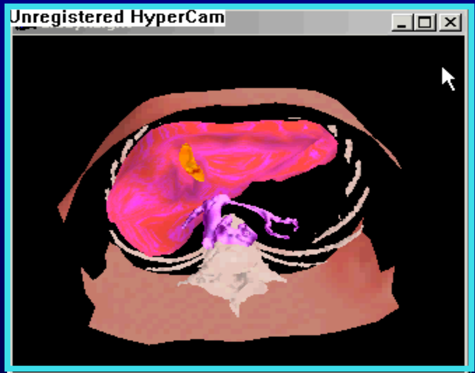
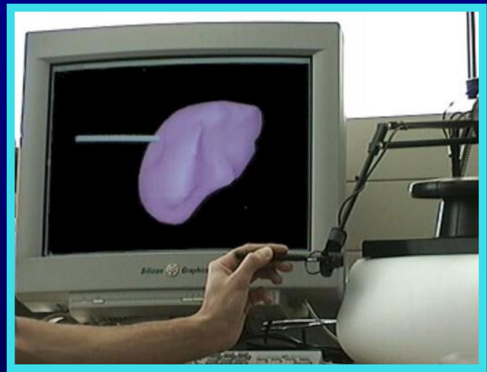
Dynamic simulation and 3D interaction

C. Mendoza, C. Laugier & O. Galizzi, F. Faure

INRIA Rhône-Alpes





Overview

1- Motivations & Problems

2- Physical modeling & Reality-based modeling

***3- 3D interaction** (virtual cutting, haptic interaction)*

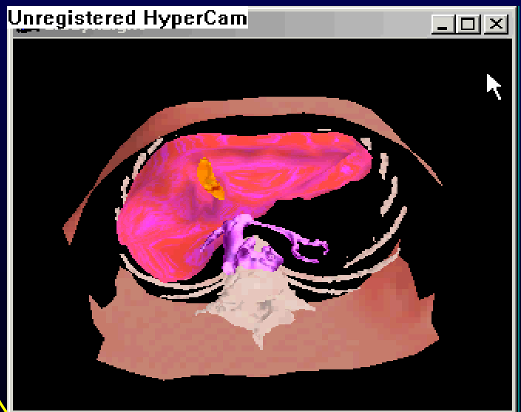
5- Conclusion & Future work

Motivations & Problems

Training surgeons for new surgical procedures (MIS)

« Virtual patient » as an alternative to conventional training *(mechanical endotrainers, animals, human patients)*

=> Very useful for training surgeons to new surgical procedures, no risk for the patient, and possibility to generate arbitrary anatomies & pathologies



… Much more difficult than flight simulators !

- **Soft tissues, non linear and heterogeneous**
- **Physical consistency** *(anatomy & bio-mechanical properties)*
- **Basic *real-time* 3D interactions:** *palpation, cutting, suturing…*

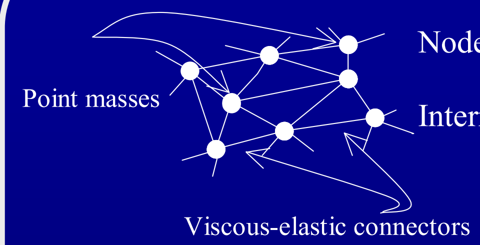
- I -
Physical modeling & Reality-based modeling

Main Approaches to physical modeling

- Mass – Spring Networks (MS-Net) *[Terzopoulos88,Aulignac99]*
- Boundary Element Methods (BEM) *[James&Pai99]*
- Long Element Methods (LEM) *[Costa&Balaniuk00] [Sundaraj&Laugier01]*
- Finite Element Models (FEM) *[Bathe96]*
  - Condensed FEM *[Cotin97]*
  - Explicit FEM *[Cotin97,O'Brien99]*
  - Multiresolution *[Debunne00]*

Physical modeling : *MS-Net & FEM principles*

MS-Network



Node dynamics : $m\ddot{x} + b\dot{x} + \sum F^{i} = \sum F^{ext}$

Internal forces : $F^{ij} = (-\lambda\,\Delta d_{ij} - \mu\,\dot{d}_{ij})k$

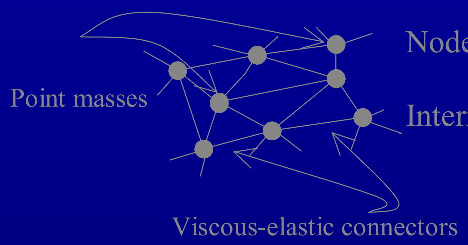
Rigidity factor Damping factor

Example of non-linear deformation :

$$F = (\lambda_1 d^3 + \lambda_2 d) + \mu\,\dot{p} \quad ;\ where \quad d = \frac{l - l_0}{l_0}$$

Physical modeling : *MS-Net & FEM principles*

MS-Network



Node dynamics : $m\ddot{x} + b\dot{x} + \sum F^{i} = \sum F^{ext}$

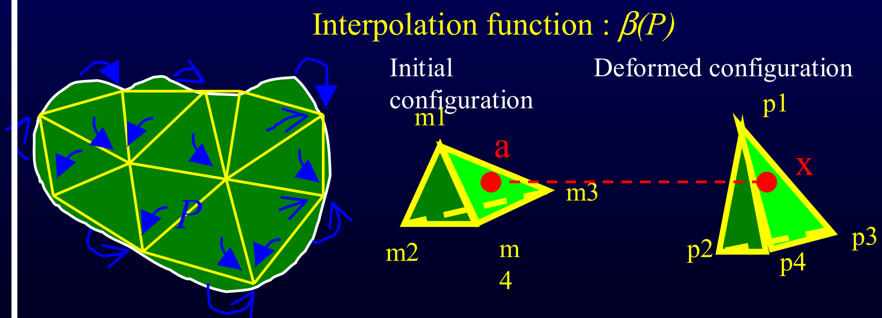
Internal forces : $F^{ij} = (-\lambda \Delta d_{ij} - \mu \dot{d}_{ij})k$

Rigidity factor Damping factor

Example of non-linear deformation :

$$F = (\lambda_1 d^3 + \lambda_2 d) + \mu \dot{p} \quad ; \quad where \quad d = \frac{l - l_0}{l_0}$$

FEM



- **Measuring deformations:** *Strain tensors ε (Almansi, Cauchy, Green…) related to displacements **U***

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial x}{\partial a_i}\frac{\partial x}{\partial a_j} - \delta_{ij}\right)$$

