Graph-Laplacian minimisation for surface smoothing in 3D finite element tetrahedral meshes

Richard Huber, Martin Holler and Kristian Bredies

University of Graz, Austria

Wels, May 11, 2016.

Context

Simulations of the heart: Requires tetrahedral mesh.

Context

Simulations of the heart: Requires tetrahedral mesh.

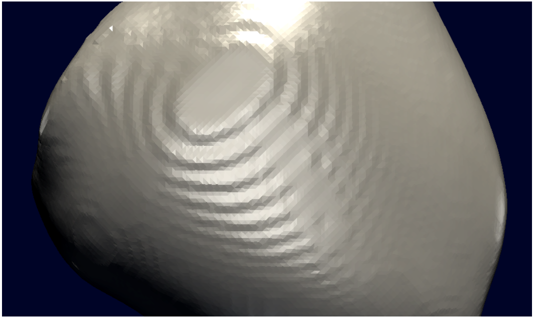
Mesh quality: Very flat tetrahedra cause errors in simulation.

Context

Simulations of the heart: Requires tetrahedral mesh.

Mesh quality: Very flat tetrahedra cause errors in simulation.

Oscillations: Artifacts occur on surface, in particular oscillations.

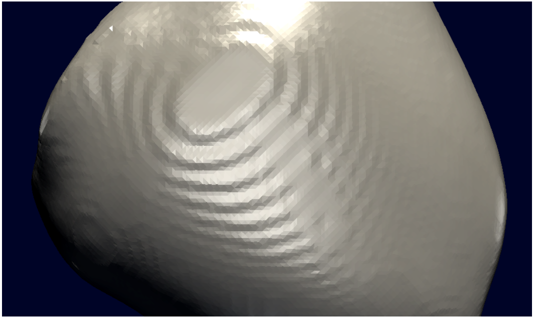


Context

Simulations of the heart: Requires tetrahedral mesh.

Mesh quality: Very flat tetrahedra cause errors in simulation.

Oscillations: Artifacts occur on surface, in particular oscillations.



Initial setting: Triangulation is given. Coordinates of vertices, edge information and masking of surface points.

Introduction

Goal: Mesh improvement for subsequent simulations.

Introduction

Goal: Mesh improvement for subsequent simulations.

Denoising & quality maintenance: Reduction of oscillations and preservation of high mesh quality.

Introduction

Goal: Mesh improvement for subsequent simulations.

Denoising & quality maintenance: Reduction of oscillations and preservation of high mesh quality.

Graph-Laplacian minimisation: Adjusting coordinates of nodes of surface tetrahedra, by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2} \|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega.$$

Introduction

Goal: Mesh improvement for subsequent simulations.

Denoising & quality maintenance: Reduction of oscillations and preservation of high mesh quality.

Graph-Laplacian minimisation: Adjusting coordinates of nodes of surface tetrahedra, by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2} \|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega.$$

Constraints: Control effect of smoothing.

Introduction

Goal: Mesh improvement for subsequent simulations.

Denoising & quality maintenance: Reduction of oscillations and preservation of high mesh quality.

Graph-Laplacian minimisation: Adjusting coordinates of nodes of surface tetrahedra, by solving

$$u^{+} \in \operatorname*{argmin}_{u \in U} \frac{1}{2} \|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega.$$

Constraints: Control effect of smoothing.

Primal-dual algorithm: Solution of resulting optimisation problem.

Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

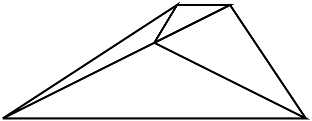
Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

Quality metrics:

Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

Quality metrics:

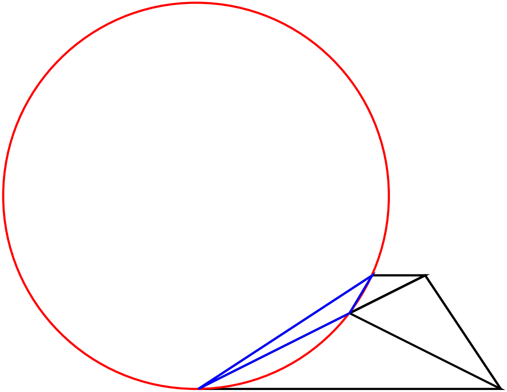
- Skewness: Ratio of tetrahedron's volume to its circumscribed ball's volume.



Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

Quality metrics:

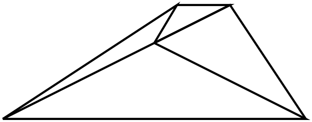
- Skewness: Ratio of tetrahedron's volume to its circumscribed ball's volume.



Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

Quality metrics:

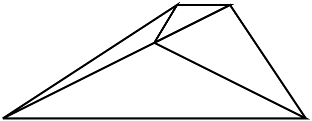
- Skewness: Ratio of tetrahedron's volume to its circumscribed ball's volume.
- Maximal and minimal angle in a tetrahedron.



Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

Quality metrics:

- Skewness: Ratio of tetrahedron's volume to its circumscribed ball's volume.
- Maximal and minimal angle in a tetrahedron.
- Ratio of maximal and minimal edge length in a tetrahedron.

