Frame-based Simulation of Complex Mechanical Structures

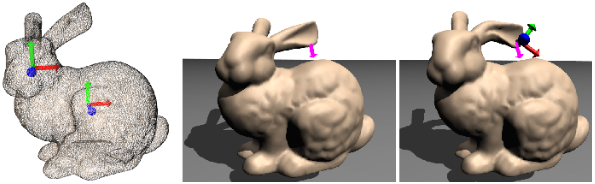
François Faure[1,2,3] Benjamin Gilles[4,2,3]
Guillaume Bousquet[1,2,3] Dinesh K. Pai[5]

[1]Grenoble Universités, [2]INRIA, [3]LJK – CNRS, [4]Tecnalia, [5]UBC

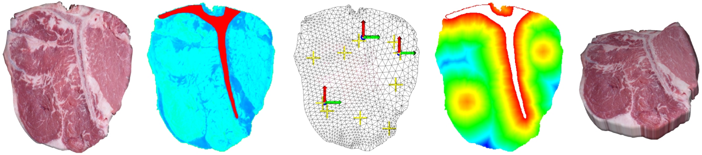
November 2012

Taken from:

- *Gilles, Bousquet, Faure, Pai*; Frame-based Elastic Models; *ACM Transactions on Graphics, 2011, 30 (2)*



- *Faure, Gilles, Bousquet, Pai*; Sparse Meshless Models of Complex Deformable Solids; *ACM Transactions on Graphics (SIGGRAPH'2011)*

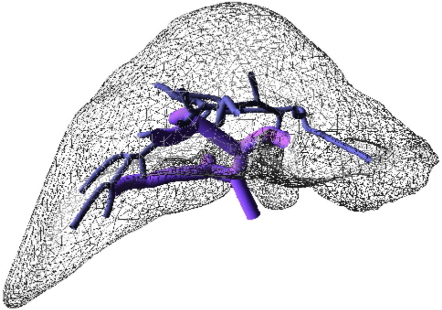


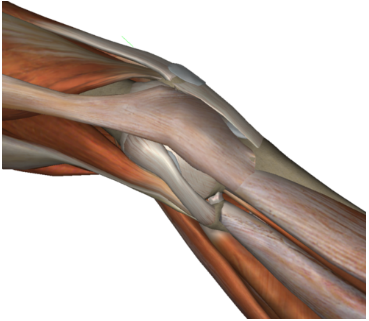
(a) *T-Bone Steak* **(b)** *Stiffness* **(c)** *Discretization* **(d)** *Compliance distance* **(e)** *Deformation*

Overview

Motivation: Interactive deformation of complex models

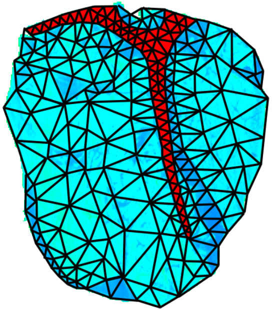
- Intricate materials





Limitations of Traditional FEM

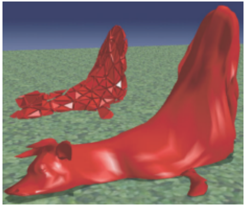
- Well-shaped elements
- One material per element
- Too many nodes !



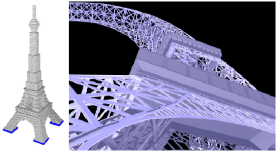
Previous Work: Interactive FEM

Embed complex geometry in a coarse mesh

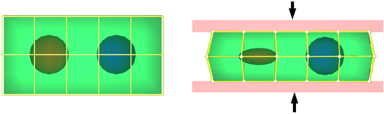
- Barycentric embedding
  [Müller 04, Sifakis 07]



- Pre-compute deformation modes
  [James 03, Barbič 05]



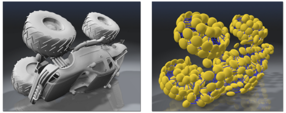
- Non-uniform elements
  [Kharevych 09, Nesme 09]



Previous Work: Frame-based Models

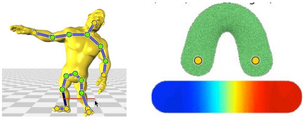
GMLS + elastons [Martin 10]

Skinning + geometry-based [Müller 11]

