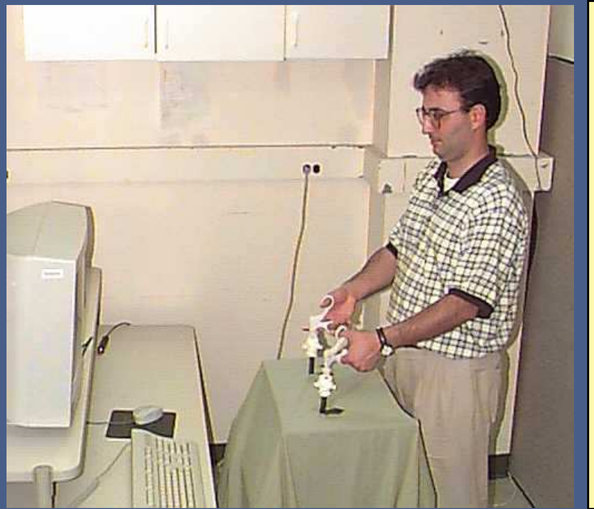
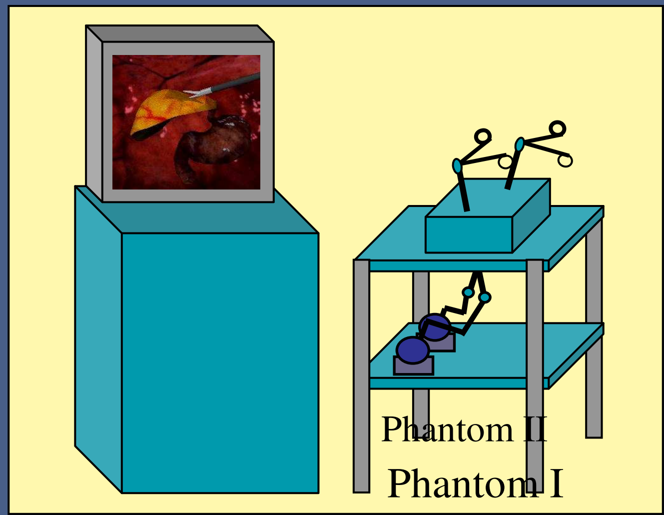
Simulation of Surgical Procedures in Virtual Environments





Cagatay Basdogan, Ph.D.
College of Engineering,
Koc University

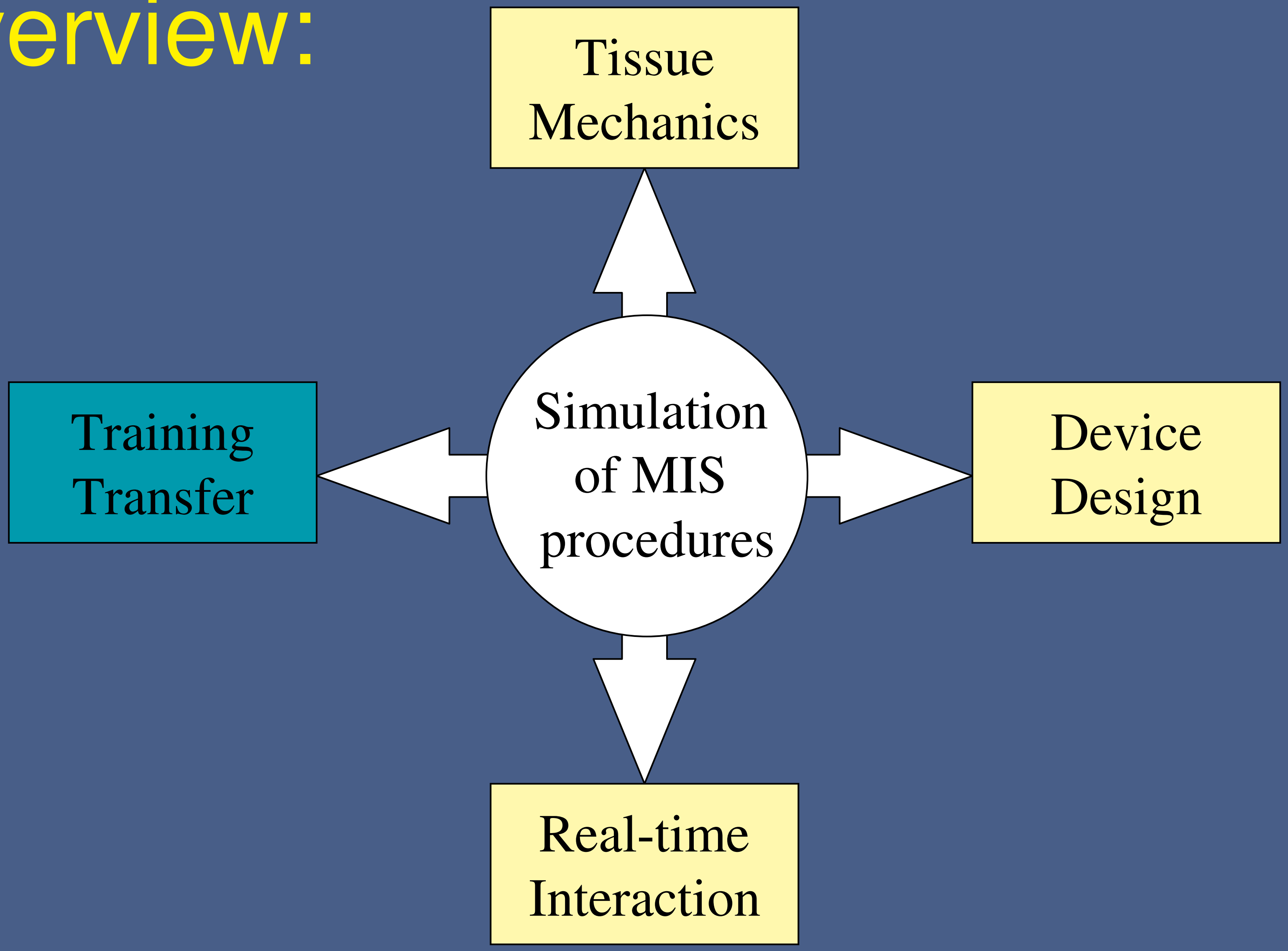
Notes & Equations & Matlab Code are online:
http://network.ku.edu.tr/~basdogan

Significance:

