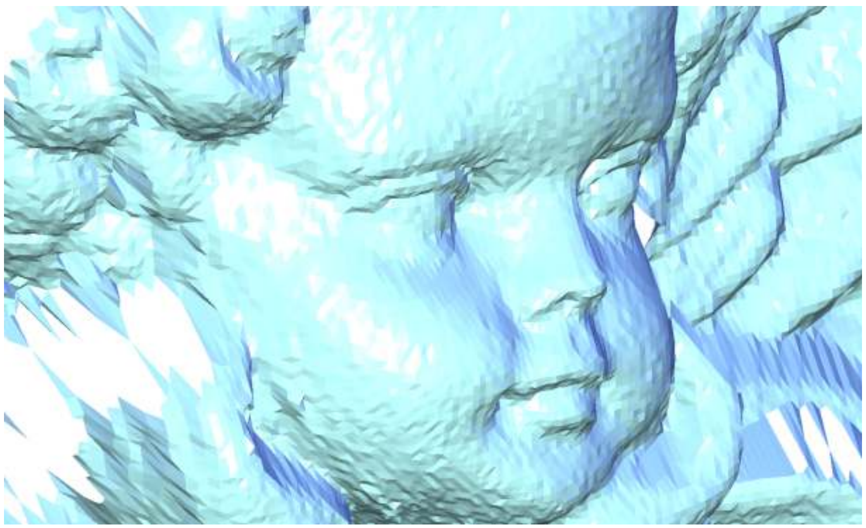
CS 523: Computer Graphics, Spring 2011

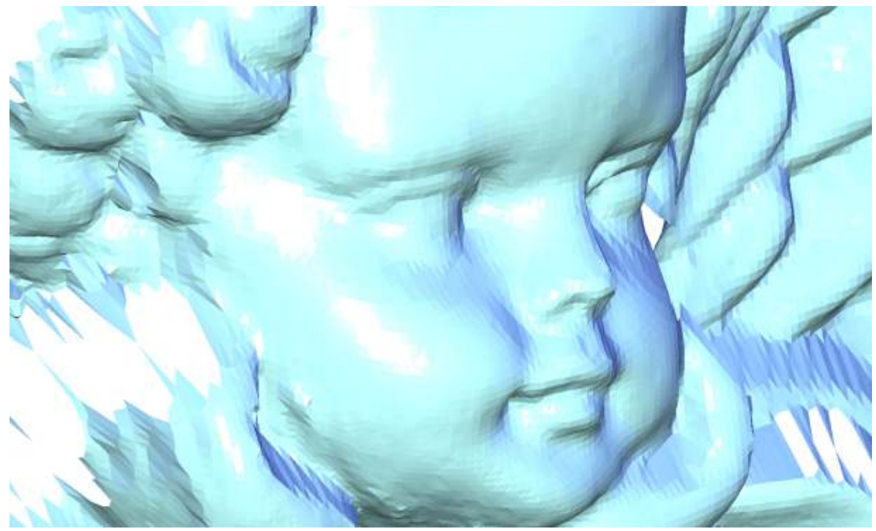
Shape Modeling

Laplacian mesh processing

Laplacian mesh optimization

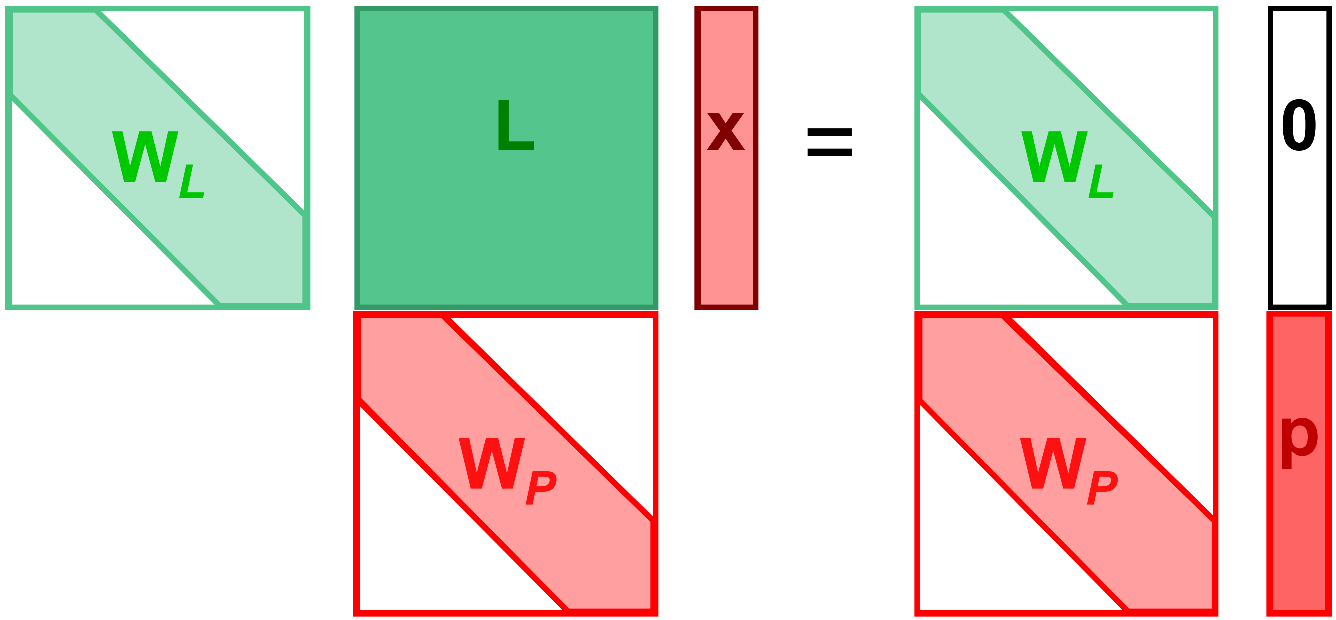
Reminder: mesh smoothing result





Laplacian mesh optimization

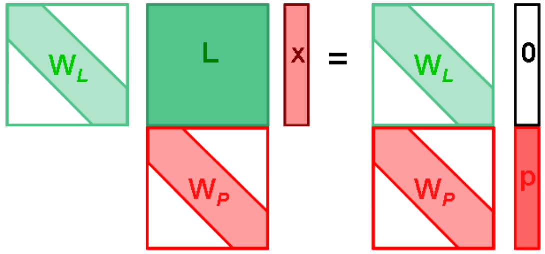
Reminder: mesh smoothing setup



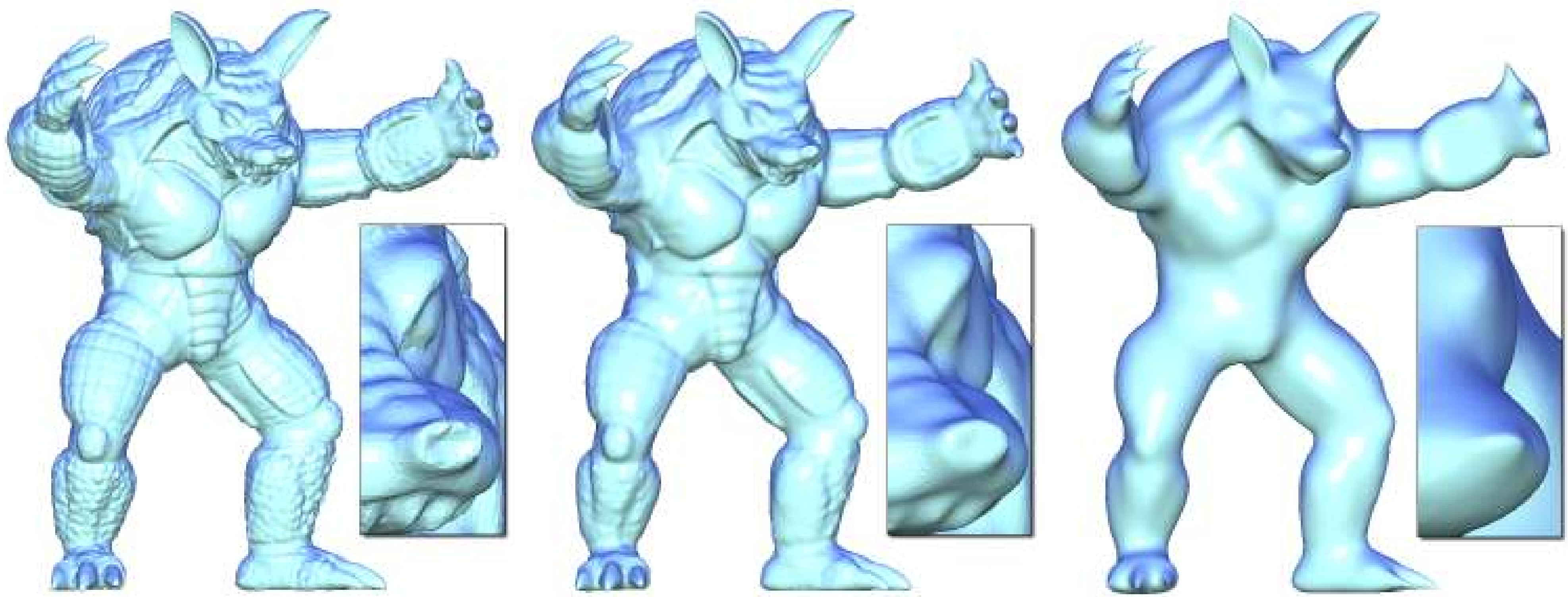
- **Mesh smoothing** $L = L_{cot}$ (outer fairness) or $L = L_{uni}$ (outer and inner fairness)
- Controlled by W_P and W_L (Intensity, Features)
- Least squares solve using normal equations

