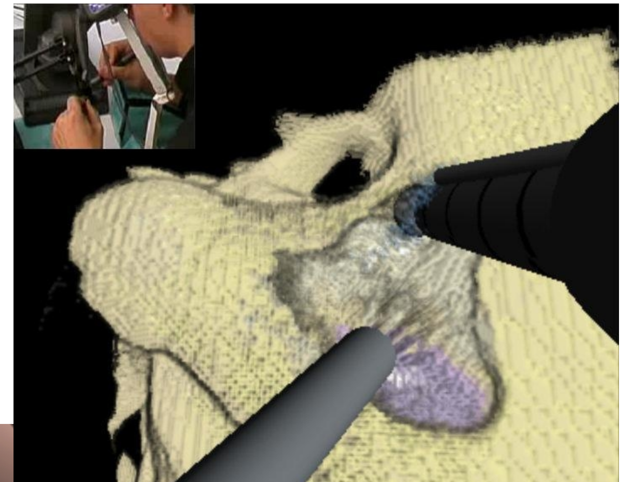
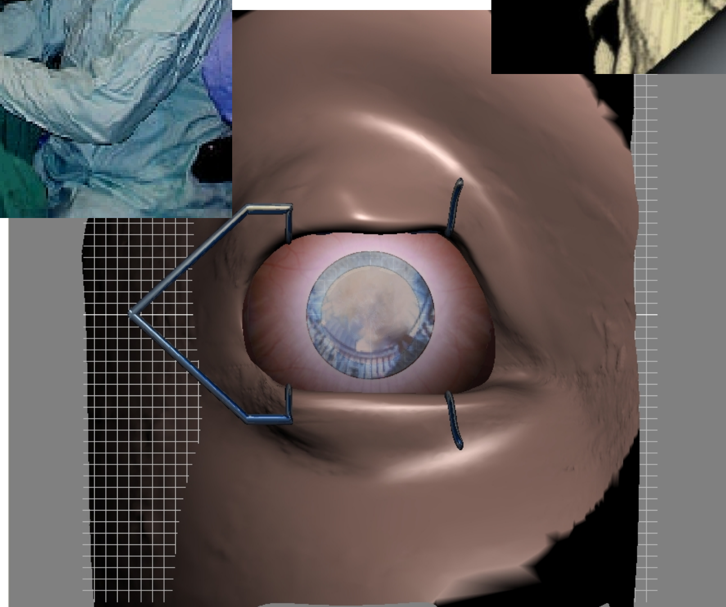
Rigid and quasi rigid body modeling for medical simulation



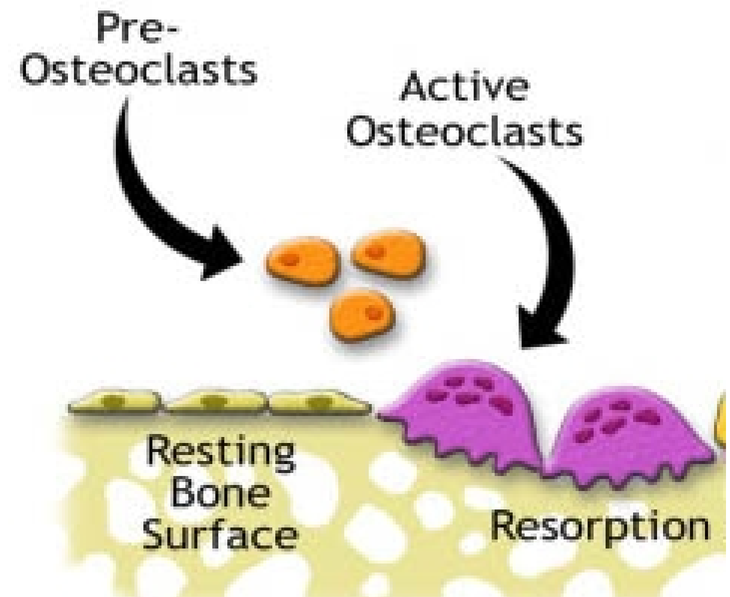


Marco Agus
CRS4, Visual Computing

3DAH Summer School
May 2008, Pula

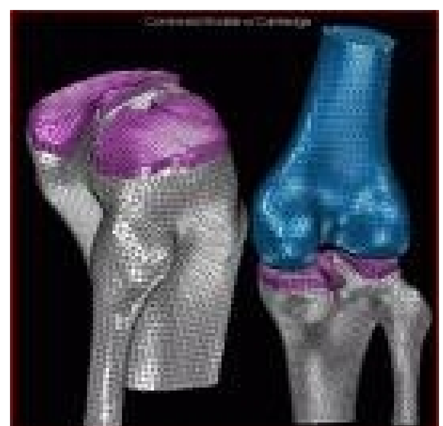
Hard tissue simulation

- Biological Hard tissue = Bone tissue
- Accurate numerical methods for
  - Bone remodeling
  - Bone fracture
  - Implants analysis
- Fast simulation methods for
  - Tissue – tool real-time interaction



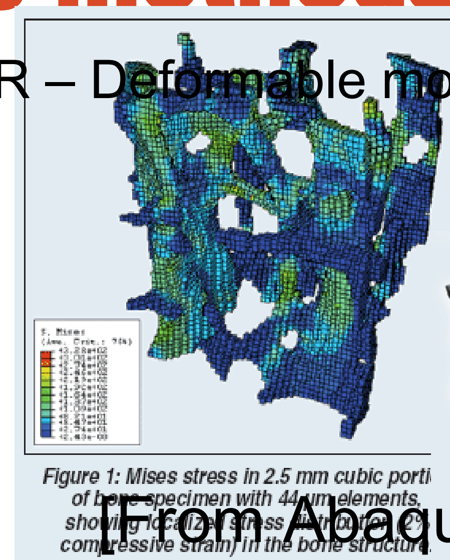
[From Stryker Osteonics]

[From University Loeven]



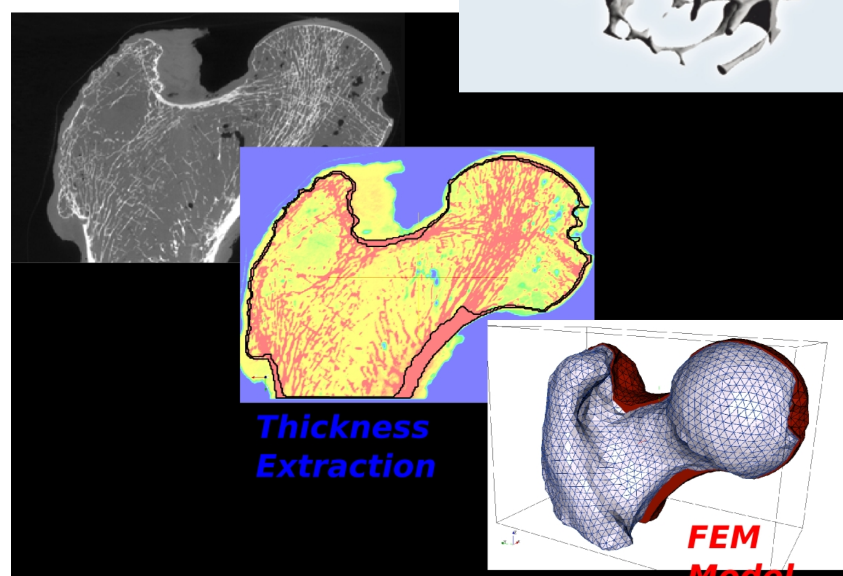
Numerical analysis methods

[STAR – Deformable models] - EG 2005



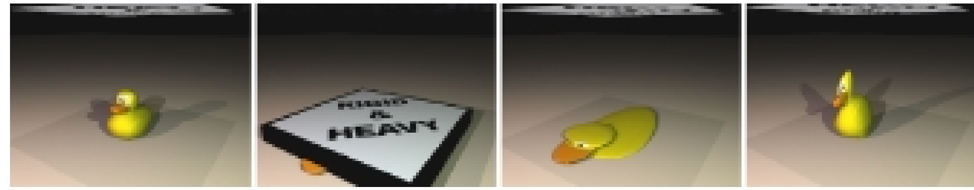
[From Abaqus Inc.]

- Lagrangian mesh based methods
  - Finite Element Method [Muller & Gross, GI 2004]
  - Boundary Element Method [James & Pai, ACM-TOG 2003]
  - Volumetric Mass-spring systems [Teschner, CGI 2004]
- Good for bone remodeling simulation and fracture simulation..
- Offline simulations..

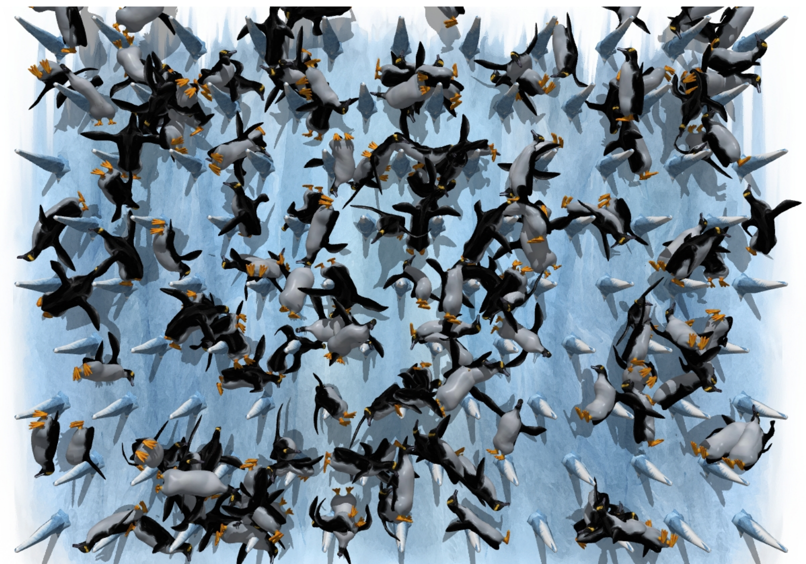


[From Un. Tuwien]