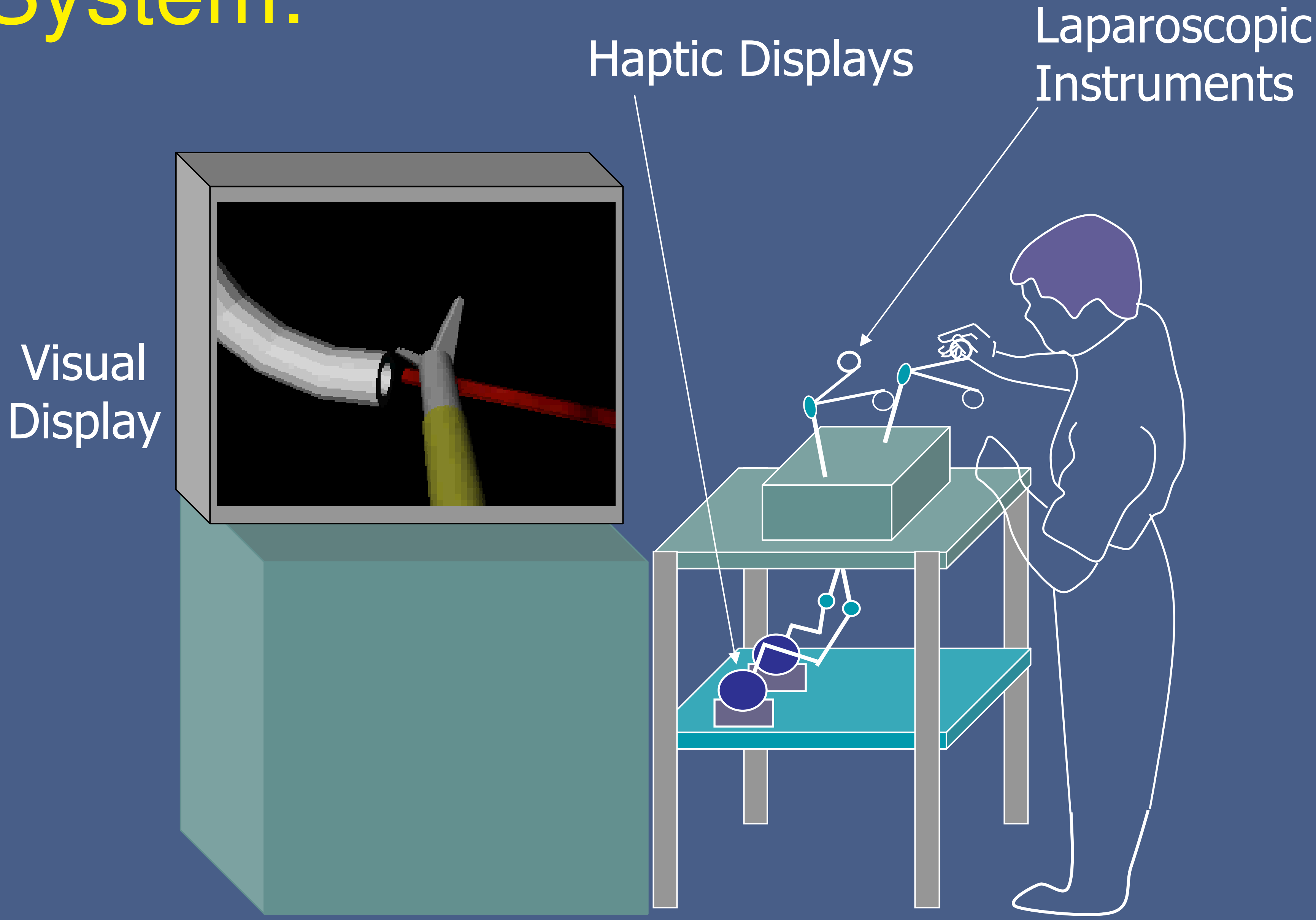
VR-based training

- to train and certify medical personnel
- to reduce the use of animals in training
- to test new surgical devices and procedures

Overview:


Tissue Mechanics
Training Transfer
Simulation of MIS procedures
Device Design
Real-time Interaction

System:


Haptic Displays
Laparoscopic Instruments
Visual Display

Challenges:

Physically-Based Modeling

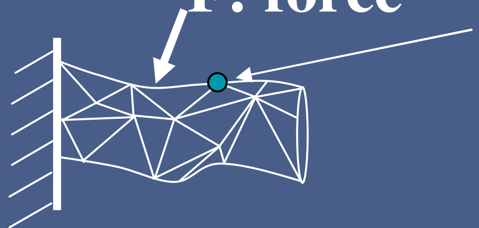
Loading via tool → Tissue/organ model → Deformation; Force on the tool

FEM ?

$$F = M\ddot{U} + C\dot{U} + KU \qquad \text{(dynamic analysis)}$$

F: force

Node (e.g. 100)



- **membrane+bending elements**
- **6-dof for each node**
- **Stiffness Matrix : K (600x600), diagonal M and C**

➡ **600 coupled equations !**

Real-time Display

- **Graphics (30 Hz)**
- **Haptics (1000 Hz)**

➡ **1 msec !**

Tissue Characteristics

- **nonlinear**
- **anisotropic**
- **hysteresis**
- **non-homogeneous**

➡ **complex !**

Topics:

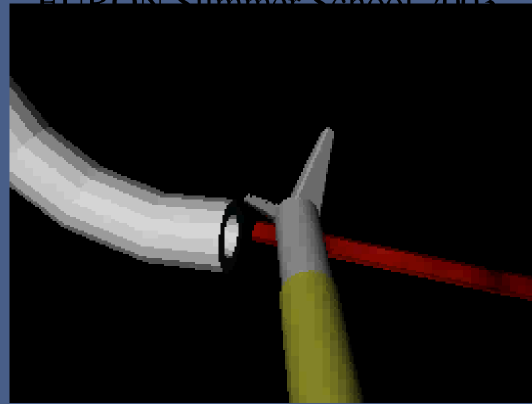
A) Collision detection and computational models of surgical instruments