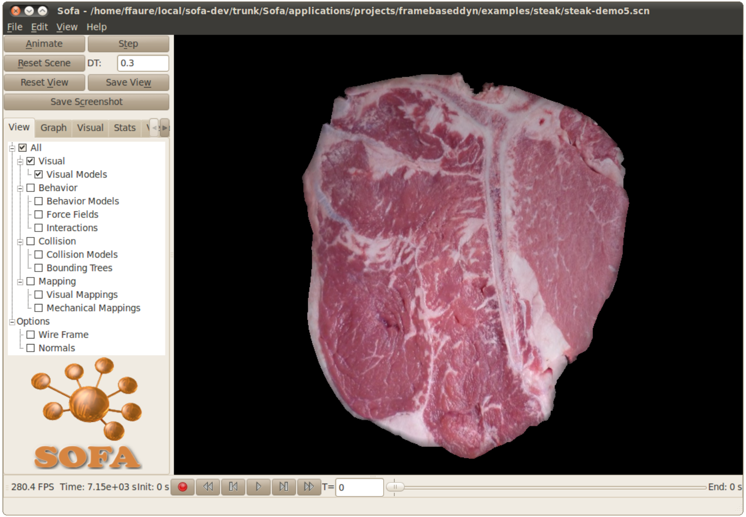


Skinning + deformation gradient [Gilles 11]

Skinning + Laplacian [Kavan 12]



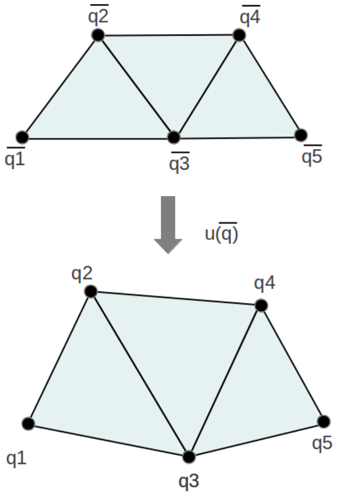
Interactive demo



Overview

Finite Element Discretization

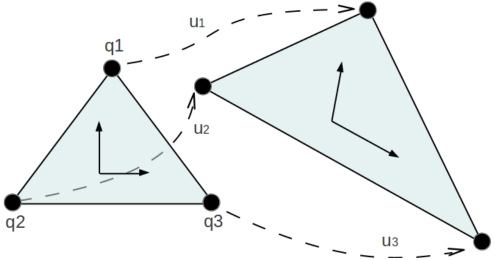
- Initial mesh nodes $\bar{\mathbf{q}} = \begin{pmatrix} \bar{q}_1 \\ \vdots \\ \bar{q}_n \end{pmatrix}$
- Node displacements $\bar{\mathbf{u}} = \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}$
- Displaced mesh nodes $\mathbf{q} = \bar{\mathbf{q}} + \mathbf{u}$



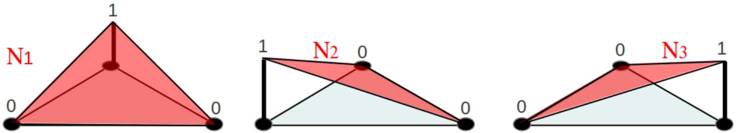
Element Interpolation

- Displacement field

$$u(p) = \sum_i N_i(p) u_i$$



- Shape functions
  - $N_i(p_i) = 1$
  - $N_i(p_{j \neq i}) = 0$

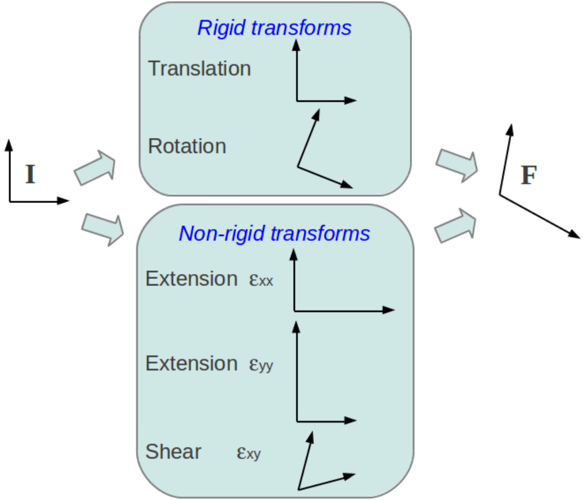


Deformation

- Deformation Gradient

$$\mathbf{F}(p) = \frac{dq(p)}{d\bar{q}}$$

- Strain measurements:
  - Cauchy: $\epsilon = (\mathbf{F} + \mathbf{F}^T)/2$
  - Green-Lagrange: $\epsilon = (\mathbf{F}^T\mathbf{F} - \mathbf{I})/2$
  - *etc.*



Elastic Forces

- Deformation energy rate
  - material properties (Hooke, StVenant-Kirchhoff, Mooney-Rivlin, etc.)