Green-Lagrange Strain Tensor
(large deformations, invariant to rotations)

- **Dealing with material information:**

*Stress tensors σ related to forces **F***

$$KU = \sum F_{ext}$$

Displacement Forces

Static Formulation

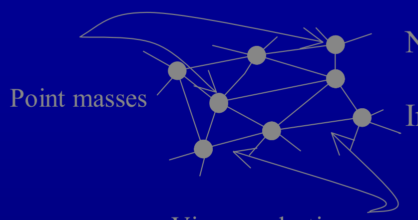
$$M\ddot{U} + D\dot{U} + KU = \sum F_{ext}$$

Mass matrix Damping matrix Stiffness matrix

Dynamic Formulation
(adding inertia & viscosity effects)

Physical modeling : *MS-Net & FEM principles (2)*

MS-Network



Node dynamics : $m\ddot{x} + b\dot{x} + \sum F^{i} = \sum F^{ext}$

Internal forces : $F^{ij} = (-\lambda \Delta d_{ij} - \mu \dot{d}_{ij})k$

Rigidity factor — Damping factor

Example of non-linear deformation :

$$F = (\lambda_1 d^3 + \lambda_2 d) + \mu \dot{p} \quad ; \quad \text{where } d = \frac{l - l_0}{l_0}$$

Explicit FEM

Consider the physics of each tetrahedron independently instead of merging them into a large matrix system => more appropriate for interactive topological modifications

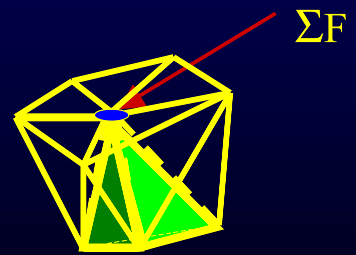
Mass Lumping: *To concentrate the effects of the system on the nodes*

[Bathe96] [Cotin97 mass-tensor][O'Brien99]

$$m_i \ddot{u} + d_i \dot{u} + k_i u = f$$

Dynamics of each node

$$F = -\frac{vol}{2}\sum_{j=1}^{4} \mathbf{p} \sum_{k=1}^{3} \sum_{l=1}^{3} \beta_{jl}\beta_{ik}\sigma_{kl}$$



System solving (ODE)

see [Daulignac 01]

- **Static Resolution**

  => Principle of « Virtual Work » : Internal and External forces perfectly balance

  – **Linear case** *(e.g. [Bro-Nielsen & Cotin 96])*

  $$K \cdot x = f_{ext}$$

  => No large strain, no Rotation, no material non-linearity

  – **Non-linear case** *(e.g. [Daulignac 01])*

  $$K(x) \cdot x = f_{ext}$$

  => Stiffness matrix K changes with displacement, making use of an iterative numerical solution, e.g. Newton-Raphson

  $$K(x_i) \cdot \Delta x_i = f_{ext} - f(x_i) = r_i$$
  $$x_{i+1} = x_i + \Delta x_i$$

System solving (ODE)

see [Daulignac 01]

- **Static Resolution**

  => Principle of « Virtual Work » : Internal and External forces perfectly balance

  – **Linear case** *(e.g. [Bro-Nielsen & Cotin 96])*

  $$K \cdot x = f_{ext}$$

  => No large strain, no Rotation, no material non-linearity

  – **Non-linear case** *(e.g. [Daulignac 01])*

  $$K(x) \cdot x = f_{ext}$$

  => Stiffness matrix K changes with displacement, making use of an iterative numerical solution, e.g. Newton-Raphson

  $$K(x_i) \cdot \Delta x_i = f_{ext} - f(x_i) = r_i$$
  $$x_{i+1} = x_i + \Delta x_i$$

- **Dynamic integration**

  $$M\ddot{x} + D\dot{x} + Kx = f_{ext}$$

  2nd order non-linear differential equation system

  Convert to 1st order system ⇒

  $$Y = \begin{pmatrix} x \\ \dot{x} \end{pmatrix} \quad \text{and} \quad f(Y) = \begin{pmatrix} \dot{x} \\ \ddot{x} \end{pmatrix}$$

  – **Explicit integration**

  Forward Euler: $Y_1 = Y_0 + h\, f(y_0)$

  s-order Runge-Kutta: $Y_1 = Y_0 + \sum_{j=1}^{s} b_j k_j$ *with:* $k_j = f\left(Y_0 + h \sum_{i=1}^{s} a_{ji} k_i\right)$

  – **Implicit integration**

  Backward Euler: $Y_1 = Y_0 + h\, f(Y_1)$

  Implicit Euler (non-linear system)

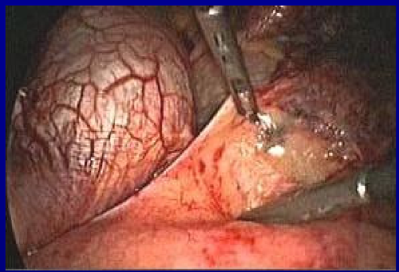
  Linearisation ⇒

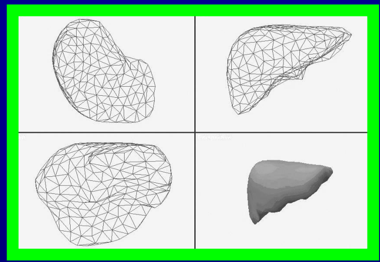
  $$\Delta Y \left[\frac{1}{h} I - \lambda \frac{\partial f}{\partial Y}\Big|_{Y = Y_0} \right] = f(Y_0)$$

  Semi-implicit Euler

Towards « Reality-Based Modeling »

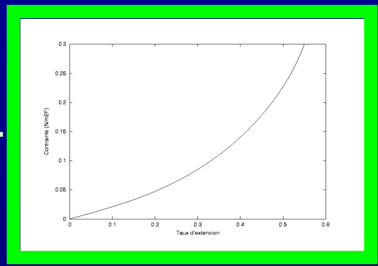
- **Medical simulator (Virtual liver)** *[Boux & Laugier 99]*