B) Physically-based modeling for simulating soft tissue behavior

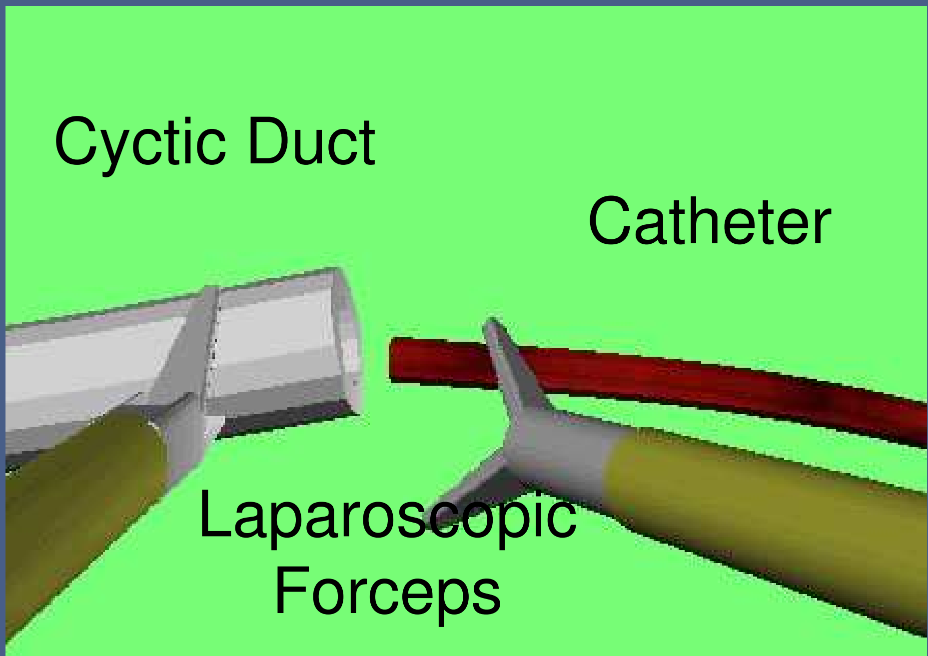
C) Haptic rendering of deformable objects

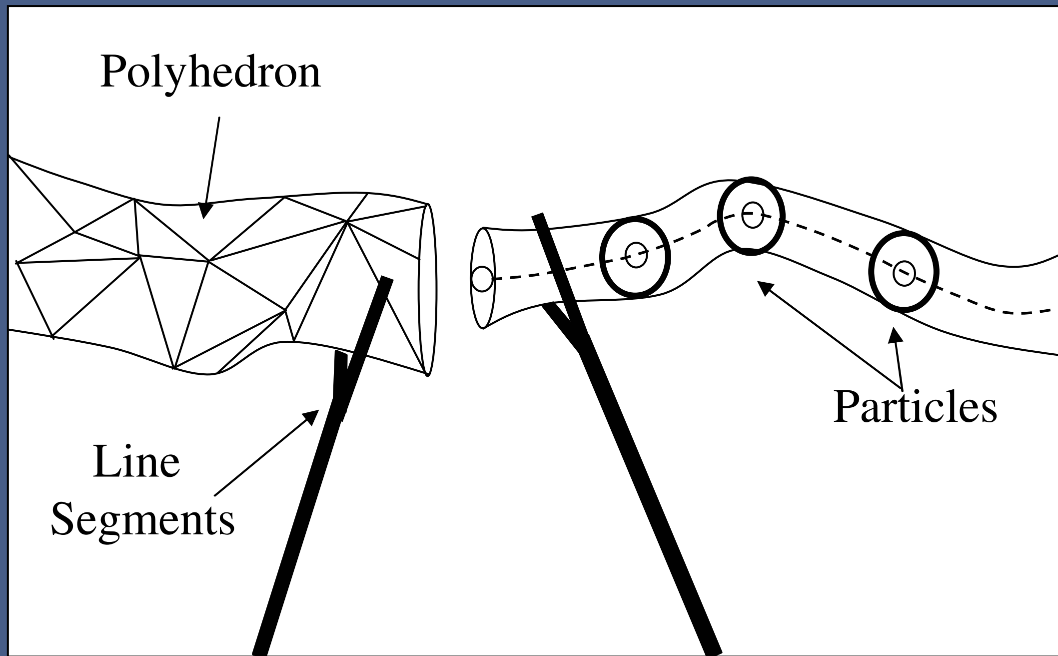
D) Software and hardware integration

Case Study:



Simulation of Catheter Insertion into the Cyctic Duct





What you see ...

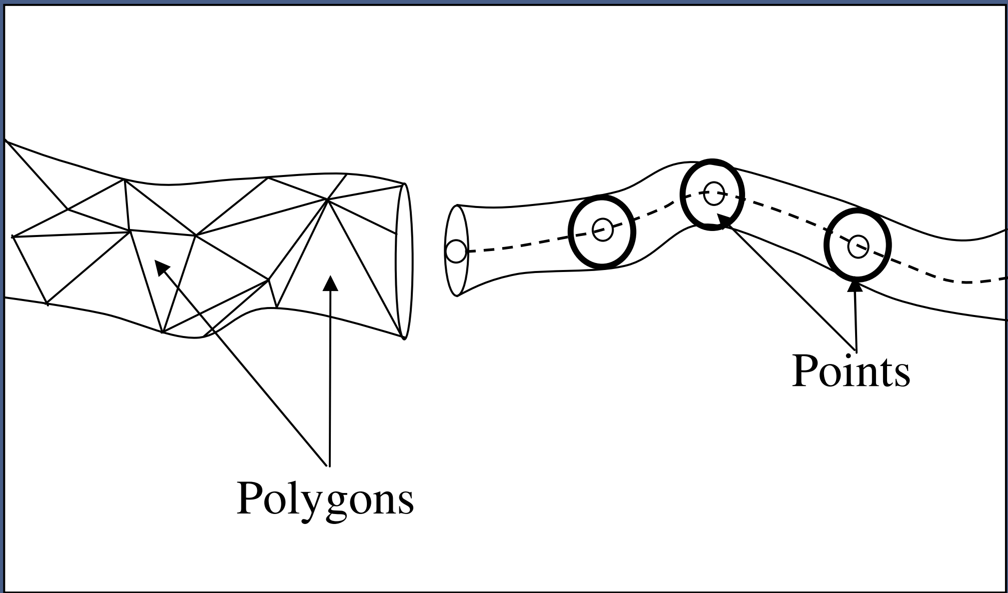
What is really happening ...

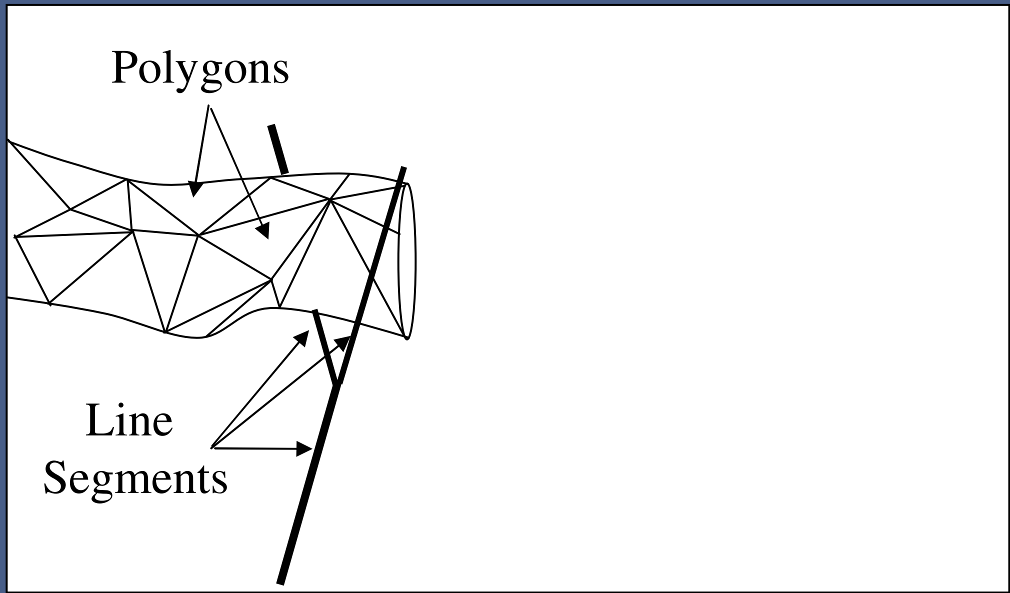
Principles of Collision Detection

(object-object is too expensive !)