Numerical analysis methods



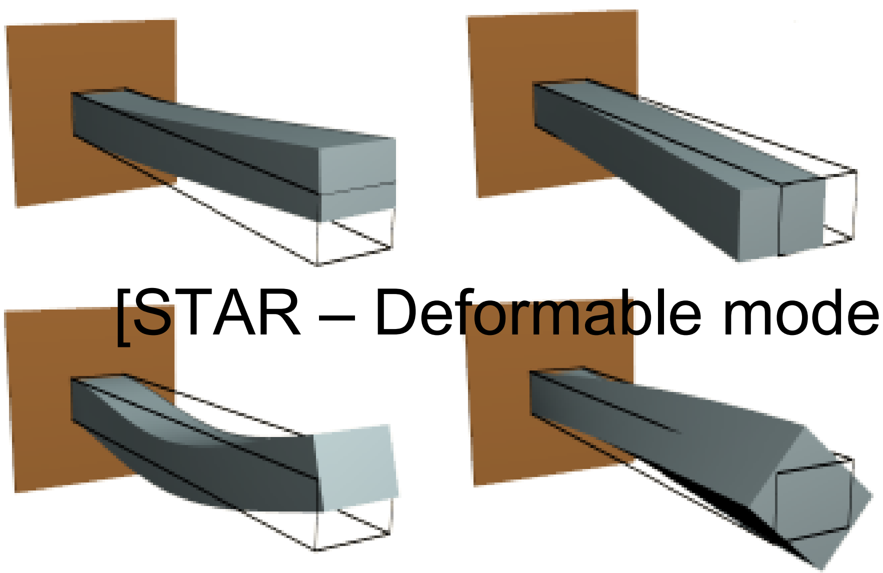
- Lagrangian mesh-free methods
  - Meshless fracturing [Pauly & al, Siggraph 2005]
  - Point-based systems and meshless deformations [Mueller & al, Siggraph 2005]
  - Fast Lattice Shape Matching [Rivers & James, Siggraph 2007]
- Trade-off between real-time constraints and realistic behavior

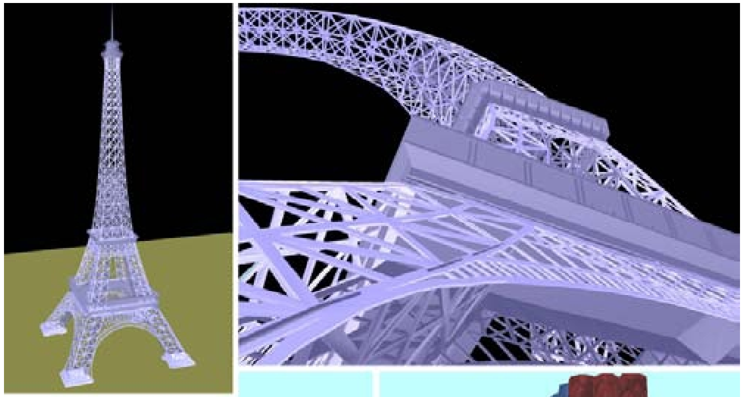


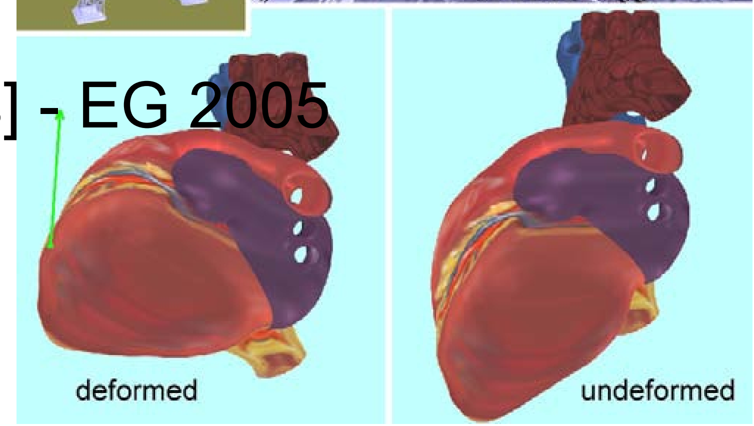
[STAR – Deformable models] - EG 2005

Numerical analysis methods

- Reduced deformation models and modal analysis
  - [James & Pai, Siggraph 2004]
  - [Barbic & James, Siggraph 2005]
- Good for bone remodeling simulation and fracture simulation..
- Offline simulations..



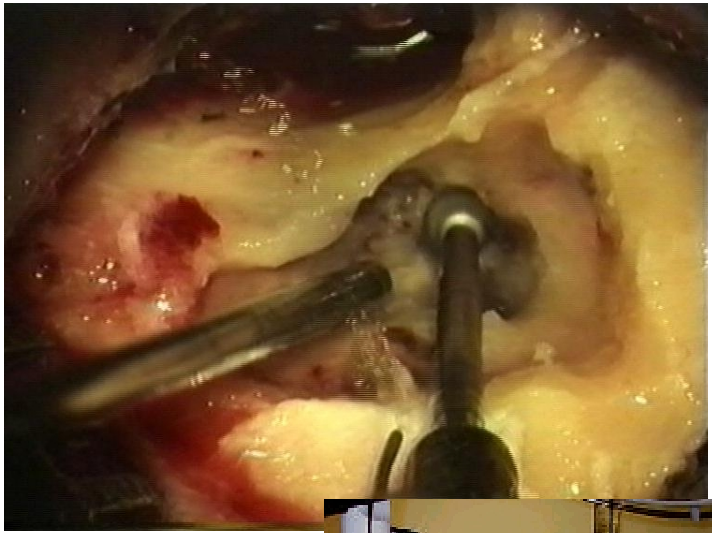




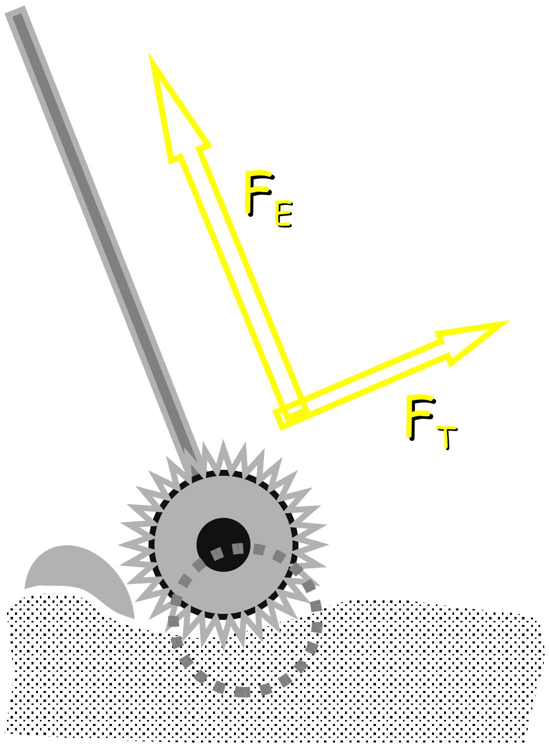
[STAR – Deformable models] - EG 2005

Tissue - tool interaction simulation

- Complex simulation
  - Collision detection
  - Surface tracking
  - Stress distribution
  - Plastic deformation
  - Break-up
- Real-time constraints impose simplified models..
- Local models are required more than global deformation analysis
- CASE studies: temporal bone, crystalline phacoemulsification



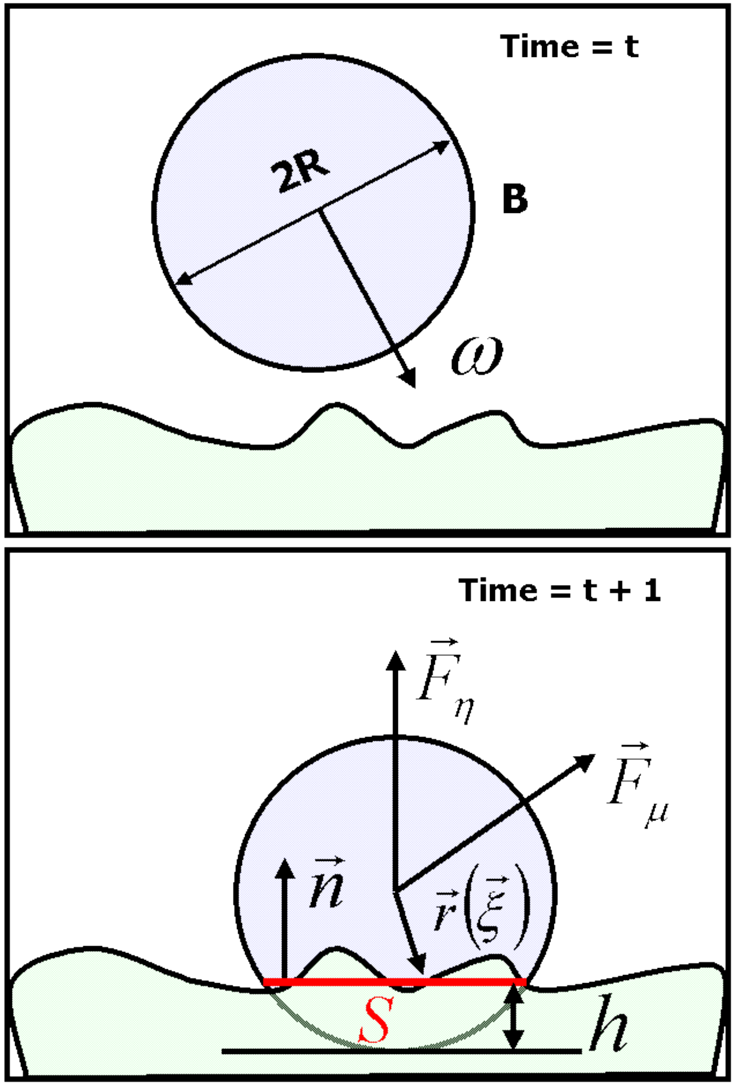
Case study: Burr-bone interaction model



- Mechanical description of the cutting of the material by a rotating burr
- Simplified model based on a limited number of parameters