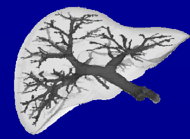


3D reconstructed model

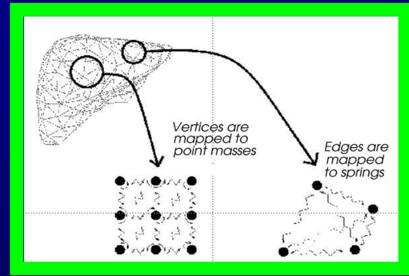
+



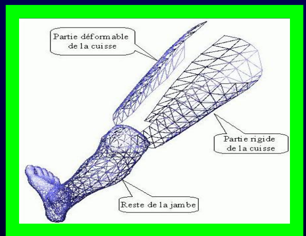
Stress-strain curve (litterature)



Three anatomic components:
- the Glisson capsule
- the Parenchyma
- the Vascular network

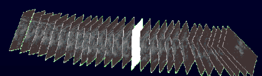


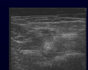
- **Echographic simulator** *Inria + Tim-c + UC-Berkeley* *[Daulignac & Laugier 00]*



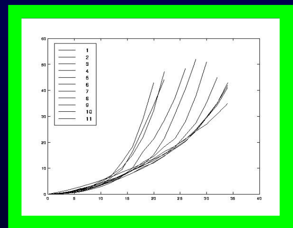
Geometric model

+





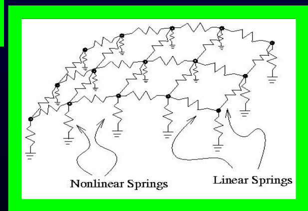
Measured data

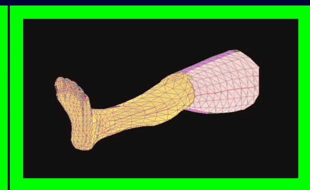


Stress-strain curves (measured)

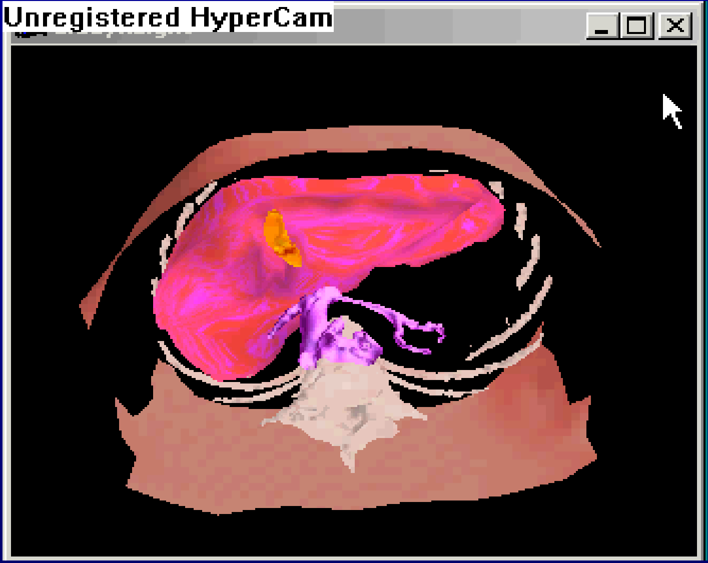
$$F = k\Delta x \quad (linear)$$

$$F = \frac{\Delta x}{a\Delta x + b} \quad (non-linear)$$

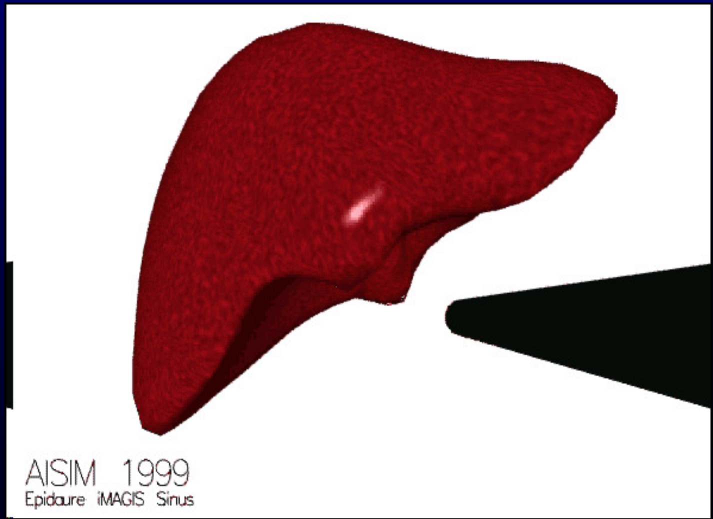




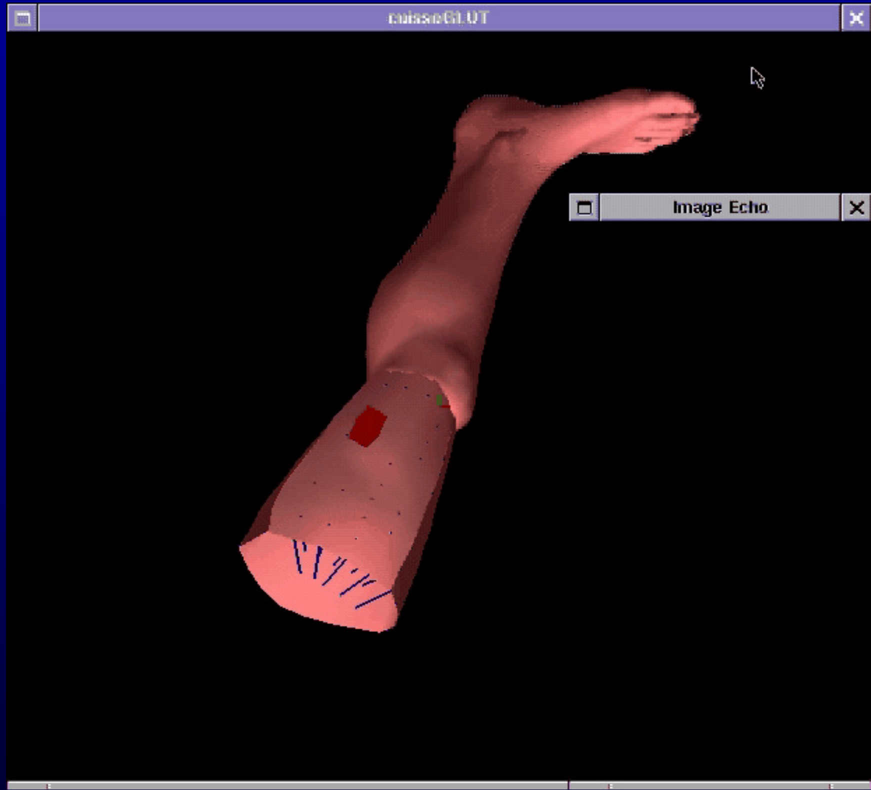
Medical simulators : *Some experimental results*