Point-Object

Line Segment - Object





(e.g. catheter - cyctic duct)

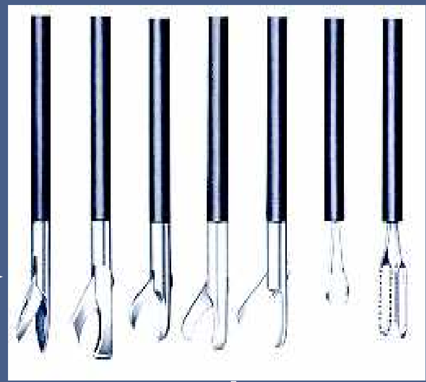
(e.g. forceps - cyctic duct)

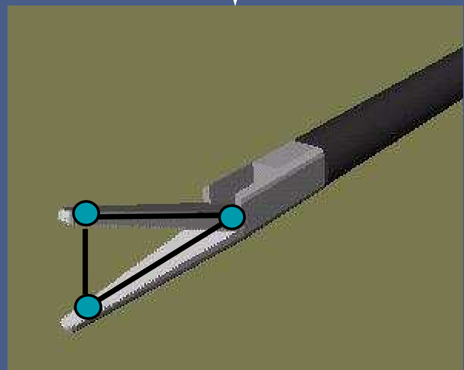
Computational Models of Laparoscopic Instruments

Group A

Point-based rendering

Group B





Ray-based rendering

Physically-based modeling for simulating soft tissue behavior

- Desired properties of deformable models
- Modeling of deformable objects
    1) particle-based
    2) FEM-based
- Implementing constraints
- Problems with particle-based techniques
- Problems with FEM techniques

Desired properties of force-reflecting deformable models

- reflect stable forces
- display smooth deformations
- handle various boundary conditions and constraints
- display “physically-based” behavior in real-time

Modeling of Deformable Objects

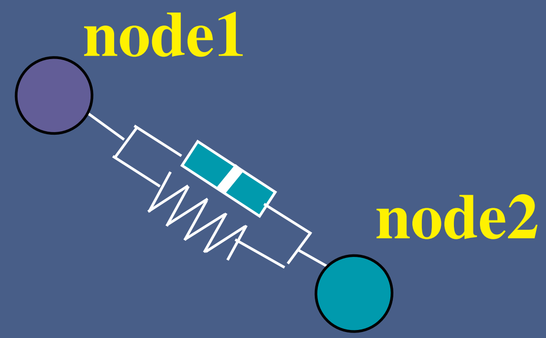
visit my web-site for the details : http://eis.jpl.nasa.gov/~basdogan

Particle-based :

$F = ma$

- F_{spring}
- $F_{damping}$
- $F_{gravity}$

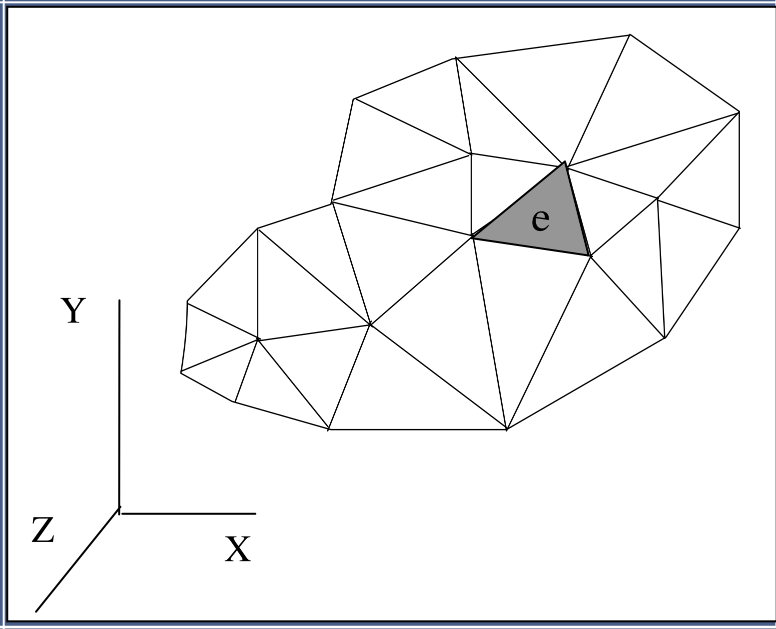
$$a(t + \Delta t) = F/m$$
$$v(t + \Delta t) = v(t) + \Delta t\, a(t + \Delta t)$$
$$p(t + \Delta t) = p(t) + \Delta t\, v(t + \Delta t)$$



FEM-based :

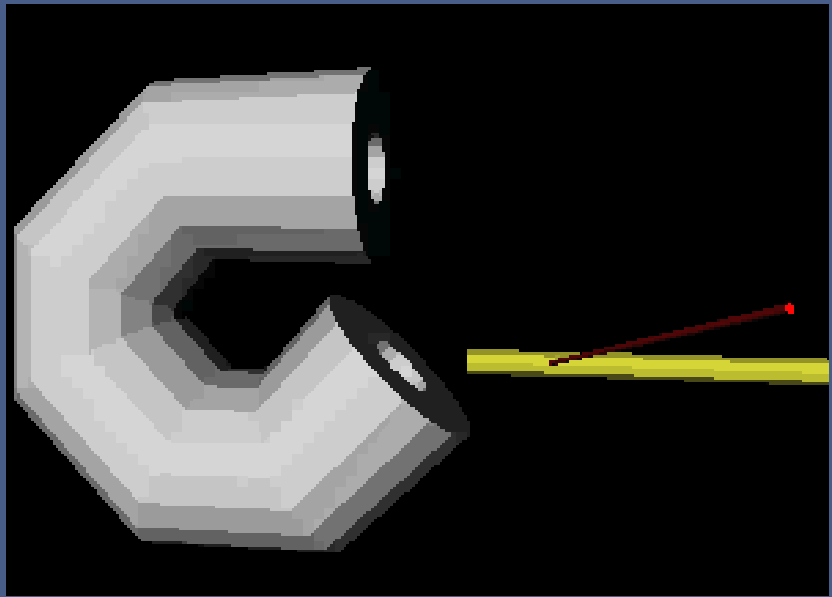
$F = KU$ (static analysis)

