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

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Mean curvature flow

Surface evolution under mean curvature flow (MCF):

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

Some references:

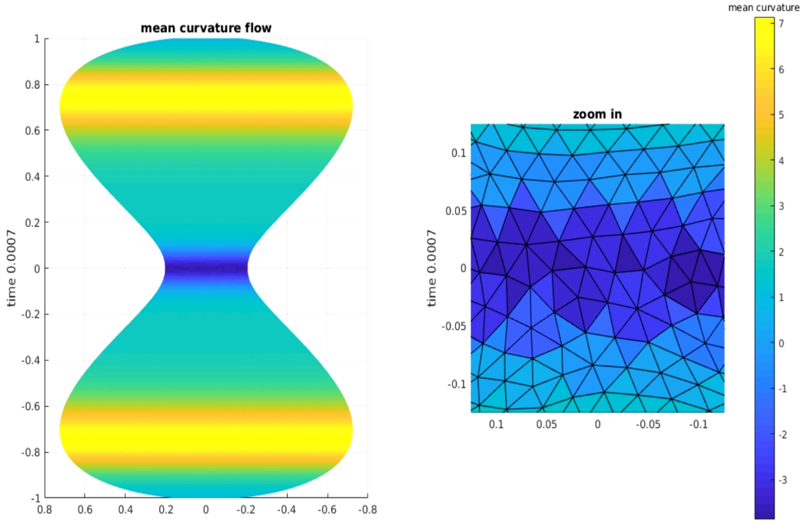
- [Huisken (1984)] – analysis;
- appears in many biological and physical models.

Some 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



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 Γ^0 is an initial surface, and

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

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

$$\partial_t x(t) = v(x(t), t) \ \Big(= \partial_t X(p, t)\Big).$$

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 Γ

- Normal vector: ν_Γ
- 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: κ_1 and κ_2, the principal curvatures, and 0 (with eigenvector ν_Γ)
- they define

$$\text{mean curvature} \qquad H = \mathrm{tr}(A) = \kappa_1 + \kappa_2,$$
$$|A|^2 = \|A\|_F^2 = \kappa_1^2 + \kappa_2^2.$$

MCF and Dziuk's algorithm

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

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

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 $\varphi \in H^1(\Gamma[X])^3$:

$$\int_{\Gamma[X]} v \cdot \varphi = -\int_{\Gamma[X]} \nabla_{\Gamma[X]} x_{\Gamma(X)} \cdot \nabla_{\Gamma[X]} \varphi.$$

+ 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:

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

Lemma [Huisken (1984)]

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

$$\partial^\bullet \nu = \Delta_{\Gamma[X]}\nu + |A|^2 \nu,$$
$$\partial^\bullet H = \Delta_{\Gamma[X]}H + |A|^2 H.$$

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

Weak form

The numerical discretization is based on a weak formulation:

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

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

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

+ ODE for positions.

Evolving surface finite elements and matrix–vector formulation

ESFEM – basic notations

- 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

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

- 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 $\mathbf{x}(t) \in \mathbb{R}^{3N}$ and discrete velocity $v_h(\cdot,t) \in S_h(\mathbf{x}(t))^3$ such that

$$\int_{\Gamma_h[\mathbf{x}]} v_h \cdot \varphi_h = -\int_{\Gamma_h[\mathbf{x}]} \nabla_{\Gamma_h[\mathbf{x}]} x_{\Gamma_h[\mathbf{x}]} \cdot \nabla_{\Gamma_h[\mathbf{x}]} \varphi_h,$$

$$\partial_t X_h = v_h,$$

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 – Dziuk's algorithm

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

$$\int_{\Gamma_h[\mathbf{x}]} v_h \cdot \varphi_h = -\int_{\Gamma_h[\mathbf{x}]} \nabla_{\Gamma_h[\mathbf{x}]} x_{\Gamma_h[\mathbf{x}]} \cdot \nabla_{\Gamma_h[\mathbf{x}]} \varphi_h,$$
$$\partial_t X_h = v_h,$$