$$A^TA \; x = A^T b$$

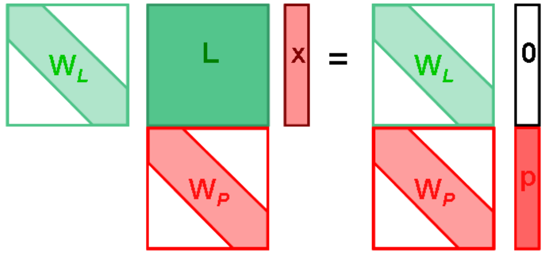
$$x = (A^TA)^{-1} \; A^T b$$



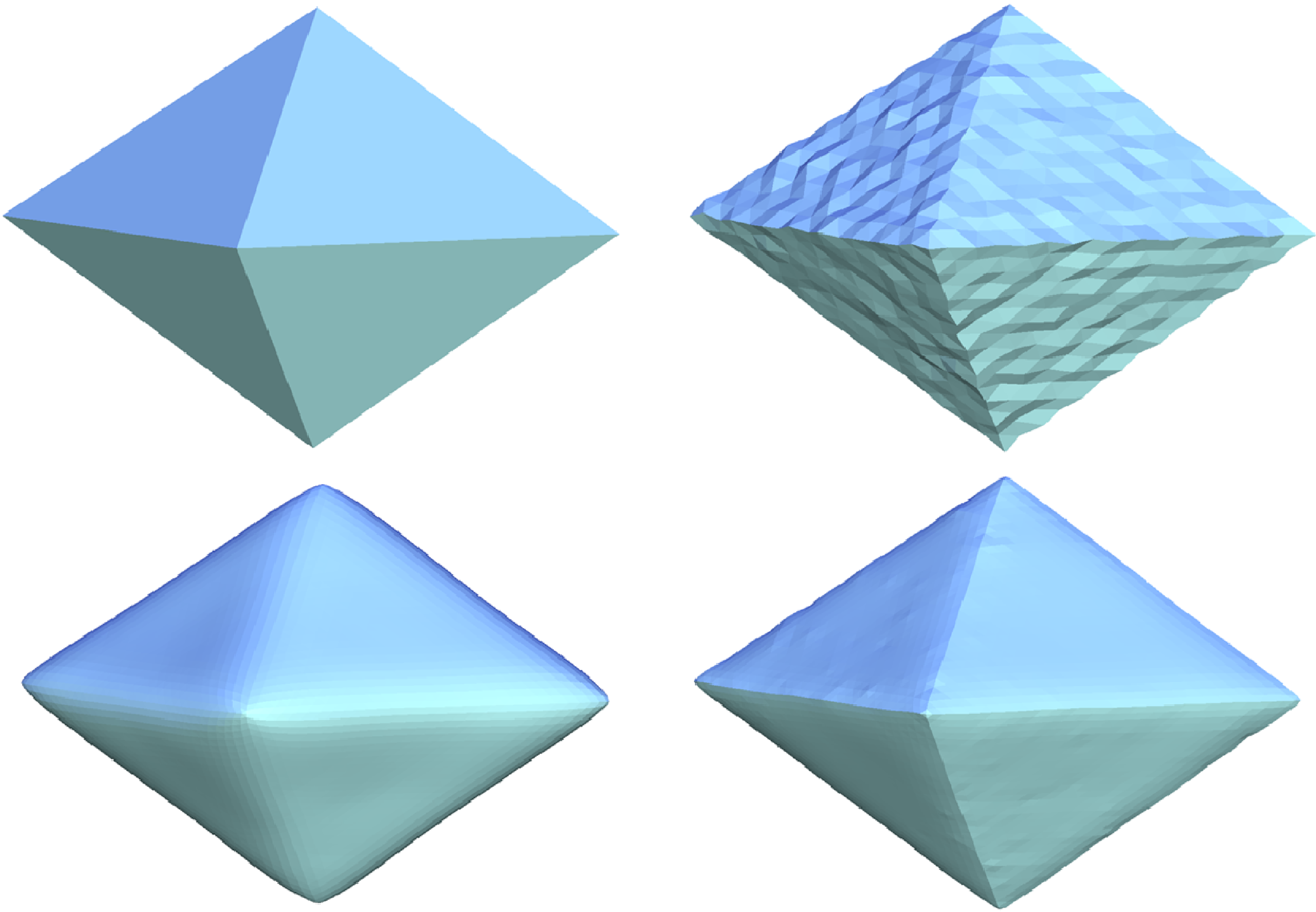
Using W_P



(a) original (173k) (c) cdf weights (s = 0.2) (f) cdf weights (s = 0.02)

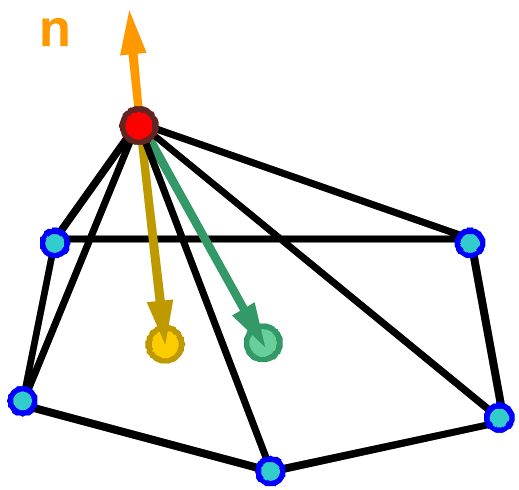


Using W_P and W_L

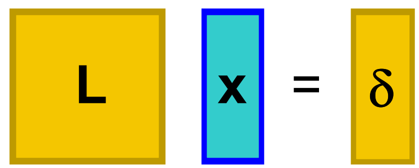


Laplacian Mesh Processing

- Discrete Laplacians



$$\boldsymbol{\delta}_i = \mathbf{x}_i - \frac{1}{\sum_{(i,j)\in E} w_{ij}} \sum_{(i,j)\in E} w_{ij}\mathbf{x}_j$$

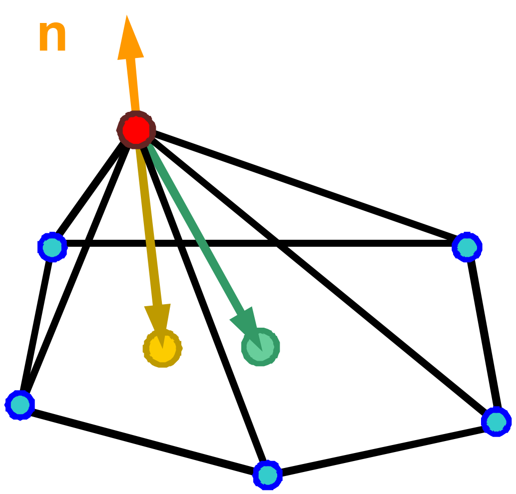


$\delta_{\mathbf{uniform}}$: $w_{ij} = 1$

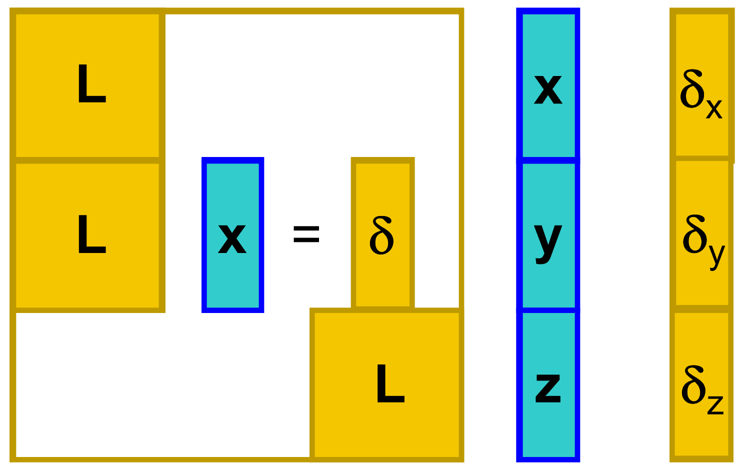
$\delta_{\mathbf{cotangent}}$: $w_{ij} = \cot \alpha_{ij} + \cot \beta_{ij}$

Laplacian Mesh Processing

- Surface reconstruction



$$\boldsymbol{\delta}_i = \mathbf{x}_i - \frac{1}{\sum_{(i,j)\in E} w_{ij}} \sum_{(i,j)\in E} w_{ij}\mathbf{x}_j$$

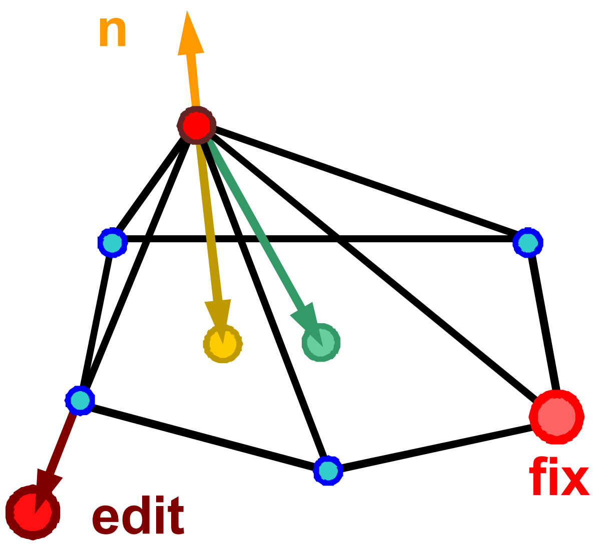


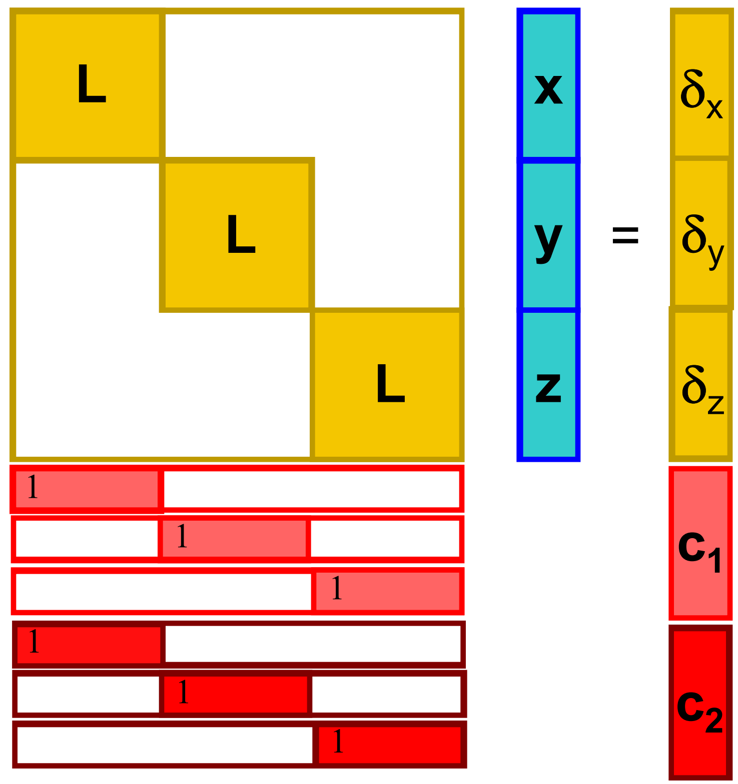
$\delta_{\mathbf{uniform}}$: $w_{ij} = 1$