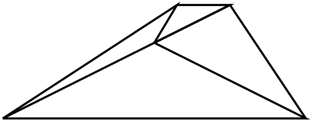


Mesh quality: Non-degenerate tetrahedra, no very flat tetrahedra.

Quality metrics:

- Skewness: Ratio of tetrahedron's volume to its circumscribed ball's volume.
- Maximal and minimal angle in a tetrahedron.
- Ratio of maximal and minimal edge length in a tetrahedron.

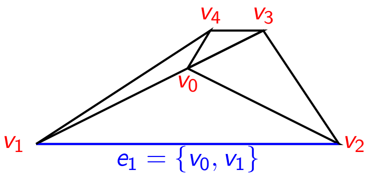
Initial mesh: Assumed to have sufficiently high mesh quality.



Notation:

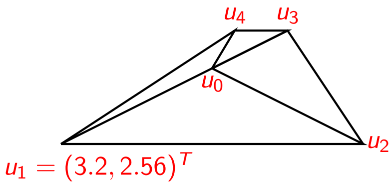
Notation:

- The surface triangulation induces a graph $G = (V, E)$ with vertices $v_0 \dots v_N$ and Edges E.



Notation:

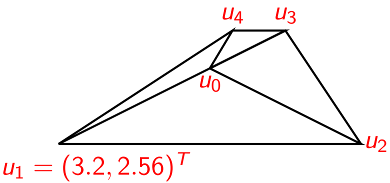
- The surface triangulation induces a graph $G = (V, E)$ with vertices $v_0 \dots v_N$ and Edges E.
- $u_0, \dots, u_N \in \mathbb{R}^3$ denotes position of surface nodes $v_0 \dots v_N$ and $(u_0, \dots, u_N) = u \in U = \mathbb{R}^{3(N+1)}$.



Notation:

- The surface triangulation induces a graph $G = (V, E)$ with vertices $v_0 \dots v_N$ and Edges E.
- $u_0, \dots, u_N \in \mathbb{R}^3$ denotes position of surface nodes $v_0 \dots v_N$ and $(u_0, \dots, u_N) = u \in U = \mathbb{R}^{3(N+1)}$.
- The graph-Laplacian operator is defined as component-wise matrix-multiplication with the matrix $\hat{\Delta} \in \mathbb{R}^{N \times N}$, given as

$$(\hat{\Delta})_{i,j} := \begin{cases} \mathrm{Deg}(v_i) & \text{if } i = j, \\ -1 & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{else,} \end{cases}$$

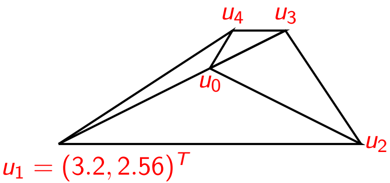


Notation:

- The surface triangulation induces a graph $G = (V, E)$ with vertices $v_0 \dots v_N$ and Edges E.
- $u_0, \dots, u_N \in \mathbb{R}^3$ denotes position of surface nodes $v_0 \dots v_N$ and $(u_0, \dots, u_N) = u \in U = \mathbb{R}^{3(N+1)}$.
- The graph-Laplacian operator is defined as component-wise matrix-multiplication with the matrix $\hat{\Delta} \in \mathbb{R}^{N \times N}$, given as

$$(\hat{\Delta})_{i,j} := \begin{cases} \mathrm{Deg}(v_i) & \text{if } i = j, \\ -1 & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{else,} \end{cases} \qquad \Delta u := \begin{pmatrix} \hat{\Delta} u^1 \\ \hat{\Delta} u^2 \\ \hat{\Delta} u^3 \end{pmatrix},$$

where $u^1, u^2, u^3 \in \mathbb{R}^N$ represents x,y and z coordinates of u.



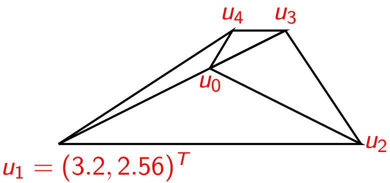
Notation:

- The surface triangulation induces a graph $G = (V, E)$ with vertices $v_0 \dots v_N$ and Edges E.
- $u_0, \dots, u_N \in \mathbb{R}^3$ denotes position of surface nodes $v_0 \dots v_N$ and $(u_0, \dots, u_N) = u \in U = \mathbb{R}^{3(N+1)}$.
- The graph-Laplacian operator is defined as component-wise matrix-multiplication with the matrix $\hat{\Delta} \in \mathbb{R}^{N \times N}$, given as

$$(\hat{\Delta})_{i,j} := \begin{cases} \mathrm{Deg}(v_i) & \text{if } i = j, \\ -1 & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{else,} \end{cases} \quad \Delta u := \begin{pmatrix} \hat{\Delta} u^1 \\ \hat{\Delta} u^2 \\ \hat{\Delta} u^3 \end{pmatrix},$$

  where $u^1, u^2, u^3 \in \mathbb{R}^N$ represents x,y and z coordinates of u.

- Mean value and curvature: $(\Delta u)_0 = 4u_0 - u_1 - u_2 - u_3 - u_4$.



Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

$$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2. \tag{1}$$

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

$$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2. \tag{1}$$

Aim at adapting idea of reducing Laplacian energy in global method.

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

(1) $$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2.$$

Aim at adapting idea of reducing Laplacian energy in global method.

(2) $$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2.$$

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

(1) $$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2.$$

Aim at adapting idea of reducing Laplacian energy in global method.

(2) $$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2.$$

Issue: Solutions are concentrated points. Mesh quality is not considered.

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

$$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2. \tag{1}$$

Aim at adapting idea of reducing Laplacian energy in global method.

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2. \tag{2}$$

Issue: Solutions are concentrated points. Mesh quality is not considered.

Constraints: Consider problem (2) under constraints with feasible set Ω guaranteeing:

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

(1) $$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2.$$

Aim at adapting idea of reducing Laplacian energy in global method.

(2) $$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2.$$

Issue: Solutions are concentrated points. Mesh quality is not considered.

Constraints: Consider problem (2) under constraints with feasible set Ω guaranteeing:

- Fit to data.

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

$$(1) \qquad u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2.$$

Aim at adapting idea of reducing Laplacian energy in global method.

$$(2) \qquad u^+ \in \underset{u \in U}{\operatorname{argmin}} \frac{1}{2}\|\Delta u\|_2^2.$$

Issue: Solutions are concentrated points. Mesh quality is not considered.

Constraints: Consider problem (2) under constraints with feasible set Ω guaranteeing:

- Fit to data.
- Maintenance of mesh quality.

Energy reduction of graph-Laplacian

Local, vertex wise gradient descent

$$u_i \leftarrow u_i - \alpha(\Delta^T \Delta u)_i \text{ which reduces } \frac{1}{2}|(\Delta u)_i|^2. \tag{1}$$

Aim at adapting idea of reducing Laplacian energy in global method.

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2. \tag{2}$$

Issue: Solutions are concentrated points. Mesh quality is not considered.

Constraints: Consider problem (2) under constraints with feasible set Ω guaranteeing:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Update surface points by solving

$$(3) \qquad u^{+} \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega,$$

for suitable feasible set Ω that incorporates the above posed properties:

Update surface points by solving

$$(3) \qquad u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega,$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.

Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.

Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.

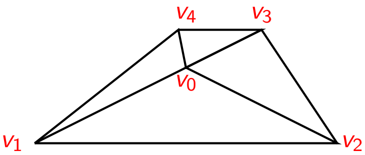
Update surface points by solving

$$(3) \qquad u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \qquad \text{subject to } u \in \Omega,$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



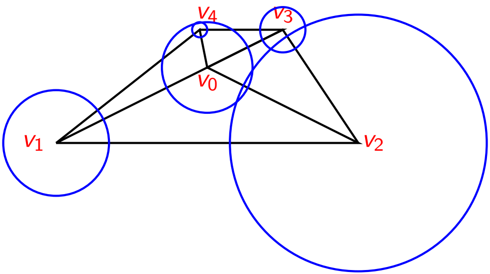
Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



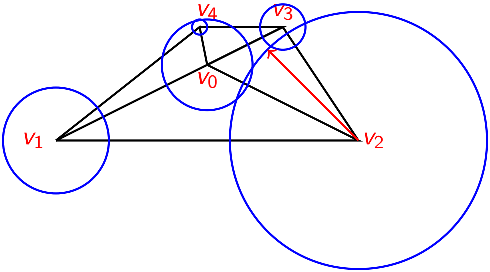
Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



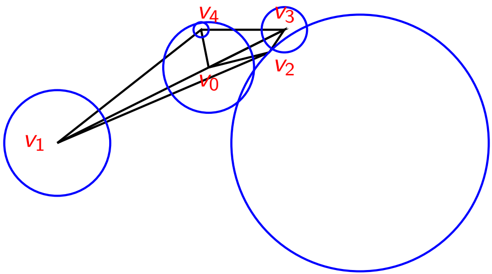
Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



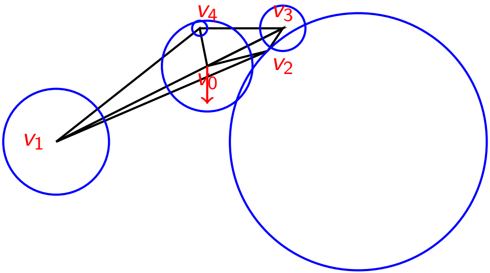
Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



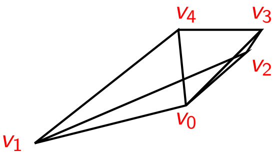
Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



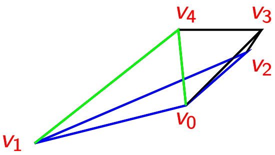
Update surface points by solving

$$u^+ \in \operatorname*{argmin}_{u \in U} \frac{1}{2}\|\Delta u\|_2^2, \quad \text{subject to } u \in \Omega, \tag{3}$$

for suitable feasible set Ω that incorporates the above posed properties:

- Fit to data.
- Maintenance of mesh quality.
- Convex, vertex wise independent representation of Ω.

