

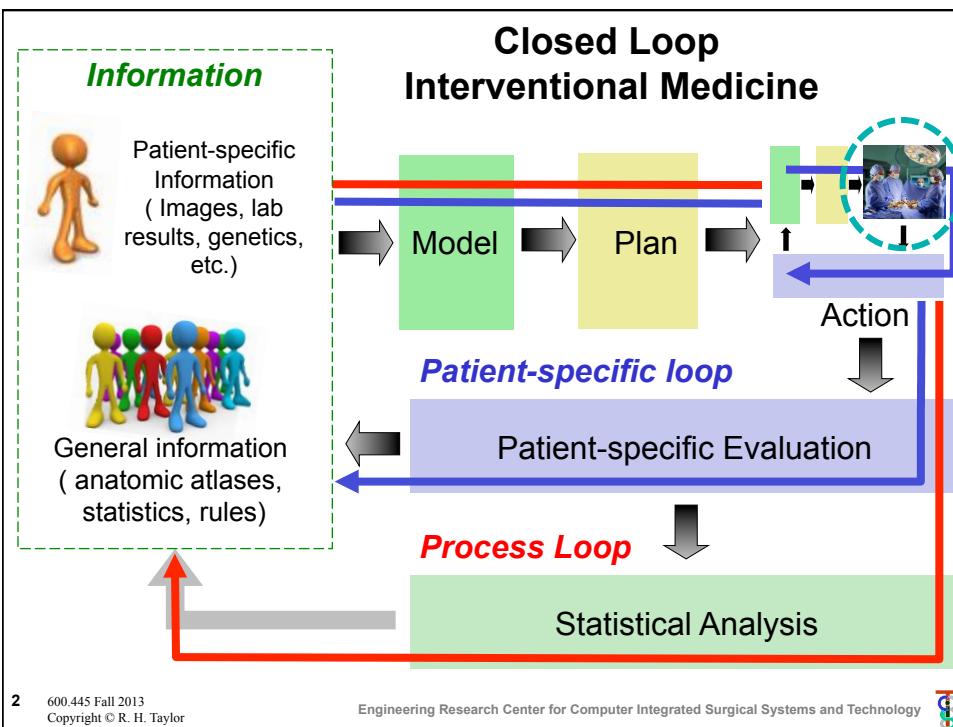
# Medical Robots, Constrained Robot Motion Control, and “Virtual Fixtures”

Russell H. Taylor

600.445

1 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Common classes of medical robots

- **Surgical “CAD/CAM” systems**
  - Goal is accurate execution of surgical plans
  - Typically based on medical images
  - Planning may be “online” or “offline”
  - Execution is often at least semi-autonomous but may still involve interaction with humans
  - Examples: Orthopaedic robots, needle placement robots, radiation therapy robots
- **Surgical “assistant” systems**
  - Emphasis is on interactive control by human
  - Human input may be through hand controllers (e.g., da Vinci), hand-over-hand (e.g., Mako, JHU “steady hand” robots)
  - Typically augmenting or supplementing human ability
  - Common applications include MIS, microsurgery
- **Note that the distinction is really somewhat arbitrary**
  - Most real systems have aspects of both.

3 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Surgical CAD/CAM: Orthopaedic Robots



Robodoc



PFS™ device “hangs on” to existing drill platforms. The PFS™ instrument has been designed for use in neurosurgical and orthopaedic applications.

Blue Belt Technologies



D. Glozman  
& M. Shoham



MAKO RIO

Mako Robotics Rio  
<http://www.makosurgical.com/>



ACROBOT surgical robot

4 600.445 Fall 2013  
Copyright © R. H. Taylor

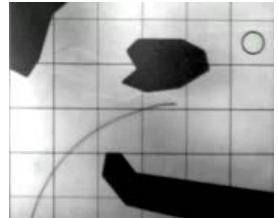
Engineering Research Center for Computer Integrated Surgical Systems and Technology



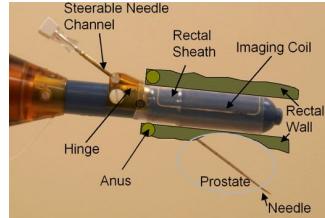
## Image-guided needle placement



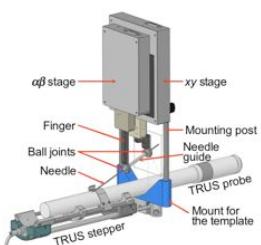
Masamune, Fichtinger, Iordachita, ...



Okamura, Webster, ...



Krieger, Fichtinger, Whitcomb, ...



Fichtinger, Kazanzides, Burdette, Song ...

5 600.445 Fall 2013  
Copyright © R. H. Taylor



Iordachita, Fischer, Hata...