$\delta_{\mathbf{cotangent}}$: $w_{ij} = \cot \alpha_{ij} + \cot \beta_{ij}$

Laplacian Mesh Processing

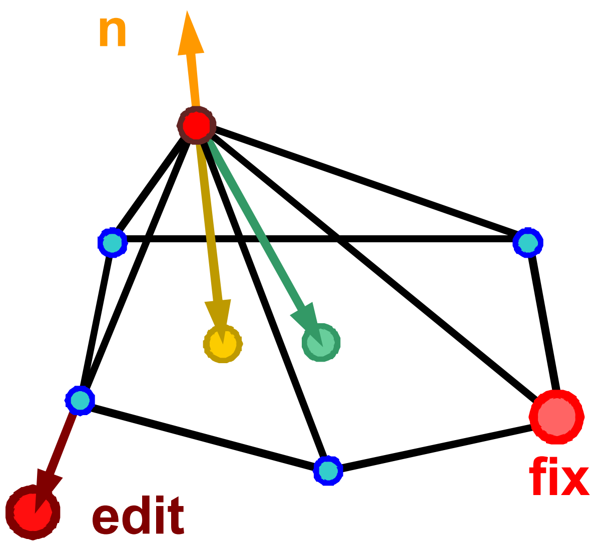
- Surface reconstruction + editing





Laplacian Mesh Processing

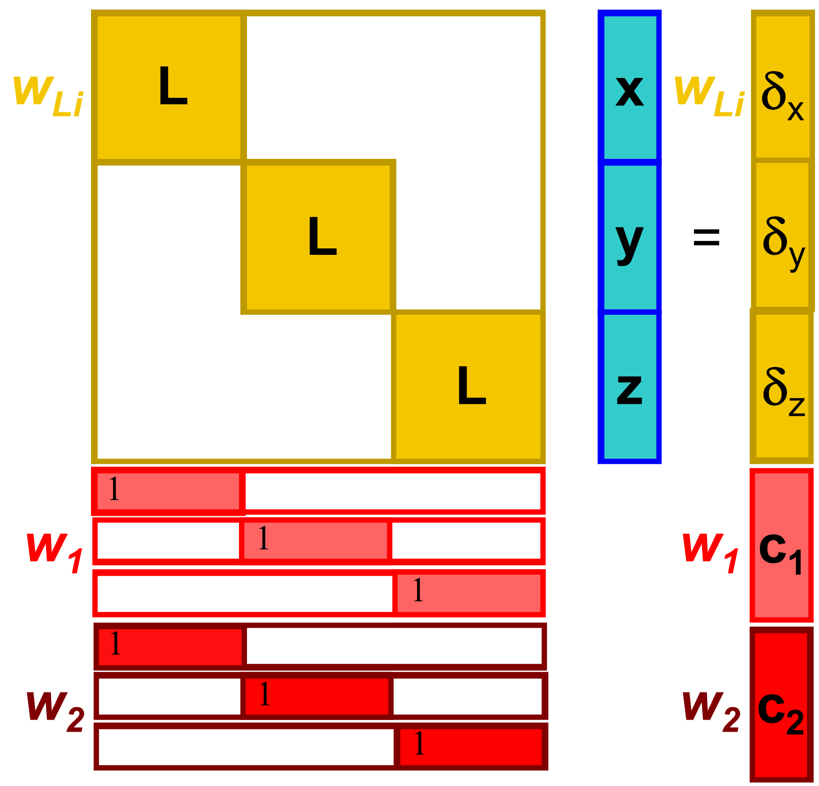
- Least-squares solution



Normal Equations

$$A^T A \, x = A^T b$$

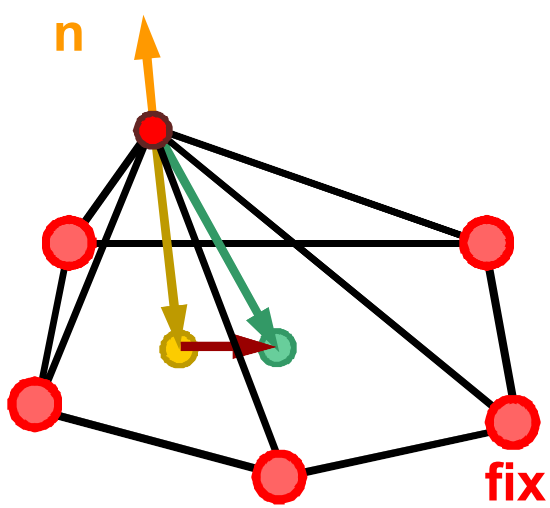
$$x = (A^T A)^{-1} A^T b$$

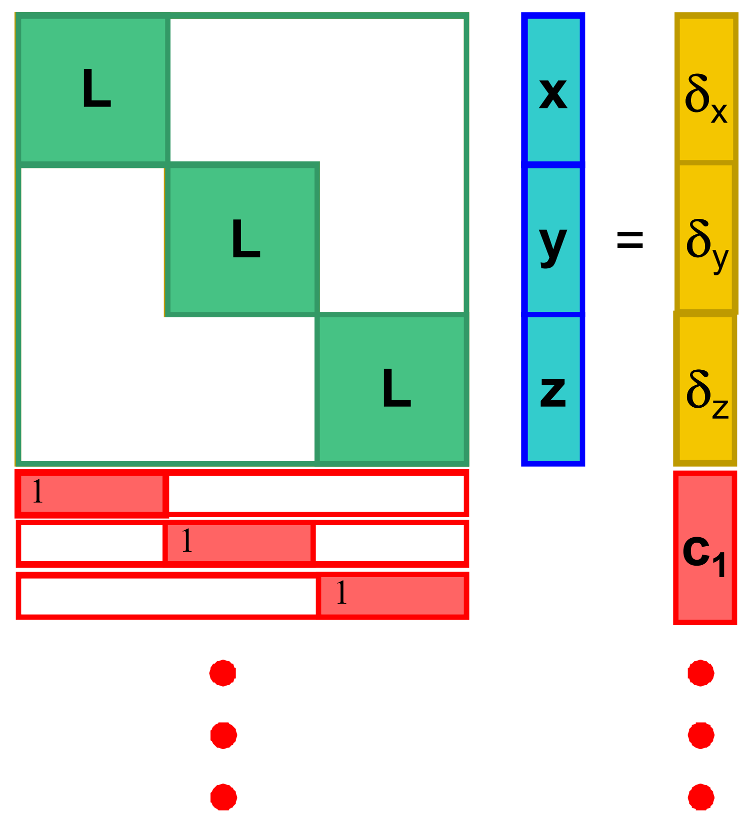


$$A \quad x \quad = \quad b$$

Laplacian Mesh Processing

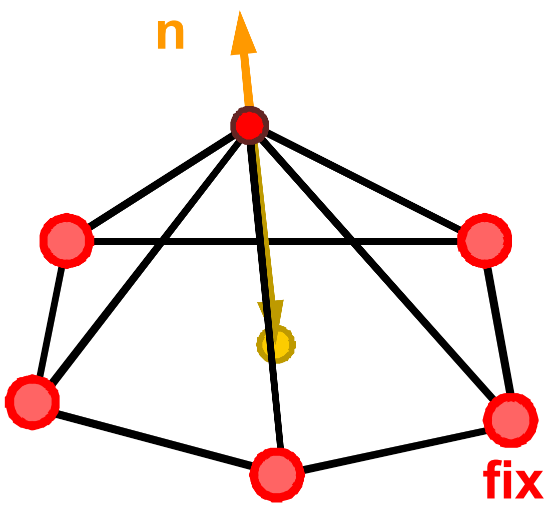
- Tangential smoothing

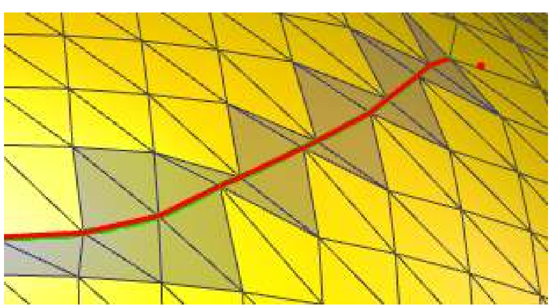