Pointwise constraints: Fixing radii r_i for v_i. Allow u_i move in *ball* with radius r_i around original location.



Adaptive constraints:

,

Adaptive constraints:

- Incorporating topology into constraints.

,

Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

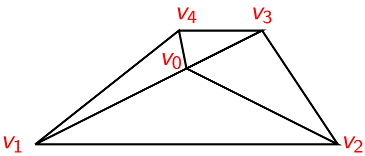
,

Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

,

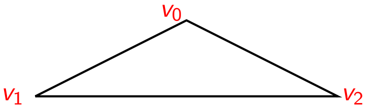


Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

,



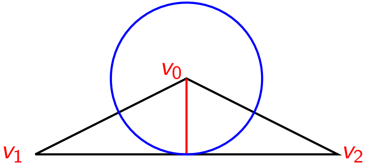
Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

- Heights determine maximal range of movement.

,



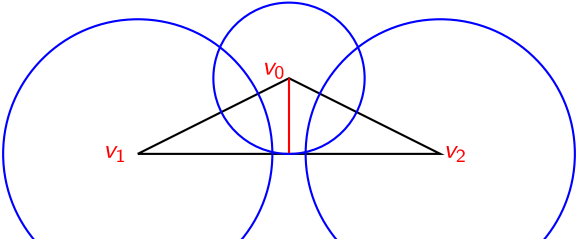
Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

- Heights determine maximal range of movement.
- Synchronous movement, further reduction of radius.

,



Adaptive constraints:

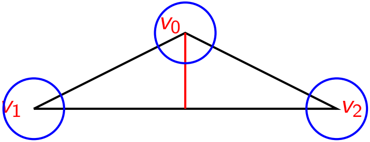
- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

- Heights determine maximal range of movement.
- Synchronous movement, further reduction of radius.

$$h_T = \min\{h\colon h \text{ height of } T\} \text{ for tetrahedron } T,$$

$$r_i = \alpha h_T \quad \text{for } 0 < \alpha < 1/2$$



Adaptive constraints:

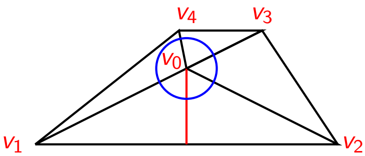
- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

- Heights determine maximal range of movement.
- Synchronous movement, further reduction of radius.
- v_0 contained in many tetrahedra.

$$h_T = \min\{h\colon h \text{ height of } T\} \text{ for tetrahedron } T,$$

$$r_i = \alpha h_T \quad \text{for } 0 < \alpha < 1/2$$



Adaptive constraints:

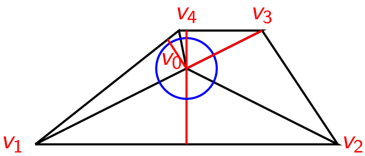
- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

- Heights determine maximal range of movement.
- Synchronous movement, further reduction of radius.
- v_0 contained in many tetrahedra.

$$h_T = \min\{h \colon h \text{ height of } T\} \text{ for tetrahedron } T,$$

$$r_i = \alpha h_T \quad \text{for } 0 < \alpha < 1/2$$



Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

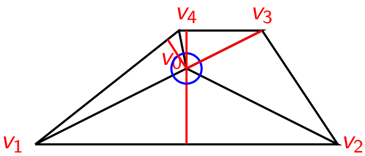
Considerations:

- Heights determine maximal range of movement.
- Synchronous movement, further reduction of radius.
- v_0 contained in many tetrahedra.

$$h_T = \min\{h\colon h \text{ height of } T\} \text{ for tetrahedron } T,$$

$$r_i = \alpha \hat{h}_i \quad \text{for } 0 < \alpha < 1/2 \text{ with } \hat{h}_i = \min\{h_T\colon v_i \in T\},$$

$$\Omega = \{u \in U\colon \|u_i - u_{0i}\| \leq r_i \text{ for } i = 1, \dots, N\}. \tag{4}$$



Adaptive constraints:

- Incorporating topology into constraints.
- Maintain high mesh quality.

Considerations:

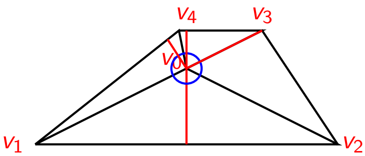
- Heights determine maximal range of movement.
- Synchronous movement, further reduction of radius.
- v_0 contained in many tetrahedra.

$$h_T = \min\{h\colon h \text{ height of } T\} \text{ for tetrahedron } T,$$

$$r_i = \alpha \hat{h}_i \quad \text{for } 0 < \alpha < 1/2 \text{ with } \hat{h}_i = \min\{h_T\colon v_i \in T\},$$

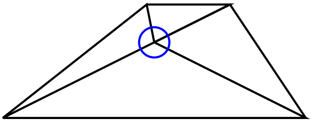
$$\Omega = \{u \in U\colon \|u_i - u_{0i}\| \le r_i \text{ for } i = 1, \dots, N\}. \tag{4}$$

- Note that also interior points are considered in the construction of constraints.



