A convergent evolving finite element algorithm for mean curvature flow of closed surfaces

Balázs Kovács

joint work with
Buyang Li (Hong Kong), Christian Lubich (Tübingen)

7 December 2018, Banff
Mean curvature flow

Surface evolution under \textit{mean curvature flow} (MCF):

\[ v = -H \nu_{\Gamma(t)}. \]

Some references:

- [Huiskens (1984)] – analysis;
- appears in many biological and physical models.

Some \textit{finite element} literature:

- [Dziuk (1990)] – first algorithm for mean curvature flow;
- [Barrett, Garcke and Nürnberg] – many schemes, with good properties;
  Both without error analysis.
- for curves or graphs much more is known due to Barrett, Deckelnick, Dziuk, Pozzi, Stinner, Styles, and many others...
An example for MCF

(mean curvature flow)

(mean curvature)

(zoom in)
Outline

- Notations, coupled system and weak from
- Evolving surface finite elements and matrix–vector formalism
- Time integration
  - Relating different surfaces
  - Stability via energy estimates
- Numerical experiments
Notations, coupled system and weak from
Evolving surfaces

Let \( \Gamma(t) \subset \mathbb{R}^3 \) be a closed surface

\[
\Gamma[X] = \Gamma(t) = \Gamma[X(\cdot, t)] = \{X(p, t) : p \in \Gamma^0\},
\]

where \( \Gamma^0 \) is an initial surface, and

\[
X : \Gamma^0 \times [0, T] \rightarrow \mathbb{R}^3 \text{ a smooth vector-field.}
\]

Consider a point \( p \in \Gamma^0 \) fixed, the surface velocity \( \nu \) satisfies, in
\( x(t) = X(p, t) \), by

\[
\partial_t x(t) = \nu(x(t), t) \quad \left( = \partial_t X(p, t) \right).
\]

The position \( x = X(p, t) \) is obtained by solving the above ODE from 0 to \( t \)
for a fixed \( p \), \( \Gamma[X(\cdot, t)] \) is a collection of such points \( x \).
Differential operators on $\Gamma$  

- Normal vector: $\nu_{\Gamma}$  
- Tangential gradient: $\nabla_{\Gamma} u = \nabla u - (\nabla u \cdot \nu_{\Gamma}) \nu_{\Gamma} : \Gamma \to \mathbb{R}^3$  
- Laplace–Beltrami operator: $\Delta_{\Gamma} u = \nabla_{\Gamma} \cdot \nabla_{\Gamma} u$  
  
  (for $u : \Gamma \to \mathbb{R}$, on a regular surface $\Gamma \subset \mathbb{R}^3$)
Geometric quantities and mean curvature $H$

- extended Weingarten map (3 × 3 symmetric matrix)
  \[ A(x) = \nabla_{\Gamma} \nu_{\Gamma}(x) \]
- with eigenvalues: $\kappa_1$ and $\kappa_2$, the principal curvatures, and 0 (with eigenvector $\nu_{\Gamma}$)
- they define
  \[
  \text{mean curvature } H = \text{tr}(A) = \kappa_1 + \kappa_2,
  \]
  \[
  |A|^2 = \|A\|_F^2 = \kappa_1^2 + \kappa_2^2.
  \]
MCF and Dziuk’s algorithm

A regular surface $\Gamma[X]$ moving under mean curvature flow satisfies:

$$
\partial_t X = \nu, \\
\nu = -H \nu_{\Gamma[X]}.
$$

Heat like equation, using that $-H \nu_{\Gamma} = \Delta_{\Gamma} x_{\Gamma}$:

$$
\partial_t X(p, t) = \Delta_{\Gamma[X]} x_{\Gamma[X]}.
$$

The algorithm [Dziuk (1990)] is based on its weak formulation, for all test functions $\varphi \in H^1(\Gamma[X])^3$:

$$
\int_{\Gamma[X]} \nu \cdot \varphi = -\int_{\Gamma[X]} \nabla_{\Gamma[X]} x_{\Gamma(X)} \cdot \nabla_{\Gamma[X]} \varphi. \\
+ \text{ODE for positions.}
$$