Explicit FEM *[Mendoza 02]*



Multi-resolution model & Rendering
[Debunne & Cany 99]



Echographic simulator using MS-Net
(deformation, haptic interaction, echographic images)
[Daulignac & Laugier 00]

- I I -

3D Interaction

Collision detection , Virtual cutting

Haptic interaction

Processing collisions *(Coldetect library)*

Download possible from : http://www.inrialpes.fr/sharp/coldetection/

- **Collision checking & Interpenetration volume evaluation**

  => To be performed at interactive rate (detecting 3D interactions, computing reaction forces)

- **Hierarchies & Pruning techniques**

  => Convex decomposition & Object hierarchies (AABB) & Pruning elements within the convex hulls intersection

  ⇒ Constant-time for «collision» of convex components (based on GJK$^+$)

  ⇒ Linear-time w.r.t the intersection of the convex hulls for « interpenetration »

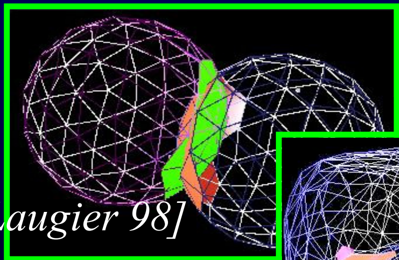
  [Joukhadar & Laugier 96] [Sundaraj & Laugier 00]

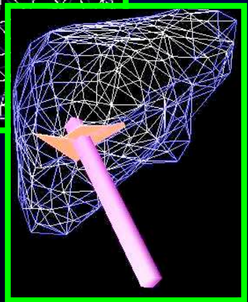
- **Collision response**

  => Non-linear penalty method [Hunt & Crossley 75][Deguet & Laugier 98]

$$\vec{F}_{collision} = -\lambda x^n - \mu \dot{x} x^n \mid n \cong 1$$

$$\mu = \frac{3}{2}\alpha\lambda \Rightarrow e = 1 - \alpha v \qquad e: \text{coeff restitution Poisson}$$

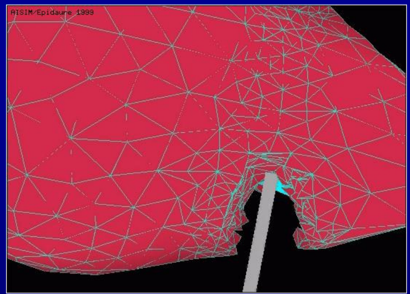


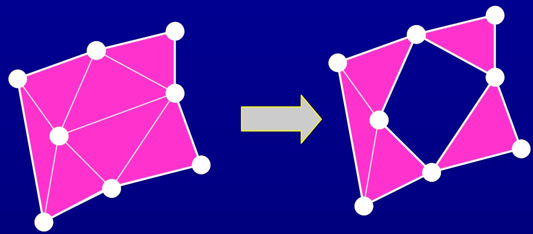


AABB : Axis Aligned bounding Boxes

Interactive Virtual Cutting : *Previous work*

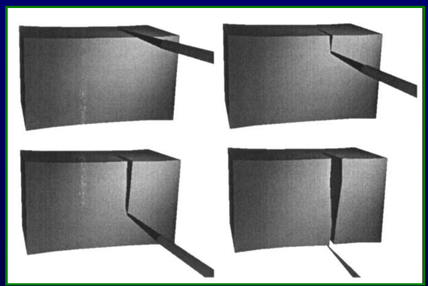
- **"Destruction" approach** *[Terzopoulos88][Norton91][Cotin97]*

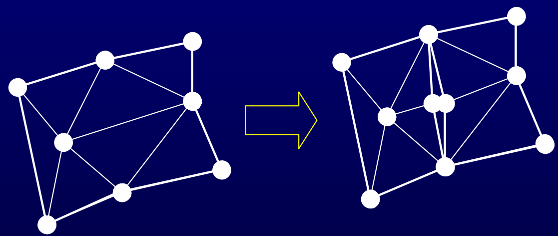




- Fine discretization is required *(increases processing time)*
- Destroy material *(not realistic in some cases, easy to implement)*

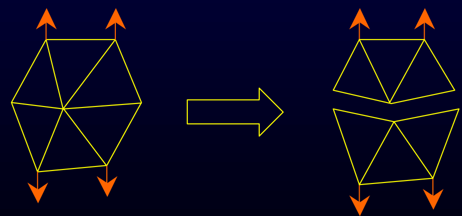
- **"Subdivision" approach** *[Bielser99][MorK00]*





- Realistic results
- Large increase # primitives *(bad for real-time computations)*

- **"Separation" approach** *[Boux00] [Mendoza01]*

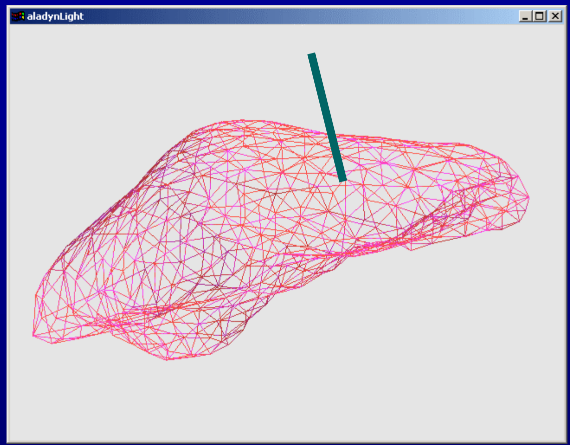


- Realistic results
- Low increase # primitives *(good for processing time & 3D interaction)*

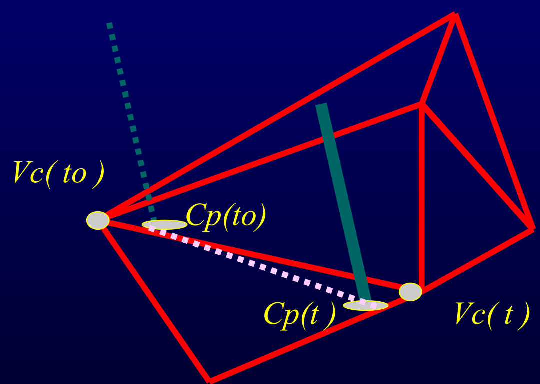
Interactive Virtual Cutting : *Our approach*