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

$$\int_{\Gamma_h[\mathbf{x}]} \nabla_{\Gamma_h[\mathbf{x}]} v_h \cdot \nabla_{\Gamma_h[\mathbf{x}]} \varphi_h^v + \int_{\Gamma_h[\mathbf{x}]} v_h \cdot \varphi_h^v$$

$$= -\int_{\Gamma_h[\mathbf{x}]} \nabla_{\Gamma_h[\mathbf{x}]} (H_h \nu_h) \cdot \nabla_{\Gamma_h[\mathbf{x}]} \varphi_h^v - \int_{\Gamma_h[\mathbf{x}]} H_h \nu_h \cdot \varphi_h^v,$$

$$\int_{\Gamma_h[\mathbf{x}]} \partial_h^\bullet \nu_h \cdot \varphi_h^\nu + \int_{\Gamma_h[\mathbf{x}]} \nabla_{\Gamma_h[\mathbf{x}]} \nu_h \cdot \nabla_{\Gamma_h[\mathbf{x}]} \varphi_h^\nu = \int_{\Gamma_h[\mathbf{x}]} |\nabla_{\Gamma_h[\mathbf{x}]} \nu_h|^2 \, \nu_h \cdot \varphi_h^\nu,$$

$$\int_{\Gamma_h[\mathbf{x}]} \partial_h^\bullet H_h \, \varphi_h^H + \int_{\Gamma_h[\mathbf{x}]} \nabla_{\Gamma_h[\mathbf{x}]} H_h \cdot \nabla_{\Gamma_h[\mathbf{x}]} \varphi_h^H = \int_{\Gamma_h[\mathbf{x}]} |\nabla_{\Gamma_h[\mathbf{x}]} \nu_h|^2 \, H_h \, \varphi_h^H,$$

+ ODE for positions.

Matrix–vector formulation

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

$$\begin{aligned} \mathbf{K}(\mathbf{x})\mathbf{v} &= \mathbf{g}(\mathbf{x}, \mathbf{u}), \\ \mathbf{M}(\mathbf{x})\dot{\mathbf{u}} + \mathbf{A}(\mathbf{x})\mathbf{u} &= \mathbf{f}(\mathbf{x}, \mathbf{u}), \\ \dot{\mathbf{x}} &= \mathbf{v}. \end{aligned}$$

As compared to [Dziuk (1990)]:

$$\begin{aligned} \mathbf{M}(\mathbf{x})\mathbf{v} + \mathbf{A}(\mathbf{x})\mathbf{x} &= 0, \\ \dot{\mathbf{x}} &= \mathbf{v}. \end{aligned}$$

Time integration:
stability and convergence

Linearly implicit full discretization

Recall the matrix–vector formulation:

$$\begin{aligned} \mathbf{K}(\mathbf{x})\mathbf{v} &= \mathbf{g}(\mathbf{x},\mathbf{u}), \\ \mathbf{M}(\mathbf{x})\dot{\mathbf{u}} + \mathbf{A}(\mathbf{x})\mathbf{u} &= \mathbf{f}(\mathbf{x},\mathbf{u}), \\ \dot{\mathbf{x}} &= \mathbf{v}. \end{aligned}$$

A non-linear coupled problem.

Linearly implicit full discretization

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

$$\begin{aligned} \mathbf{K}(\widetilde{\mathbf{x}}^n)\mathbf{v}^n &= \mathbf{g}(\widetilde{\mathbf{x}}^n, \widetilde{\mathbf{u}}^n), \\ \mathbf{M}(\widetilde{\mathbf{x}}^n)\dot{\mathbf{u}}^n + \mathbf{A}(\widetilde{\mathbf{x}}^n)\mathbf{u}^n &= \mathbf{f}(\widetilde{\mathbf{x}}^n, \widetilde{\mathbf{u}}^n), \\ \dot{\mathbf{x}}^n &= \mathbf{v}^n, \end{aligned}$$

with

discrete derivative:: $\dot{\mathbf{x}}^n = \frac{1}{\tau}\sum_{j=0}^{q} \delta_j \mathbf{x}^{n-j}$, and

extrapolated value: $\widetilde{\mathbf{x}}^n = \sum_{j=0}^{q-1} \gamma_j \mathbf{x}^{n-1-j}$.

Stability

Relating different surfaces – I.

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

Intermediate surfaces:

$$\mathbf{e} = \mathbf{x} - \mathbf{y} \leftrightsquigarrow \Gamma_h^\theta = \Gamma_h(\mathbf{y} + \theta\mathbf{e}) \quad (\theta \in [0,1]),$$

and the corresponding error:

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

Relating different surfaces:

$$\mathbf{w}^T(\mathbf{M}(\mathbf{x}) - \mathbf{M}(\mathbf{y}))\mathbf{z} = \int_0^1 \int_{\Gamma_h^\theta} w_h^\theta(\nabla_{\Gamma_h^\theta} \cdot e_h^\theta) z_h^\theta \, \mathrm{d}\theta,$$

$$\mathbf{w}^T(\mathbf{A}(\mathbf{x}) - \mathbf{A}(\mathbf{y}))\mathbf{z} = \int_0^1 \int_{\Gamma_h^\theta} \nabla_{\Gamma_h^\theta} w_h^\theta \cdot (D_{\Gamma_h^\theta} e_h^\theta) \nabla_{\Gamma_h^\theta} z_h^\theta \, \mathrm{d}\theta,$$