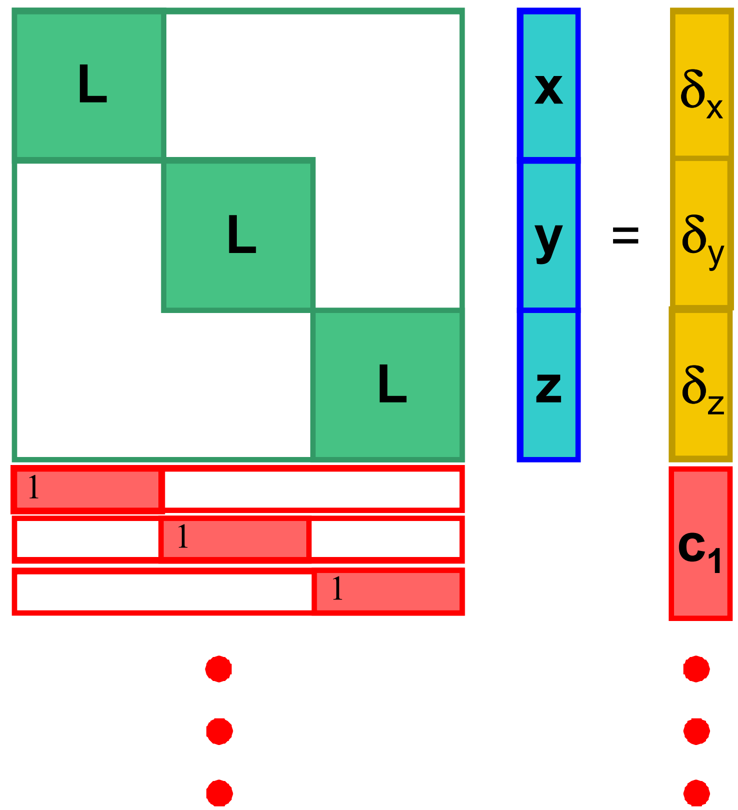


Laplacian Mesh Processing

- Tangential smoothing

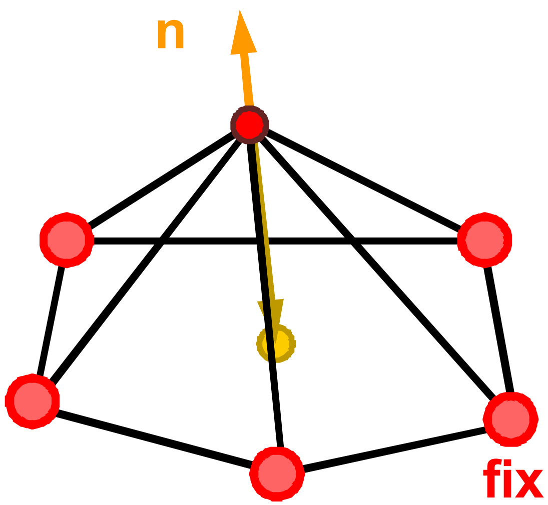


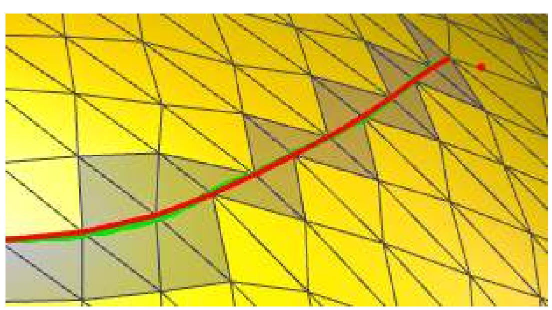


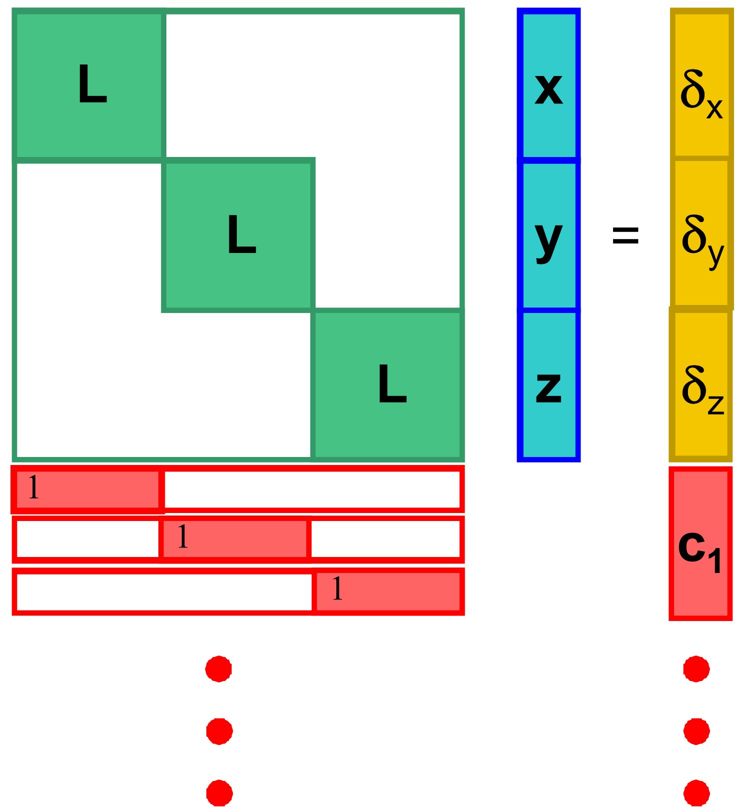


Laplacian Mesh Processing

- Tangential smoothing

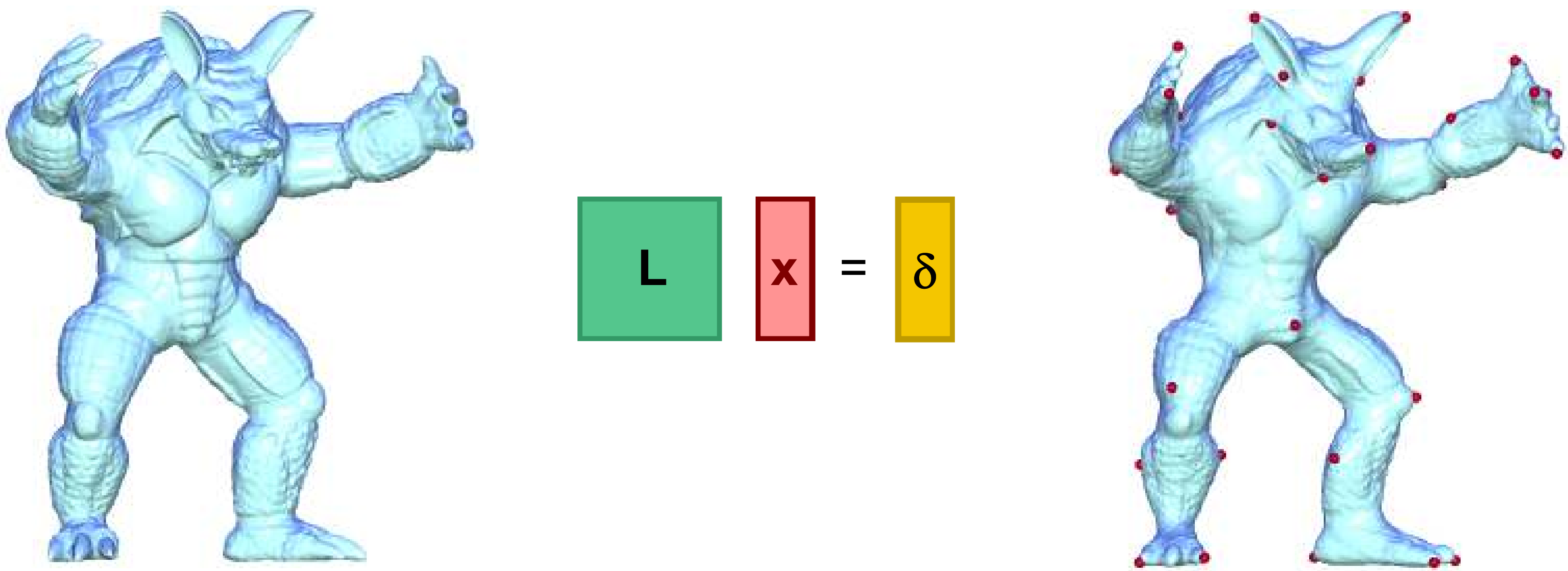






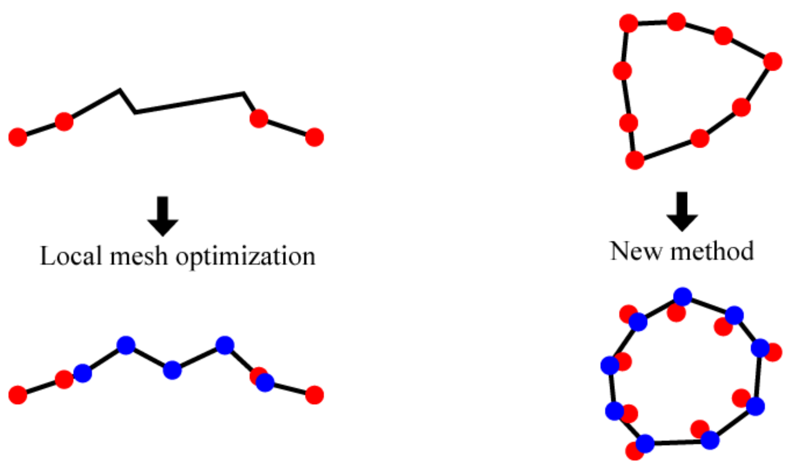
Idea

- Can we use such a system for **global** optimization ?



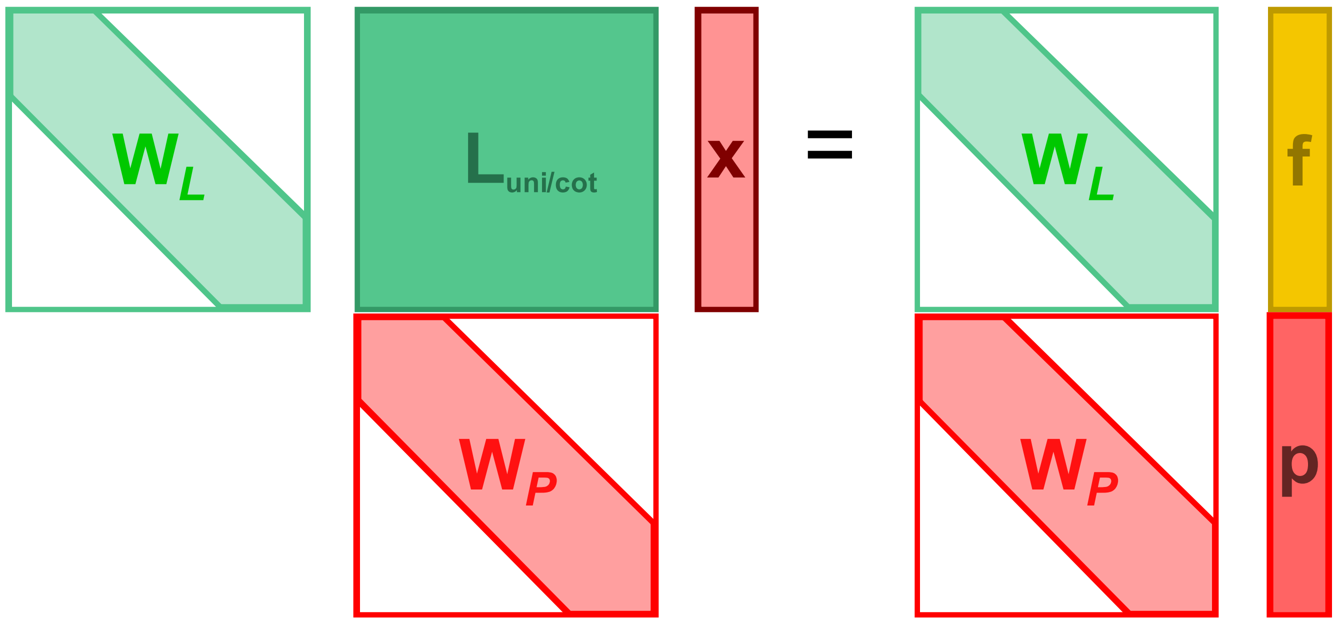
One solution

- All vertices are (weighted) anchors



- Preserves global shape
- Uses existing LS framework
- Anchor + Laplacian weights determine result

