerical experiments parameters of computations are shown in figure captions.

Again starting from an initial ellipse with half-axes ratio 1:3 we numerically compute its evolution by the surface diffusion. Fig. 3a shows the result with and Fig. 3b without asymptotically uniform tangential redistribution (AUTR). In Tables 1 and 2 we show how precisely the encompassed area is preserved (which is one of the analytical properties of surface diffusion flow with $\delta = 1$ and $b = 0$) during the evolution. The main observation here is the fact that when using AUTR we obtain significantly lower error and the method with AUTR has approximately the second order of accuracy which is not the case for computations without AUTR. In these experiments we used coupling $\tau \approx h^2$, cf. [9–11,27,

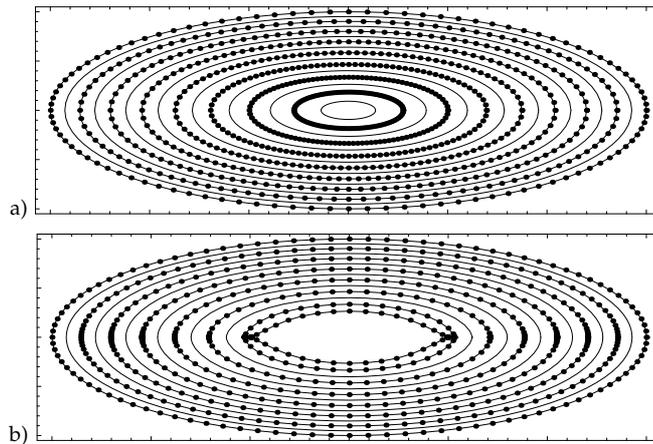


Figure 1: Affine invariant shrinking evolution of an initial ellipse ($b = k^{1/3}, \delta = 0, F = 0$); a) with asymptotically uniform redistribution ($\kappa_1 = 3$); b) without redistribution. Numerical parameters $n = 100, \tau = 0.001$. Time steps $j = 0, 200, \dots, 1400$ are plotted using lines and grid points, while time steps $j = 100, 300, \dots$, up to $j = 1500$ a); and $j = 1100$ b) are plotted using lines.

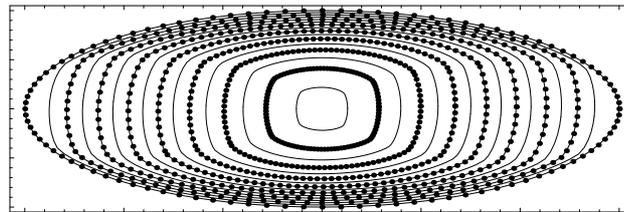


Figure 2: Anisotropic shrinking evolution of an initial ellipse with AUTR; $b = (1 - 0.9 \cos(4\nu - \pi))k, \delta = 0, F = 0$. Numerical and AUTR parameters: $n = 100, \tau = 0.001, \kappa_1 = 3$. Time steps $j = 0, 200, \dots, 1400$ are plotted using lines and grid points, while time steps $j = 100, 300, \dots, 1500$ are plotted using lines.

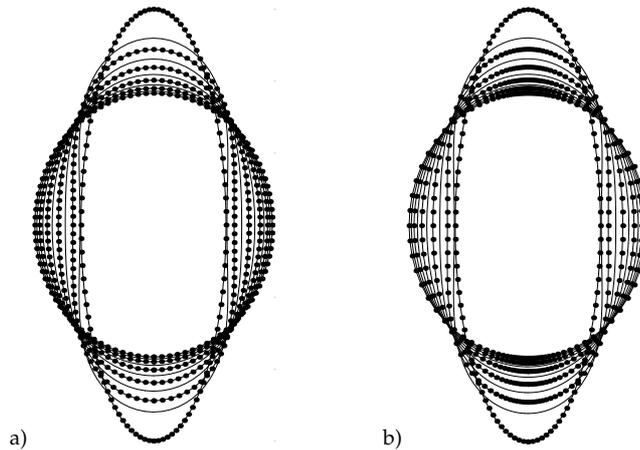


Figure 3: Surface diffusion flow ($\delta = 1, b = 0, F = 0$) of an initial ellipse with a) and without tangential redistribution b). Numerical and redistribution parameters: $n = 100, \tau = 0.001, \kappa_1 = 10$. Time steps $j = 0, 400, \dots, 2000$ are plotted using lines and grid points, while time steps $j = 200, 600, \dots, 1800$ are plotted using lines.