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:

$$\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y}+\mathbf{e})} \leq c\,\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y})},$$
$$\|\nabla_{\Gamma_h^\theta} w_h^\theta\|_{L^p(\Gamma_h^\theta)} \leq c_p\,\|\nabla_{\Gamma_h^0} w_h^0\|_{L^p(\Gamma_h^0)},$$

etc. . . . , and

$$\mathbf{w}^T(\mathbf{M}(\mathbf{x}) - \mathbf{M}(\mathbf{y}))\mathbf{z} \leq c\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y})}\|\mathbf{z}\|_{\mathbf{M}(\mathbf{y})},$$
$$\mathbf{w}^T(\mathbf{M}(\mathbf{x}) - \mathbf{M}(\mathbf{y}))\mathbf{w} \leq c\|e_h^0\|_{W^{1,\infty}(\Gamma_h[\mathbf{y}])}\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y})}^2,$$

etc. . . .

Under the important condition on e: $\|e_h^0\|_{W^{1,\infty}(\Gamma_h[\mathbf{y}])} \leq \frac{1}{2}$.

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:

$$\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y}+\mathbf{e})} \leq c\,\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y})},$$
$$\|\nabla_{\Gamma_h^\theta} w_h^\theta\|_{L^p(\Gamma_h^\theta)} \leq c_p\,\|\nabla_{\Gamma_h^0} w_h^0\|_{L^p(\Gamma_h^0)},$$

etc. . . . , and

$$\mathbf{w}^T(\mathbf{M}(\mathbf{x}) - \mathbf{M}(\mathbf{y}))\mathbf{z} \leq c\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y})}\|\mathbf{z}\|_{\mathbf{M}(\mathbf{y})},$$
$$\mathbf{w}^T(\mathbf{M}(\mathbf{x}) - \mathbf{M}(\mathbf{y}))\mathbf{w} \leq c\|e_h^0\|_{W^{1,\infty}(\Gamma_h[\mathbf{y}])}\|\mathbf{w}\|_{\mathbf{M}(\mathbf{y})}^2,$$

etc. . . .

Under the important condition on e: $\|e_h^0\|_{W^{1,\infty}(\Gamma_h[\mathbf{y}])} \leq \frac{1}{2}$.

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}.

Illustrate using a simple case

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

$$
\begin{aligned}
(\dot{u}(t), \varphi) + (Au(t), \varphi) &= (f(t), \varphi), \\
u(0) &= u_0 .
\end{aligned}
$$

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

$$
\frac{\mathrm{d}}{\mathrm{d}t}|u|^2 + \|u\|^2 \leq c\|f\|_*^2, \tag{a}
$$

$$
|\dot{u}|^2 + \frac{\mathrm{d}}{\mathrm{d}t}\|u\|^2 \leq c|f|^2, \tag{b}
$$

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: $0 < \eta < 1$):

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

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

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

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

Where is the multiplier?

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: $0 < \eta < 1$):

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

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 $\dot{u}^n$:

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

Which yields a pointwise stability estimate in the strong norm.

Convergence of the full discretisation

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, v, ν, H) be sufficiently smooth (i.e. H^{k+1}).
Then for sufficiently small h and τ 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 $0 \leq t \leq T$:

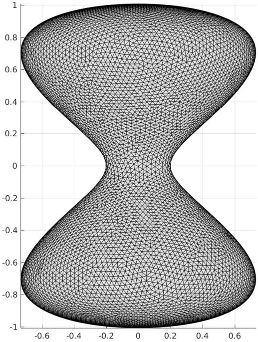
$$\begin{aligned}
\|(x_h^n)^L - \mathrm{id}_{\Gamma(t_n)}\|_{H^1(\Gamma(t_n))^3} &\leq C(h^k + \tau^q), \\
\|(v_h^n)^L - v(\cdot, t_n)\|_{H^1(\Gamma(t_n))^3} &\leq C(h^k + \tau^q), \\
\|(\nu_h^n)^L - \nu(\cdot, t_n)\|_{H^1(\Gamma(t_n))^3} &\leq C(h^k + \tau^q), \\
\|(H_h^n)^L - H(\cdot, t_n)\|_{H^1(\Gamma(t_n))} &\leq C(h^k + \tau^q).
\end{aligned}$$