Taylor, Masamune, Susil, Patriciu, Stoianovici,...  
Engineering Research Center for Computer Integrated Surgical Systems and Technology

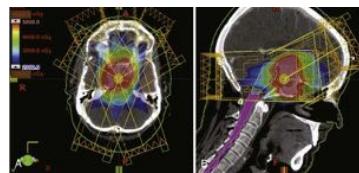


## Image Guided Radiotherapy

- Radiation source mounted on robotic arm
- Automatic segmentation of targets
- Automated planning radiation beam path
- Image guide patient motion compensation for more accurate radiation targeting



Cyberknife



Slide credit: Howie Choset + RHT



Varian Trilogy System

[http://www.varian.com/us/oncology/radiation\\_oncology/trilogy/](http://www.varian.com/us/oncology/radiation_oncology/trilogy/)

6 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Common classes of medical robots

- Surgical “CAD/CAM” systems
  - Goal is accurate execution of surgical plans
  - Typically based on medical images
  - Planning may be “online” or “offline”
  - Execution is often at least semi-autonomous but may still involve interaction with humans
  - Examples: Orthopaedic robots, needle placement robots, radiation therapy robots
- Surgical “assistant” systems
  - Emphasis is on interactive control by human
  - Human input may be through hand controllers (e.g., da Vinci), hand-over-hand (e.g., Mako, JHU “steady hand” robots)
  - Typically augmenting or supplementing human ability
  - Common applications include MIS, microsurgery
- Note that the distinction is really somewhat arbitrary
  - Most real systems have aspects of both

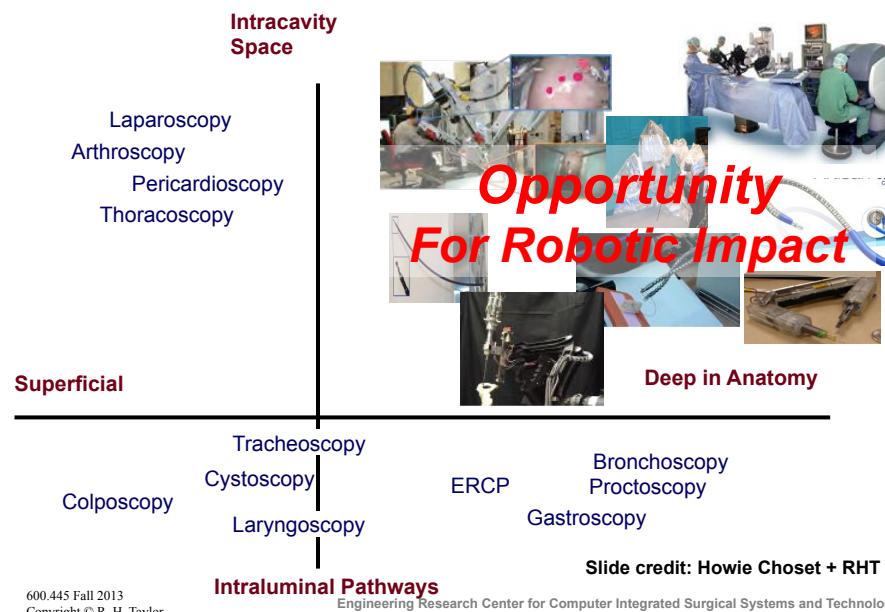
7

600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## MIS Landscape



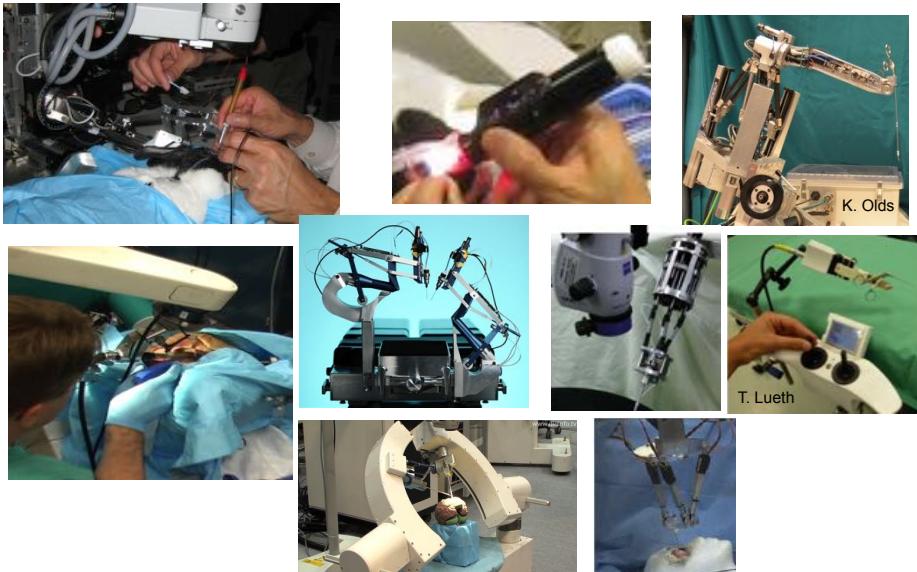
8

600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Precision Augmentation



9 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Common classes of medical robots