[Agus et al] - IEEE-VR 2003

Burr –bone contact model



$$m_0 = \int_{r<R} dr^3 \rho(r)$$

$$\vec{m}_1 = \int_{r<R} dr^3 \rho(\vec{r})\vec{r}$$

$$\vec{n} = -\frac{\vec{m}_1}{|\vec{m}_1|}$$

$$m_0 = \pi \rho_0 R^3 \left(\frac{h}{R}\right)^2 \left(1 - \frac{h}{3R}\right)$$

$$\vec{F}_e = c_e R^2 \left(\frac{h}{R}\right)^{\frac{3}{2}} \vec{n}$$

Burr-bone contact model

First two moments of the bone mass density contained in B.

$$m_0 = \int_{r<R} dr^3 \rho(\vec{r})$$

$$\vec{m}_1 = \int_{r<R} dr^3 \rho(\vec{r})\vec{r}$$

Local normal to the surface.

$$\vec{n} = -\frac{\vec{m}_1}{|\vec{m}_1|}$$

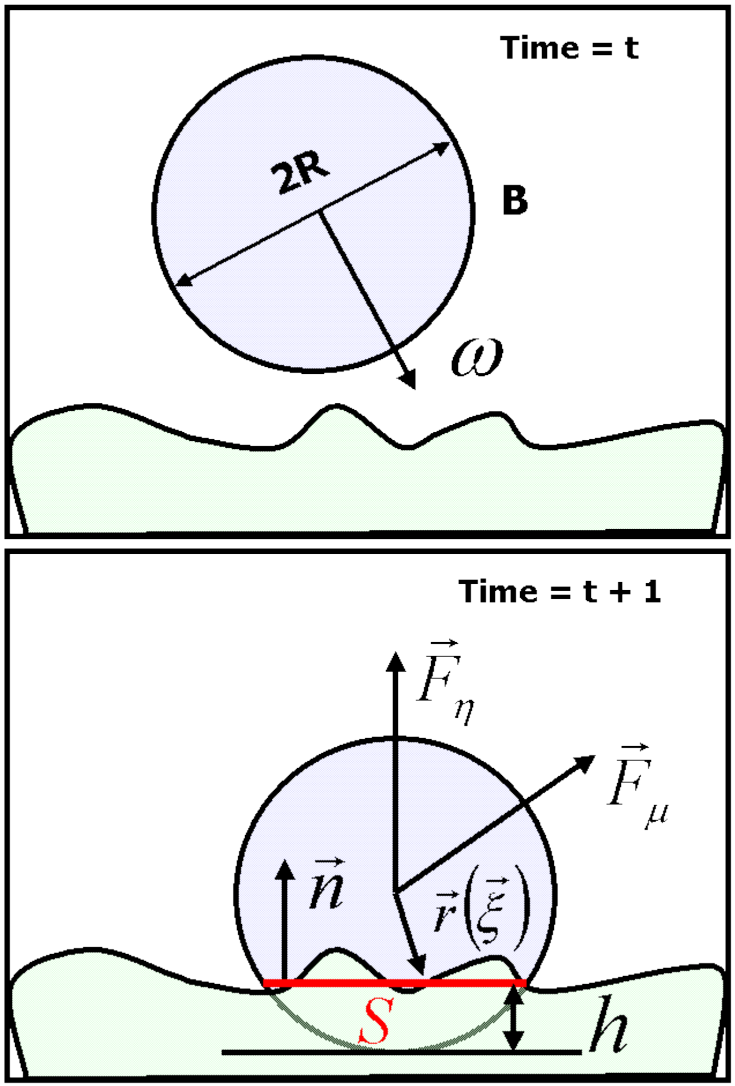
Equation for "penetration depth".

$$m_0 = \pi \rho_0 R^3 \left(\frac{h}{R}\right)^2 \left(1 - \frac{h}{3R}\right)$$

Elastic response of the bone to the impinging burr, as from classical contact theory.

$$\vec{F}_e = c_e R^2 \left(\frac{h}{R}\right)^{\frac{3}{2}} \vec{n}$$

Burr-bone contact model – Force Evaluation



$$a = (C_1 R)^{\frac{1}{3}} F_e^{\frac{1}{3}}$$

$$P(\vec{\xi}) = -\frac{3}{2\pi a^2}\sqrt{1-\frac{|\vec{\xi}|^2}{a^2}}\,\vec{F}_e$$

$$\vec{F}_\mu = \mu \int_{\xi < a} d\sigma\, P(\vec{\xi})\, \frac{\vec{r}(\vec{\xi}) \times \vec{\omega}}{|\vec{r}(\vec{\xi})||\vec{\omega}|}$$

$$\vec{F}_T = \vec{F}_e + \vec{F}_\mu$$

Burr-bone contact model Erosion modeling

ASSUMPTION

All the power spent by working against the frictional forces on a "contact surface" element goes toward the erosion of the bone material in contact with the surface.

$$\mu P(\vec{\xi})\,\omega\, r(\vec{\xi}) \left(1 - \left(\frac{\vec{r}(\vec{\xi}) \cdot \vec{\omega}}{\left|\vec{r}(\vec{\xi})\right| \cdot \left|\vec{\omega}\right|} \right)^2 \right) d\sigma = \alpha\, \varphi(\vec{\xi})\, d\sigma$$

Discretized description – Force evaluation

ASSUMPTION

The voxels are approximated with spheres of the same volume, centered at the voxel center. The intersection volume, surface and distance between voxels and burr bit are thus simplified.

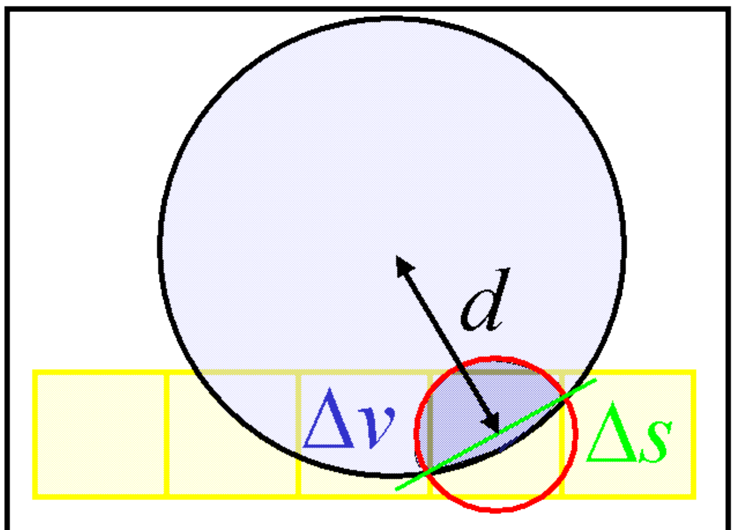
$$\Delta v(d) = \frac{\pi}{12}\left(d^3 - 6\left(R^2 + \eta^2\right)d + 8(R^3 + \eta^3) - 3\left(\eta^2 - R^2\right)\frac{1}{d} \right)$$

$$\Delta \sigma(d) = \frac{\pi}{4}\left(2\left(R^2 + \eta^2\right) - d^2 - \left(\eta^2 - R^2\right)\frac{1}{d^2} \right)$$

$$r(d) = \frac{1}{2}d + \frac{r^2 - \eta^2}{2}\frac{1}{d}$$

Discretized description – Force evaluation

A direct translation of the force contact model will transform integrals in sums over the voxels that have non null intersection with the burr bit.



$$\vec{M}^* = \sum_i \Delta v(|\vec{c}_i|)\chi_i$$

$$\vec{M}_1^* = \sum_i \Delta v(|\vec{c}_i|)\chi_i \vec{c}_i$$

$$\vec{F}_\mu = \mu \sum_i \Delta \sigma (|\vec{c}_i|) P(\xi_i) \frac{\vec{c}_i \times \vec{\omega}}{|\vec{c}_i||\vec{\omega}|}$$

Discretized description – Erosion modeling

The power spent by the frictional forces on a voxel is

$$\mu P(\vec{\xi}_i)\omega c_i\left(1-\left(\frac{\vec{c}_i \cdot \vec{\omega}}{|\vec{c}_i||\vec{\omega}|}\right)^2\right)\Delta\sigma_i = \alpha\varphi_i\Delta\sigma_i$$

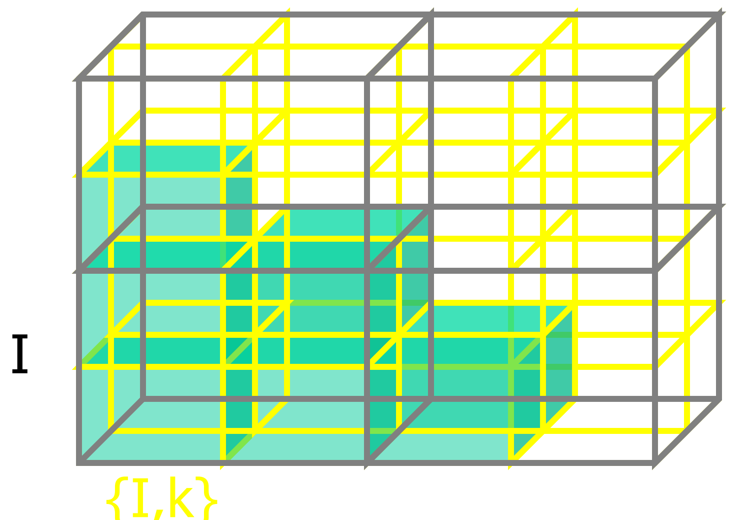
$$\Delta M_i = \Delta t \Delta\sigma\varphi_i$$

An 8 bit counter is associated with each voxel, representing the voxel density, and decreased by a value proportional to the "assumed" amount of removed mass.

Multi-scale spatial description

Multiresolution volumetric description, obtained by partitioning the volume of interest using an octree. The leaves refer to the scene voxels while the coarsest level is the whole scene.