Numerical experiments

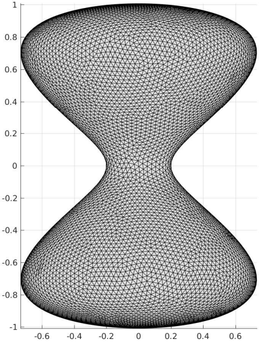
Comparison

time = 0

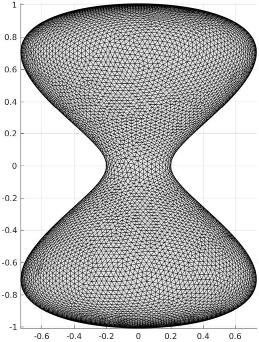
Dziuk's algorithm



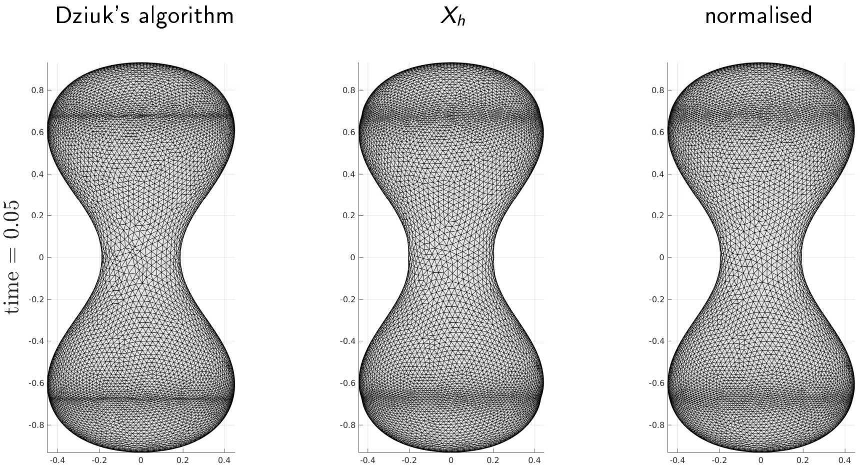
X_h



normalised

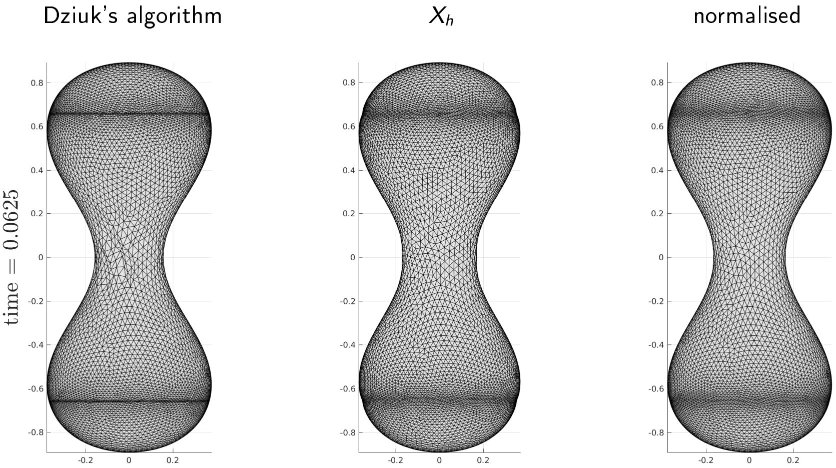


Comparison



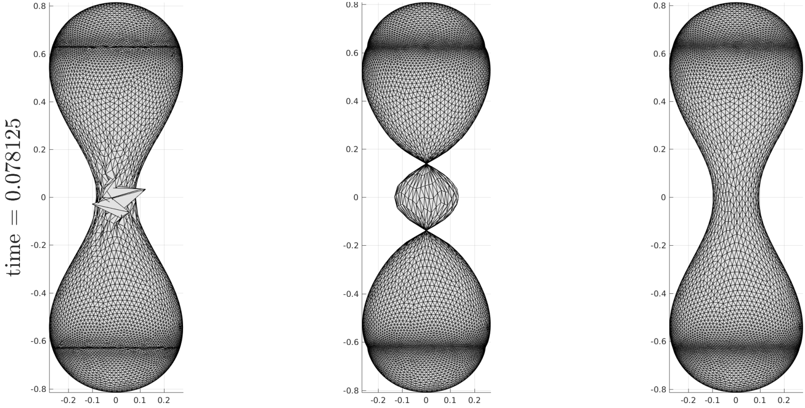
Dziuk's algorithm
X_h
normalised
time = 0.05
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-0.4
-0.2
0
0.2
0.4