Simple and elegant algorithm but, unfortunately, no convergence result.
The analysts approach

A regular surface \( \Gamma[X] \) moving under mean curvature flow satisfies:

\[
\partial_t X = \nu,
\]

\[
\nu = -H \nu_{\Gamma[X]}.
\]

**Lemma [Huisken (1984)]**

For a regular surface \( \Gamma[X] \) moving under mean curvature flow, the normal vector and the mean curvature satisfy

\[
\partial \cdot \nu = \Delta_{\Gamma[X]} \nu + |A|^2 \nu,
\]

\[
\partial \cdot H = \Delta_{\Gamma[X]} H + |A|^2 H.
\]

**Coupled system: fundamental for analysis, but were not used for numerics.**
Weak form

The numerical discretization is based on a weak formulation:

$$\int_{\Gamma[X]} \nabla_{\Gamma[X]} \nu \cdot \nabla_{\Gamma[X]} \varphi^v + \int_{\Gamma[X]} \nu \cdot \varphi^v = -\int_{\Gamma[X]} \nabla_{\Gamma[X]} (H \nu) \cdot \nabla_{\Gamma[X]} \varphi^v - \int_{\Gamma[X]} H \nu \cdot \varphi^v,$$

$$\int_{\Gamma[X]} \partial \cdot \nu \cdot \varphi^v + \int_{\Gamma[X]} \nabla_{\Gamma[X]} \nu \cdot \nabla_{\Gamma[X]} \varphi^v = \int_{\Gamma[X]} |\nabla_{\Gamma[X]} \nu|^2 \nu \cdot \varphi^v,$$

$$\int_{\Gamma[X]} \partial \cdot H \varphi^H + \int_{\Gamma[X]} \nabla_{\Gamma[X]} H \cdot \nabla_{\Gamma[X]} \varphi^H = \int_{\Gamma[X]} |\nabla_{\Gamma[X]} \nu|^2 H \varphi^H,$$

+ ODE for positions.
Evolving surface finite elements and matrix-vector formulation
We collect the evolving nodes into the vector $\mathbf{x}(t) \in \mathbb{R}^{3N}$.

The nodes $\mathbf{x}(t) \in \mathbb{R}^{3N}$ determine the approximation

$$\Gamma[\mathbf{X}(\cdot, t)] \approx \Gamma[\mathbf{x}(t)].$$

Nodal basis functions, of degree $k$, $\phi_j[\mathbf{x}]$ span the evolving finite element space $S_h(\mathbf{x})$ on $\Gamma_h[\mathbf{x}]$. 
Spatial semi-discretisation – Dziuk’s algorithm

Find the nodal vector $x(t) \in \mathbb{R}^{3N}$ and discrete velocity $v_h(\cdot, t) \in S_h(x(t))^3$ such that

$$\int_{\Gamma_h[x]} v_h \cdot \varphi_h = -\int_{\Gamma_h[x]} \nabla \Gamma_h[x] \times \Gamma_h[x] \cdot \nabla \Gamma_h[x] \varphi_h,$$

$$\partial_t X_h = v_h,$$

for all $\varphi_h(\cdot, t) \in S_h(x(t))^3$, with $X_h(\cdot, t) = \sum_{j=1}^{N} x_j(t) \phi_j[x(0)]$.

Matrix–vector formulation:

The mass and stiffness matrices are denoted by $M(x)$ and $A(x)$

$$M(x)v + A(x)x = 0,$$

$$\dot{x} = v.$$
Spatial semi-discretisation – Dziuk’s algorithm

Find the nodal vector $\mathbf{x}(t) \in \mathbb{R}^{3N}$ and discrete velocity $\mathbf{v}_h(\cdot, t) \in S_h(\mathbf{x}(t))^3$ such that

$$
\int_{\Gamma_h[\mathbf{x}]} \mathbf{v}_h \cdot \varphi_h = - \int_{\Gamma_h[\mathbf{x}]} \nabla \Gamma_h[\mathbf{x}] \times \Gamma_h[\mathbf{x}] \cdot \nabla \Gamma_h[\mathbf{x}] \varphi_h,
$$

$$
\partial_t X_h = \mathbf{v}_h,
$$

for all $\varphi_h(\cdot, t) \in S_h(\mathbf{x}(t))^3$, with $X_h(\cdot, t) = \sum_{j=1}^{N} x_j(t) \phi_j[\mathbf{x}(0)]$.