$$dW(\epsilon)$$

- Total deformation energy: integrate across the volume

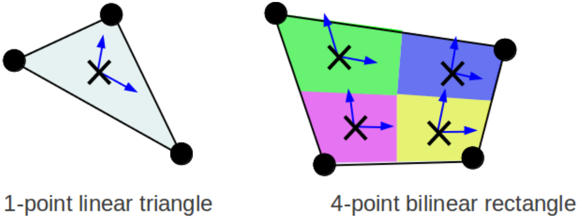
$$W = \int_{\mathcal{V}} dW(p)$$

- Forces derive from the potential energy:

$$\mathbf{f} = -\frac{\partial W}{\partial \mathbf{q}}$$

Space Integration

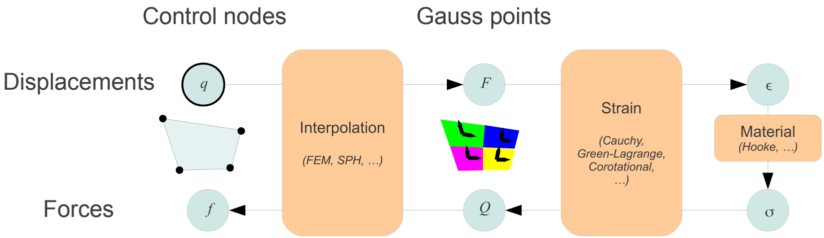
- Compute values at Gauss Points



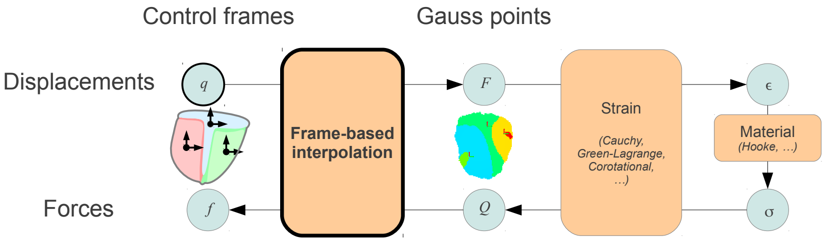
- Approximate integral as weighted sum

$$\mathbf{f} = -\frac{\partial W}{\partial \mathbf{q}} \simeq -\sum_{p \in G} \frac{\partial W(p)}{\partial \mathbf{q}} \mathcal{V}_p$$

Summary: pipeline

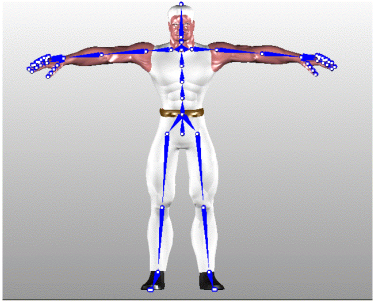
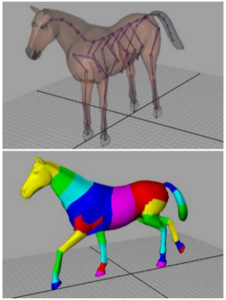
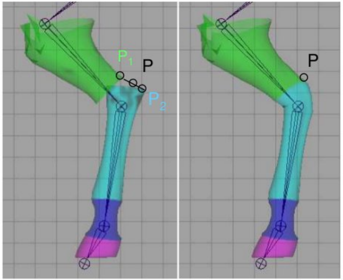


Summary: pipeline



An alternative deformation field: Computer Graphics Skinning

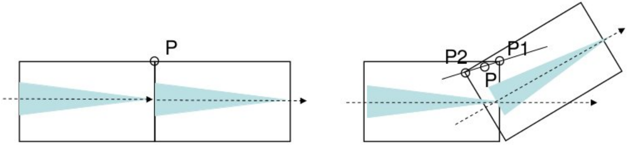
- attach a deformable body to an articulated skeleton

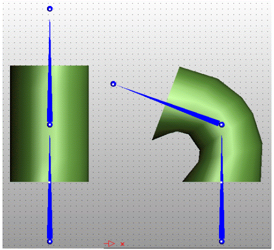
- Continuous deformation field
- No volumetric mesh, arbitrarily few control nodes
- Allows to derive continuous media mechanics ?

Linear Blend Skinning

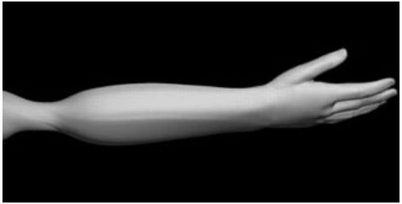
- Linear Blend Skinning: blend rigid motions



$$\mathbf{P} = w1^{*}\mathbf{P1} + w2^{*}\mathbf{P2}$$



- Artefacts with large deformations

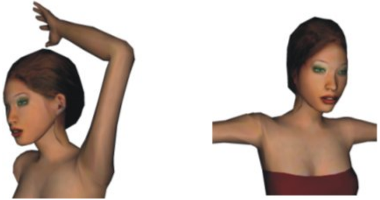


Non-linear Blend Skinning

- Dual Quaternion Skinning [Kavan07]

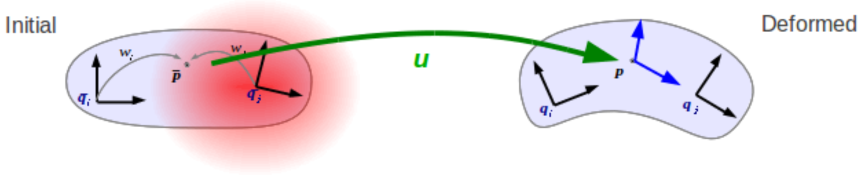
linear:

non-linear:



Frame-based discretization

- Replace polygon interpolation with rigid motion blending

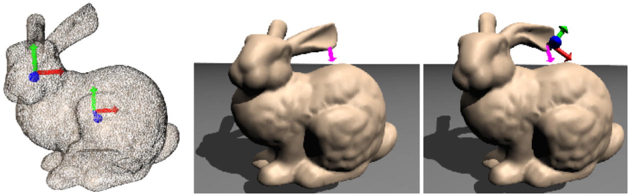


- Same pipeline as FEM
- Shape functions ?
- Integration points ?

Overview

Frame-based Elastic Models

Gilles, Bousquet, Faure, Pai; Frame-based Elastic Models; *ACM Transactions on Graphics, 2011, 30 (2)*

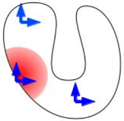


Shape Functions

- Harmonic weights
  [Joshi07]

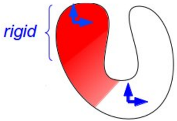
- + Smooth
- + Interpolating
- − Slow convergence

- Weights based on geodesic distance



- + Fast frame insertion
- + Tunable support
- − Approximating

- Custom weights
  e.g. mix rigid / deformable



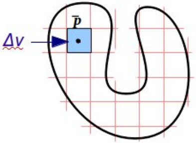
- + No numerical stiffening

Space integration

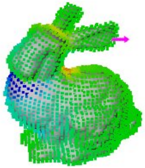
Spatial integration :

$$W = \int_V W(\bar{\boldsymbol{p}})dv \approx \sum W(\bar{\boldsymbol{p}})\Delta v$$

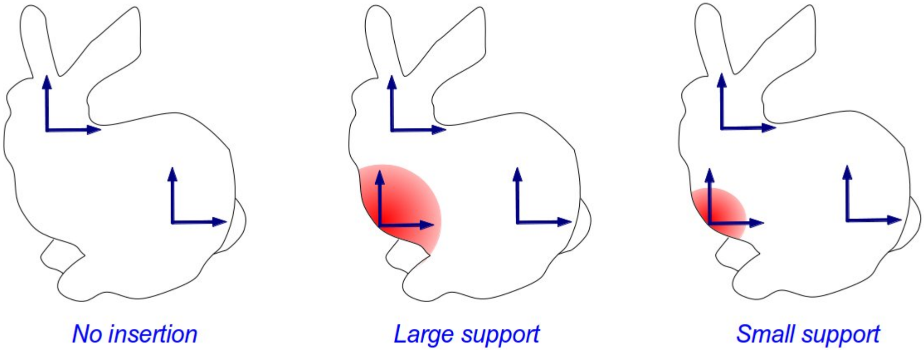
Approximation using midpoint rule



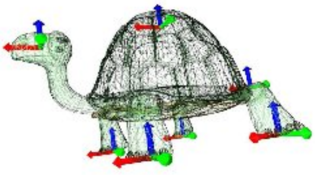
Voxelization

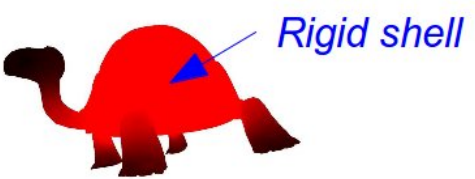


Adaptivity



Material aware shape function



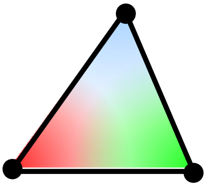


Overview

Shape functions = blending functions = weight functions

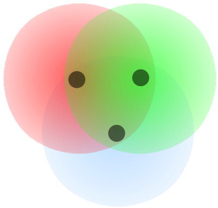
- F.E.M.: attached to each (node,element)

$$u(x) = \sum_i \phi_i(x) u_i$$



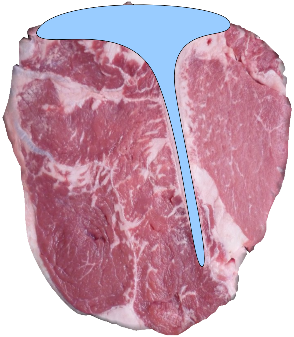
- Meshless:
  - *kernel function* attached to each node
  - Shape functions = kernel functions normalized in each point

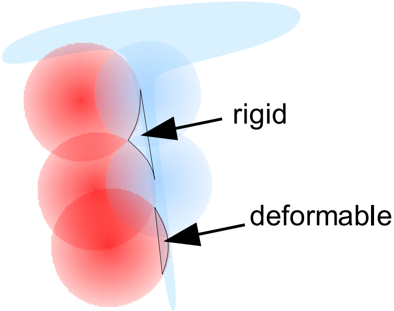
$$u(x) = \frac{\sum_i \phi_i(x) u_i}{\sum_i \phi_i(x)}$$



