I N R I A

Institut National de Recherche en Informatique et en Automatique

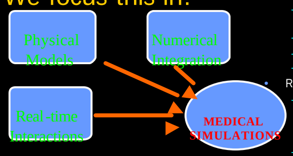
National Institute for Research in Computer Science and Control

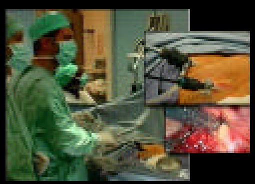
A Public French Scientific and Technological Institut under the auspices of the Ministry of Research and the Ministry of Industry

Kenneth SUNDARAJ

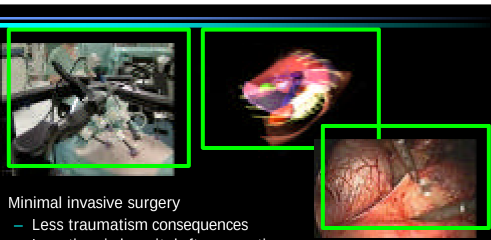
Physical Models, Resolution Methods and Real-time Interactions for Soft Tissue Simulations in Medical Simulators

We focus this in:





- Motivations
  - Education, training & planning
  - Security
  - Cost & ethics
  - Certification
- Requirements for Realism
  - Volumetric models of the human body that capture the internals of organs, etc
  - Bio-mechanical descriptions of the properties of different body parts
  - Interactive, real-time, realistic deformation of volumetric models
  - Topology modifications
  - Haptic Interaction
  - Collision detection (haptic device to organs, organ to organ)
  - Real-time rendering of volumetric objects

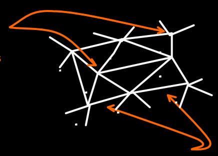


- Minimal invasive surgery
  - Less traumatism consequences
  - Less time in hospital after operations
  - Increased techniques for medical students
    - *No more use of corpses or animals (less expensive, non-unique utilisation; non ethical problems)*
    - *Medical simulator : eternal patient; rares pathologies simulation; quantitative evaluation methods*
    - *Virtual reality techniques: visual advantages, touching sensations*

ORGANIZATION

- **Physical Models**
  - **Mass-spring models**
  - **Finite element method (FEM)**
  - **Long element method (LEM)**
- NumericalResolution
  - Dynamic Solution
    - Explicit Integration
    - Implicit Integration
  - Static Solution
- Real-time Interactions
  - Collision Detection
  - Haptic Interaction
  - Topology Modification

Mass-Spring Networks (MSN)

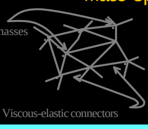


Node dynamics : $m\ddot{x} + b\dot{x} + \sum F^{i} = \sum F^{ext}$

Internal forces : $F^{ij} = (-l\,D d_{ij} - m\dot{d}_{ij})k$

Rigidity factor   Damping factor

Mass-Spring Networks : Example

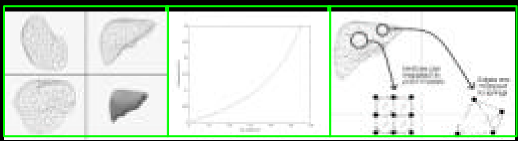


Node dynamics : $m\ddot{x} + b\dot{x} + \sum F^{i} = \sum F^{ext}$

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

Rigidity factor   Damping factor

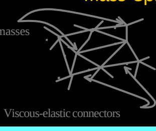
Virtual liver : Glisson capsule & Paremchyma *[Boux & Laugier 99]*



3D reconstructed model   Stress-strain curve (litterature)

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

Mass-Spring Networks : Example



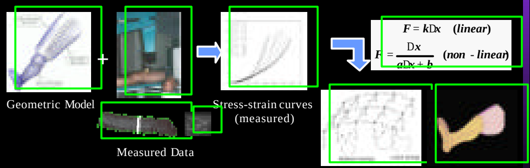
Node dynamics : $m\ddot{x} + b\dot{x} + \sum F^{i} = \sum F^{ext}$

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

Rigidity factor   Damping factor

Virtual Thigh : Echographic Simulator *[d'Aulignac & Laugier 00]*

INRIA + TimC + UC-Berkeley



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

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

Mass-Spring Networks : Example

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

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

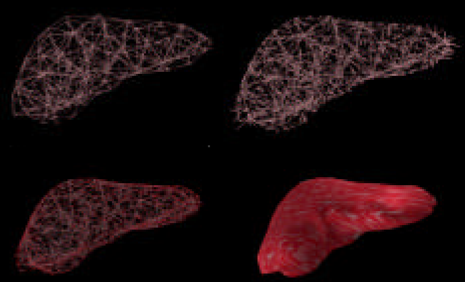
Rigidity factor   Damping factor

Quite easy to implement, Real-time, Topological modifications possible

BUT

Difficult to tune model parameters, Physical accuracy not guaranteed (it depends on the network topology)