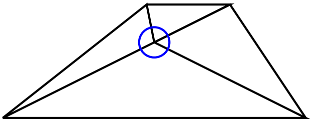
Restrictive constraints:

Restrictive constraints: Little movement allowed.



Restrictive constraints: Little movement allowed.

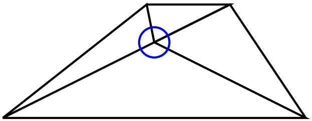
Reiteration of method: Start again with updated constraints.



Restrictive constraints: Little movement allowed.

Reiteration of method: Start again with updated constraints.

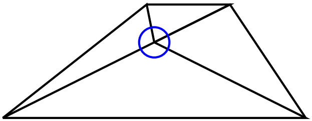
- More flexibility and movement.



Restrictive constraints: Little movement allowed.

Reiteration of method: Start again with updated constraints.

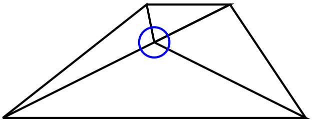
- More flexibility and movement.
- Constraints ensure mesh quality.



Restrictive constraints: Little movement allowed.

Reiteration of method: Start again with updated constraints.

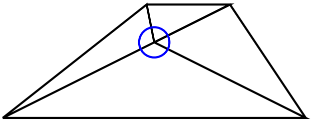
- More flexibility and movement.
- Constraints ensure mesh quality.
- Many iterations affect quality.



Restrictive constraints: Little movement allowed.

Reiteration of method: Start again with updated constraints.

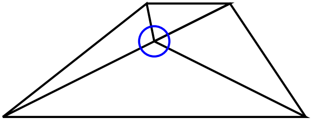
- More flexibility and movement.
- Constraints ensure mesh quality.
- Many iterations affect quality.
- 2-5 outer iteration.



Restrictive constraints: Little movement allowed.

Reiteration of method: Start again with updated constraints.

- More flexibility and movement.
- Constraints ensure mesh quality.
- Many iterations affect quality.
- 2-5 outer iteration.
- Improves results.



Outline

1 Preliminaries

2 The optimisation problem

3 Numerical solution

4 Discussion of results

Primal-dual algorithm: Global solution via iterative method.

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u \in \Omega} G(Au) \quad \Longleftrightarrow \quad \min_{u \in U} G(Au) + I_\Omega(u)$$

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u\in\Omega} G(Au) \quad \Longleftrightarrow \quad \min_{u\in U} G(Au) + I_\Omega(u)$$
$$\Longleftrightarrow \quad \min_{u\in U} \max_{w\in U} \langle w, Au\rangle - G^*(w) + I_\Omega(u).$$

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u\in\Omega} G(Au) \iff \min_{u\in U} G(Au) + I_\Omega(u)$$
$$\iff \min_{u\in U}\max_{w\in U} \langle w, Au\rangle - G^*(w) + I_\Omega(u).$$

Specific saddle-point problem for $A = \Delta$ and $G = \frac{1}{2}\|\cdot\|_2^2$

$$\min_{u\in U}\max_{w\in U} \quad \langle w, \Delta u\rangle - \frac{1}{2}\|w\|_2^2 + I_\Omega(u). \tag{5}$$

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u\in\Omega} G(Au) \iff \min_{u\in U} G(Au) + I_\Omega(u)$$
$$\iff \min_{u\in U}\max_{w\in U} \langle w, Au\rangle - G^*(w) + I_\Omega(u).$$

Specific saddle-point problem for $A = \Delta$ and $G = \frac{1}{2}\|\cdot\|_2^2$

(5)
$$\min_{u\in U}\max_{w\in U} \quad \langle w, \Delta u\rangle - \frac{1}{2}\|w\|_2^2 + I_\Omega(u).$$

Proposition

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u\in\Omega} G(Au) \iff \min_{u\in U} G(Au) + I_\Omega(u)$$
$$\iff \min_{u\in U}\max_{w\in U} \langle w, Au\rangle - G^*(w) + I_\Omega(u).$$

Specific saddle-point problem for $A = \Delta$ and $G = \frac{1}{2}\|\cdot\|_2^2$

$$\min_{u\in U}\max_{w\in U} \quad \langle w, \Delta u\rangle - \frac{1}{2}\|w\|_2^2 + I_\Omega(u). \tag{5}$$

Proposition

1. *The original optimisation problem* (3), $\min_{u\in\Omega} F(u) = \frac{1}{2}\|\Delta u\|_2^2$, *has a solution which however is not unique.*

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u\in\Omega} G(Au) \iff \min_{u\in U} G(Au) + I_\Omega(u)$$
$$\iff \min_{u\in U}\max_{w\in U} \langle w, Au\rangle - G^*(w) + I_\Omega(u).$$

Specific saddle-point problem for $A = \Delta$ and $G = \frac{1}{2}\|\cdot\|_2^2$

$$\min_{u\in U}\max_{w\in U} \quad \langle w, \Delta u\rangle - \frac{1}{2}\|w\|_2^2 + I_\Omega(u). \tag{5}$$

Proposition

1. *The original optimisation problem* (3), $\min_{u\in\Omega} F(u) = \frac{1}{2}\|\Delta u\|_2^2$, *has a solution which however is not unique.*
2. *Also, the saddle-point problem* (5) *with feasible set* Ω *defined as in* (4) *admits at least one solution.*

Primal-dual algorithm: Global solution via iterative method.