Laplacian mesh processing framework



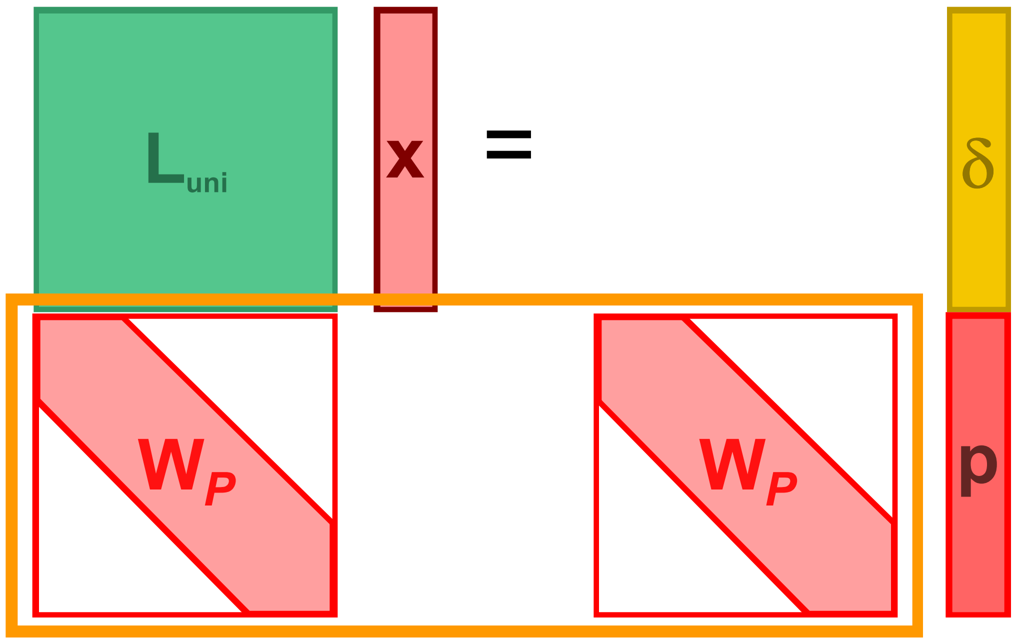
- **Detail preserving tri shape optimization** for $L = L_{uni}$ and $f = \delta_{cot}$ (similar to local optimization)
- **Mesh smoothing** $L = L_{cot}$ (outer fairness) or $L = L_{uni}$ (outer and inner fairness) and $f = 0$

Application:
Triangle shape optimization

Global vertex relocation

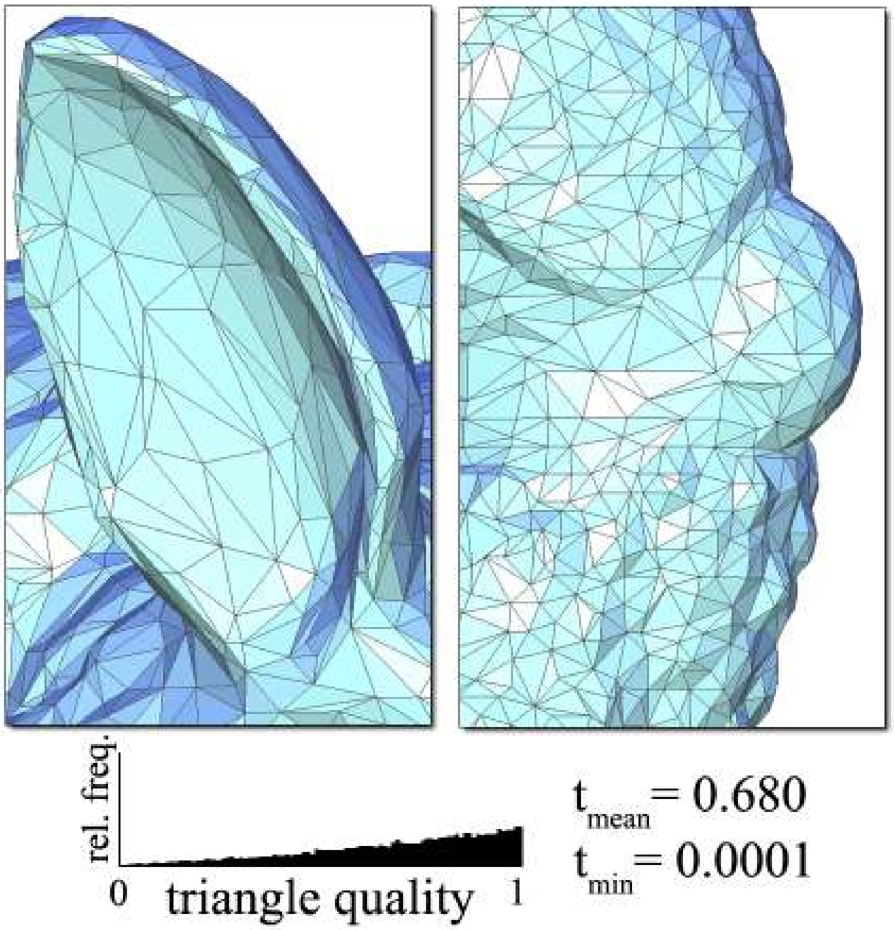
Triangle shape Optimization

By global vertex relocation

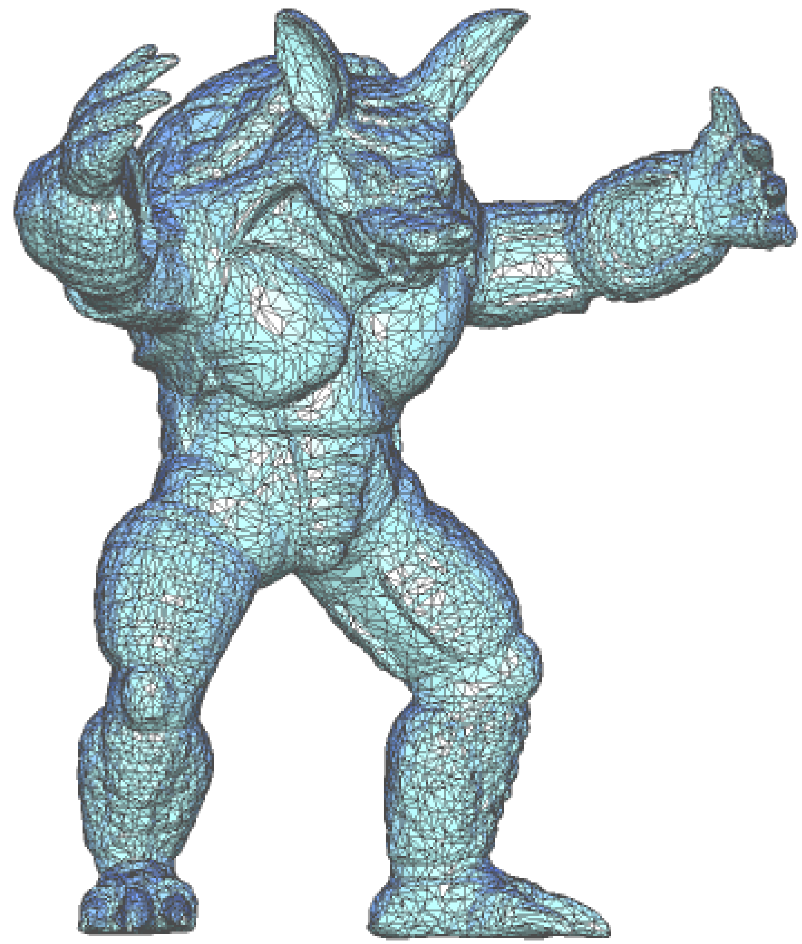


- **Detail preserving tri shape optimization** for $L = L_{uni}$ and $f = \delta_{cot}$ (similar to local optimization)