Finite Element Method



Finite Elements Method (FEM)

- Green-Lagrange *Strain Tensor* : $\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial x}{\partial a_i}\cdot\frac{\partial x}{\partial a_j}\right) - \delta_{ij}$, $\delta_{ij} = \begin{cases}1 : i = j\\ 0 : i \neq j\end{cases}$

  a : a point in the underformed object
  x : corresponding point in the deformed object

- 2nd Piola-Kirchhoff *Stress Tensor* :

$$S = S^{(E)}_{ij} + S^{(V)}_{ij} = \sum_{k=1}^{3}\sum_{l=1}^{3} C_{ijkl}E_{kl} + \sum_{k=1}^{3}\sum_{l=1}^{3} D_{ijkl}V_{kl}$$

Elastic stress   Viscous stress   Stiffness matrix (relation between stress & strain)   Damping matrix

- For Hookean and Isotropic material : $S^{(E)}_{ij} = \sum_{k=1}^{3} \lambda E_{kk}\delta_{ij} + 2\mu E_{ij}$, $\delta_{ij} = \begin{cases}1 : i = j\\ 0 : i \neq j\end{cases}$

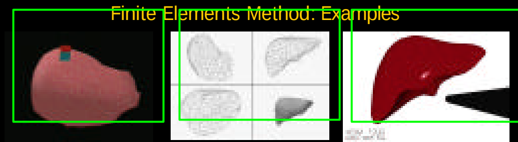
  Lamé coefficients

- 2nd order differential equation to solve :

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

Mass matrix   Damping matrix   Stiffness matrix

Finite Elements Method: Examples



Non-linear model *[d'Aulignac 01]*   Multi-resolution *[Debunne & Cany00]*

Physical realism (based on continuum mechanics)

BUT

Computationally more expensive *(~4 times slower than MSN)*, Pre-calculation and matrix condensation techniques often used for making interaction possible, Large deformations & Topological modifications difficult to process

Long Element Method (LEM)

Coop. Stanford

- **Based on continuum mechanics, while having a lower complexity than FEM**
- **Using bulk variables** *(pressure, density, volume ...)* **instead of stress-strain classical representations** *(forces, inertia, masses ...)*
- **Using "Pascal's Principle"** *(constant pressure in the object)* **& Volume conservation**

Non-linear elastic deformations of objects "filled" with some uncompressible liquid (good approximation for soft biological tissues)

"Pascal's Principle": *Any change in pressure is transmitted undiminished throughout the entire contained fluid*

LEM : Basic Idea

- Pressure & Stress:

  $P = F / A \qquad \varepsilon = E\,\Delta L / L$

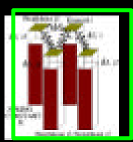
- Static equilibrium condition : *external pressure = internal pressure*

  $P_{atm} + E\,\Delta L / L = P_{fluid} + \rho g d$

  where $P_{ext} = P_{atm} + E\,\Delta L / L$ and $P_{int} = P_{fluid} + \rho g d$

- By introducing $\Delta P = P_{fluid} - P_{atm}$

  we obtain : $E\,\Delta L / L - \Delta P = \rho g d$



- Surface tension P_{st} :

  P_{st} is due to the neighboring long elements

  $P^i_{st} = \sum k_{ij}\,\Delta L_{ij} / A_i$ for all neighboring j

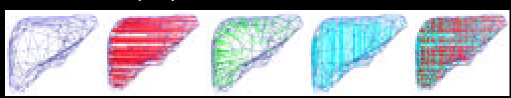
  $E\,\Delta L / L + P_{st} - \Delta P = \rho g d$

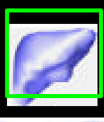
  Stress term, Surfurce Tension term, Pressure term, Gravity term

- Volume conservation : $\sum A_i\,\Delta L_i = 0$

LEM: Results & Characteristics

- **Object is discretized into long elements**
- **Virtual human liver :** *~400 facets, 1200 LE*
- **Model construction (preprocessing phase) : *450ms*** *on a PC Linux 450MHz*
- **Real-time simulation & interactions *(50Hz)***