General saddle-point problem for linear A and G:

$$\min_{u\in\Omega} G(Au) \quad \Longleftrightarrow \quad \min_{u\in U} G(Au) + I_\Omega(u)$$
$$\Longleftrightarrow \quad \min_{u\in U} \max_{w\in U} \langle w, Au\rangle - G^*(w) + I_\Omega(u).$$

Specific saddle-point problem for $A = \Delta$ and $G = \frac{1}{2}\|\cdot\|_2^2$

(5) $$\min_{u\in U} \max_{w\in U} \quad \langle w, \Delta u\rangle - \frac{1}{2}\|w\|_2^2 + I_\Omega(u).$$

Proposition

1. *The original optimisation problem* (3), $\min_{u\in\Omega} F(u) = \frac{1}{2}\|\Delta u\|_2^2$, *has a solution which however is not unique.*
2. *Also, the saddle-point problem* (5) *with feasible set* Ω *defined as in* (4) *admits at least one solution.*
3. *Further, for any saddle-point* (u^+, w^+) *of* (5), u^+ *is a solution of the original minimisation problem* (3).

Specific iteration directives:

Specific iteration directives: Simplicity of constraints leads to

$$(6) \qquad \left\{ \vphantom{\begin{matrix} a \\ b \\ c \\ d \end{matrix}} \right.$$

Specific iteration directives: Simplicity of constraints leads to

$$(6) \qquad \begin{cases} w \leftarrow \frac{(w+\sigma\Delta\bar{u})}{(1+\sigma)} \\ \\ \\ \\ \end{cases}$$

- Matrix-vector multiplication.

Specific iteration directives: Simplicity of constraints leads to

$$(6)\qquad \begin{cases} w \leftarrow \frac{(w+\sigma\Delta\bar{u})}{(1+\sigma)} \\ \bar{u} \leftarrow P_\Omega(u - \tau\Delta w) \end{cases}$$

- Matrix-vector multiplication.
- With $P_\Omega(u)$ pointwise projection.

Specific iteration directives: Simplicity of constraints leads to

$$
\begin{cases}
w \leftarrow \frac{(w+\sigma\Delta\bar{u})}{(1+\sigma)} \\
\bar{u} \leftarrow P_\Omega(u - \tau\Delta w) \\
u \leftarrow 2\bar{u} - u \\
(u, \bar{u}) \leftarrow (\bar{u}, u)
\end{cases} \tag{6}
$$

- Matrix-vector multiplication.
- With $P_\Omega(u)$ pointwise projection.
- Simple operations, vectorisation, fast execution.

Outline

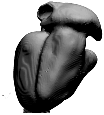
1 Preliminaries

2 The optimisation problem

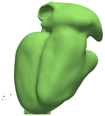
3 Numerical solution

4 Discussion of results

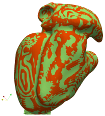
Visual results Example 1:



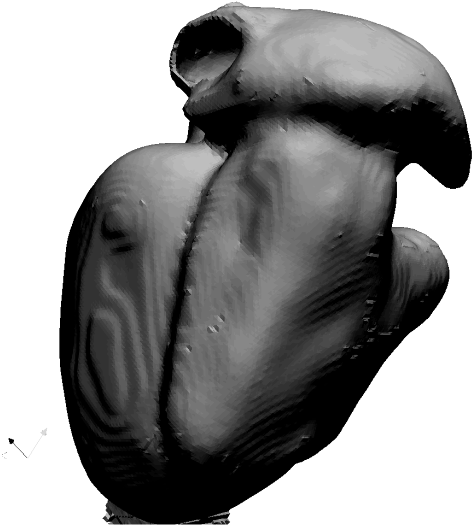
original



smoothed

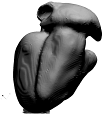


overlay

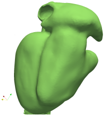


Oscillations:

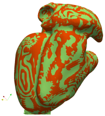
Visual results Example 1:



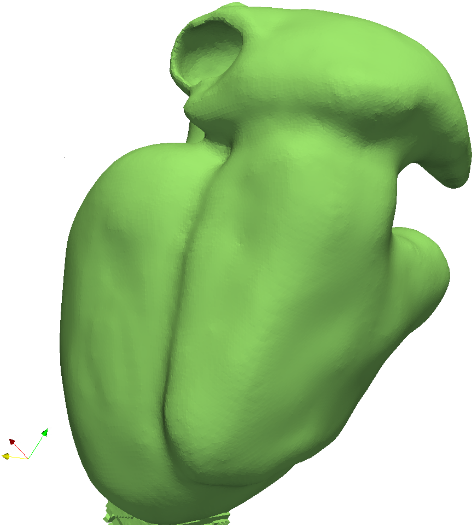
original



smoothed

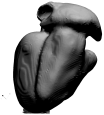


overlay

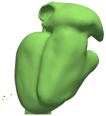


Smoothed:

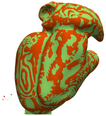
Visual results Example 1:



original



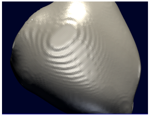
smoothed



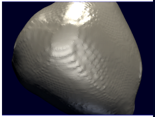
overlay

Overlay:

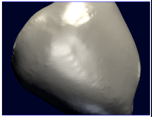
Visual results Example 1:



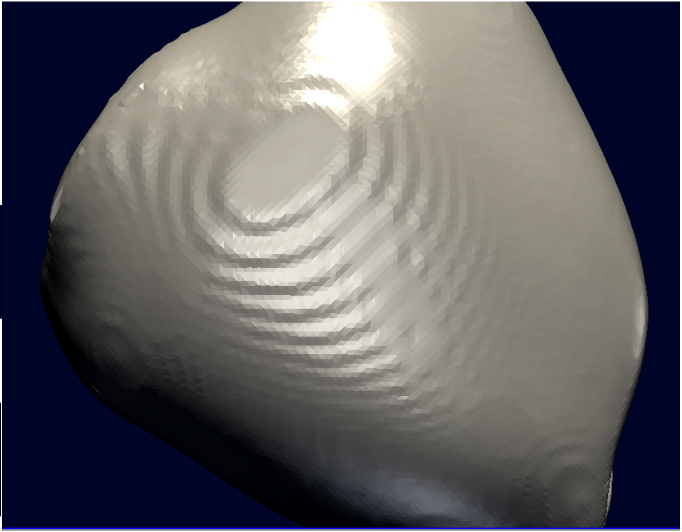
original



smoothed

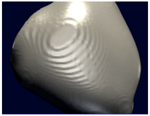


overlay

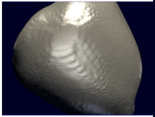


Oscillations:

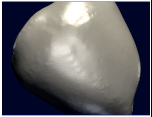
Visual results Example 1:



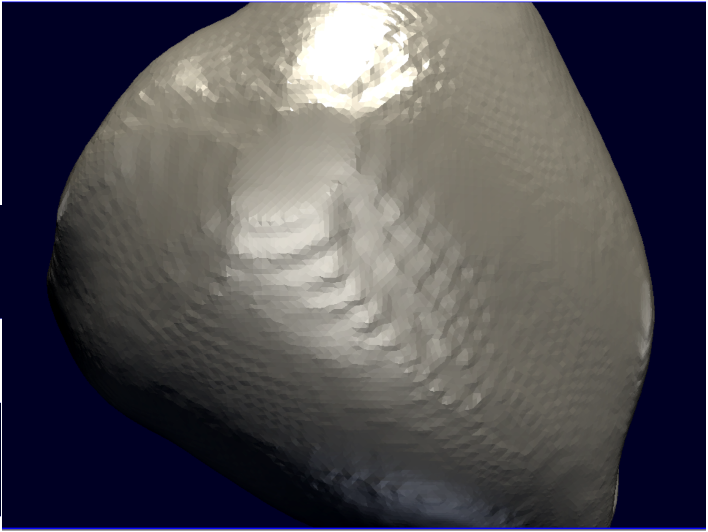
original



smoothed

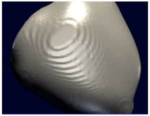


overlay

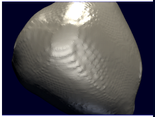


Smoothed:

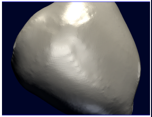
Visual results Example 1:



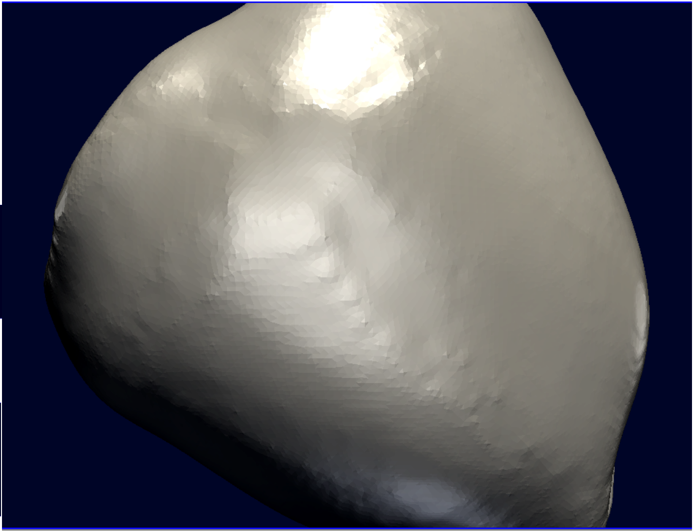
original



smoothed



overlay



Overlay:

Quantitative results

Example 2: Outer iterations 2, inner iterations 200, $\alpha = 1/3$.

Quantitative results

Example 2: Outer iterations 2, inner iterations 200, $\alpha = 1/3$.

Skewness P: Volume ratio to circumscribed ball.

Quantitative results

Example 2: Outer iterations 2, inner iterations 200, $\alpha = 1/3$.