Matrix–vector formulation:

The mass and stiffness matrices are denoted by $\mathbf{M}(\mathbf{x})$ and $\mathbf{A}(\mathbf{x})$

$$
\mathbf{M}(\mathbf{x}) \mathbf{v} + \mathbf{A}(\mathbf{x}) \mathbf{x} = 0,
$$

$$
\dot{\mathbf{x}} = \mathbf{v}.
$$
Spatial semi-discretisation – coupled system

Find the unknown nodal vector $x(t) \in \mathbb{R}^{3N}$ and the unknown finite element functions $u_h(\cdot, t) \in S_h(x(t))$ and $v_h(\cdot, t) \in S_h(x(t))^3$ such that, for all $\varphi_h(\cdot, t) \in S_h(x(t))$, with $\partial_h \varphi_h = 0$, and for all $\psi_h(\cdot, t) \in S_h(x(t))^3$

\[
\int_{\Gamma_h[x]} \nabla \Gamma_h[x] v_h \cdot \nabla \Gamma_h[x] \varphi_h^v + \int_{\Gamma_h[x]} v_h \cdot \varphi_h^v = 0,
\]

\[
\int_{\Gamma_h[x]} \partial_h v_h \cdot \varphi_h^v + \int_{\Gamma_h[x]} \nabla \Gamma_h[x] v_h \cdot \nabla \Gamma_h[x] \varphi_h^v = \int_{\Gamma_h[x]} |\nabla \Gamma_h[x] v_h|^2 v_h \cdot \varphi_h^v,
\]

\[
\int_{\Gamma_h[x]} \partial_h \varphi_h^H + \int_{\Gamma_h[x]} \nabla \Gamma_h[x] H_h \cdot \nabla \Gamma_h[x] \varphi_h^H = \int_{\Gamma_h[x]} |\nabla \Gamma_h[x] v_h|^2 H_h \varphi_h^H,
\]

+ ODE for positions.
Matrix–vector formulation

Upon setting \( u = (n, H)^T \in \mathbb{R}^{4N} \) and \( K(x) = M(x) + A(x) \), the semi-discrete problem is equivalent to the following differential algebraic system:

\[
K(x)v = g(x, u), \\
M(x)\dot{u} + A(x)u = f(x, u), \\
\dot{x} = v.
\]

As compared to [Dziuk (1990)]:

\[
M(x)v + A(x)x = 0, \\
\dot{x} = v.
\]
Time integration:

stability and convergence
Linearly implicit full discretization

Recall the matrix–vector formulation:

\[ K(x)v = g(x, u), \]
\[ M(x)\ddot{u} + A(x)u = f(x, u), \]
\[ \dot{x} = v. \]

A non-linear coupled problem.
Linearly implicit full discretization

Linearly implicit $q$-step backward difference formulae (BDF):

\[
K(\tilde{x}^n)v^n = g(\tilde{x}^n, \tilde{u}^n),
\]
\[
M(\tilde{x}^n)\dot{u}^n + A(\tilde{x}^n)u^n = f(\tilde{x}^n, \tilde{u}^n),
\]
\[
\dot{x}^n = v^n,
\]

with

**discrete derivative:**
\[
\dot{x}^n = \frac{1}{\tau} \sum_{j=0}^{q} \delta_j x^{n-j}, \quad \text{and}
\]

**extrapolated value:**
\[
\tilde{x}^n = \sum_{j=0}^{q-1} \gamma_j x^{n-1-j}.
\]
Stability
Relating different surfaces – I.

Let \( x \in \mathbb{R}^{3N} \) and \( y \in \mathbb{R}^{3N} \) be two vectors which define the surfaces \( \Gamma_h(x) \) and \( \Gamma_h(y) \).