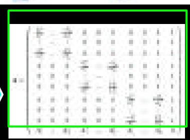


Hypotheses:

- Constant Gravity & Fluid density : $\rho_i g_i d_i = 0.01$
- Isotropic material : $E^x = E^y = E^z = constant$
- Uniform surface tension : $k_{ij} = constant$

$$(E_i / L_i)\Delta L_i - \sum_{neighbor\ j} (k_{ij} / A_i)\Delta L_{ij} - \Delta P = \rho_i g d_i$$

$$\sum_{all\ element\ i} A \Delta L_i = 0 \quad \text{(volume conservation)}$$

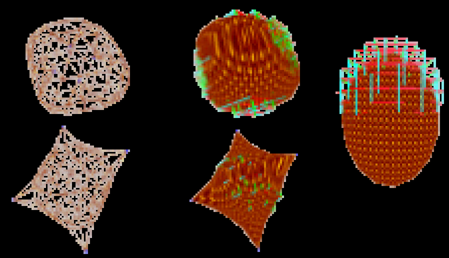


N+1 unknowns : $\Delta L_{i\,(i=1,n)}$, ΔP

Sparse matrix solved using the bi-conjugate gradient method

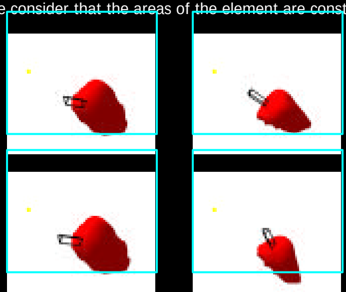
Small Deformations

- Deformations that are less than 5% strain.
- Poking and Pinching *[Sundaraj & Laugier, IROS01].*



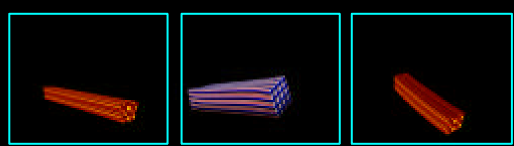
Small Deformations

- In this simulation, we do not update the state matrix.
- We consider that the areas of the element are constant.



Large Deformations

- In this simulation, the state matrix is updated
- Stretching, Bending & Twisting *[Sundaraj & Laugier, IROS02]*



<u>ORGANIZATION</u>

- PhysicalModels
  - Mass-spring models
  - Finite element method (FEM)
  - Long element method (LEM)
- **Numerical Resolution**
  - **Dynamic Solution**
    - **Explicit Integration**
    - **Implicit Integration**
  - **Static Solution**
- Real-time Interactions
  - Collision Detection
  - Haptic Interaction
  - Topology Modification

Dynamic integration

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

2nd order non-linear differential equation system

Convert to 1st order system → $Y = \begin{pmatrix} x \\ \dot{x} \end{pmatrix}$ *and* $f(Y) = \begin{pmatrix} \dot{x} \\ \ddot{x} \end{pmatrix}$

- **Explicit integration**: *forward Euler, Runge-Kutta*

h: timestep
Y1: next state
Y0: current state

  - ✓ Forward Euler (truncated Taylor serie) $Y_1 = Y_0 + h\, f(y_0)$

OK if first time derivative almost constant during h

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

Tries to approximate the s time derivatives

- **Implicit integration** : *backward Euler*

$Y_1 = Y_0 + h\, f(Y_1)$

Implicit Euler (non-linear system)
Evaluate the derivative at the next step

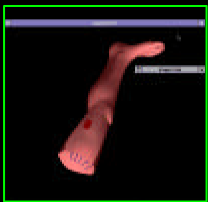
Linearisation →

$$DY\left[\frac{1}{h} I - l \left.\frac{\partial f}{\partial Y}\right|_{Y=Y_0}\right] = f(Y_0)$$

Semi-implicit Euler

Jacobian matrix at the current configuration

Dynamic integration: Experimental results



Explicit integration : *MSNet, cubic springs*

[Boux & Laugier 99]

Virtual liver :
370 Facets, 1151 Tetrahedra, 3399 Springs
150Hz on a SGI Onyx2

Implicit integration : *MSN, non-linear*

[d'Aulignac & Laugier 00]

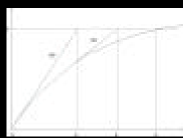
Echographic simulator :
370 non-linear Springs
100Hz on an Octane 175 Mhz

Static Resolution

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

$$M\ddot{x} + D\dot{x} + Kx = f_{ext} \quad \Rightarrow \quad K \cdot x = f_{ext}$$