Table 1: An ellipse evolving by the surface diffusion using AUTR, same parameters as in Fig. 3a. We report errors in area evolution, experimental order of convergence (EOC) in this quantity and computational time (CPU) for refined discretization parameters.

n	τ	# of steps	area error	EOC	CPU (sec)
25	0.016	125	0.0774		0.02
50	0.004	500	0.0202	1.93	0.43
100	0.001	2000	0.0052	1.89	2.94
200	0.00025	8000	0.0014	1.91	40.21

Table 2: An ellipse evolving by the surface diffusion without tangential redistribution, same parameters as in Fig. 3b. Same quantities as above are reported.

n	τ	# of steps	area error	EOC	CPU (sec)
25	0.016	125	0.2922		0.11
50	0.004	500	0.1056	1.46	4.66
100	0.001	2000	0.0320	1.72	198
200	0.00025	8000	0.0155	1.04	1741

28], the exact area at any time moment is $A_e = 3\pi$ and the area error is computed as

$$\|e_n^m\| = \left(\sum_{j=1}^m (A^j - A_e)^2 \tau \right)^{\frac{1}{2}}, \quad (4.1)$$

where

$$A^j = \frac{1}{2} \sum_{i=1}^n \det(x_i^j, x_i^j - x_{i-1}^j)$$

gives the encompassed area of the polygonal computed curve at the j -th time step.

In Tables 1 and 2 we also report computational times achieved on a standard 2.2GHz laptop. They justify why the method with AUTR is referred to as a *fast* method. For standard curve resolutions, with 100 or 200 grid points, we get that one time step takes 0.0014 respectively 0.0050 second. Such fast CPU times are obtained due to tangential redistribution which not only stabilized the computations but also improve cyclic penta-diagonal iteration matrix properties in the SOR iterative method (the relaxation parameter was set to 1.6). Such CPU times are obtained also in further experiments presented in this section. Of course, one has to multiply them by the number of time steps which may lead to overall long computations when computing long time behaviors of curve evolutions shown e.g. in Fig. 8.

The next set of experiments is focused on the backward mean curvature flow with expanding constant force. The flow is regularized by various strengths of the Willmore flow. Starting with an initial ellipse with half-axes ratio 3:1 we can observe that in the case of a strong regularization the backward mean curvature flow is dominated by the elastic relaxation due to the Willmore surface energy (see Fig. 4a). On the other hand,

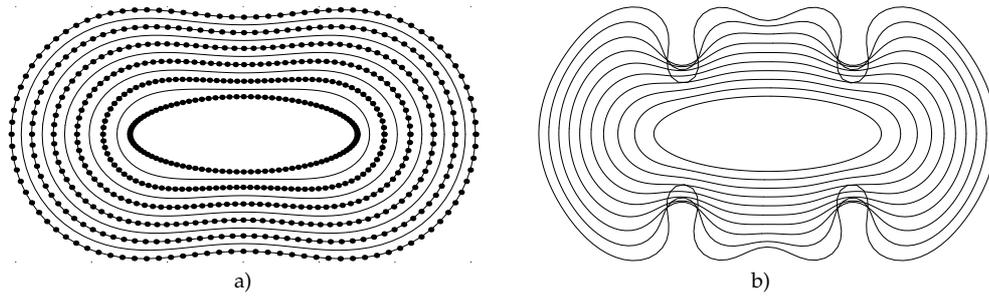


Figure 4: Backward mean curvature flow with negative external force regularized by the Willmore surface energy ($b = -k - \frac{1}{2}\delta k^3, F = -1$). a) a strong regularization effect with $\delta = 1$; b) a weak regularization $\delta = 0.1$. Numerical and AUTR parameters: $n = 100, \tau = 0.001, \kappa_1 = 10$. In subfigure a) the time steps $j = 0, 400, \dots, 2000$ are plotted using lines and grid points, while time steps $j = 200, 600, \dots, 1800$ are plotted only with lines. In subfigure b) the time steps $j = 0, 200, \dots, 1800$ are plotted.