$F = M\ddot{U} + C\dot{U} + KU$ (dynamic analysis)



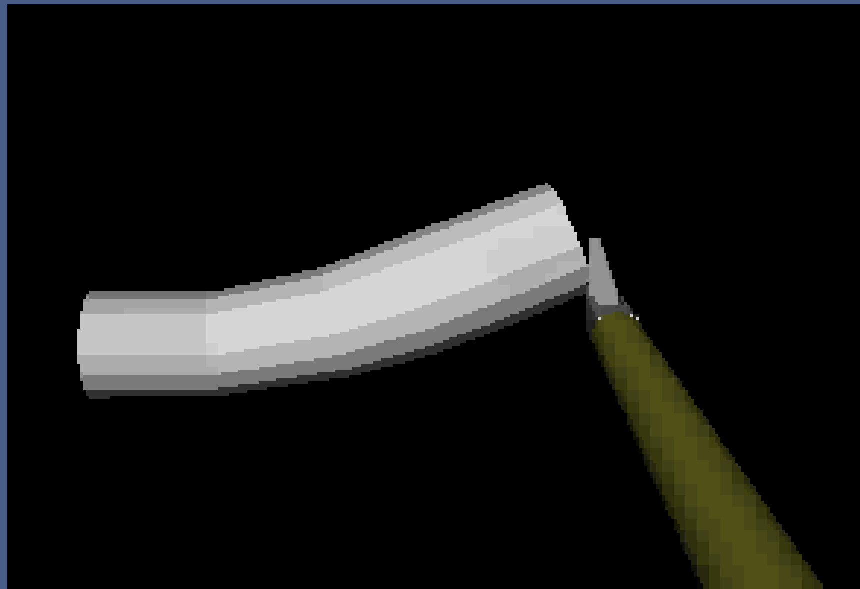
Comparison

Particle-based



easy to implement, flexible

FEM-based



comprehensive

Hybrid Approach

Constraints

(see my SIGGRAPH'99 Course Notes for details, http://eis.jpl.nasa.gov/~basdogan)

Examples:

- a node is fixed in 3D space
- a node is constrained to stay on a path
- curvature constraint
- constant volume

Implementation:

1) Particle-based models (Ref: Witkin/Baraff, SIGGRAPH'97 Notes)

   a) Penalty

   b) Lagrange multipliers

2) FEM

Problems with Particle-Based Techniques

A) Adding damping to stabilize oscillations

B) Adding constraints

C) Too many elements

→ stiffer system → $\Delta t \downarrow$ → # of iterations ↑

adaptive step size?

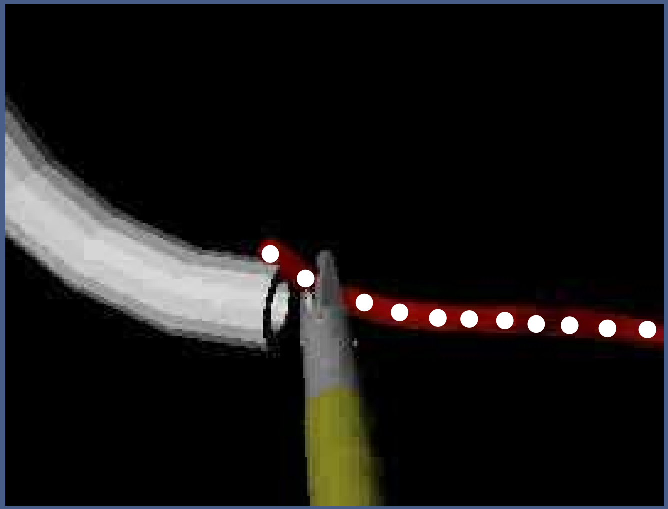
D) Too few elements → difficult to preserve volume

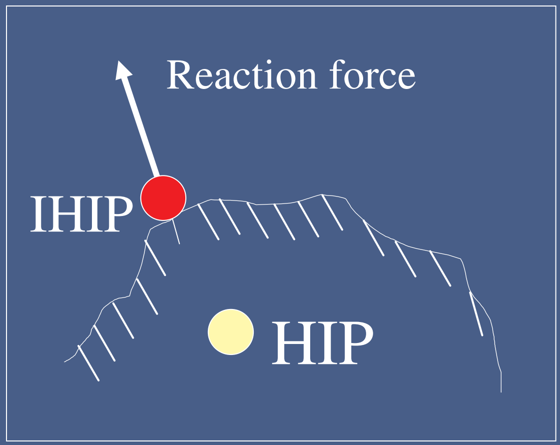
E) Non-homogeneous distribution of elements → finer adjustment of spring and damper coefficients

Problems with FEM Techniques

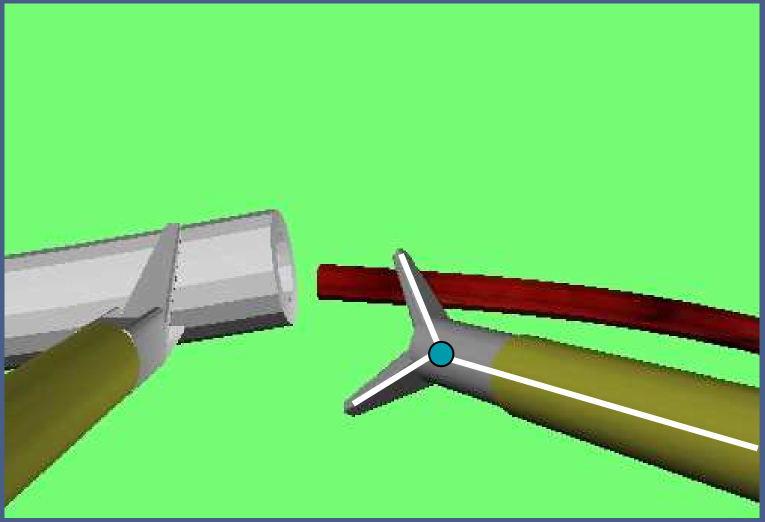
		requires
A)	Change in topology	→ Re-meshing
B)	Dynamic analysis	→ Modeling approximations
C)	Matrix inversion	→ Pre-computation
D)	Memory allocation	→ Simplifications in the geometry