Initialization: starting from leaves, for each octree block I, local values of moments m_o^I and $\vec{m}_1^I$ are computed, with the following rules:



$$m_o^I = \sum_k m_0^{\{I,k\}}$$

$$\vec{m}_1^I = \sum_k \left[\vec{r}_I^{\,k} m_0^{\{I,k\}} + \vec{m}_1^{\{I,k\}} \right]$$

Multi-scale spatial description

Moments estimation: at each haptic cycle the octree is descended until we find blocks fully contained or partially intersecting the burr sphere.

Blocks fully contained → contributions to moments are added.

Blocks partially intersecting → if block size > l refinement otherwise partial volume contributions are added

$$\Delta m_0 = \frac{\Delta V}{V_I} m_o^I$$

$$\Delta \vec{m}_1 = \frac{\Delta V}{V_I}\left(\vec{R}_b - \vec{R}_c\right)\left(m_0^I + \vec{m}_1^I \cdot \left(\vec{R}_c - \vec{R}^I\right) + O\left(\left(l/R\right)^2\right)\right)$$

Multi-scale erosion

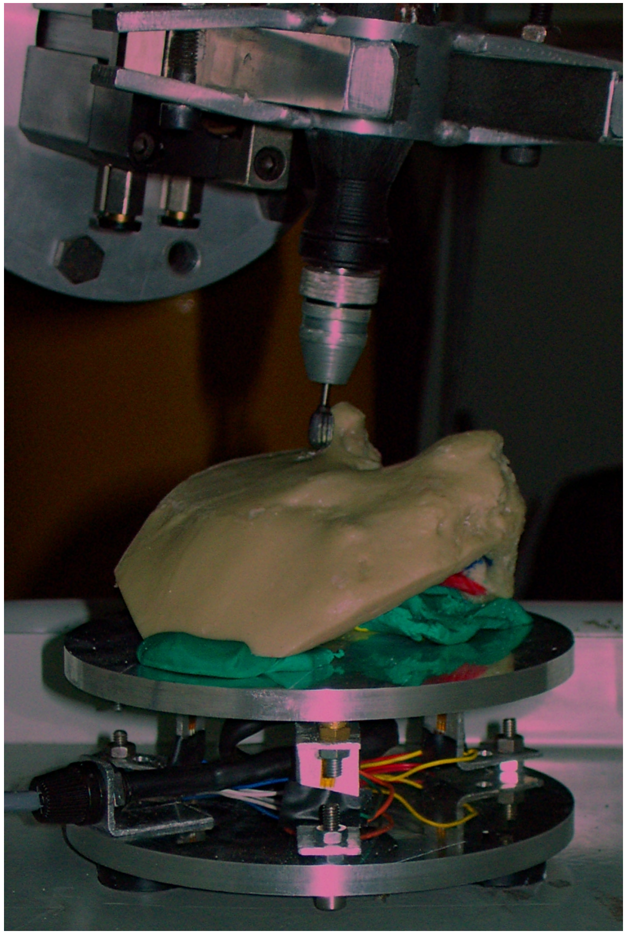
Erosion modeled as a position dependent erosion rate described by f, an erosion shape function, constrained to have a max for r/R=0 and to be null for r/R>1.

$$\frac{d\rho}{dt} = \alpha f\left(\frac{r}{R}\right)\rho\,(\vec{r})$$

In_current implementation, to accomodate a wide range of erosion rates with 8 bits, the rate of erosion is converted to a probability that the value of the voxel at position r will be reduced by one at next step, and used in a Roussian roulette scheme.

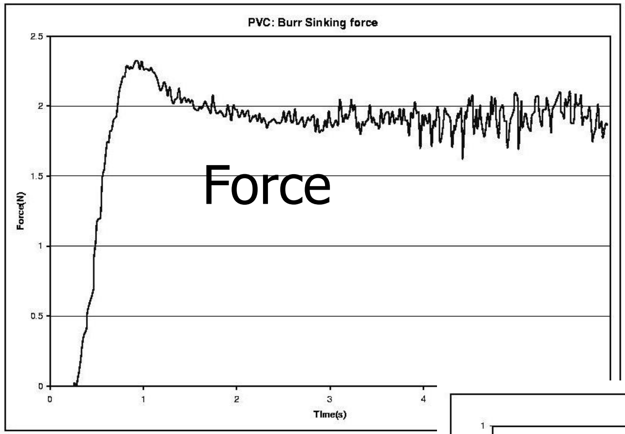
The erosion model is contained in the octree descent algorithm, and the moment updates are pulled from octree leaves up to the root.

Fitting parameters to experimental data

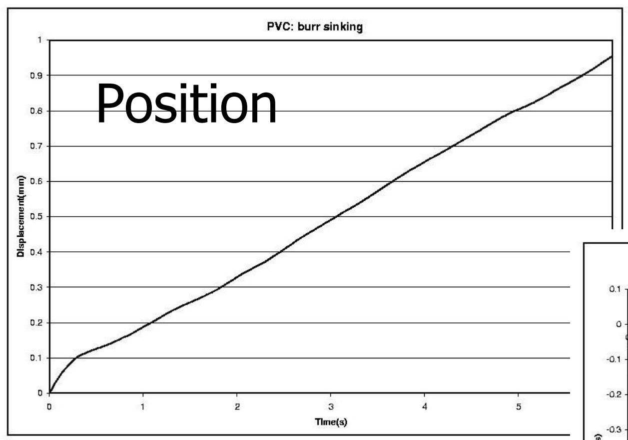


- **Experimental facility:** an arm robot and a mono-directional load cell.
- **Reference experiment**: vertical descent at constant force, performed in the experimental setup and its virtual analogue.
- **Materials tested**: Pettigrew synthetic model, temporal bone, PVC K70 resin.

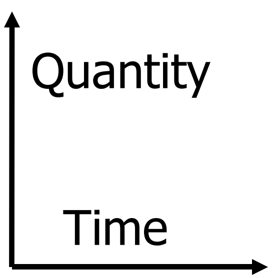
Experimental procedure

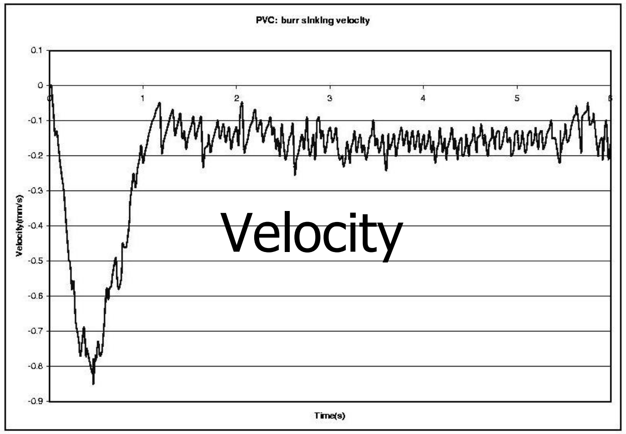


Apply a constant force along the rotating tool axis, measure and record tool positions.



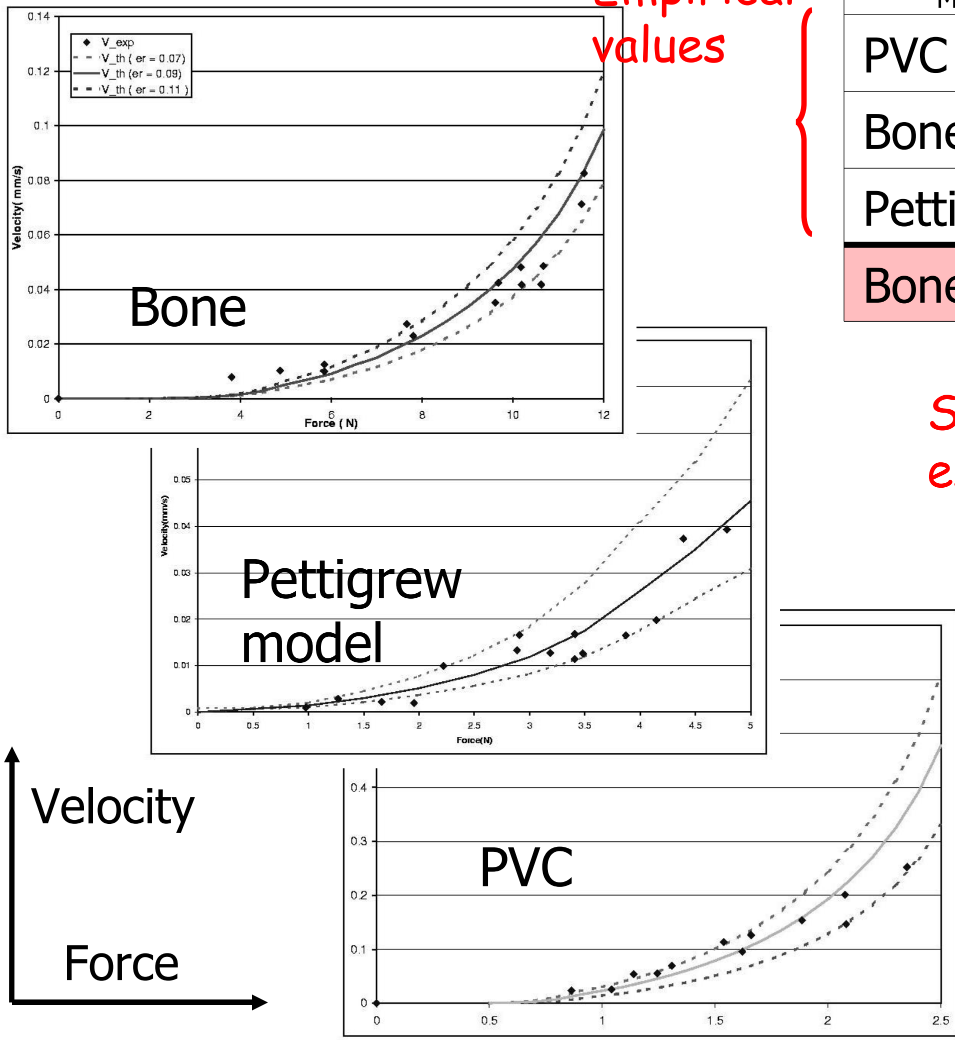
Sample data from PVC sessions





Velocity essentially const, and used to characterize the behaviour of the material.