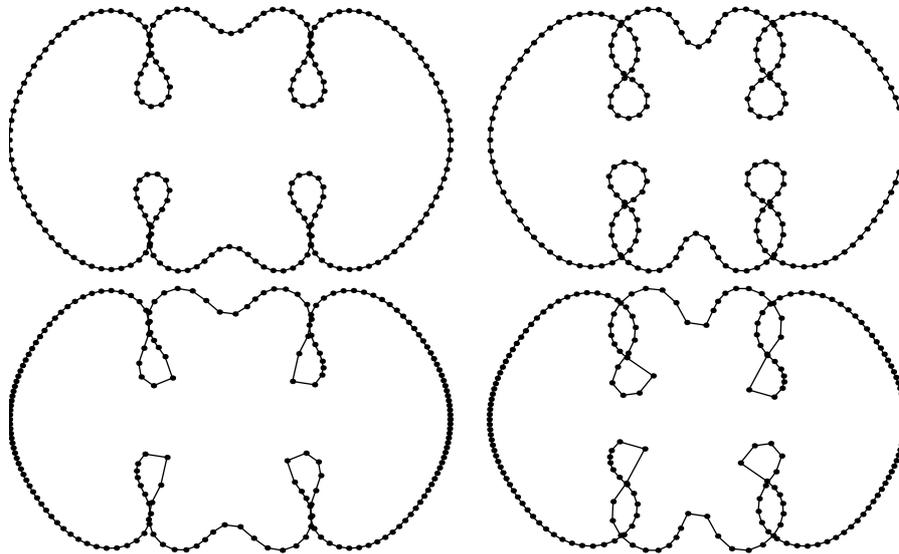


Figure 5: Comparison of continuation of the evolution from Fig. 4b computed with AUTR (upper row) and with redistribution preserving relative local length (bottom row), respectively. In both cases we show time steps $j = 2000$ (left) and $j = 2200$ (right).

if we decrease the fourth order regularization by taking smaller values of the parameter $\delta > 0$, the nonconvex parts formed during evolution are attenuated (see Fig. 4b) and even curve selfintersections may occur (see Figs. 5 and 6) as it can be expected in the backward in time diffusion process.

In Fig. 5 we also show an important role of AUTR in such nontrivial experiments. The asymptotically uniform tangential redistribution keeps very good curve resolution even in cases of several subsequent curve selfintersections. On the other hand, if we consider for example the well-known redistribution preserving the relative local length [18, 21, 25], obtained by taking $\omega = 0$ in (2.11) and (3.1) then this method is not capable to

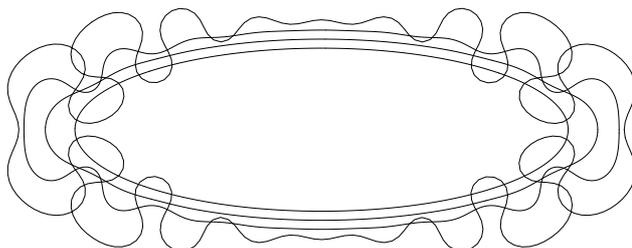


Figure 6: Regularized backward curvature driven flow with $b = -k - \frac{1}{2}\delta k^3$, $F = -1$ and even weaker regularization with $\delta = 0.01$. Redistribution parameter: $\kappa_1 = 10$ and numerical parameters: $n = 400$, $\tau = 0.0001$, are used. Time steps $j \in \{0, 1000, 2000, 3000\}$ are plotted.