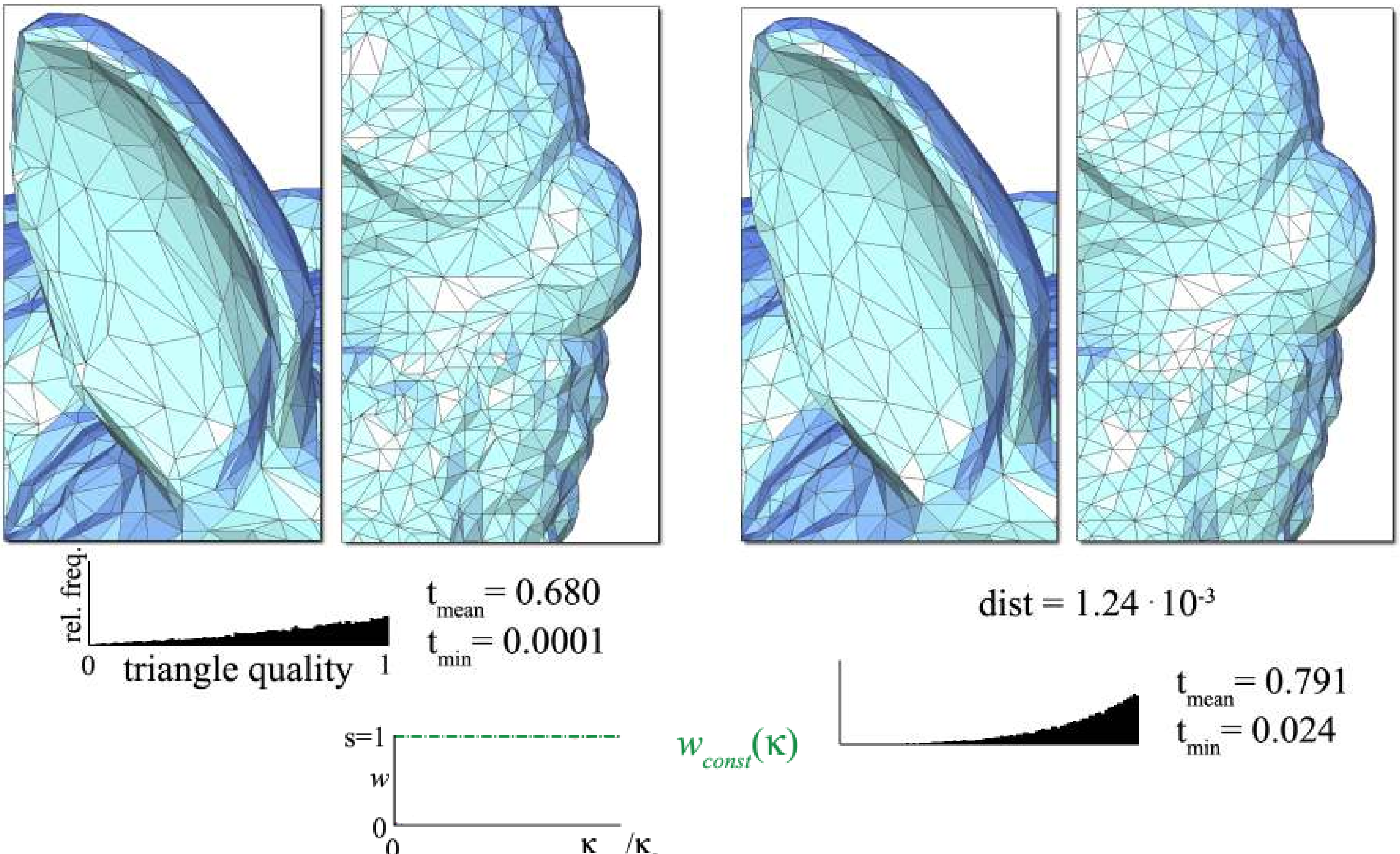
Positional Weights



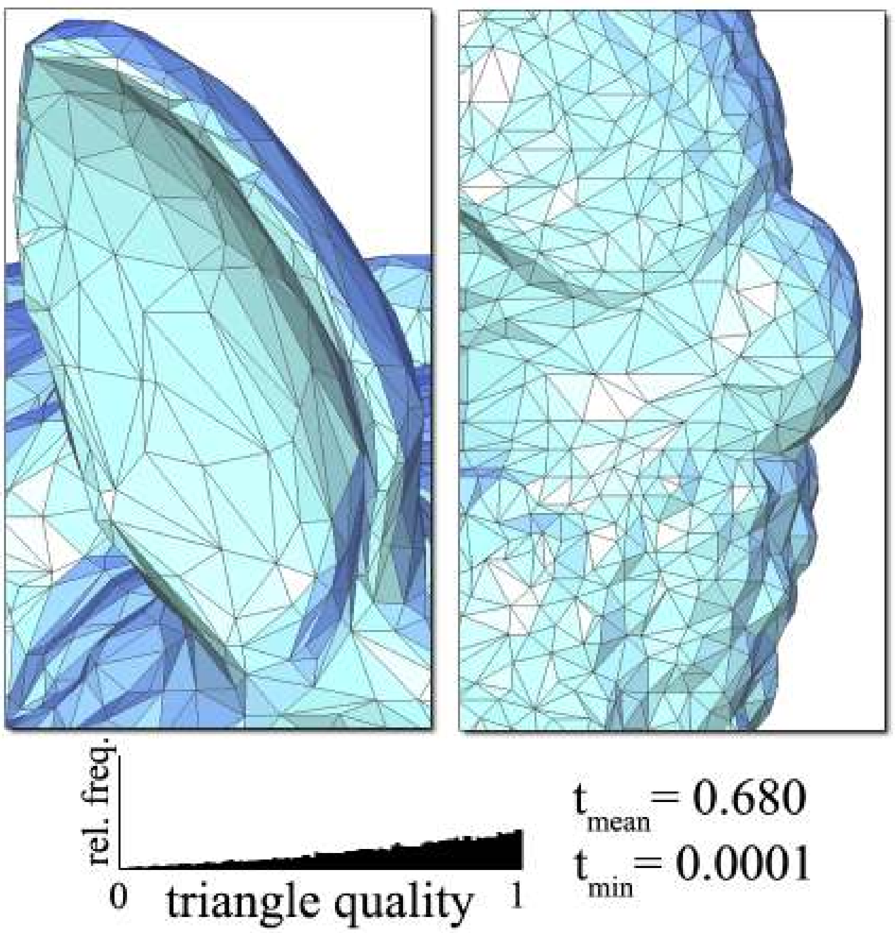
(a) original (17k)