Experimental results



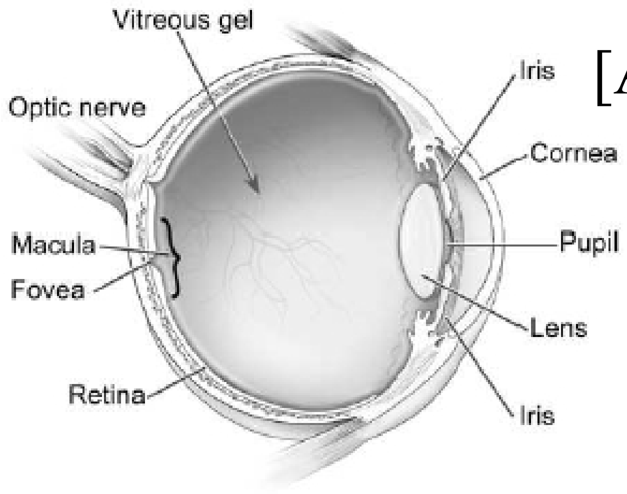
Empirical values

Material	c_{el}	α_{er}
PVC	0.5	0.26 ± 10%
Bone	2.1	0.09 ± 10%
Pettigrew	1.5	0.12 ± 10%
Bone(ENT)	1.5	0.1

Subjective expert values

Expert selected values are consistent with the values measured for bone and Pettigrew models.

Case study: Cataract simulation



[Agus et al, VRIPHYS-2006]

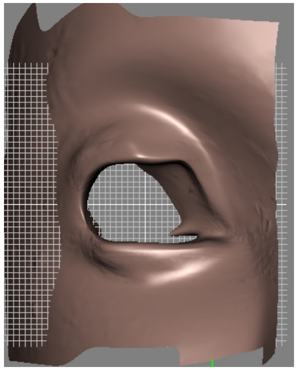
- TARGET: substitute the clouded crystalline lens with a an IOL (Intra-ocular lens)
- Phaco-emulsification:
  - Fracturing the lens nucleus with phaco, cross-shaped
  - Removing out fragments
- Cristalline is a quasi-rigid object, fragments need to be simulated with multi-body dynamics

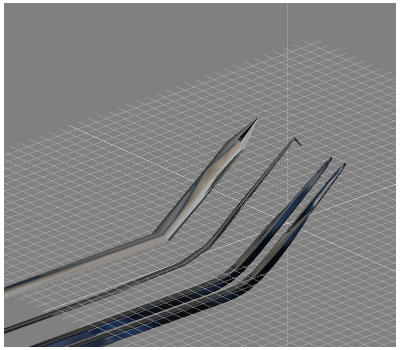
System Overview

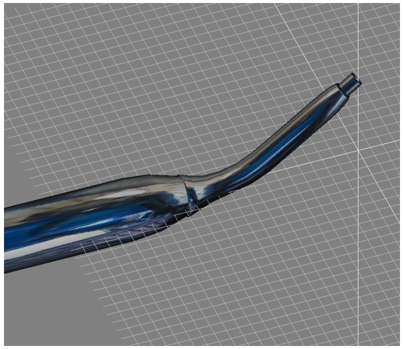
- Simulator must model visual appearance and behavior of various components:

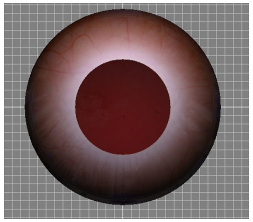
- Tools:
  - Hook
  - Forceps
  - Cutter
  - Phaco

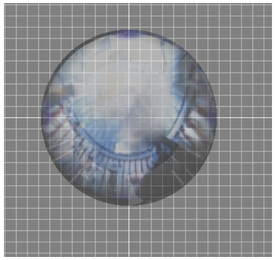
- Anatomical Models:
  - Face
  - Eyeball
  - Posterior Camera
  - *Crystalline*
  - *Membrane*
  - Iris
  - *Cornea*







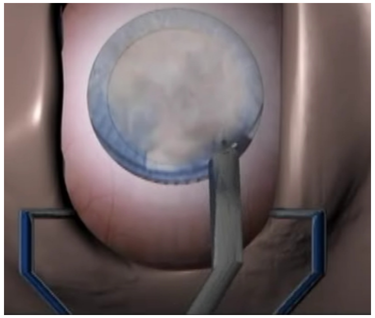


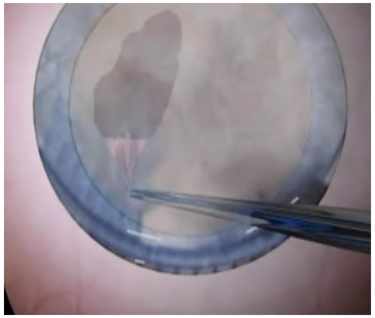


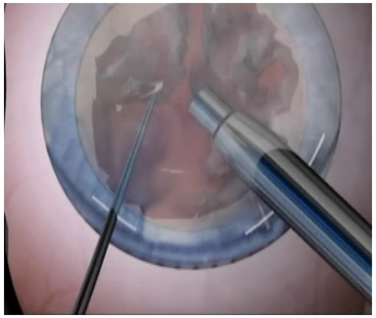
Real Time Constraint

Main Dynamic Behaviors

- **Cornea**
  - Must respond to cutting
- **Membrane**
  - Must respond to tearing
- **Crystalline**
  - Must respond to eroding

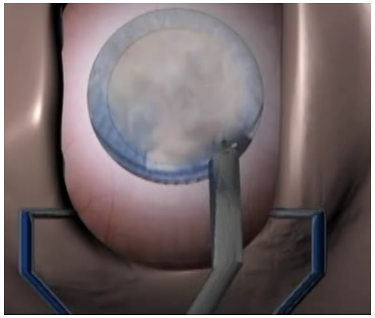


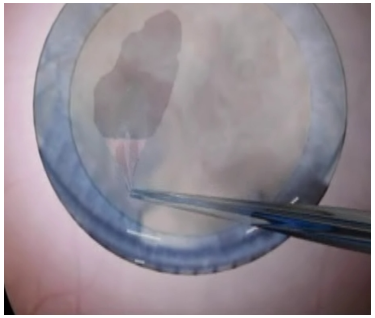


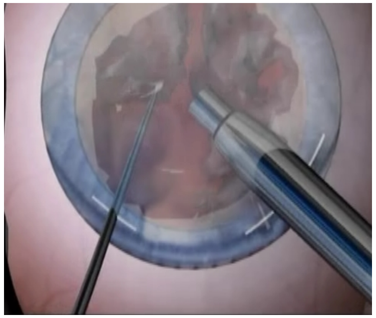


Main Dynamic Behaviors

- **Cornea**
  - Must respond to cutting
- **Membrane**
  - Must respond to tearing
- **Crystalline**
  - Must respond to eroding

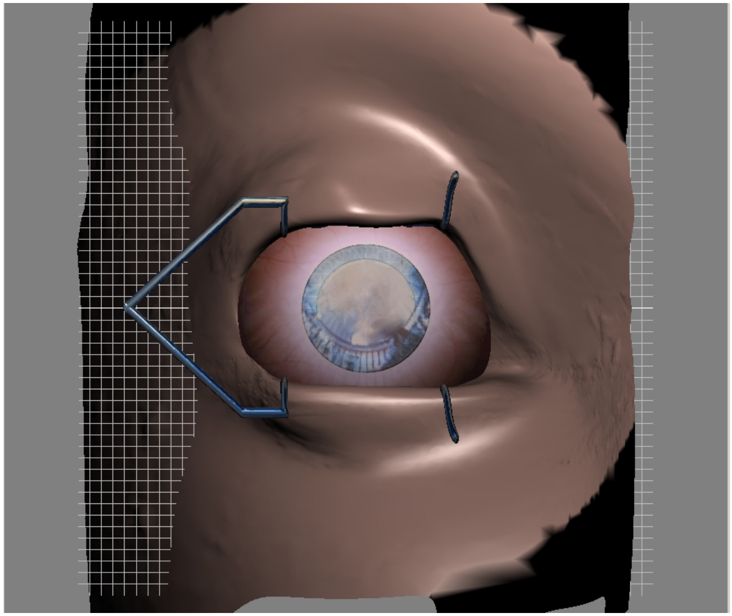






Simulator Structure: decoupled simulation

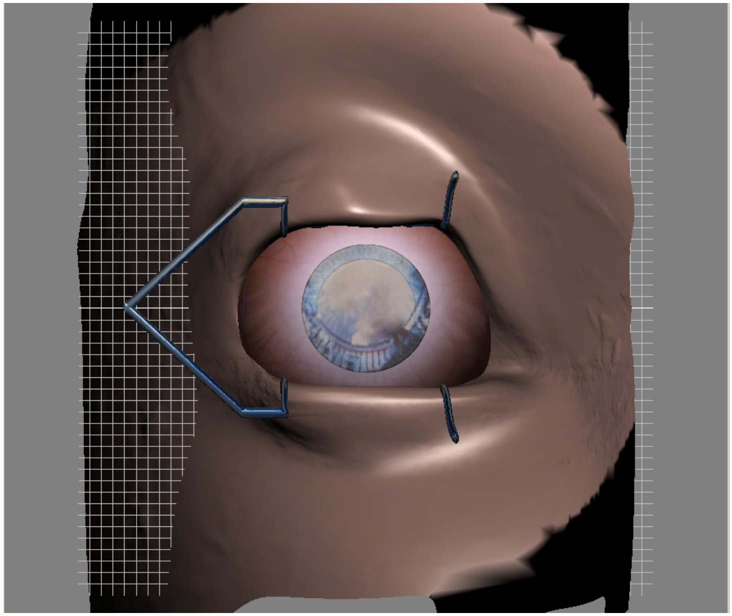
- Fast Subsystem
  - Tool Tracking
  - Cornea Behavior
  - Membrane Behavior
  - Crystalline Behavior
- Slow Subsystem
  - Rendering
- Anatomical Model