Limitations of Radial Basis Functions

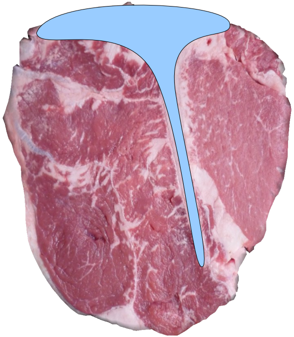
- Large spheres: overlap artifacts

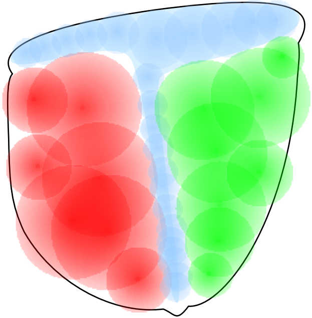




Limitations of Radial Basis Functions

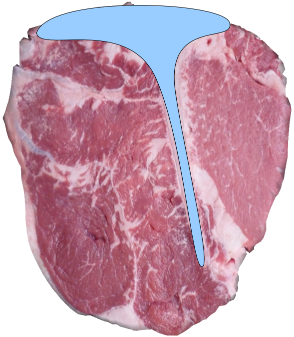
- Large spheres: overlap artifacts
- Small spheres: too many nodes

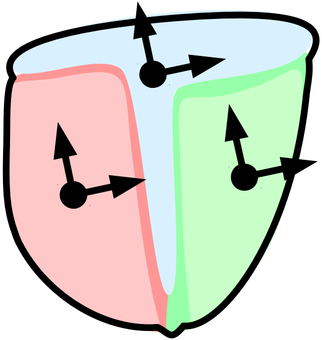




Anisotropic Functions

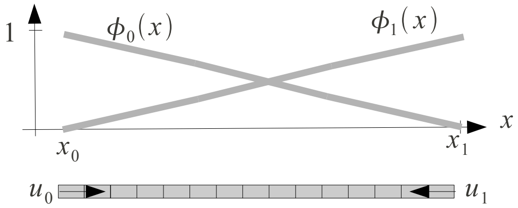
- Control function overlap using:
  - A few anisotropic functions





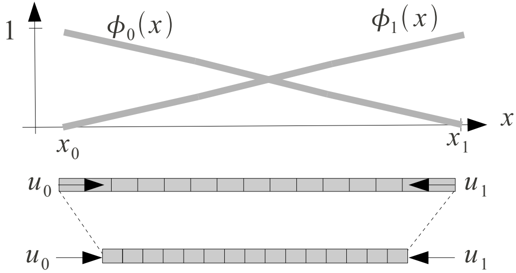
Shape Functions and Material

- Standard shape functions are geometry-based: $1 - dist/size$



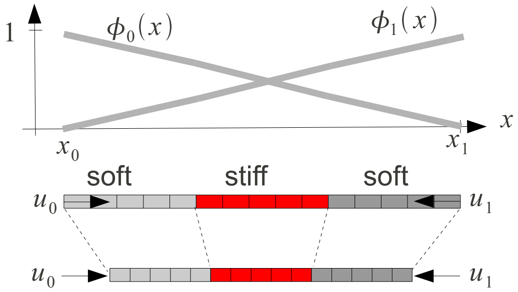
Shape Functions and Material

- Standard shape functions are geometry-based: $1 - dist/size$
- they are well suited for **homogeneous** materials,



Shape Functions and Material

- Standard shape functions are geometry-based: $1 - \mathit{dist}/\mathit{size}$
- they are well suited for **homogeneous** materials,
- but not for **heterogeneous** materials,
    - wrong deformation
    - wrong element stiffness



Compliance distance

- Distance scaled by inverse stiffness:

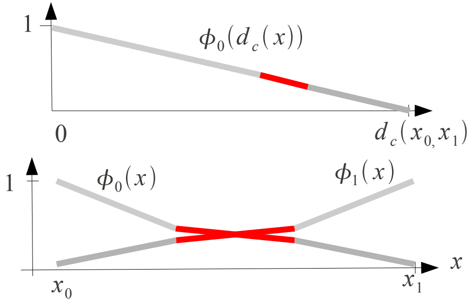
  $$d_c(a, b) = \int_{x_a}^{x_b} \underbrace{\frac{1}{E(x)}}_{compliance} |dx|$$