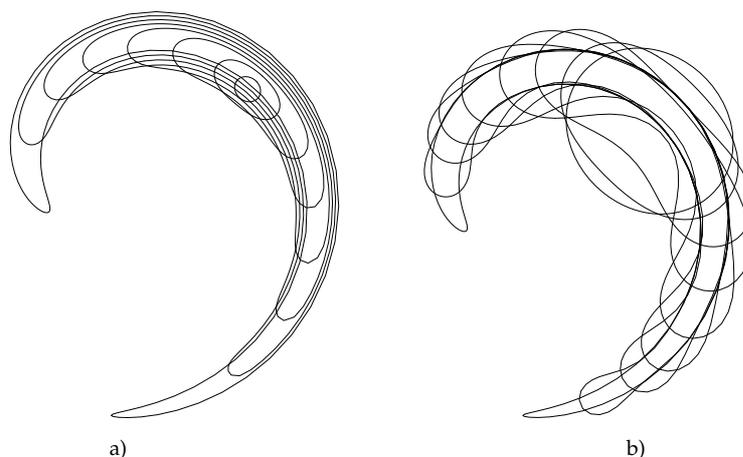


Figure 7: a) mean curvature flow with $b(k, \nu) = k$, $\delta = 0$, $F = 0$; b) surface diffusion flow with $b = 0$, $\delta = 1$, $F = 0$ of an initial spiral. Numerical and ATR parameters: $n = 100$, $\kappa_1 = 10$ and $\tau = 10^{-6}$ a) and $\tau = 10^{-10}$ b). Time steps $j \in \{0, 1, 2, 3, 4, 5, 6, 6.5\} \times 10^4$ a) and $j \in \{0, 1, 5, 10, 20, 40, 60, 80, 100\} \times 10^4$ b) are plotted.

handle this situations properly. If we consider $\delta = 0.01$ the backward diffusion effects are strongly dominating. In Fig. 6 we can observe very fast shape "enrichment" soon after even a nonconvexification of the evolving curve. It is worth to note that such experiments would be impossible without incorporating the tangential stabilization into the direct Lagrangian computational approach.

The last set of experiments is devoted to evolution of an initial spiral given by

$$x_1(u) = a \cos b, \quad x_2(u) = a \sin b,$$

$$a = 0.5 e^{-1 - \frac{1}{2} \sin(2\pi u)} - 0.025 \cos(2\pi u), \quad b = 10 \arctan(1 + 0.5 \sin(2\pi u)),$$

plotted in Fig. 8d. Again the presence of the tangential redistribution (ATR) has a stabilizing effect on all numerical computations. Without redistribution a parametric approach collapses soon. The evolution by the mean curvature and surface diffusion is shown in Fig. 7a and Fig. 7b. The backward curve diffusion regularized by a different strength of the Willmore flow is presented in Fig. 8a-c.