- **Linear case** *(e.g. [Bro-Nielsen & Cotin 96])*
  - ✓ Pre-inversion of K (if enough space)
  - ✓ No large strain, no Rotation
  - ✓ No material non-linearity



- **Non-linear case** *(e.g. [Daulignac 01])*
  - ✓ Stiffness matrix K changes with displacement
  - ✓ A non-linear iterative numerical solution is required, *e.g. Newton- Raphson*

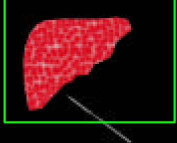
$$K(x_i) \cdot Dx_i = f_{ext} - f(x_i) = r_i$$
$$x_{i+1} = x_i + Dx_i$$

Residual force (to minimize)

OK if excitation frequency < 1/3 of object natural frequency
Sufficient for well-damped biological tissues

Static Resolution : Experimental results

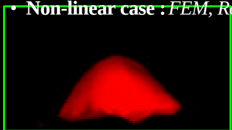
- **Linear case :** *LEM, Small deformation & Volume conservation*



Virtual liver :
800 facets, 2400 LEM
50Hz on a PC 450MHz

[Sundaraj, Laugier & Costa 01]

- **Non-linear case :** *FEM, Rotational invariance*



Virtual liver :
1157 tetrahedra, 60Hz on a SGI Onyx2
[d'Aulignac 01]

Pseudo-dynamic simulation

Quasi-Dynamic Resolution

- In some soft tissue simulation, a quasi-dynamic solution can be adapted. Then, we have a system of type :

$$A(x)\,x = b$$

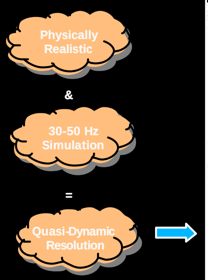
- Linear case (small deformations): **A** is constant.

Solution : LU , CG or Gauss methods.

- Non-Linear case (large deformations): **A** is not constant and needs to be updated at each time-step.

Solution :
  - 2 nested iterative schemes.
  - Iteratively use of *Successive Over Relaxation* (SOR).
  - Full or modified *Newton-Raphson* schemes can be used at each iteration.

Resolution Techniques (Quasi-Dynamique)



Algorithm 1 Newton-Raphson (NR) iteration

```
while true do
  update state matrix, K
  calculate pressure and gravity effects, U
  for i = 0 to n do
    SOR iteration for K · x = U
  end for
end while
```

Algorithm 2 Quasi-dynamic graphical display (25Hz)

```
t_last = t
while true do
  perform NR iteration
  if t > t_last + 0.04s then
    display configuration
    t_last = t
  end if
end while
```

ORGANIZATION

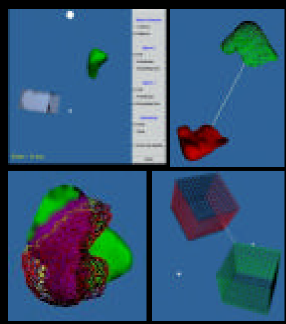
- PhysicalModels
  - Mass-spring models
  - Finite element method (FEM)
  - Long element method (LEM)
- NumericalResolution
  - Dynamic Solution
    - Explicit Integration
    - Implicit Integration
  - Static Solution
- **Real-time Interactions**
  - **Collision Detection**
  - **Haptic Interaction**
  - **Topology Modification**

Collision Detection

- A very crucial aspect of real-time interactions. In most simulators about 90% of the portion of a time step is spent on detecting interference.
- The main problem is to detect in interactive time, interference between virtual geometrical models.
- The problem becomes more complex when applied to human organs because these objects are deformable.
- In addition, we may need the following information:
  - *distance computation*
  - *contact localization*
- Algorithms need to be efficient and applicable to convex and concave objects.

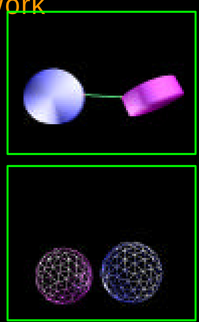
Algorithms for Collision Detection

- We have implemented the following algorithms:
  - Distance Computation *[K. Sundaraj et al., IROS 2000]*
  - Collision Test using AABB hierarchy
  - Contact Localization *[K. Sundaraj & C. Laugier, ICARCV 2001]*
- Implemented for convex and concave objects
- *Our experience: Hardware Open GL seems better for rigid objects but for high resolution deformable objects, we have noticed degradation in performance .*



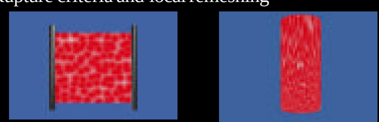
Current Work

- We are currently working on a implementation of interference detection for deformable objects.
- This will include provisions for
  - Large Global Deformations
  - Auto Collisions
  - Topology Modifications
- We hope to release a MESH COLLISION LIBARARY (MeshCol) in the near future.
- Our aim is to minimize memory utilization and maximize computation efficiency.