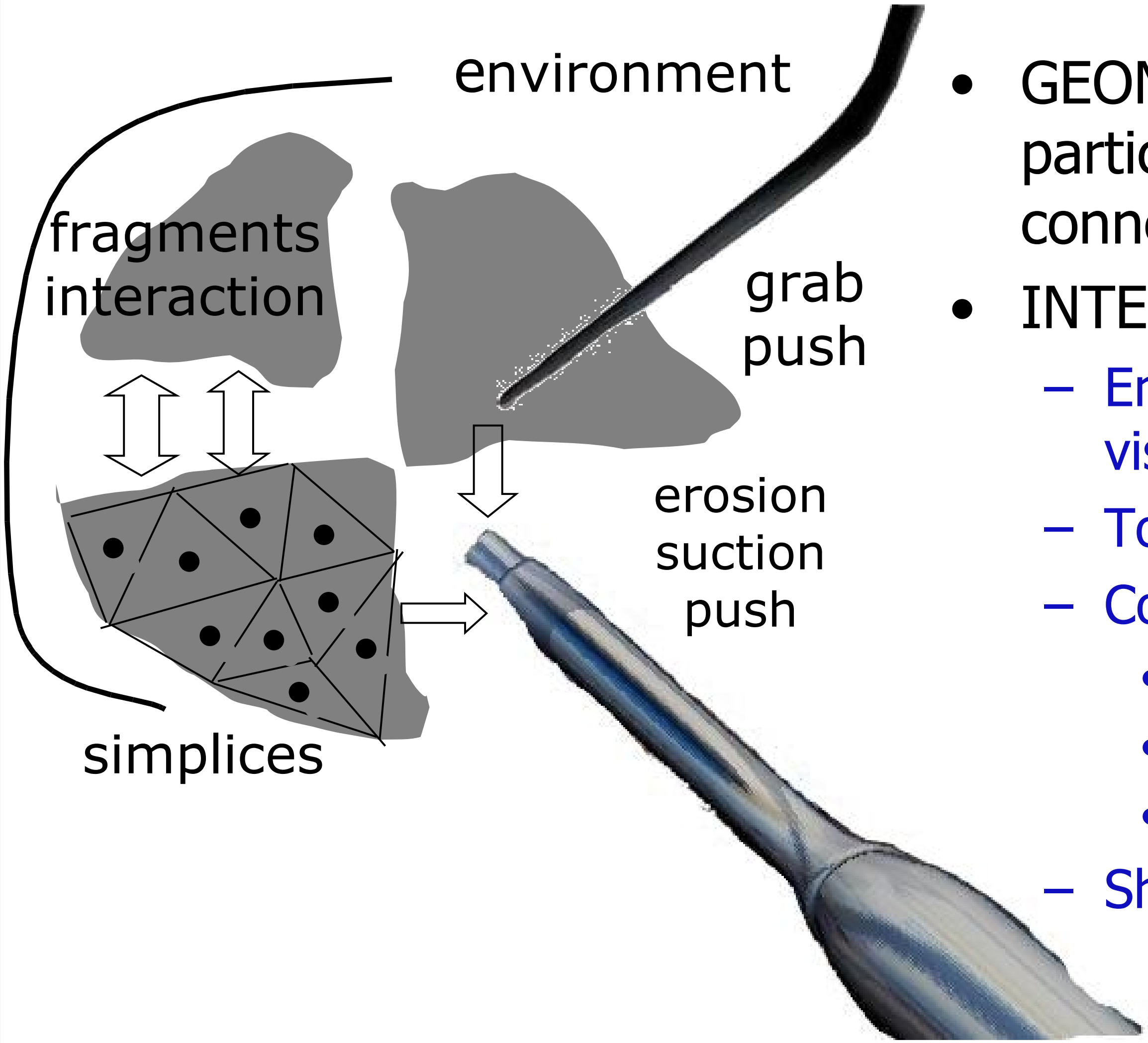
Simulator Structure: decoupled simulation

- Fast Subsystem
  - Tool Tracking
  - Cornea Behavior
  - Membrane Behavior
  - Crystalline Behavior
- Slow Subsystem
  - Rendering

- Anatomical Model



Phacoemulsification simulation



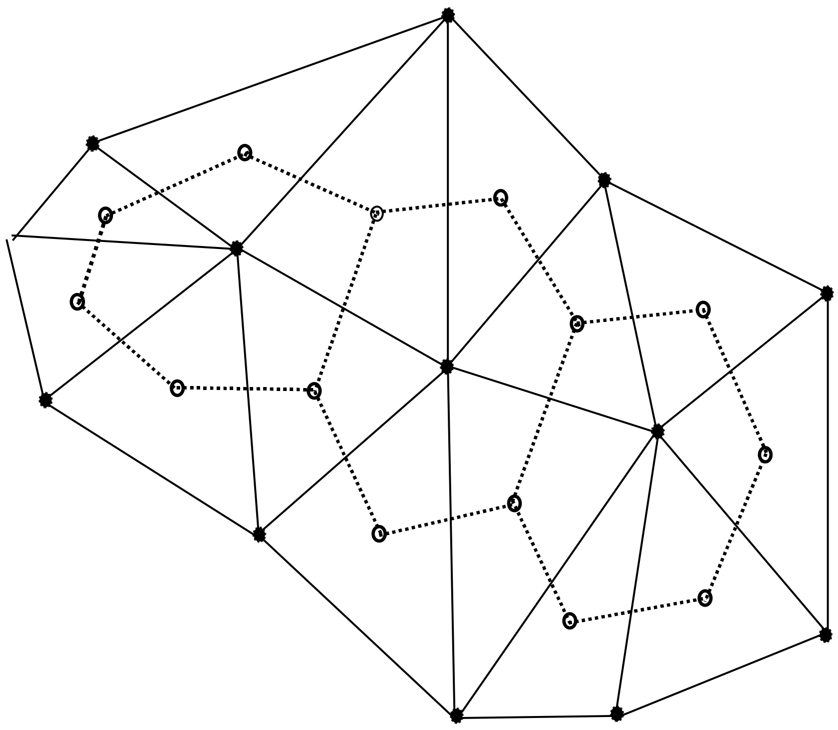
- GEOMETRIC DESCRIPTION: particle simplices with connectivity
- INTERACTIONS:
  - Environment (gravity + viscosity)
  - Tools interaction
  - Collision detection
    - Fragments – fragments
    - Fragments – tools
    - Fragments – environment
  - Shape matching correction

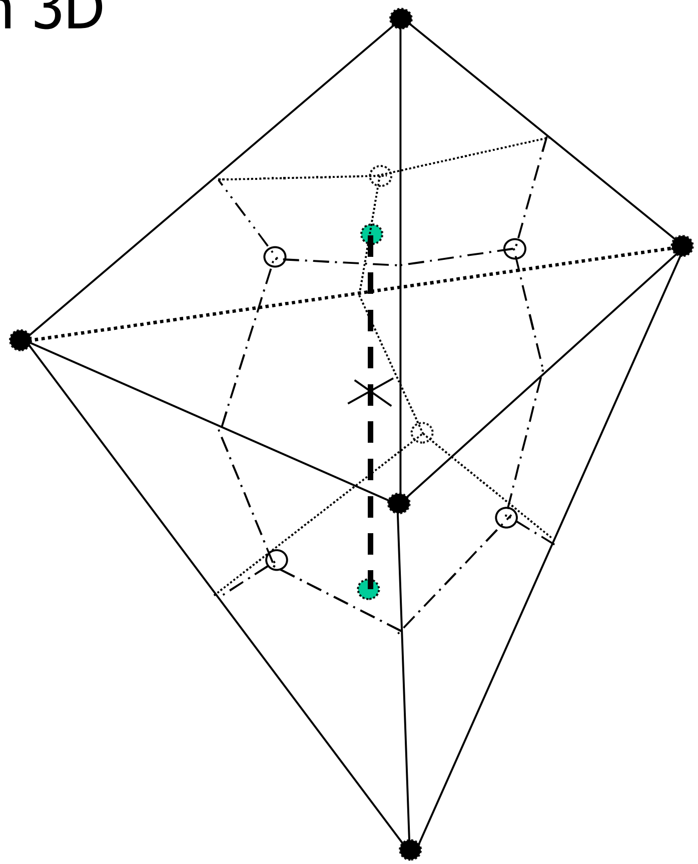
Main design decisions

- Combine mesh-based and particle-based techniques
- Tetrahedral mesh
  - Easy and fast to render
  - Maintains fragment coupling
- Particle-based simulation
  - Shape matching correction good for quasi-rigid bodies
  - Uses info from tetrahedral mesh to evolve fragments independently