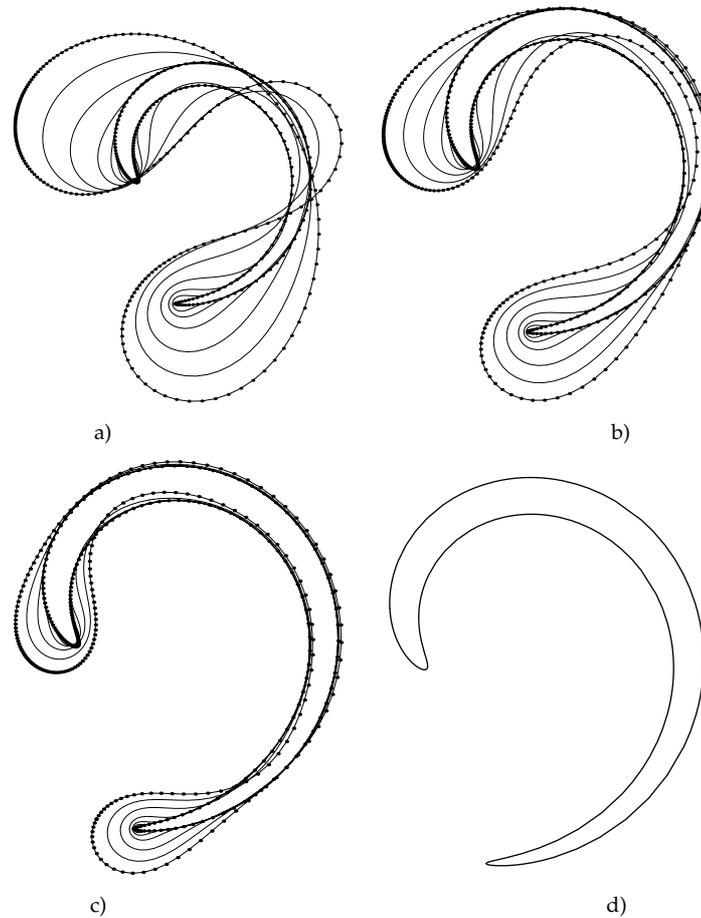


Figure 8: Backward mean curvature flow regularized by the Willmore flow ($b(k, \nu) = -k - \frac{1}{2}\delta k^3, F = 0$) with different strengths $\delta = 1$ a), $\delta = 0.1$ b), $\delta = 0.01$ c) of the same initial spiral d). Numerical and AUTR parameters: $n = 200, \tau = 10^{-12}, \kappa_1 = 100$. Time steps: $j = 0, 10^5, 10^6, 10^7, 10^8, 4 \cdot 10^8, 10^9$ are plotted. Initial and the last step plotted with grid points.

5 Conclusions

A new direct Lagrangian method stabilized by a suitable tangential redistribution has been presented for the case of general plane curve evolution models. An evolved curve is driven in the normal direction by a combination of the fourth order terms related to the intrinsic Laplacian of curvature, second order terms related to the curvature, first order terms related to anisotropy and tangential redistribution and by a given external velocity field. We showed how a proper choice of a tangential velocity can stabilize and speed up computations. Nontrivial numerical experiments justified applicability and numerical stability of the approximation scheme in the anisotropic mean curvature flow, surface diffusion and the Willmore flow and even in the case of backward curve diffusion slightly

regularized by the fourth order terms. Further study of the scheme from the numerical analysis point of view as well as derivation of analytical quantities to which the numerical results can be compared in the above mentioned models will be the objective of our future research.

Acknowledgments

This work was supported by grants: VEGA 1/0269/09, APVV-0351-07, APVV-RPEU-0004-07 (K. Mikula and M. Balažovjeh) and APVV-0247-06 (D. Ševčovič).

References

- [1] S. B. Angenent, M. E. Gurtin, Multiphase thermomechanics with an interfacial structure 2. Evolution of an isothermal interface, *Arch. Rat. Mech. Anal.*, 108 (1989), 323-391.
- [2] J. W. Barrett, H. Garcke, R. Nürnberg, A Parametric Finite Element Method for Fourth Order Geometric Evolution Equations, *Journal of Computational Physics*, 222 (2007), 441-467.
- [3] J. W. Barrett, H. Garcke, R. Nürnberg, On the variational approximation of combined second and fourth order geometric evolution equations, *SIAM Journal on Scientific Computing*, 29/3 (2007), 1006-1041.
- [4] M. Beneš, M. Kimura, P. Pauš, D. Ševčovič, T. Tsujikawa, S. Yazaki, Application of a curvature adjusted method in image segmentation, *Bulletin of Inst. of Mathematics, Academia Sinica, New Series*, 3(4) (2008), 509-524.
- [5] M. Beneš, K. Mikula, T. Oberhuber and D. Ševčovič, Comparison study for Level set and Direct Lagrangian methods for computing Willmore flow of closed planar curves, *Computing and Visualization in Science* (2009), DOI 10.1007/s00791-008-0112-2.
- [6] J. W. Cahn, J. E. Taylor, Surface motion by surface diffusion, *Acta Metallica Materiala*, 42 (1994), 1045-1063.
- [7] V. Caselles, R. Kimmel, G. Sapiro, Geodesic active contours, *International Journal of Computer Vision*, 22 (1997), 61-79.
- [8] K. Deckelnick, Weak solutions of the curve shortening flow, *Calc. Var. Partial Differ. Equ.*, 5 (1997), 489-510.
- [9] G. Dziuk, Convergence of a semi discrete scheme for the curve shortening flow, *Mathematical Models and Methods in Applied Sciences*, 4 (1994), 589-606.
- [10] G. Dziuk, Discrete anisotropic curve shortening flow, *SIAM J. Numer. Anal.*, 36 (1999), 1808-1830.
- [11] G. Dziuk, E. Kuwert, R. Schatzle, Evolution of elastic curves in \mathbb{R}^n : existence and computation, *SIAM J. Math. Anal.*, 33 (2002), 1228-1245.
- [12] L. Euler, *Methodus inveniendi lineas curvas: Additamentum I, De curvis elasticis*, Opera Omnia, Zürich: Orell Fassli, Ser. 1, 24 (1952), 231-297.
- [13] P. Frolkovič, K. Mikula, Flux-based level set method: a finite volume method for evolving interfaces, *Applied Numerical Mathematics*, Vol. 57, No. 4 (2007), 436-454.
- [14] P. Frolkovič, K. Mikula, High-resolution flux-based level set method, *SIAM J. Sci. Comp.*, 29/2 (2007), 579-597.
- [15] M. Gage, R. S. Hamilton, The heat equation shrinking convex plane curves, *J. Diff. Geom.*, 23 (1986), 69-96.