Skewness P: Volume ratio to circumscribed ball.

Percentiles of P	1%	5%	10%
Original mesh	0.0900	0.2350	0.3348
First iteration	0.1136	0.2360	0.3195
Second iteration	0.0907	0.1999	0.2761

Quantitative results

Example 2: Outer iterations 2, inner iterations 200, $\alpha = 1/3$.

Skewness P: Volume ratio to circumscribed ball.

Percentiles of P	1%	5%	10%
Original mesh	0.0900	0.2350	0.3348
First iteration	0.1136	0.2360	0.3195
Second iteration	0.0907	0.1999	0.2761

Determinant ratio Θ: Measures change of size and orientation.

Quantitative results

Example 2: Outer iterations 2, inner iterations 200, $\alpha = 1/3$.

Skewness P: Volume ratio to circumscribed ball.

Percentiles of P	1%	5%	10%
Original mesh	0.0900	0.2350	0.3348
First iteration	0.1136	0.2360	0.3195
Second iteration	0.0907	0.1999	0.2761

Determinant ratio Θ: Measures change of size and orientation.

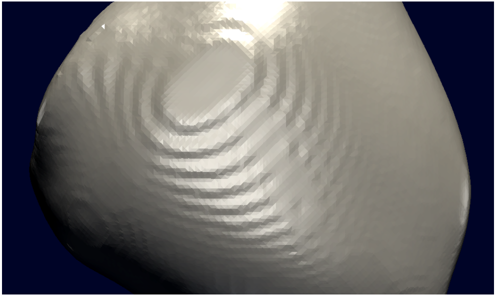
Percentiles of Θ	1%	5%
1. Iteration	0.6486	0.7407
2. Iteration	0.5280	0.6470

Outlook

- Update of interior points.

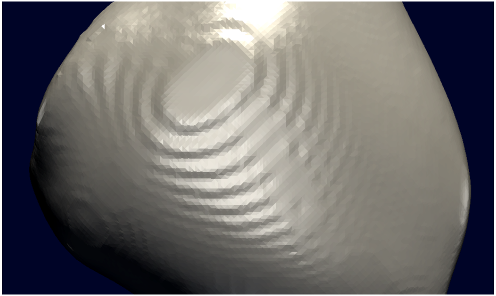
Outlook

- Update of interior points.



Outlook

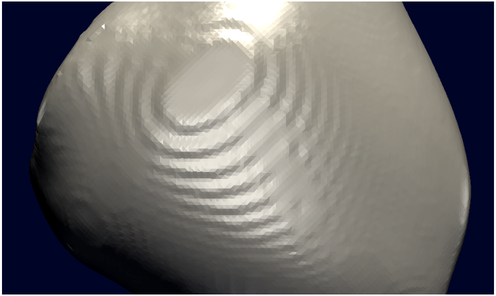
- Update of interior points.



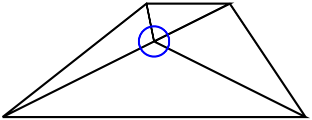
- More general constraints.

Outlook

- Update of interior points.

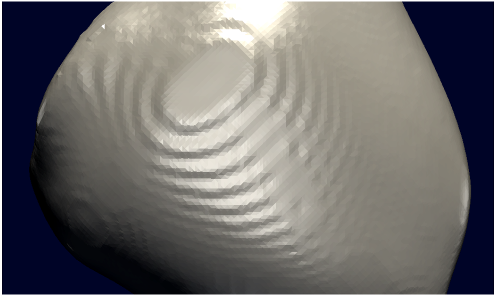


- More general constraints.

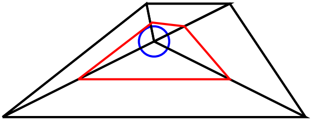


Outlook

- Update of interior points.



- More general constraints.



Thank you for your attention!

Thank you for your attention!

- M. Botsch and L. Kobbelt and M. Pauly and L. Alliez and B. Lévy, *Polygon mesh processing*, CRC press, Taylor and Francis, 2010

- A. Chambolle and T. Pock , *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of Mathematical Imaging and Vision, 40(1):120-145, 2010.

- A. Crozier, C. M. Augustin, A. Neic, A. J. Prassl, M. Holler, T. E. Fastl, A. Hennenmuth, K. Bredies, T. Kuehne, M. J. Bishop, S. A. Niederer, and G. Plank, *Image-based personalization of cardiac anatomy for coupled electromechanical modeling*, Annals of Biomedical Engineering, 44(1):58-70, 2016.

- B. M. Klingne and J. R. Shewchuk, *Aggressive tetrahedral mesh improvement*, Proceedings of the 16th International Meshing Roundtable: 3-23, 2008, Springer

- X. Lu and Z.Deng and W. Chen, *A Robust Scheme for Feature-Preserving Mesh Denoising*, IEEE Transactions on Visualization and Computer Graphics, 22 (3): 1181- 1194, 2016

- H. Zhang and C. Wu and J. Zhang and J. Deng, *Variational mesh denoising using total variation and piecewise constant function space*,IEEE Transactions on Visualization and Computer Graphics 21 (7): 873 - 886, 2015