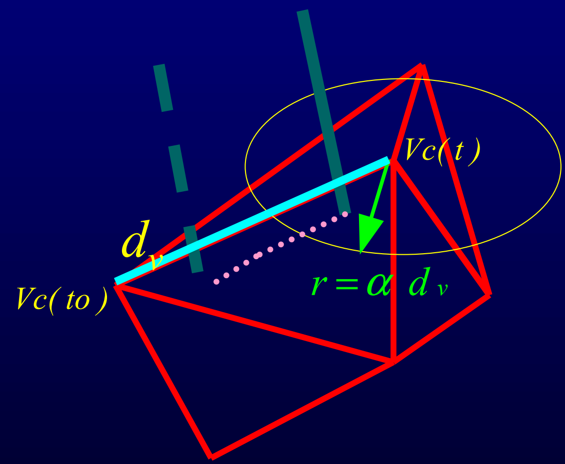
- **Step 1: Breaking the material**
  - ✓ Geometrical criteria *(cutting attempt & neighborhood condition)*
  - ✓ Physical criteria
- **Step 2: Select & Separate tetrahedrons**
  - ✓ Separation approach
  - ✓ Singularities processing (connection with zero area)
- **Step 3: Local Remeshing**
  - ✓ Avoiding degenerated tetrahedra

Step1: Breaking the material *(Geometric criteria)*



=> *Interpretate* the movements of the virtual tool on the *surface* of the object





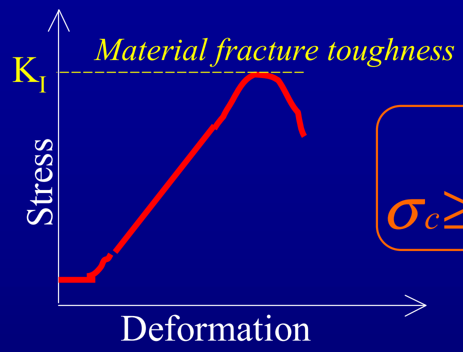
Potential cutting attempt

$$v_c(t) \neq v_c(t_0)$$

Neighborhood condition
(avoiding degenerated cuts)

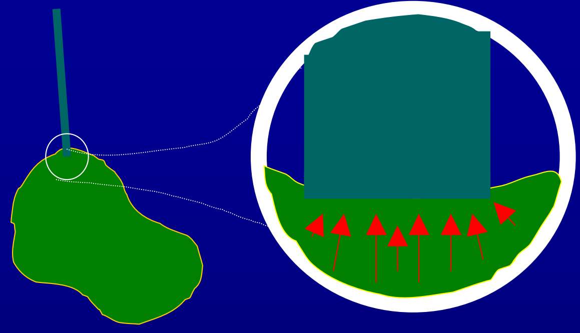
$$C_p(t) \in Neighbor\ (v_c(t))$$

Step1: Breaking the material *(Physical criteria)*



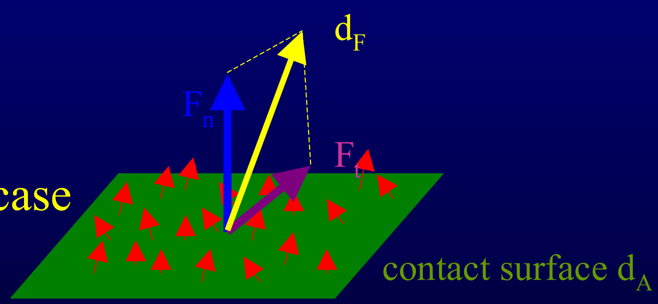
Breaking condition :

$\sigma_c \geq K_I \qquad \sigma_c$ = *Cutting stress*

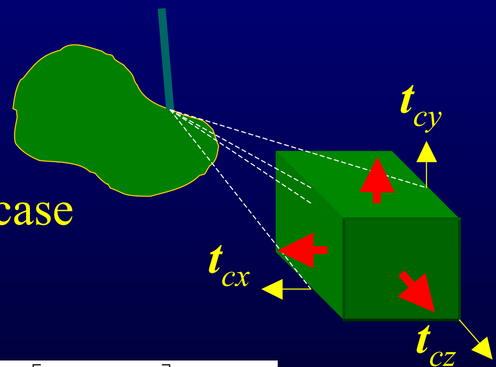


Virtual scalpel & Object interaction => Internal stresses

2D case



3D case



Contact Surface => Sharpness factor (κ)

Normal stress (σ), Shear stress (τ)

$$t_c = \frac{1}{\kappa}(\sigma \vec{n}_1 + \tau \vec{n}_2) \qquad \sigma = \lim_{d_A \to 0} \frac{|F_n|}{d_A} \qquad \tau = \lim_{d_A \to 0} \frac{|F_t|}{d_A}$$

Cutting traction vector

$$\begin{bmatrix} tc_x \\ tc_y \\ tc_z \end{bmatrix} = \begin{bmatrix} \frac{\sigma_{xx}}{\kappa} & \frac{\tau_{xy}}{\kappa} & \frac{\tau_{xz}}{\kappa} \\ \frac{\tau_{yx}}{\kappa} & \frac{\sigma_{yy}}{\kappa} & \frac{\tau_{yz}}{\kappa} \\ \frac{\tau_{zx}}{\kappa} & \frac{\tau_{zy}}{\kappa} & \frac{\sigma_{zz}}{\kappa} \end{bmatrix} \begin{bmatrix} \vec{n}_1 & \vec{n}_2 & \vec{n}_3 \end{bmatrix} = T_c = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix} \begin{bmatrix} \vec{n}_1 & \vec{n}_2 & \vec{n}_3 \end{bmatrix}$$