Comparison

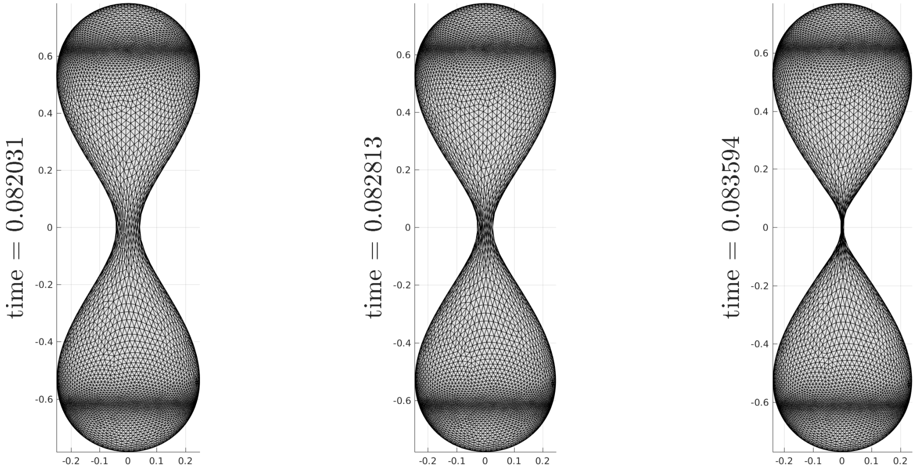

Dziuk's algorithm
X_h
normalised
time = 0.0625
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-0.2
0
0.2

Comparison

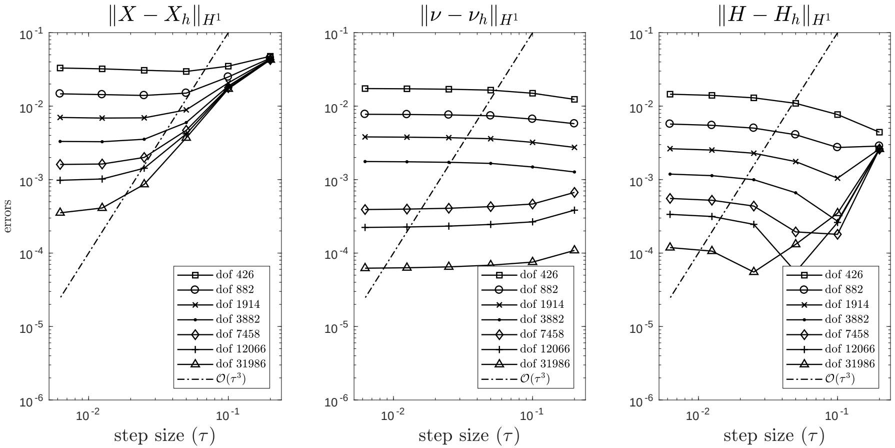
Dziuk's algorithm | X_h | normalised



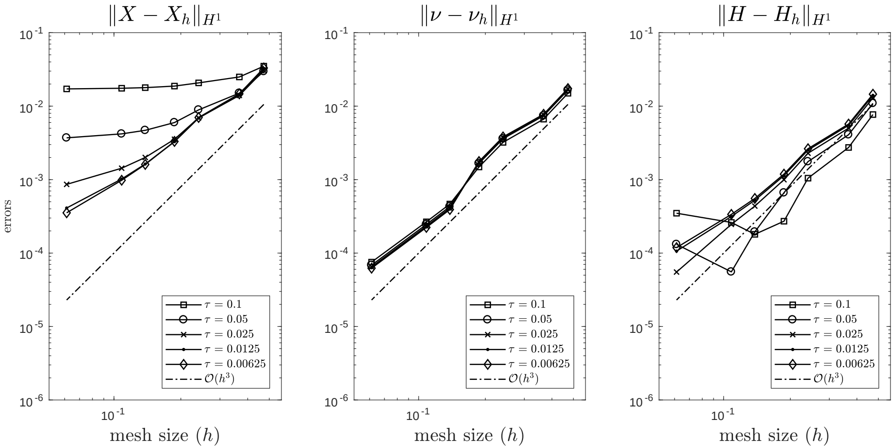
The normalised algorithm – singularity



Convergence – in time



Convergence – in space



Thank you for your attention!