Geometric description

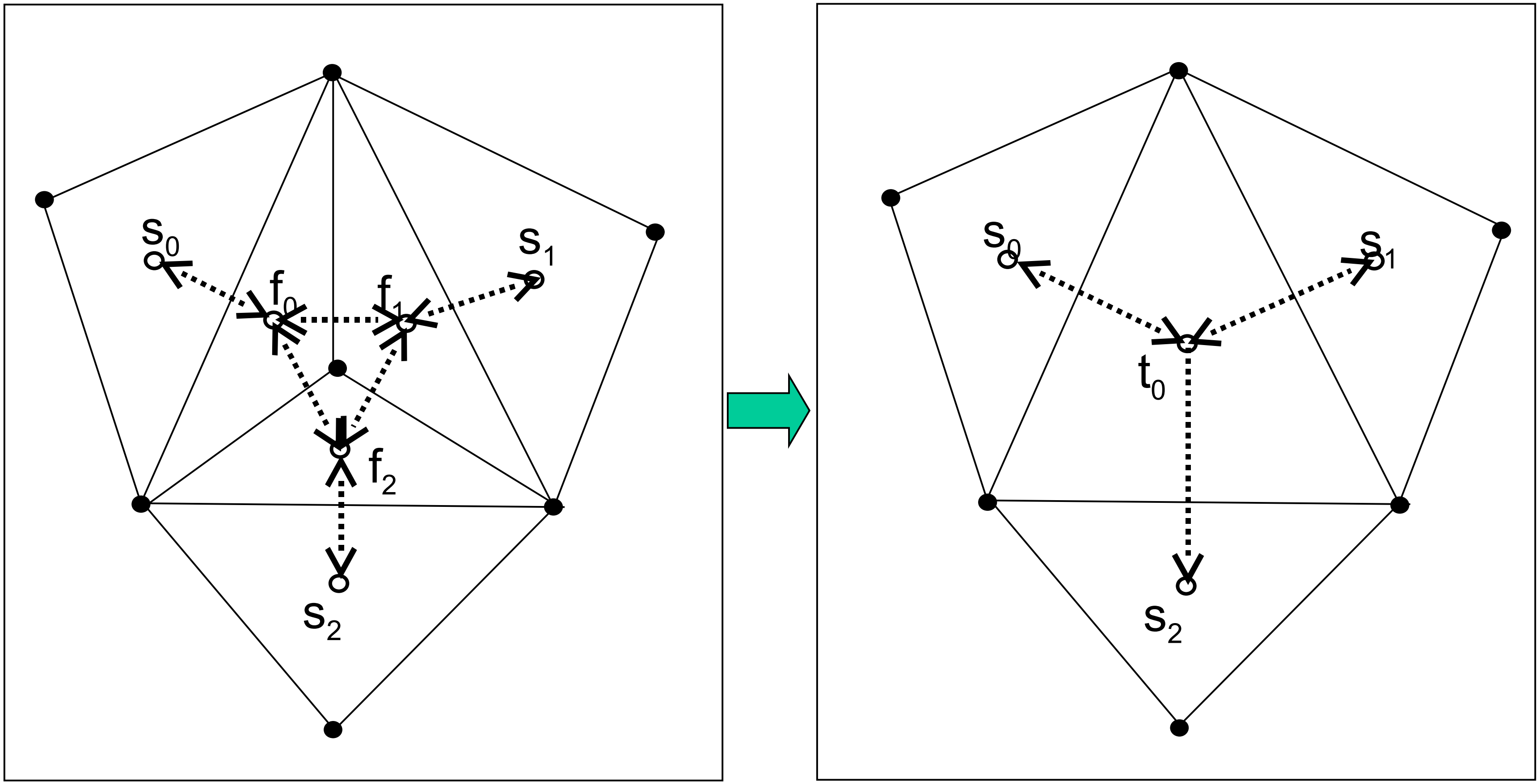
- Simplex construction starting from a tetrahedron mesh: triangulation for rendering is easy to obtain
- Removal is trivial in 2D, complex in 3D