Compliance distance

- Distance scaled by inverse stiffness:

$$d_c(a,b) = \int_{x_a}^{x_b} \underbrace{\frac{1}{E(x)}}_{compliance} |dx|$$

- Shape functions linear wrt. compliance distance



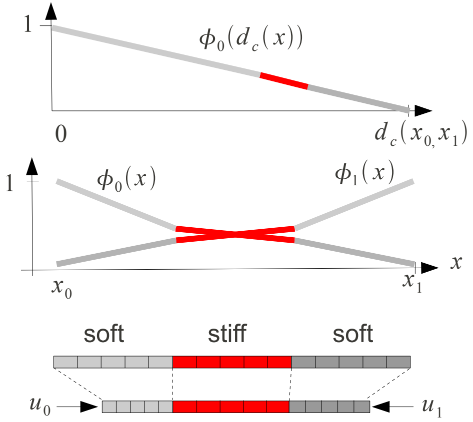
Compliance distance

- Distance scaled by inverse stiffness:

$$d_c(a, b) = \int_{x_a}^{x_b} \underbrace{\frac{1}{E(x)}}_{compliance} |dx|$$

- Shape functions linear wrt. compliance distance

- Exact stiffness and displacement in 1d
- Distance computations only !



Overview

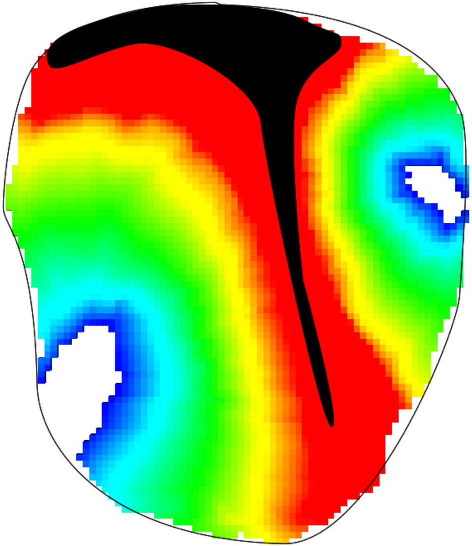
- Motivation and Previous Work
- Background
- Frame-based elastic models
- Compliance distance
- Material-aware kernel functions
- Simulation
- Results

Kernel function in 2d and 3d

Node inside the bone

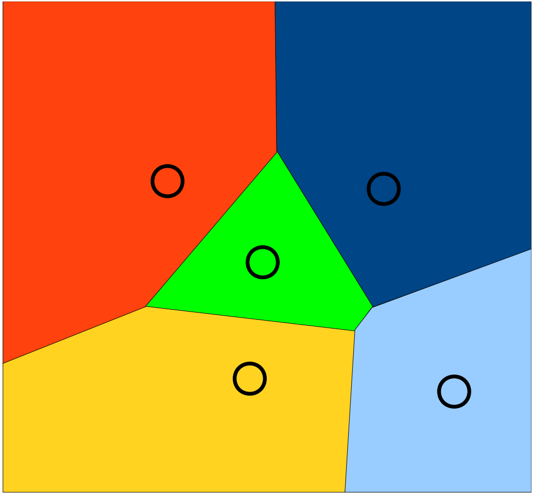
- Weight values
  - at given node: 1
  - at other nodes: 0

- Depend on:
  - compliance distance
  - positions of the other nodes



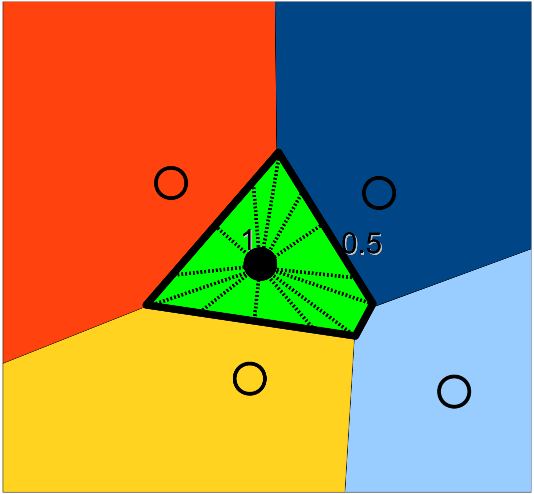
Voronoi-based interpolation

- Decompose space in cells based on distance



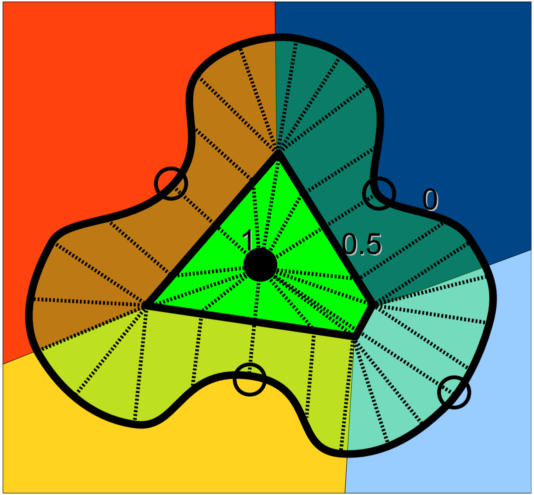
Voronoi-based interpolation

- Decompose space in cells based on distance
- Interpolate the weight along the shortest paths inside the cells
  - Cell center: 1
  - Cell edge: 0.5