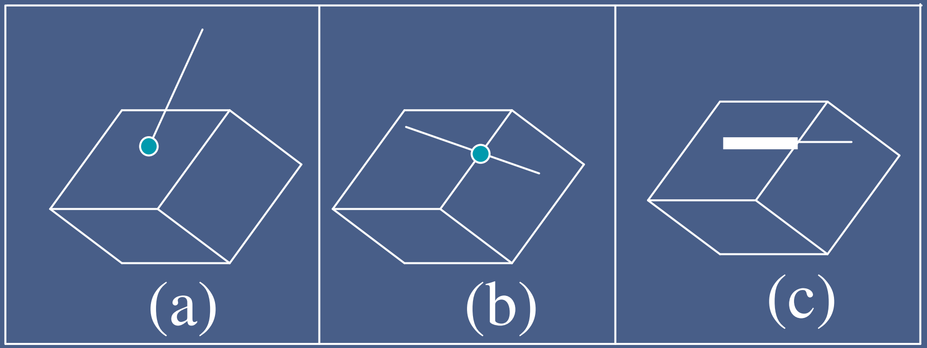
Point-based

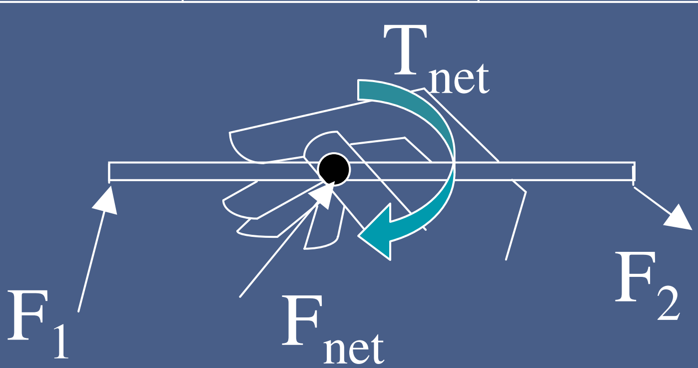




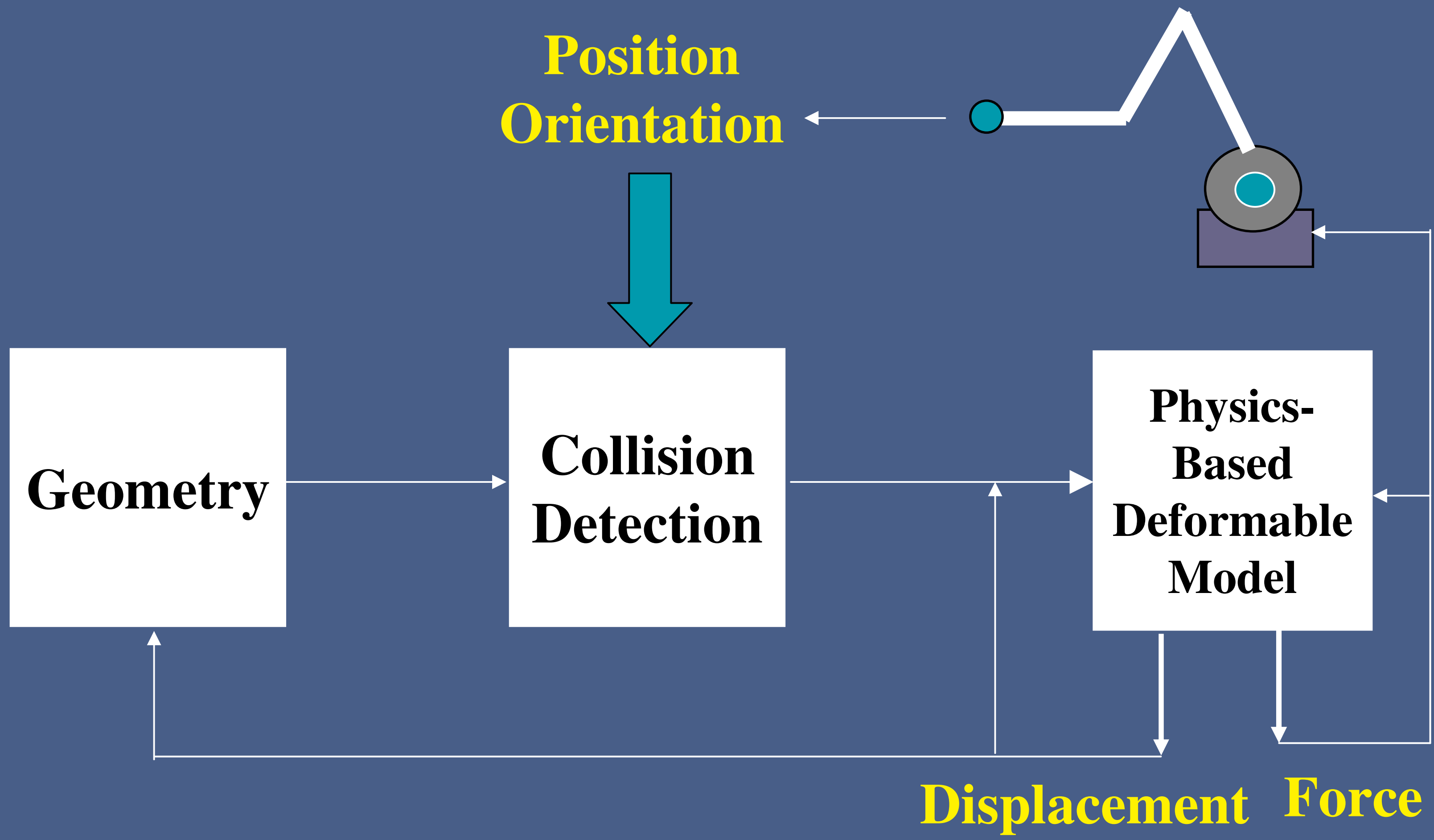
Ray-based







Computational Architecture



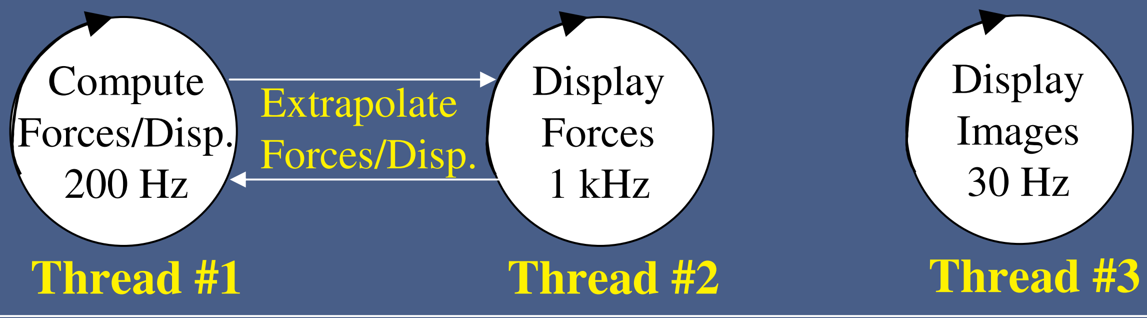
D) Software and Hardware Integration: tips and tricks