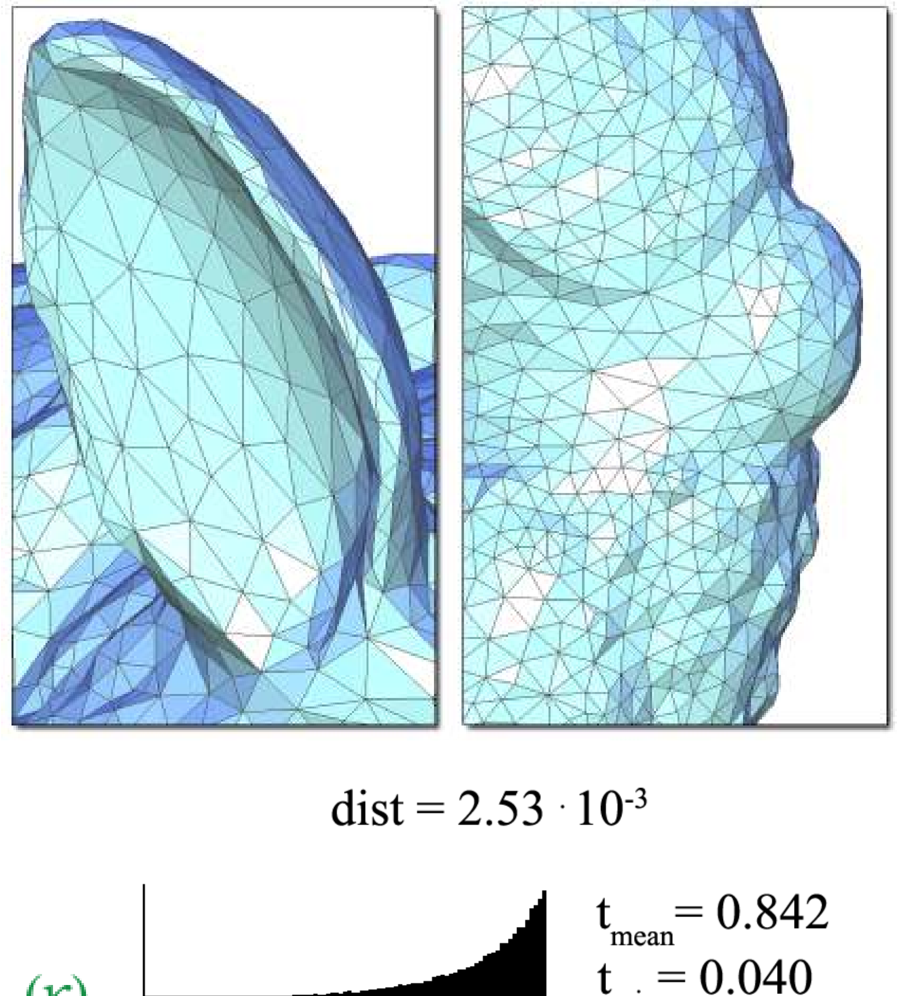


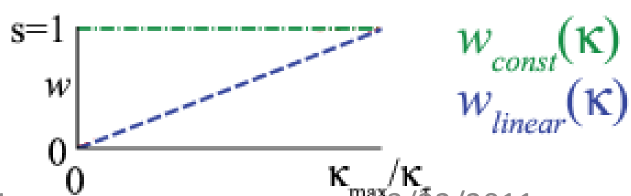
Constant Weights



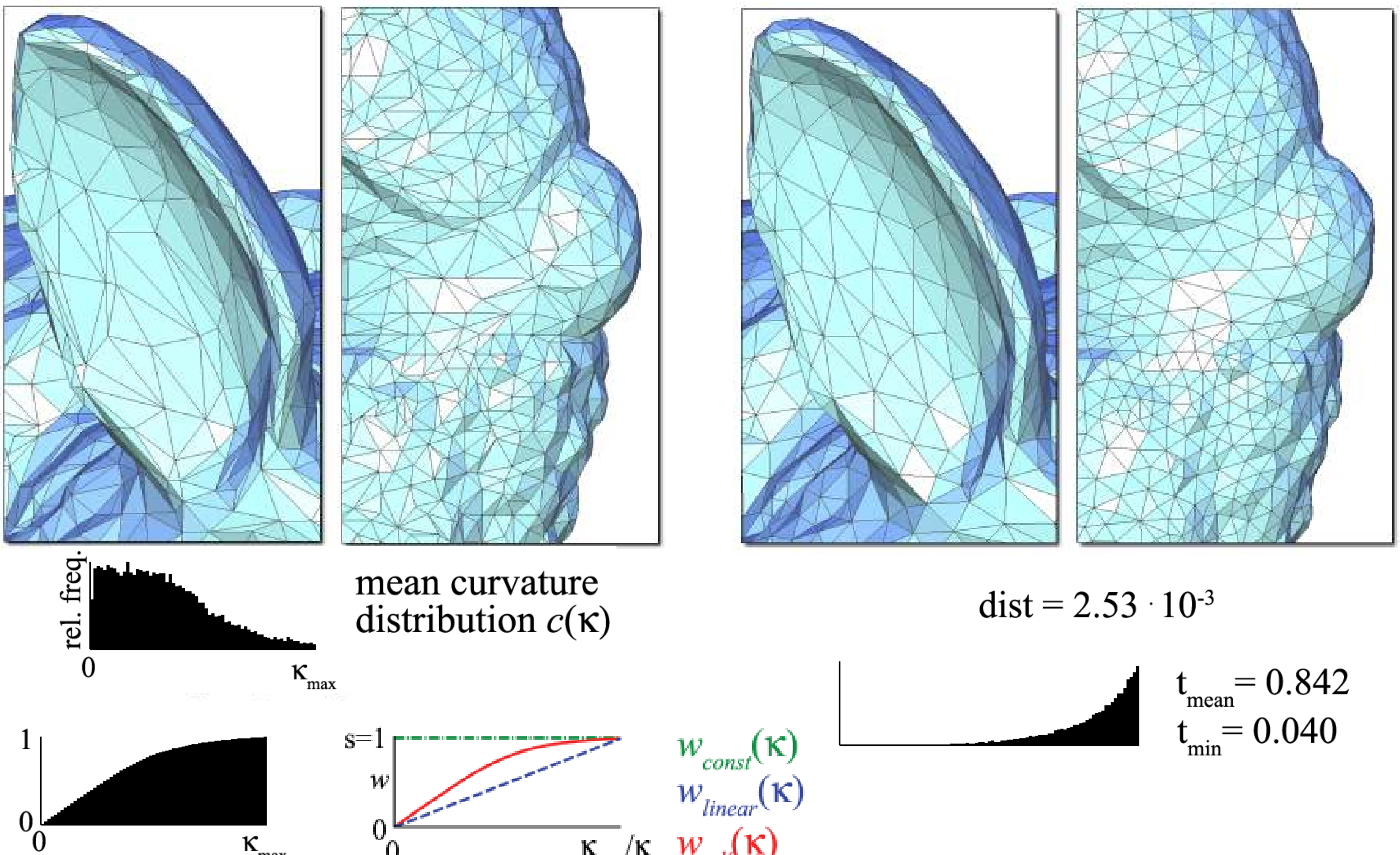
Linear Weights





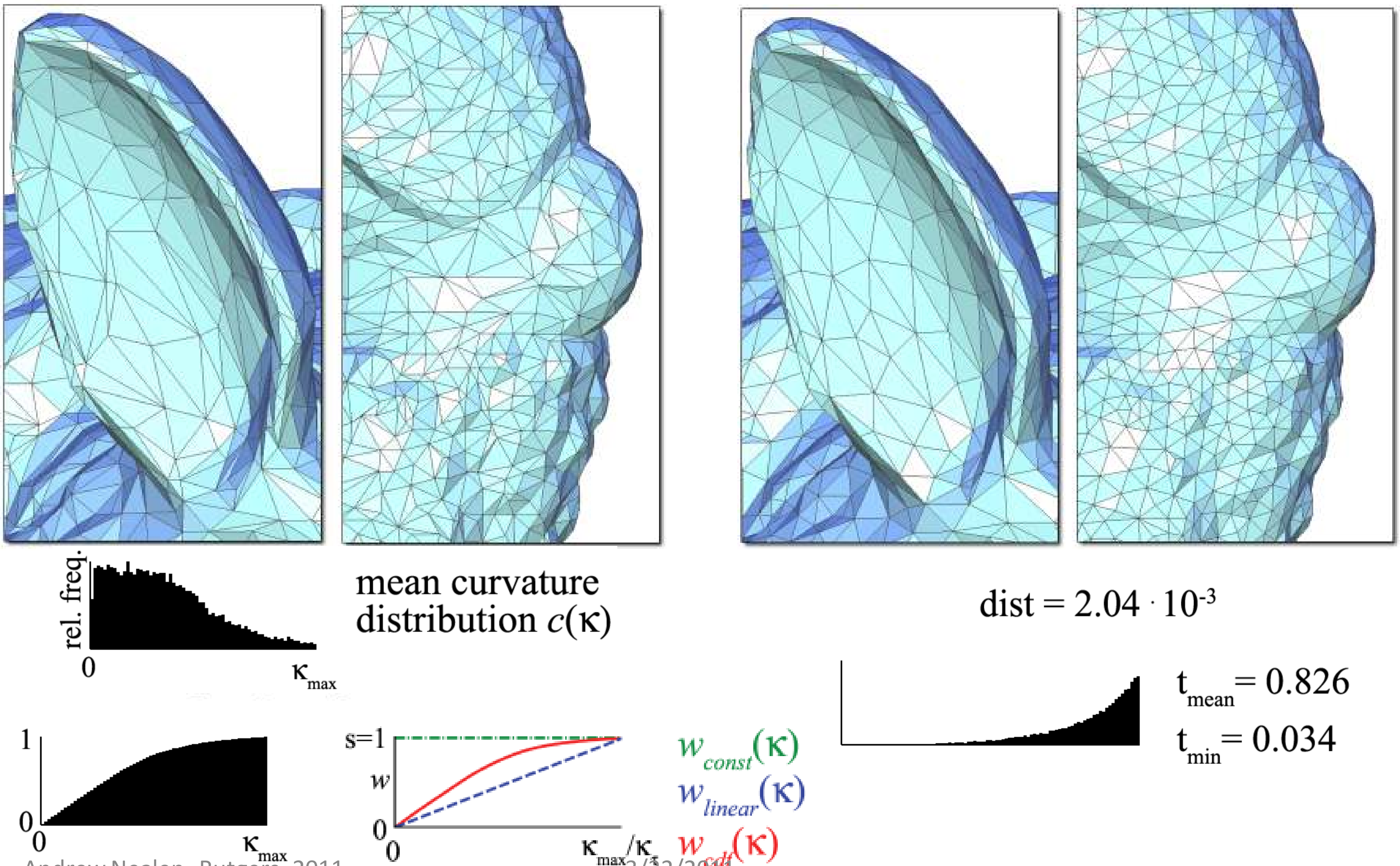


CDF Weights



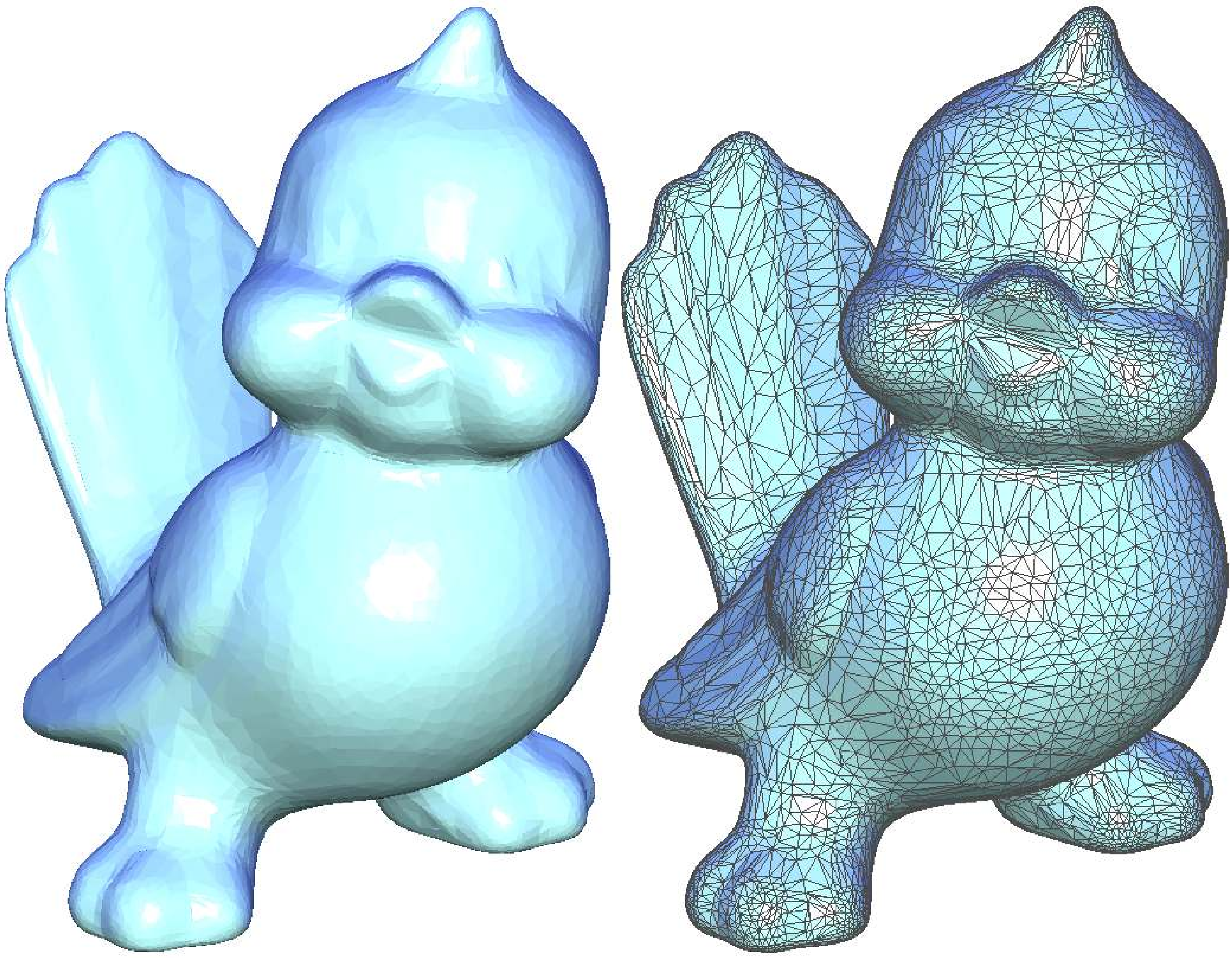
rel. freq.
0
κ_max
mean curvature distribution c(κ)
dist = 2.53 · 10^-3
t_mean = 0.842
t_min = 0.040
1
0
0
κ_max
s=1
w
0
0
κ_max/κ_τ
w_const(κ)
w_linear(κ)
w_cdf(κ)