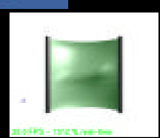
Topology modifications

- 2D & 3D tearing *[Boux-de-Casson-00][Boux-de-Casson, Laugier-00]*
  Rupture criteria and local remeshing



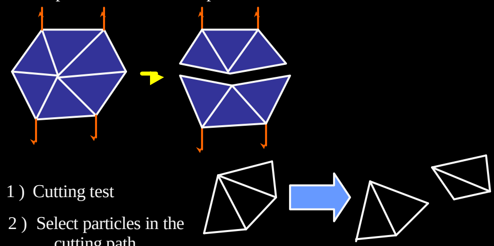
- 2D cutting *[Boux-de-Casson-00]*
  [Mendoza, Boux-de-Casson,Laugier-01]
  - No subdivision,
  - No destruction

  ⇒ *Separating the simulation elements*



Our approach : Main idea (2D & 3D)

1. Separate the simulation primitives

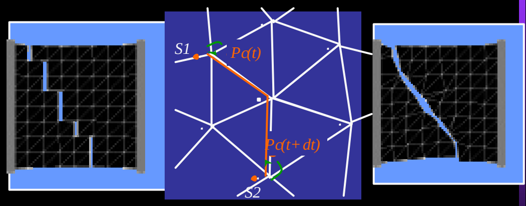


1) Cutting test

2) Select particles in the cutting path

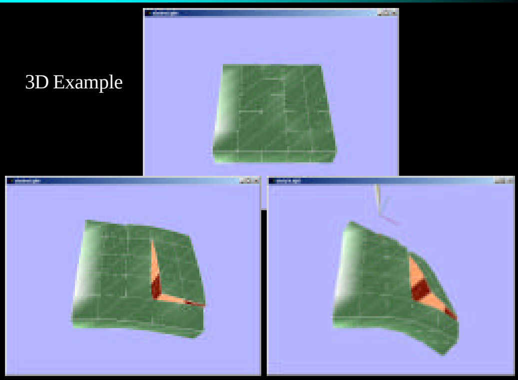
3) Change the topology

Projection of the selected particles into the cutting path

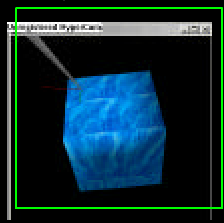


2D Example

3D Example

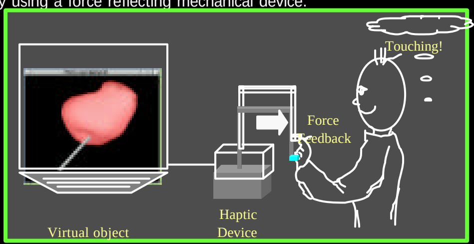


Cutting a deformable cube with 100 tetrahedrons using non-linear explicit finite elements.



Haptic Interaction

Haptics is the process of touching a virtual object by using a force reflecting mechanical device.