- Programming tips to speed up your computations
- Modeling tips to speed up your computations
- Simulation set-up

Programming tips to speed up your computations

- Synchronize your haptic and graphic loops through a shared database

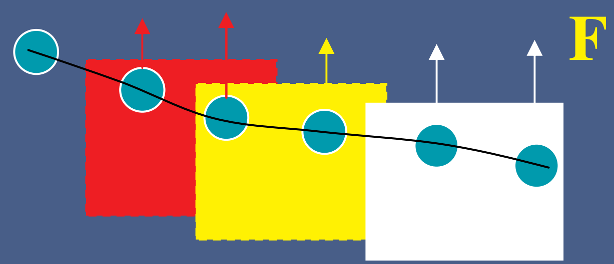
- Construct a multi-layered computing structure if possible



- Construct a hierarchical data structure
- Update your geometric coordinates less frequently

Δt_{haptic} = 0.001 sec (display forces)

$\Delta t_{iteration}$ = 0.01 sec (update coordinates)



Modeling tips to speed up your computations

- consider local deformation
- take advantage of single point interactions
- condense your matrices in FEM
- consider modeling approximations for dynamic FEM analysis
- pre-compute (matrices, unit displacements/force)
- loosely couple your force and deformation model
- take advantage of human perceptual limitations

local deformation

```
r = |vertex[i].coord - Collision Point |;
        if ( r < R_deformation )
            vertex[i].frozen = yes;
```

point-based interactions

$$U = K_i^{-1} F_i$$

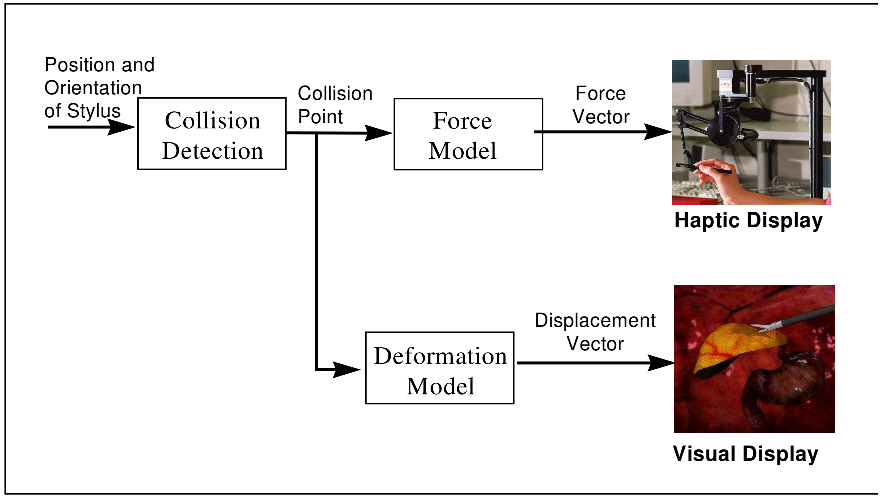
Multiply "i" is the i-th column of K^{-1} matrix and i-th entry of force vector

condensation

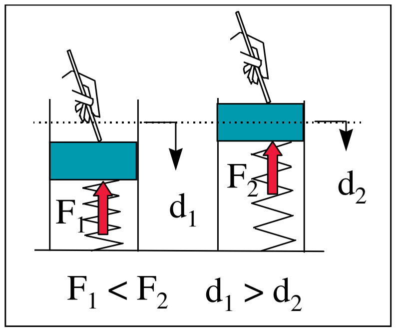
$$\begin{bmatrix} K_{MM} & K_{MS} \\ K_{SM} & K_{SS} \end{bmatrix} \begin{bmatrix} U_M \\ U_S \end{bmatrix} = \begin{bmatrix} F_M \\ F_S \end{bmatrix}$$

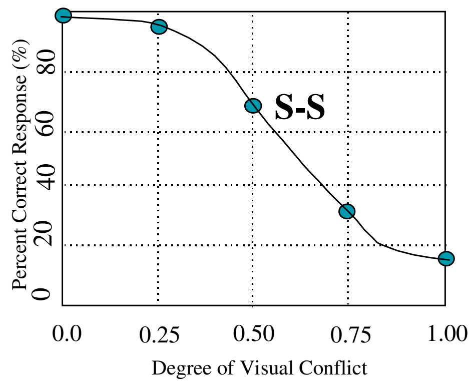
- eliminate internal elements
- eliminate rotational dofs

loose coupling of force and displacement models



human perceptual limitations





Modeling approximations:

Modal Analysis

$F = M\ddot{U} + C\dot{U} + KU$ (global coordinates, N COUPLED equations !)

$U = \Phi X$

$f_i = \ddot{X}_i + \alpha_i \dot{X}_i + \omega^2_i X_i$ $i = 1, \ldots, N$ (modal coordinates, DECOUPLED !)

modal reduction (R << N)

solve for $X^R, \dot{X}^R, \ddot{X}^R$ → $U, \dot{U}, \ddot{U}$

Compute F

Spectral Lanczos Decomposition

$$E'(t) = F_o \frac{1}{K'}(1 - e^{\frac{-\alpha t}{2}} Cos(\sqrt{K' - (\alpha^2/4)I}\ t) + \frac{\alpha}{2}\frac{1}{\sqrt{K' - (\alpha^2/4)I}} e^{\frac{-\alpha t}{2}} Sin(\sqrt{K' - (\alpha^2/4)I}\ t)$$

K' **(not diagonal !)**
NxN

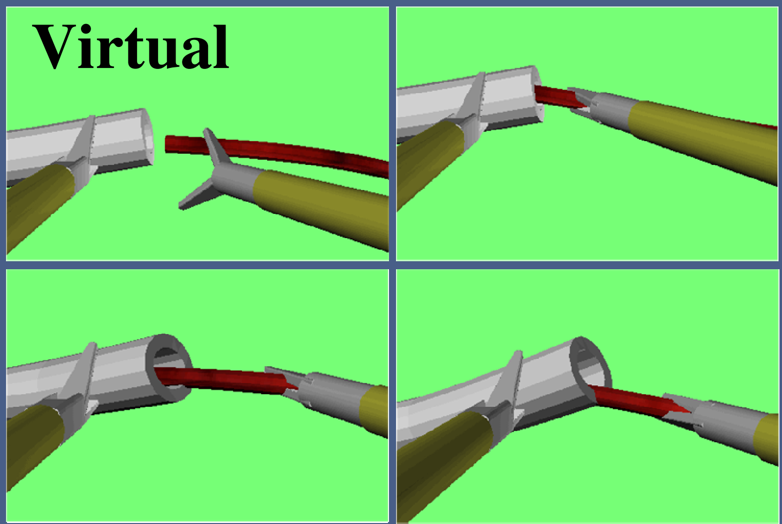
Lanczos Decomposition

$$E'(t) = F_o QV \left[\frac{1}{\Lambda}(1 - e^{\frac{-\alpha t}{2}} Cos(\sqrt{\Lambda - (\alpha^2/4)}\ t) + \frac{\alpha}{2}\frac{1}{\sqrt{\Lambda - (\alpha^2/4)}} e^{\frac{-\alpha t}{2}} Sin(\sqrt{\Lambda - (\alpha^2/4)}\ t) \right] V^T e_1$$