CDF Weights

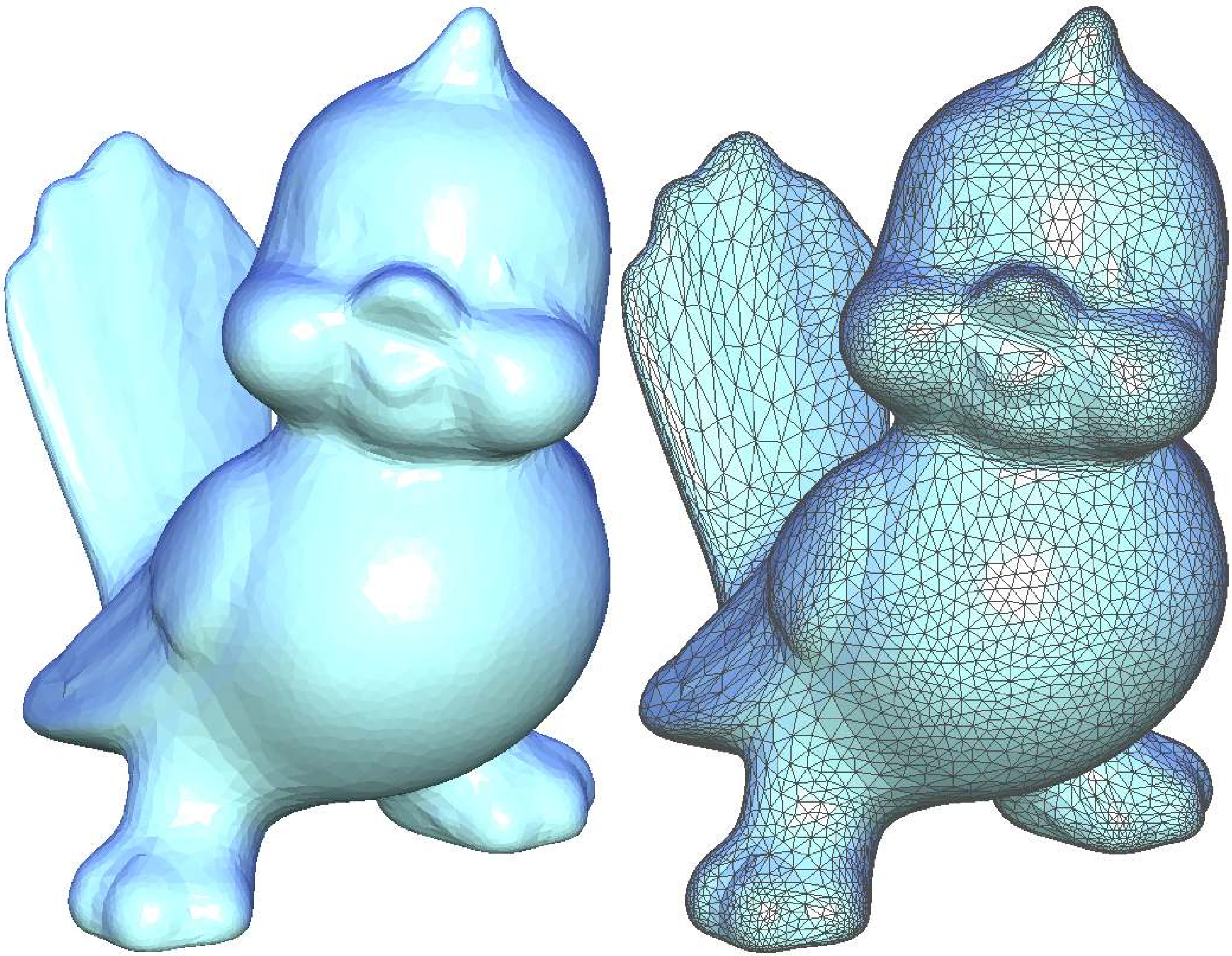


rel. freq.
0
κ_{max}
mean curvature distribution $c(\kappa)$
dist = $2.04 \cdot 10^{-3}$
$t_{mean} = 0.826$
$t_{min} = 0.034$
1
0
0
κ_{max}
s=1
w
0
0
$\kappa_{max}/\kappa_{\tau}$
$w_{const}(\kappa)$
$w_{linear}(\kappa)$
$w_{cdf}(\kappa)$

Original



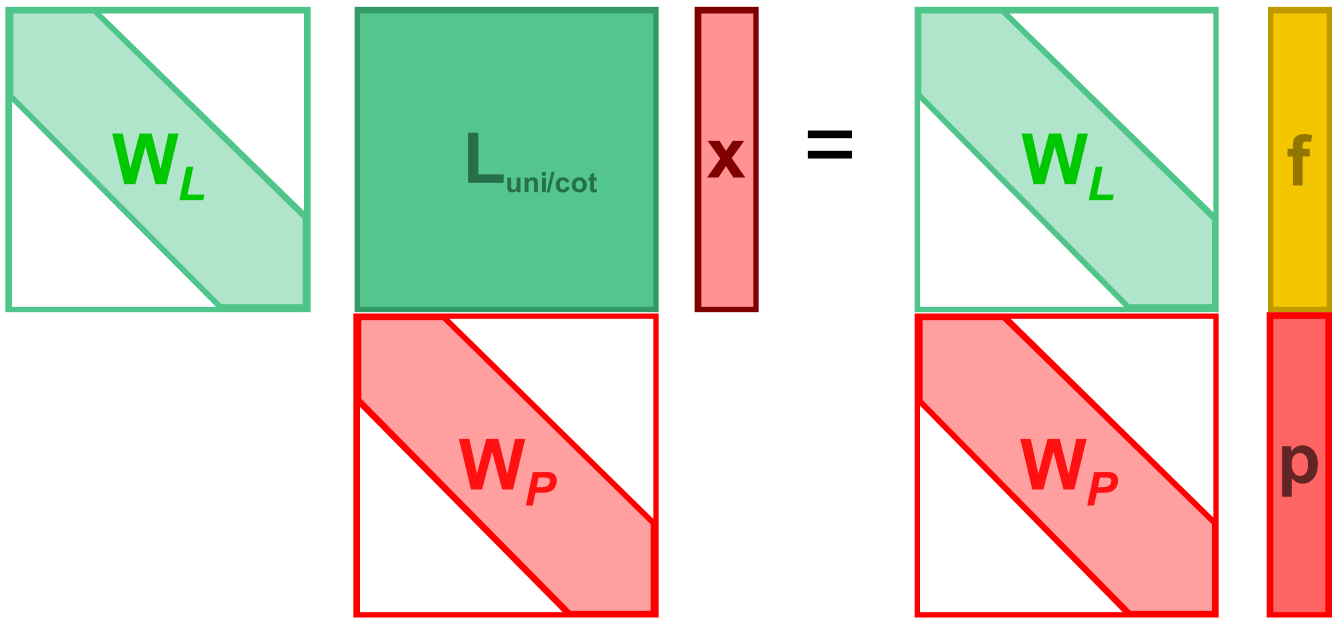
Tri Shape Optimization



Application:
Detail preserving mesh editing

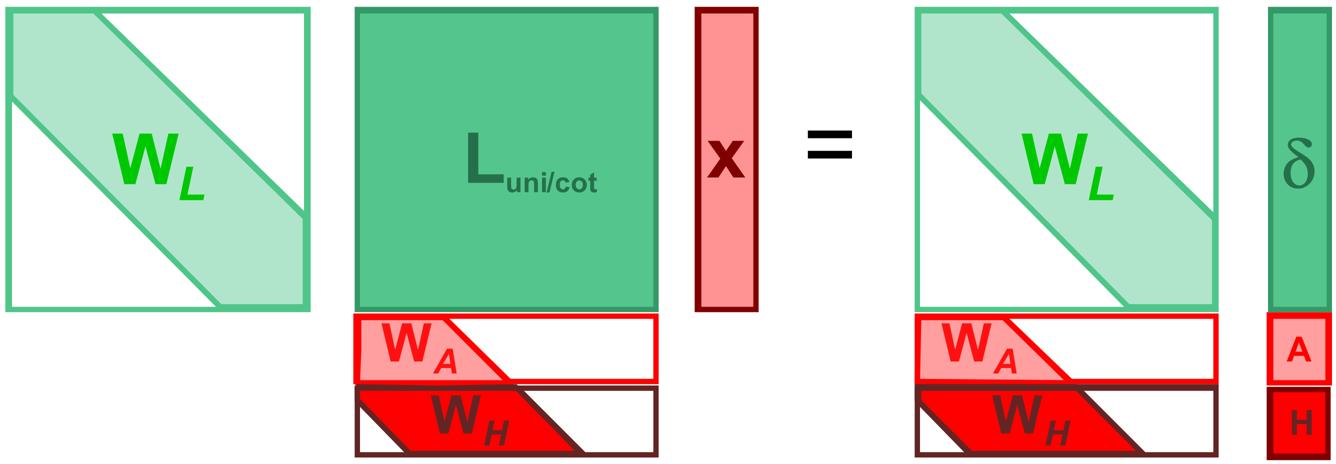
Retain local features
as much as possible

Laplacian mesh processing framework



- **Detail preserving mesh editing** for $L = L_{\text{uni } \mathbf{or} \text{ cot}}$ and $f = \delta_{\text{uni } \mathbf{or} \text{ cot}}$

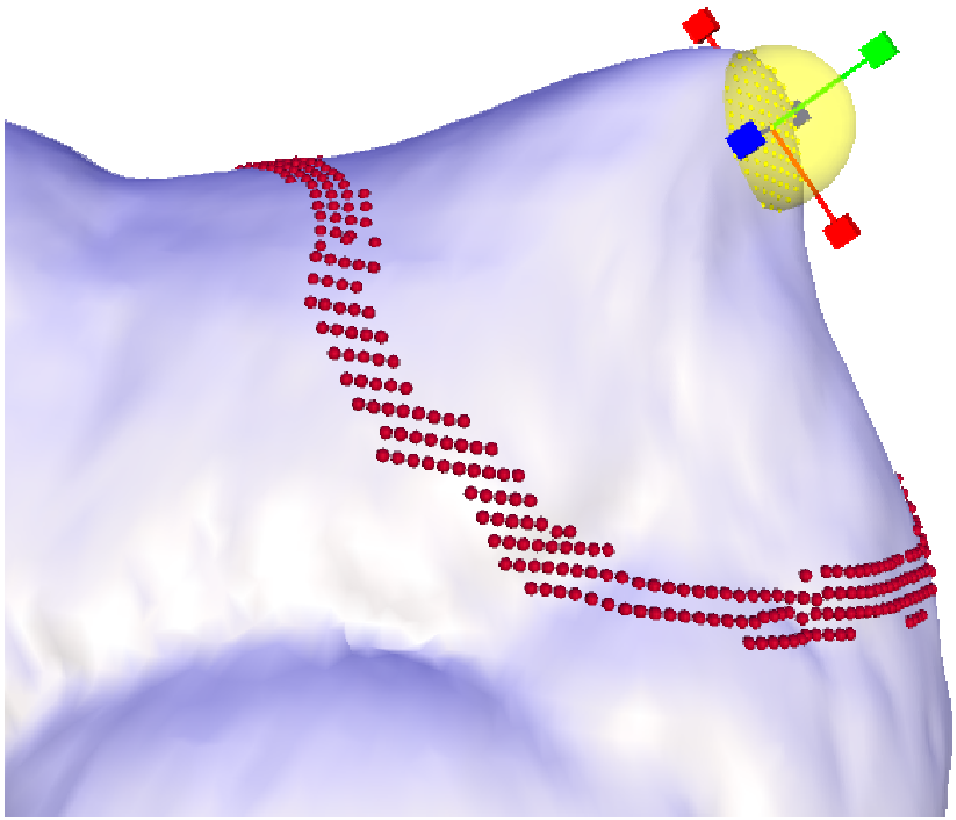
Laplacian surface editing framework



- **Detail preserving mesh editing** for $L = L_{\text{uni } \mathbf{or} \text{ cot}}$ and $f = \delta_{\text{uni } \mathbf{or} \text{ cot}}$
- using a **subset** of the mesh, padded by anchor vertices **A** and using vertices **H** as the deformation control handle

Laplacian surface editing framework

- Region of interest (ROI) is bounded by a belt of static anchors
- Manipulation through handle vertices



Why local Laplacian coordinates?

- Local detail representation – enables **detail preservation** through various modeling tasks
- Representation with **sparse** matrices
- Efficient **linear** surface reconstruction