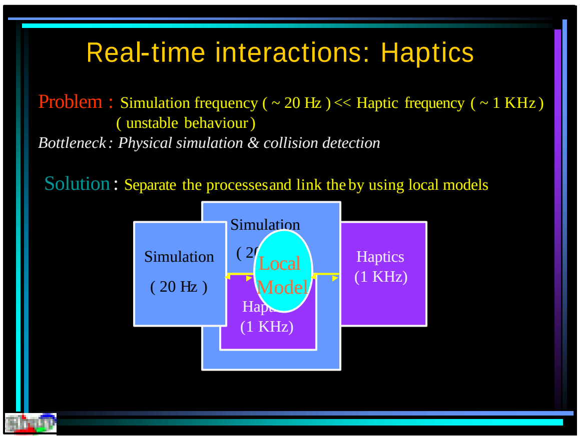


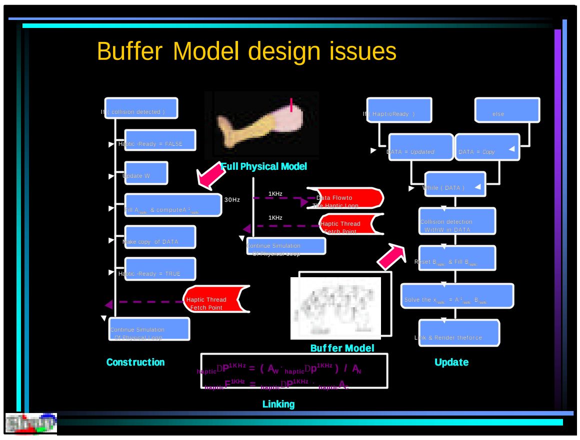
Main difficulties

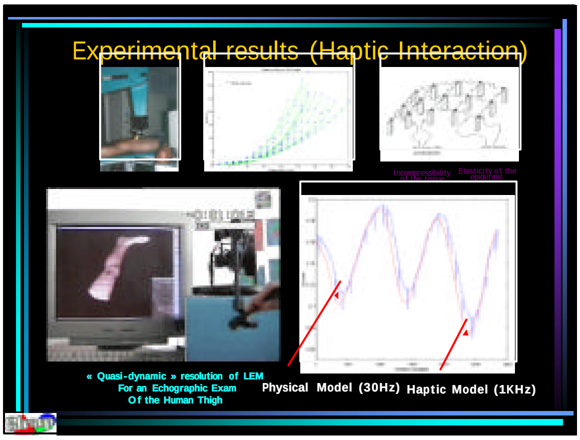
- **Mechanical instabilities:**
  - **Haptic interaction creates a "feedback loop" where physical energy flows bi-directionally between the user and the system** *Various factors may generate contact instabilities and oscillatory behaviors (because of the "non-passive" nature of the discrete implementation)*
  - **A solution for improving the "mechanical stability" consists in *"decoupling"* the haptic device control problem and the virtual scene generation**

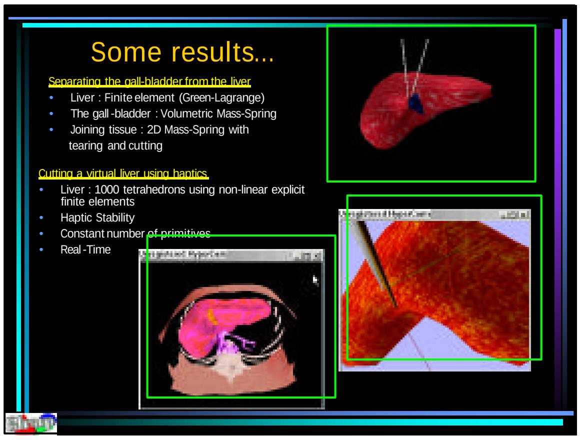
    Artificial coupling [Colgate 93], God-object [Zilles & Salisbury 94], Proxy [Ruspini & Khatib 97]

- **High frequency:**
  - **Perceptually convincing Haptic rendering also depends on *"high control bandwidths"*** *(Human sense of touch requires frequencies ranging from 300hz to 10Khz).* **This high frequency strongly limit the kind of application we can interface**
  - **A solution is to apply :**
    1- **A specific collision checker** *(simple and fast enough)*
    2- **A specific force estimation method***(constraint-based, simple "physics")*


Real-time interactions: Haptics
Problem : Simulation frequency (~ 20 Hz) << Haptic frequency (~ 1 KHz)
(unstable behaviour)
Bottleneck : Physical simulation & collision detection
Solution : Separate the processes and link the by using local models
Simulation
Simulation
(20 Hz)
Local
Model
Haptics
(1 KHz)
Haptics
(1 KHz)


Buffer Model design issues
Full Physical Model
30Hz
1KHz
1KHz
Buffer Model
Construction
Update
Linking


Experimental results (Haptic Interaction)
« Quasi-dynamic » resolution of LEM
For an Echographic Exam
Of the Human Thigh
Physical Model (30Hz)
Haptic Model (1KHz)


Some results...
Separating the gall-bladder from the liver
• Liver : Finite element (Green-Lagrange)
• The gall-bladder : Volumetric Mass-Spring
• Joining tissue : 2D Mass-Spring with tearing and cutting
Cutting a virtual liver using haptics
• Liver : 1000 tetrahedrons using non-linear explicit finite elements
• Haptic Stability
• Constant number of primitives
• Real-Time


Interested to know more...
• Our research team
– http://www.inrialpes.fr/sharp