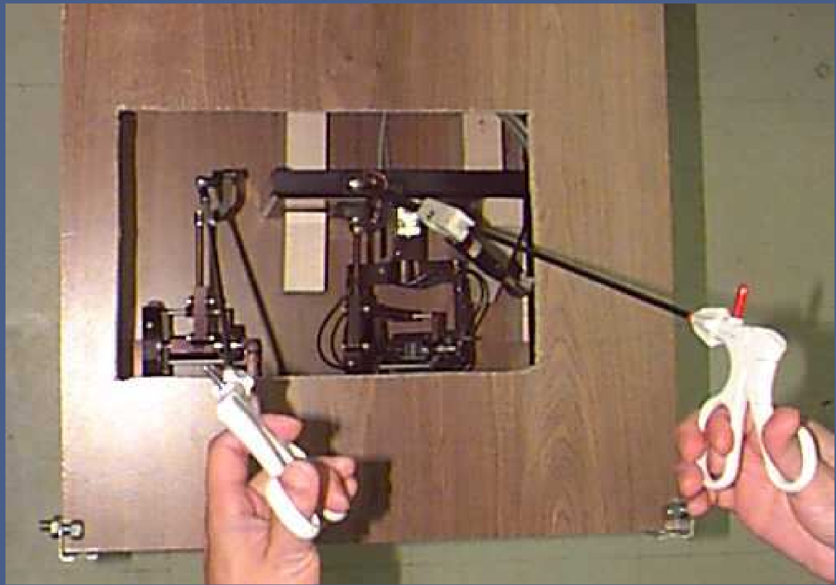
Λ **(diagonal !)**
MxM

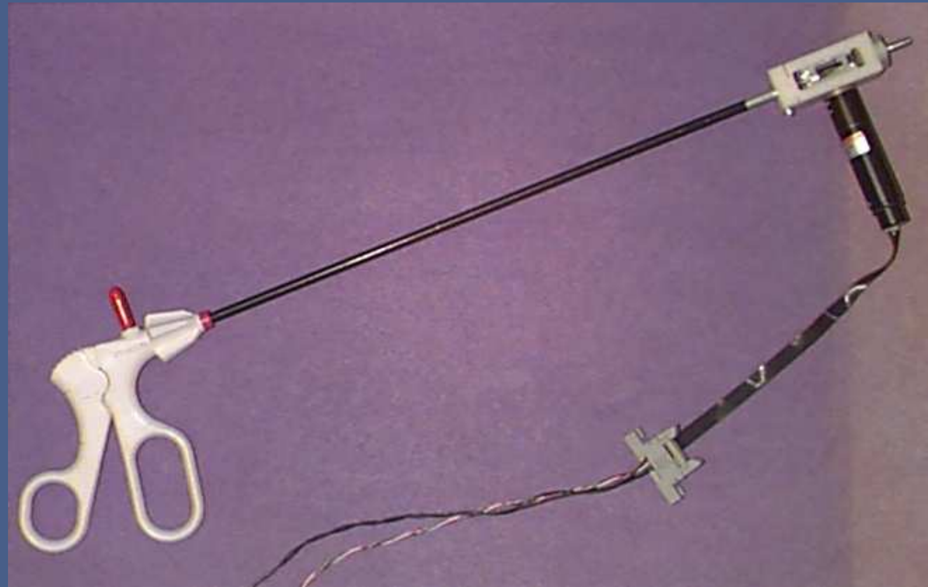
(M << N)

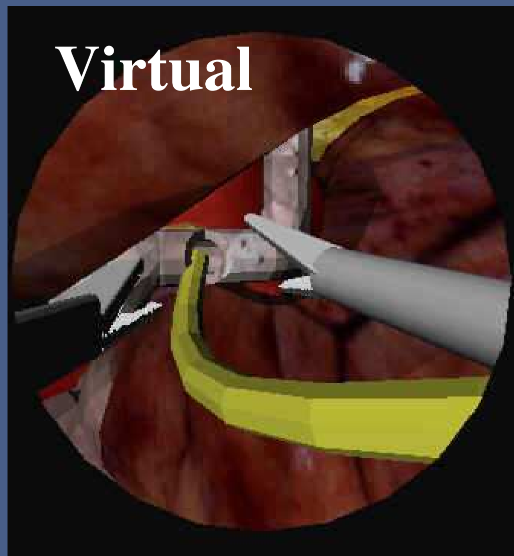
Simulation of Catheter Insertion

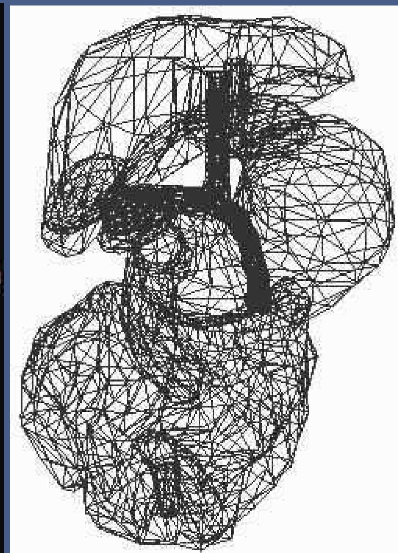


Simulation Set-Up









Geometric Model

tutorial notes are available online:
Siggraph'99, MMVR'00, MMVR'01

http://network.ku.edu.tr/~cbasdogan

Acknowledgement:

- haptic rendering
  Chih-Hao Ho, Mandayam Srinivasan, MIT
- design of force reflecting grippers
  Ela Ben-Ur, Mark Ottensmeyer, Ken Salisbury, MIT
- discussions on finite element modeling
  Suvranu De, MIT
- discussions on medical procedures, videos, and several OR visits
  Steve Small, Steve Dawson, David Rattner, MGH-Harvard Medical School
- laparoscopic trainer, several visits to animal facility
  Cynthia Barlow (Harvard Medical School)
- set-up
  Chih-Hao Ho, MIT