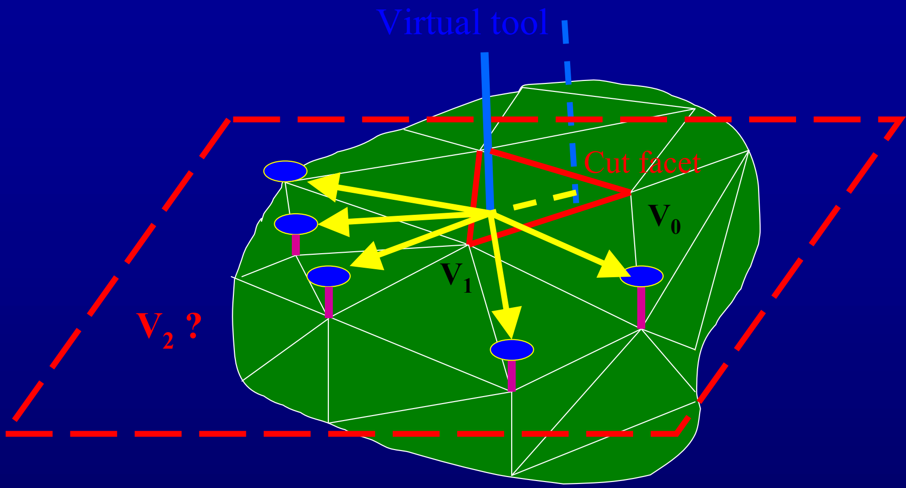
$\sigma 1$ $\sigma 2$ $\sigma 3$ *Principal stresses*

Cutting Stress

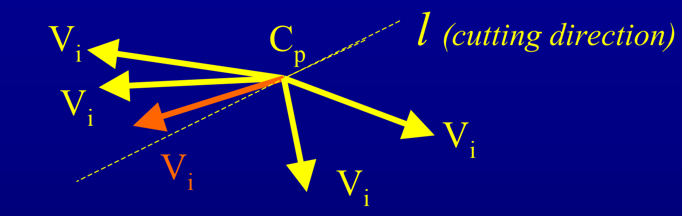
$$\sigma_c = \frac{1}{\zeta} \sigma_{max}$$

ζ : Damage

Step2: Select & Separate the tetrahedra

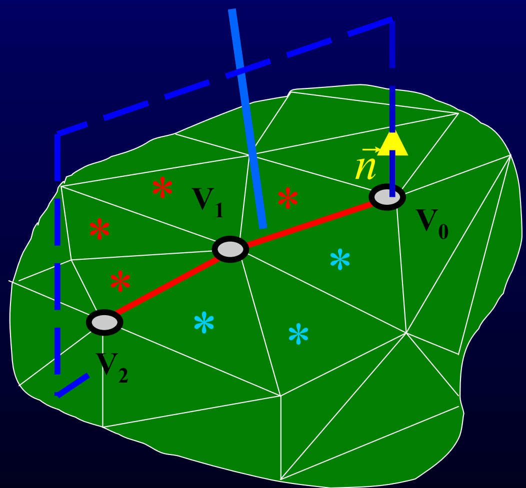


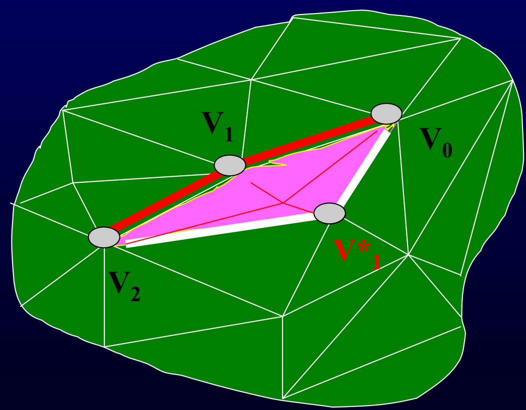
(1) Selecting the next involved node V_2



$V_2 = V_i$ *such that* $\min_i \langle V_i C_p,\ l \rangle$

(2) Selecting & Separating the involved tetrahedra



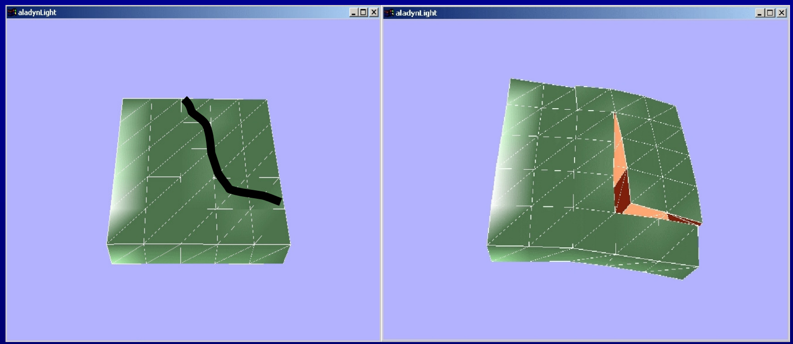


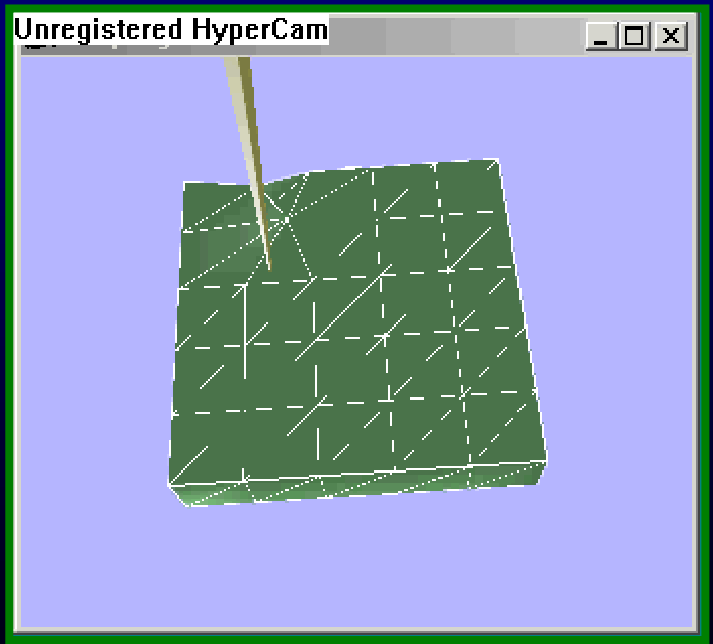
Mass of V_1 divided by two

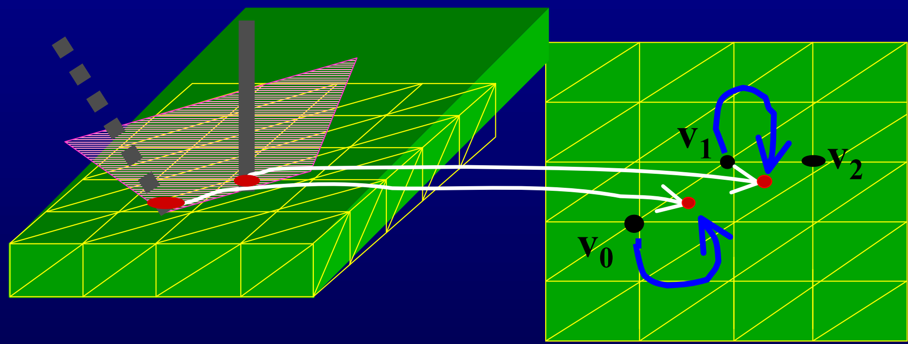
Step3: Local remeshing

=> Creating new facets & Modifying the tetrahedra shapes

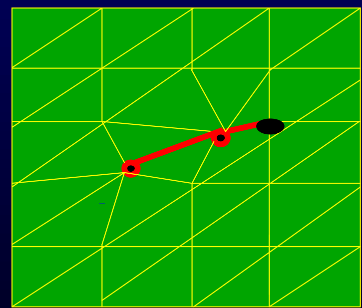
(for reflecting the profile of the cut)