Intermediate surfaces:

\[
e = x - y \iff \Gamma_h^\theta(y + \theta e) \quad (\theta \in [0, 1]),
\]

and the corresponding error:

\[
e_h^\theta = \sum_{j=1}^{N} e_j \phi_j[y + \theta e].
\]

Relating different surfaces:

\[
w^T(M(x) - M(y))z = \int_{0}^{1} \int_{\Gamma_h^\theta} w_h^\theta (\nabla_{\Gamma_h^\theta} \cdot e_h^\theta) z_h^\theta \, d\theta,
\]

\[
w^T(A(x) - A(y))z = \int_{0}^{1} \int_{\Gamma_h^\theta} \nabla_{\Gamma_h^\theta} w_h^\theta \cdot (D_{\Gamma_h^\theta} e_h^\theta) \nabla_{\Gamma_h^\theta} z_h^\theta \, d\theta,
\]
Relating different surfaces – II.

We proved **six technical lemmas**, and techniques form [K., Li, Lubich and Power (2017)], which relate different evolving surfaces with one another. For example:

\[ \| w \|_{M(y+e)} \leq c \| w \|_{M(y)}, \]
\[ \| \nabla_{\Gamma^\theta_h} w^\theta_h \|_{L^p(\Gamma^\theta_h)} \leq c_p \| \nabla_{\Gamma^0_h} w^0_h \|_{L^p(\Gamma^0_h)}, \]

etc. . . . , and

\[ w^T (M(x) - M(y)) z \leq c \| w \|_{M(y)} \| z \|_{M(y)}, \]
\[ w^T (M(x) - M(y)) w \leq c \| e^0_h \|_{W^{1,\infty}(\Gamma^0_h[y])} \| w \|_{M(y)}^2, \]

etc. . . .

**Under the important condition on e:** \[ \| e^0_h \|_{W^{1,\infty}(\Gamma^0_h[y])} \leq \frac{1}{2}. \]
Relating different surfaces – II.

We proved six technical lemmas, and techniques form [K., Li, Lubich and Power (2017)], which relate different evolving surfaces with one another. For example:

\[ \|w\|_{M(y+e)} \leq c \|w\|_{M(y)}, \]

\[ \| \nabla_{\Gamma^0} w^\theta_h \|_{L^p(\Gamma^0)} \leq c_p \| \nabla_{\Gamma^0} w^0_h \|_{L^p(\Gamma^0)}, \]

etc. . . , and

\[ w^T (M(x) - M(y))z \leq c \|w\|_{M(y)} \|z\|_{M(y)}, \]

\[ w^T (M(x) - M(y))w \leq c \|e^0_h\|_{W^{1,\infty}(\Gamma^0)} \|w\|^2_{M(y)}, \]

etc. . .

Under the important condition on \(e\):

\[ \|e^0_h\|_{W^{1,\infty}(\Gamma^0)} \leq \frac{1}{2}. \]
Stability

A key issue is to establish a pointwise bound on the $W^{1,\infty}$ norm of the errors.

(i) Obtain pointwise $H^1$ norm error estimates at time $t_n$;

(ii) Using an inverse estimate to establish bounds in the $W^{1,\infty}$ norm;

(iii) Repeat for $t_{n+1}$. 

B. Kovács (Tübingen)
Illustrate using a simple case

Consider (in the usual Hilbert space setting) the parabolic problem:

\[(\dot{u}(t), \varphi) + (Au(t), \varphi) = (f(t), \varphi),\]
\[u(0) = u_0.\]

**Energy estimates, testing with** $u$ **and** $\dot{u}$:

\[
\frac{d}{dt} |u|^2 + \|u\|^2 \leq c \|f\|^2, \tag{a}
\]
\[
|\dot{u}|^2 + \frac{d}{dt} \|u\|^2 \leq c |f|^2, \tag{b}
\]

then integrate in time.
Energy estimates for BDF methods

Using G-stability of [Dahlquist (1978)] and the multiplier techniques of [Nevanlinna and Odeh (1981)]:

Testing with multiplier $u^n - \eta u^{n-1}$ (A-stable: $\eta = 0$, $A(\alpha)$-stable: $0 < \eta < 1$):

$$(\dot{u}^n, u^n - \eta u^{n-1}) + (Au^n, u^n - \eta u^{n-1}) = (f^n, u^n - \eta u^{n-1}). \quad (a)$$

for PDEs: [Lubich, Mansour and Venkataraman (2013)], [Akrivis and Lubich (2015)], …

Testing with $\dot{u}^n$:

$$(\dot{u}^n, \dot{u}^n) + (Au^n, \dot{u}^n) = (f^n, \dot{u}^n). \quad (b)$$

Where is the multiplier?
Energy estimates for BDF methods

Using \textit{G-stability} of [Dahlquist (1978)] and the \textit{multiplier techniques} of [Nevanlinna and Odeh (1981)]:

Testing with multiplier $u^n - \eta u^{n-1}$ ($A$-stable: $\eta = 0$, $A(\alpha)$-stable: $0 < \eta < 1$):

$$
(\dot{u}^n, u^n - \eta u^{n-1}) + (Au^n, u^n - \eta u^{n-1}) = (f^n, u^n - \eta u^{n-1}). \quad (a)
$$

for PDEs: [Lubich, Mansour and Venkataraman (2013)], [Akrivis and Lubich (2015)], ...

Subtract the equations at time $t_{n-1}$ from at time $t_n$, and test with $\dot{u}^n$:

$$
(\dot{u}^n - \eta \dot{u}^{n-1}, \dot{u}^n) + (Au^n - \eta Au^{n-1}, \dot{u}^n) = (f^n - \eta f^{n-1}, \dot{u}^n). \quad (b)
$$

Which yields a pointwise stability estimate in the strong norm.
Consider the full discretisation of the coupled mean curvature flow problem using ESFEM of polynomial degree \( k \geq 2 \) and linearly implicit BDF method with \( q \leq 5 \).

Let the solutions \((X, \nu, \nu, H)\) be sufficiently smooth (i.e. \( H^{k+1} \)). Then for sufficiently small \( h \) and \( \tau \) satisfying (with \( C_0 > 0 \) fixed arbitrary)

\[
\tau^q \leq c_0 h \quad \text{if} \quad q \leq 2, \quad \text{and} \quad \tau \leq C_0 h \quad \text{if} \quad 3 \leq q \leq 5,
\]

the following estimates hold for \( 0 \leq t \leq T \):

\[
\left\| \left( x_h^n \right)^L - \text{id}_{\Gamma(t_n)} \right\|_{H^1(\Gamma(t_n))} \leq C(h^k + \tau^q),
\]

\[
\left\| \left( \nu_h^n \right)^L - \nu(\cdot, t_n) \right\|_{H^1(\Gamma(t_n))} \leq C(h^k + \tau^q),
\]

\[
\left\| \left( \nu_h^n \right)^L - \nu(\cdot, t_n) \right\|_{H^1(\Gamma(t_n))} \leq C(h^k + \tau^q),
\]

\[
\left\| \left( H_h^n \right)^L - H(\cdot, t_n) \right\|_{H^1(\Gamma(t_n))} \leq C(h^k + \tau^q).
\]
Numerical experiments
Comparison

Dziuk’s algorithm

$X_h$

normalised
Comparison

Dziuk’s algorithm

$X_h$

normalised
Comparison

Dziuk’s algorithm

$X_h$

normalised
Comparison

Dziuk’s algorithm

$X_h$

normalised

time = 0.078125
The normalised algorithm – singularity
Convergence – in time

\| X - X_h \|_{H^1} \n
\| v - v_h \|_{H^1} \n
\| H - H_h \|_{H^1} \n
\begin{align*}
\text{step size } (\tau) & \quad \text{step size } (\tau) & \quad \text{step size } (\tau)
\end{align*}
Convergence — in space

\[ \| X - X_h \|_{H^1} \]

\[ \| \nu - \nu_h \|_{H^1} \]

\[ \| H - H_h \|_{H^1} \]

- mesh size \((h)\)

B. Kovács (Tübingen)
Thank you for your attention!