- [16] M. Grayson, The heat equation shrinks embedded plane curves to round points, *J. Diff. Geom.*, 26 (1987), 285-314.
- [17] A. Handlovičová, K. Mikula, F. Sgallari, Semi-implicit complementary volume scheme for solving level set like equations in image processing and curve evolution, *Numer. Math.*, 93 (2003), 675-695.
- [18] T.Y. Hou, J. Lowengrub, M. Shelley, Removing the stiffness from interfacial flows and surface tension, *J. Comput. Phys.*, 114 (1994), 312-338.
- [19] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, *International Journal of Computer Vision*, 1 (1987), 321-331.
- [20] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, A. Yezzi, Conformal curvature flows: from phase transitions to active vision, *Arch. Rational Mech. Anal.*, 134 (1996), 275-301.
- [21] M. Kimura, Numerical analysis for moving boundary problems using the boundary tracking method, *Japan J. Indust. Appl. Math.*, 14 (1997), 373-398.
- [22] K. Mikula, J. Kačur, Evolution of convex plane curves describing anisotropic motions of phase interfaces, *SIAM J. Sci. Comput.*, 17 (1996), 1302-1327.
- [23] K. Mikula, Solution of nonlinear curvature driven evolution of plane convex curves, *Applied Numerical Mathematics*, 21 (1997), 1-14.
- [24] K. Mikula, D. Ševčovič, Solution of nonlinearly curvature driven evolution of plane curves, *Applied Numerical Mathematics*, 31 (1999), 191-207.
- [25] K. Mikula, D. Ševčovič, Evolution of plane curves driven by a nonlinear function of curvature and anisotropy, *SIAM J. Appl. Math.*, 61 (2001), 1473-1501.
- [26] K. Mikula, D. Ševčovič, A direct method for solving an anisotropic mean curvature flow of planar curve with an external force, *Mathematical Methods in Applied Sciences*, 27(13) (2004), 1545-1565.
- [27] K. Mikula, D. Ševčovič, Computational and qualitative aspects of evolution of curves driven by curvature and external force, *Comput. Visual. Sci.*, 6 (2004), 211-225.
- [28] K. Mikula, D. Ševčovič, Tangentially stabilized Lagrangian algorithm for elastic curve evolution driven by intrinsic Laplacian of curvature, *ALGORITMY 2005, Conference on Scientific Computing, Vysoke Tatry-Podbanske, Slovakia, March 13-18, 2005, Proceedings of contributed papers and posters (2005)*, 32-41.
- [29] K. Mikula, D. Ševčovič, Evolution of curves on surface driven by the geodesic curvature and external force, *Applicable Analysis*, Vol. 85, No. 4 (2006), 345-362.
- [30] S. Osher, R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer-Verlag, 2003.
- [31] S. Osher, J. Sethian, Fronts propagating with curvature dependent speed: algorithm based on Hamilton-Jacobi formulation, *J. Comput. Phys.*, 79 (1988), 12-49.
- [32] G. Sapiro, A. Tannenbaum, On affine plane curve evolution, *J. Funct. Anal.*, 119 (1994), 79-120.
- [33] J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Material Science*, Cambridge University Press, New York, 1999.
- [34] D. Ševčovič, S. Yazaki, On a motion of plane curves with a curvature adjusted tangential velocity, *Proceedings of Equadiff 2007 Conference (2008)*. To appear.
- [35] S. Yazaki, On the tangential velocity arising in a crystalline approximation of evolving plane curves, *Kybernetika*, Vol. 43, No. 6 (2007), 913-918.