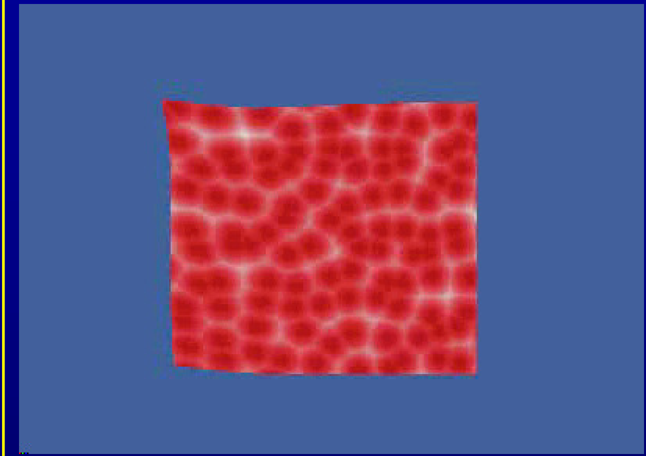




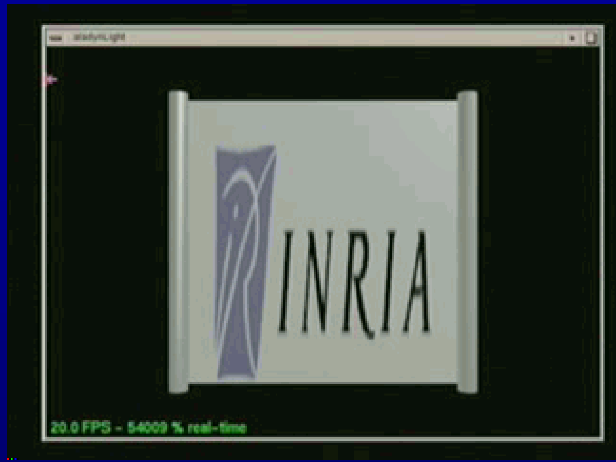
Modify accordingly the interpolation matrix



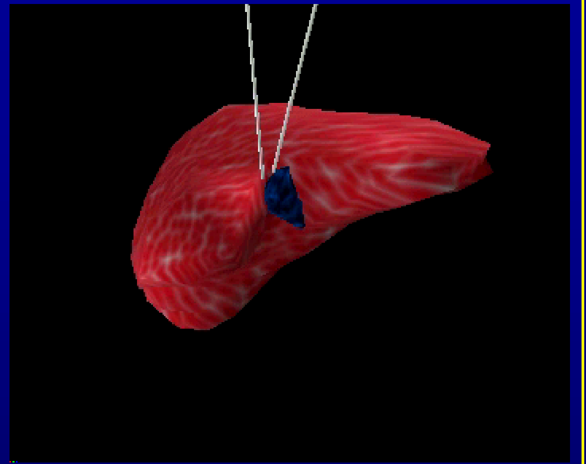
Interactive Virtual Cutting : *Experimental results*



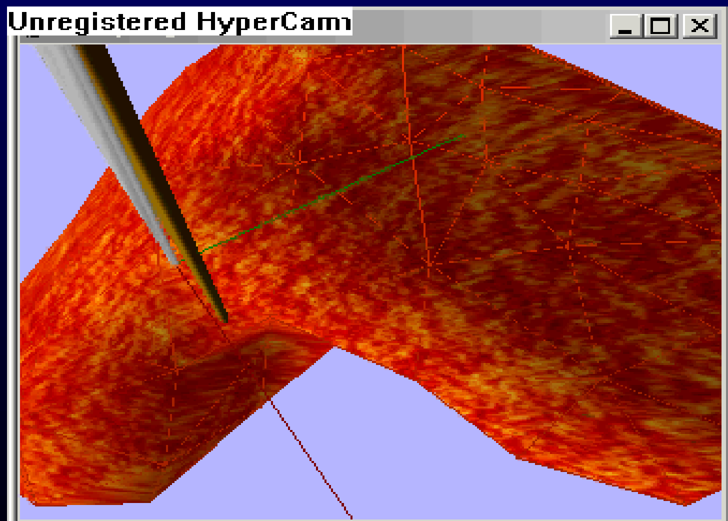
Tearing a 2D skin (MS-net)
[Boux & Laugier 00]



Cutting a flag using an haptic interface (MS-net)
[Boux & Laugier 00]

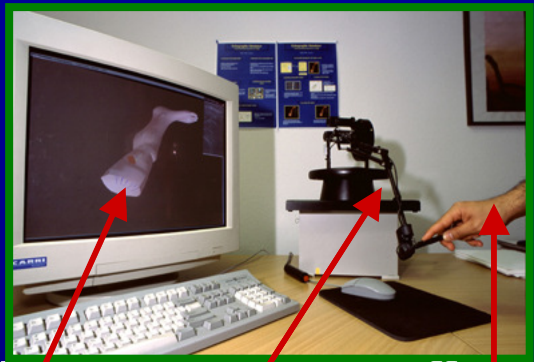


Separating the Gall-bladder from the liver (FEM + MS-net)
[Boux & Laugier & Mendoza 01]



Cutting a deformable volume using an haptic interface
[Mendoza & Laugier 03]