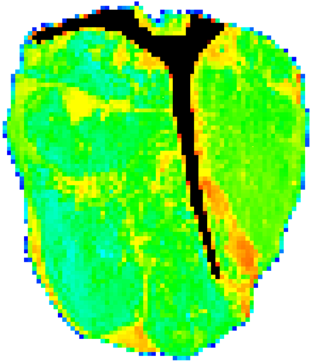
Voronoi-based interpolation

- Decompose space in cells based on distance
- Interpolate the weight along the shortest paths inside the cells
  - Cell center: 1
  - Cell edge: 0.5
- Extrapolate outside
  - Neighbor centers: 0



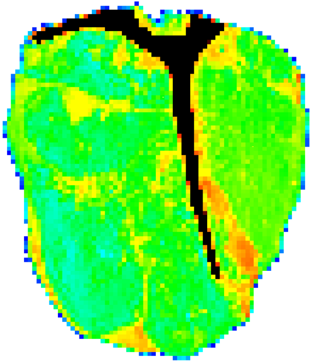
Example

data

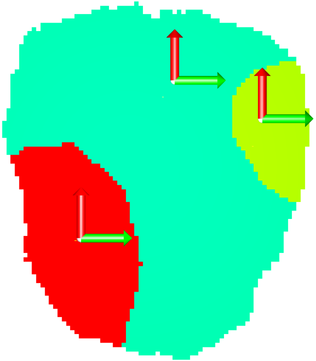


Example

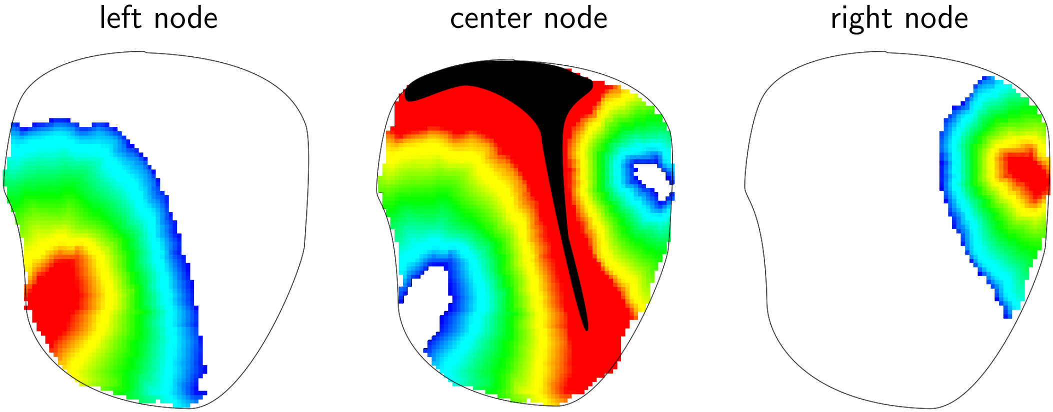
data



voronoi (3 nodes)



Example: blending functions



Automatic node distribution

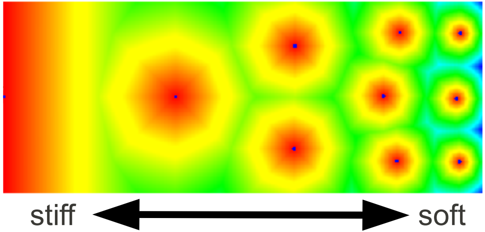
- Lloyd relaxation

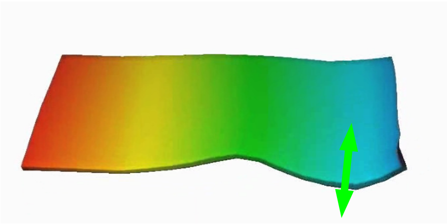
  Repeat:

  - compute Voronoi cells
  - move the nodes to the cell centers
  - until convergence