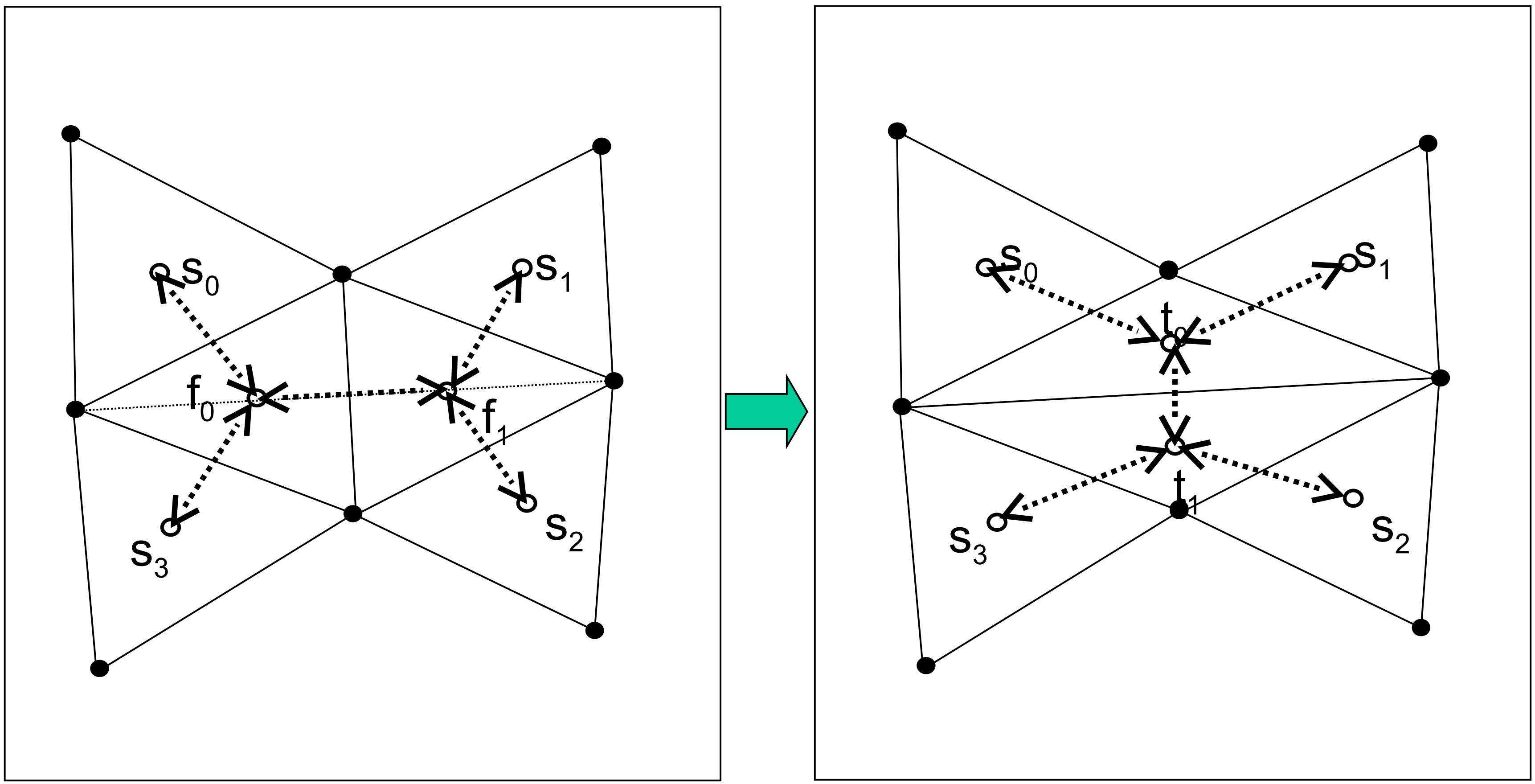
Simplex removal primitive

- Three external faces



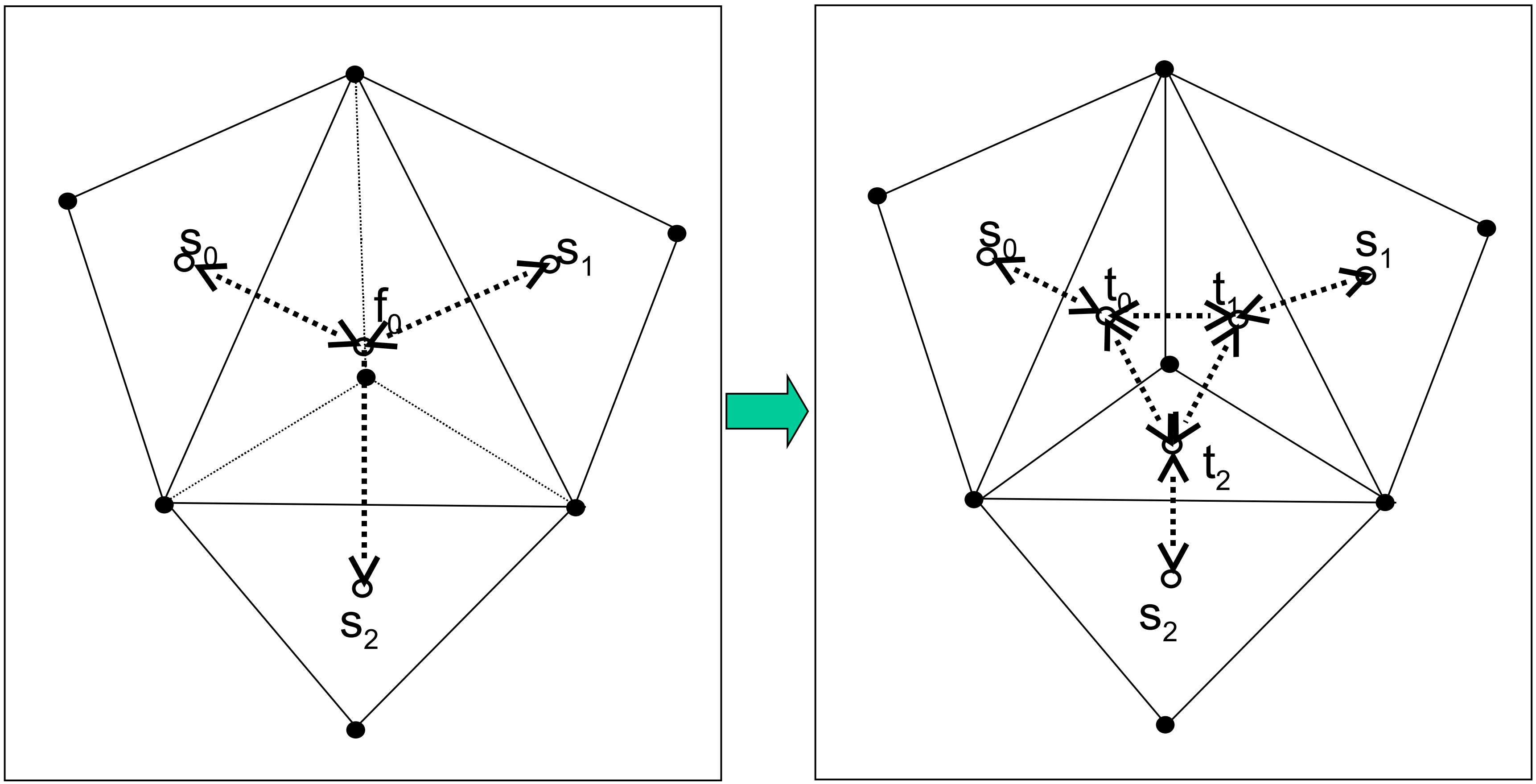
Simplex removal primitive

- Two external faces



Simplex removal primitive

- One external face

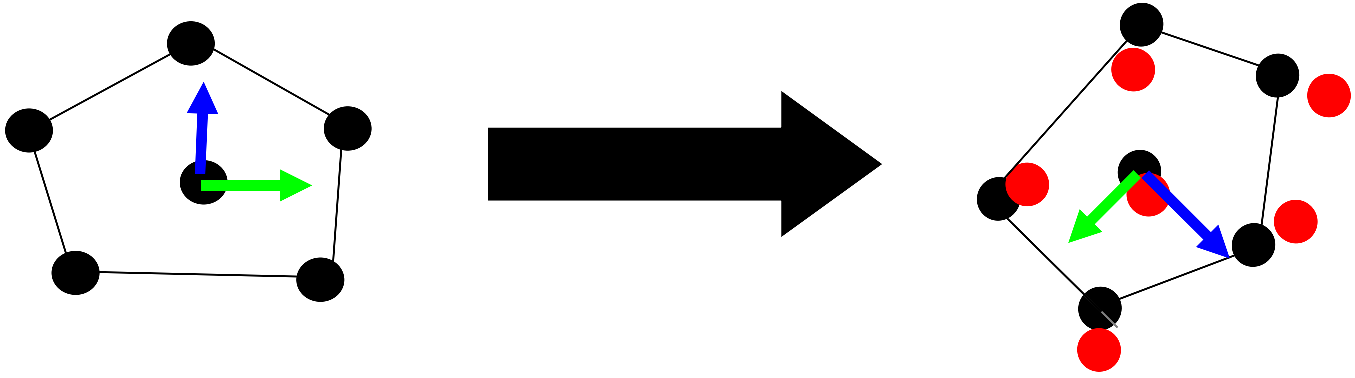


Evolution algorithm

- Applying environment forces to particles
- Crystalline lens erosion:
  - remove particles in phaco influence zone
  - update simplex topology
  - recognize independent fragments
- Collision detection:
  - between independent fragments,
  - between particles and boundaries (capsulae and cornea)
  - between particles and tools
- Restore original quasi-rigid shape of independent fragments by employing a shape-based correction technique

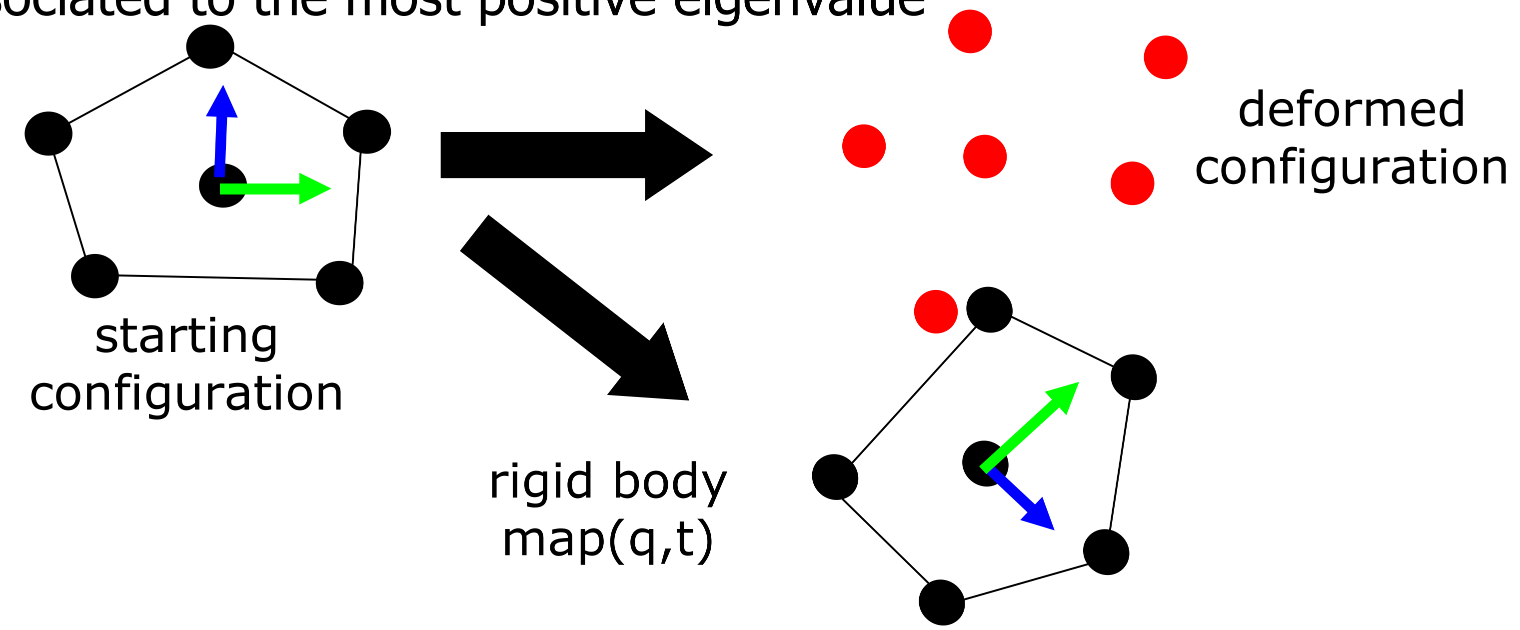
Shape matching correction

- Quasi-static meshless shape matching correction algorithm (Muller 2005)
- DRAWBACKS:
  - Numerical approach not able to handle fragments containing very few cells
  - Extracting orthonormal matrix from best-fit linear transform is not the best solution for the absolute orientation problem



Shape matching correction

- Absolute orientation problem can be solved in closed form, numerically stable (Horn 1988), as originally applied in photogrammetry
- Solve a least square error problem, where unit quaternions are used to represent rotations
- Solution of the problem is the eigenvector of a 4X 4 matrix associated to the most positive eigenvalue



Absolute orientation problem solution

- Given a particle system m_i passing from an undeformed state $\{a_i\}$ to a deformed state $\{b_i\}$ involved by dynamics, recover the rigid original configuration by finding the optimal rigid body map M(R,t)
- Solve the following MLS problem:

Residual error

$$e_i = b_i - R(a_i) - t$$

Minimization problem

$$\min_{R,t} \Psi(R,t) = \sum_i \|e_i\|^2$$

Absolute orientation problem solution

- Expressing rotations by employing unit quaternions leads to a closed form solution:

Barycentric coordinates

$$\alpha_i = a_i - \hat{a}, \beta_i = b_i - \hat{b}$$

Translation offset

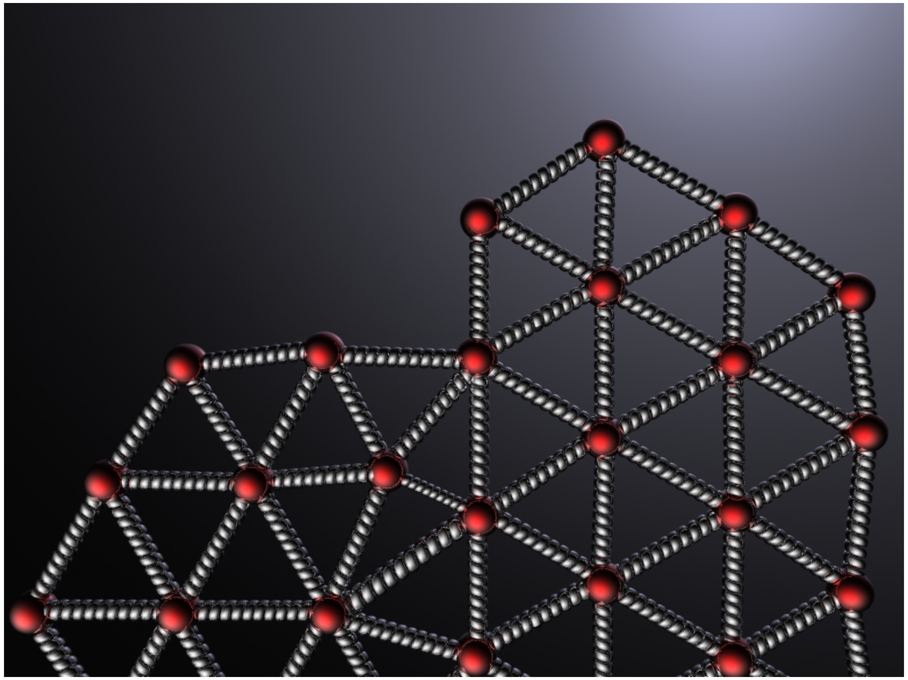
$$t = \hat{b} - R(\hat{a})$$

$$\min \psi = \max \sum_i \beta_i R(\alpha_i) = \max \tilde{q}^T N \tilde{q}$$

q is the eigenvector associated to the max eigenvalue of the symmetric matrix N

Methods and Tools (1/2): capsulorhexis simulation

- Mass-spring network over triangular mesh
- Each mass particle could be anchored,scripted or free
- Tools implemented through collision detection and scripting routines
- Accumulated acceleration for each particle includes:
  - Gravity
  - Environment viscosity
  - Spring contributions



Results

- Prototype developed in C++ on OpenSceneGraph toolkit
- Our current configuration is the following:
  - a single-processor PIV/1.5 GHz for the high-frequency device tracking task
  - a dual-processor PIV/2.2 GHz with 2 GB RAM and a NVIDIA GeForce 6800 for the low frequency tasks
  - two Phantom haptic devices connected to the single processor PC provide 6DOF tracking
  - N-vision VB30 binocular display

- Performance of the prototype is sufficient to meet the timing constraints for display and the overall realism of the simulation is considered sufficient for training purposes

Conclusions

- Phisycally based models of soft tissues
- Physical simulation of hard tissues
  - Accurate numerical methods for analysis (bone remodeling, bone fracture analysis)
  - Fast simulation methods for training (real-time bone deburring, phaco-emulsification)
- Trade-off between accurate modeling and simulation (off-line analysis) and fast simulation (training)

That's all, folks...

- More info: www.crs4.it/
- Q&A: Your turn...