Haptic interaction : *Problem & Previous work*



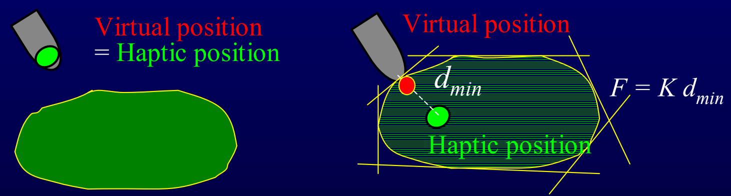
Surgery Simulator : Deformations & 3D-Interactions (*collisions, topology changes*) at **~ 30 Hz**

Difference frequency rate problem

Touch Sense : Need forces at **300 Hz to 10 KHz**

Penalty method

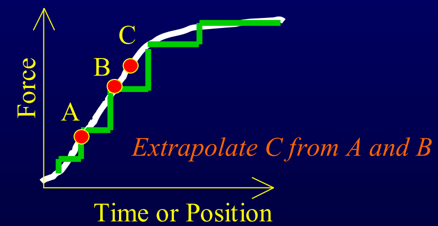
[Zilles&Salisbury94][Ruspini&Khatib97][Balaniuk&Laugier99]



=> Only based on surface & distance information

Extrapolation method

[Ellis et al. 97] [Piccibonno01]



=> Numerical errors, Sensitive to discontinuities

Low order models

Norton equivalent [Astley&Hayward97], Model reduction [Cavasoglu& Tendick00], Multiresolution [Debunne00]

=> Precomputations not suitable for topological changes

Haptic rendering : *Volumetric Buffer Model*

[Mendoza & Laugier 03]

Simulation Loop

Physical model

Collision detection
Deformations
Topological modifications

Haptic Loop

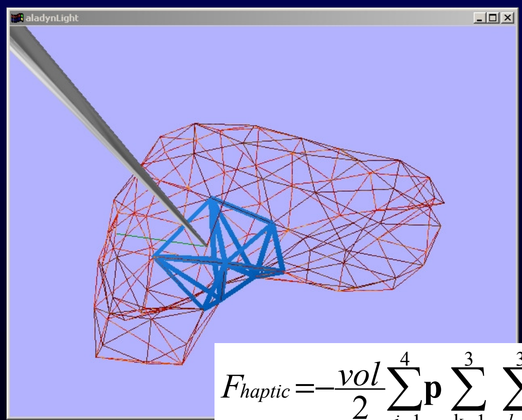
Buffer model

Tool-Object collisions
&
Haptic force computation

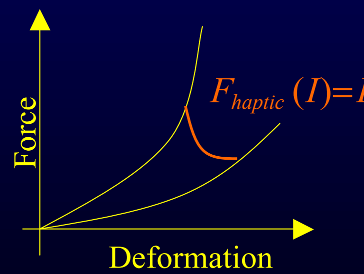
Graphical Rendering

Volumetric Buffer Model (VBM)

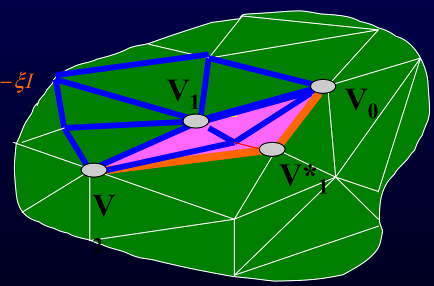
- A more simple *explicit FEM model* constructed from the original model
- Allows the computation of realistic contact forces at haptic frequencies
  (the haptic device interacts with the VBM instead of the complete physical model)



$$F_{haptic} = -\frac{vol}{2}\sum_{j=1}^{4}\mathbf{p}\sum_{k=1}^{3}\sum_{l=1}^{3}\beta_{jl}\beta_{ik}\sigma_{kl}$$

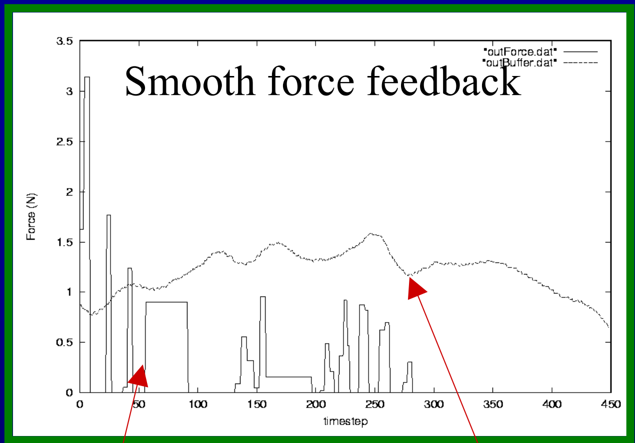


[Mahvash & Hayward 01]



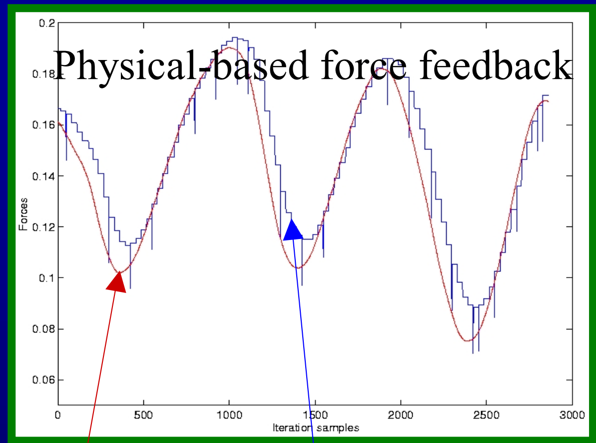
Haptic rendering of cutting operations

Haptic rendering : *Experimental results*



Force without a buffer model

Force with a buffer model



Haptic forces ~ Physical simulation forces

Virtual liver & Haptic interaction

Conclusion

- **Reality-Based Modeling of soft tissues**
  - ✓ Several types of physical models can be used depending of the mechanical characteristics of the application (MS-Net, FEM …)
  - ✓ Explicit FEM more appropriate when interactive topological modifications are required

    => Realistic, Large deformations (Green-Lagrange tensor), Real-time

- **3D interaction & Virtual cutting**
  - ✓ Efficient & Robust Collision processing for rigid objects and soft tissues *(Coldetect)*
  - ✓ Interactive Cutting procedures

    => Geometric & Physical criteria, Controling the complexity growth using the separation approach (real-time performances), Local remeshing (realism)

- **Realistic haptic interaction**
  - ✓ Volumetric Buffer-Model based on explicit FEM

    => Local model constructed from the original model, Contact & Force feedback generated at the haptic frequency, Smooth & Realistic Force feedback