- Example: non-uniform stiffness

  More samples in the more deformable regions





Overview

Motivation and Previous Work

Background

Frame-based elastic models

Compliance distance

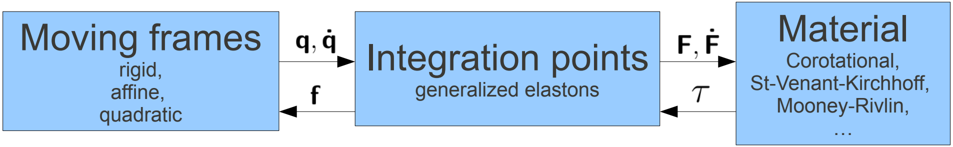
Material-aware kernel functions

Simulation

Results

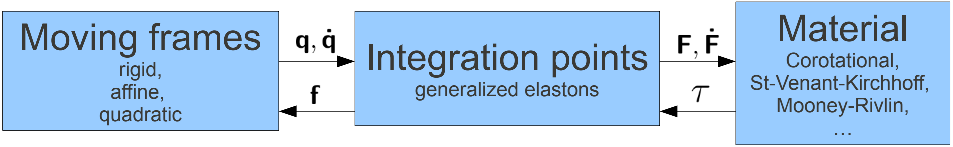
Implementation

- Modular force computation pipeline



Implementation

- Modular force computation pipeline

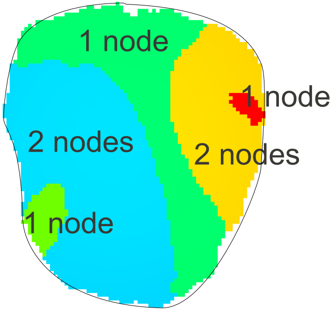


- Implementation in the SOFA open-source framework
  - implicit integration, GPU collision
  - Available in the public version: September 2011 - www.sofa-framework.org

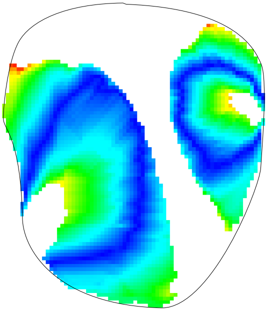
Space Integration

- Generalized elastons
- Exact integration of linear functions
- Regions with as-linear-as possible weight functions:
  - influenced by the same nodes

5 points



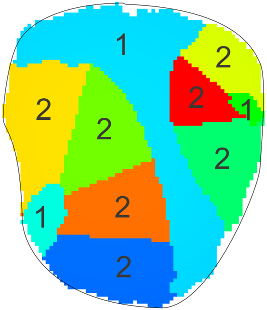
linearity error (center node)



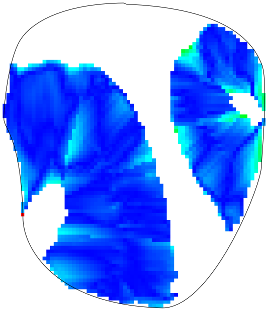
Space Integration

- Generalized elastons
- Exact integration of linear functions
- Regions with as-linear-as possible weight functions:
  - influenced by the same nodes
  - subdivided to reduce linearity error

10 points



linearity error (center node)

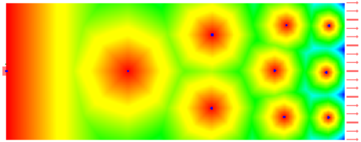


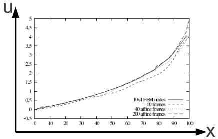
Overview

Validation

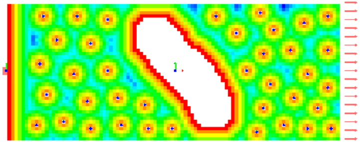
Comparison with high resolution F.E.M. solutions

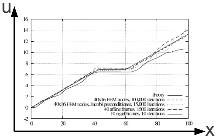
- continuously changing material

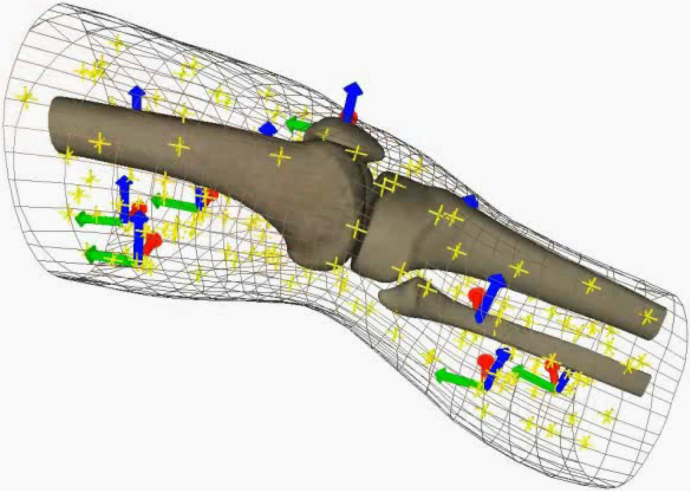




- stiff inclusion







10 affine frames, 200 samples, 10Hz

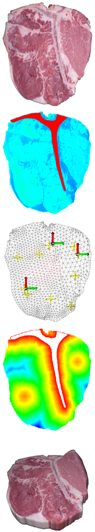
Conclusion

New anisotropic shape functions

- Material-aware
- Detailed stiffness maps in coarse displacement fields
- Interactive simulation of complex models.
- Not restricted to frame-based

Future work

- Conditions for convergence
- Fine control of weight functions
- Adaptivity



Thanks !

- ERC - Passport for Liver Surgery - *FP7, ICT-2007.5.3* PASSPORT
- NSERC
- Canada Research Chairs Program
- Human Frontier Science Program