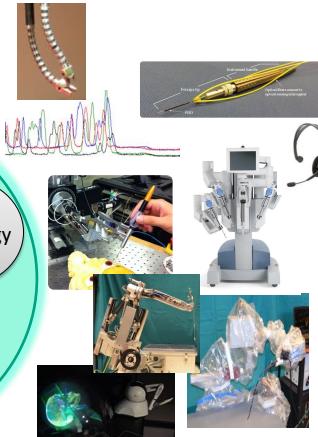
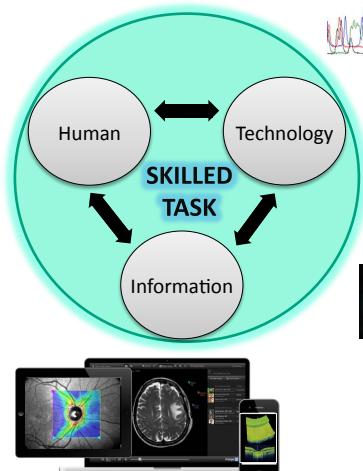
- **Surgical “CAD/CAM” systems**
  - Goal is accurate execution of surgical plans
  - Typically based on medical images
  - Planning may be “online” or “offline”
  - Execution is often at least semi-autonomous but may still involve interaction with humans
  - Examples: Orthopaedic robots, needle placement robots, radiation therapy robots
- **Surgical “assistant” systems**
  - Emphasis is on interactive control by human
  - Human input may be through hand controllers (e.g., da Vinci), hand-over-hand (e.g., Mako, JHU “steady hand” robots)
  - Typically augmenting or supplementing human ability
  - Common applications include MIS, microsurgery
- **Note that the distinction is really somewhat arbitrary**
  - Most real systems have aspects of both.

10 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Goal: “Partnership to transcend human limitations”



Slide credit: Marcin Balicki  
11 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Complementary Capabilities

### Humans

- Excellent judgment & reasoning
- Excellent optical vision
- Cannot see through tissue
- Do not tolerate ionizing radiation
- Limited precision, hand tremor
- No stereotactic accuracy
- Moderately strong
- High dexterity (“human” scale)
- Big hands and bodies
- Reasonable force sensitivity
- Must rely on memory of preoperative plans and data

### Robots

- No judgment
- Limited vision processing
- Can use x-rays, other sensors
- Do not mind radiation
- High precision
- High stereotactic accuracy
- Variable strength
- Dexterity at different scales
- Variable sizes
- Can sense very small forces
- Can be programmed to use preoperative plans and data

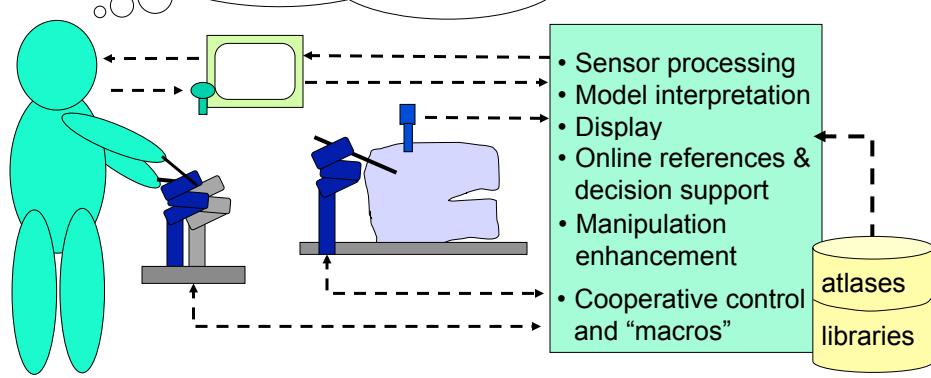
12 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Surgical Assistant Systems

- Situation assessment
- Task strategy & decisions
- Sensory-motor coordination



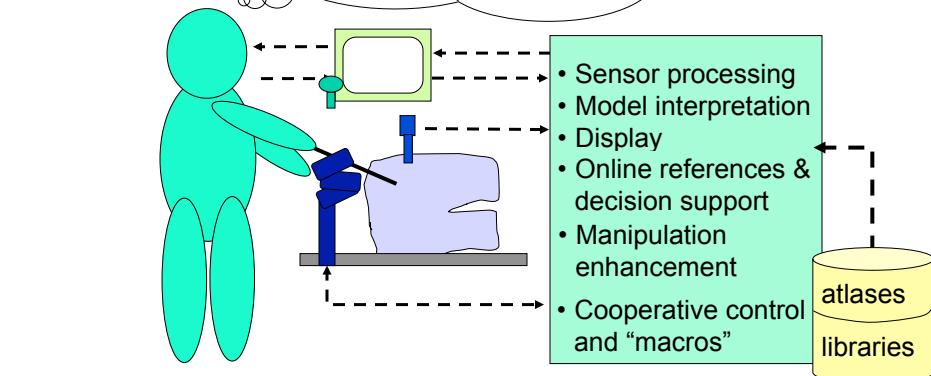
13 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Surgical Assistant Systems

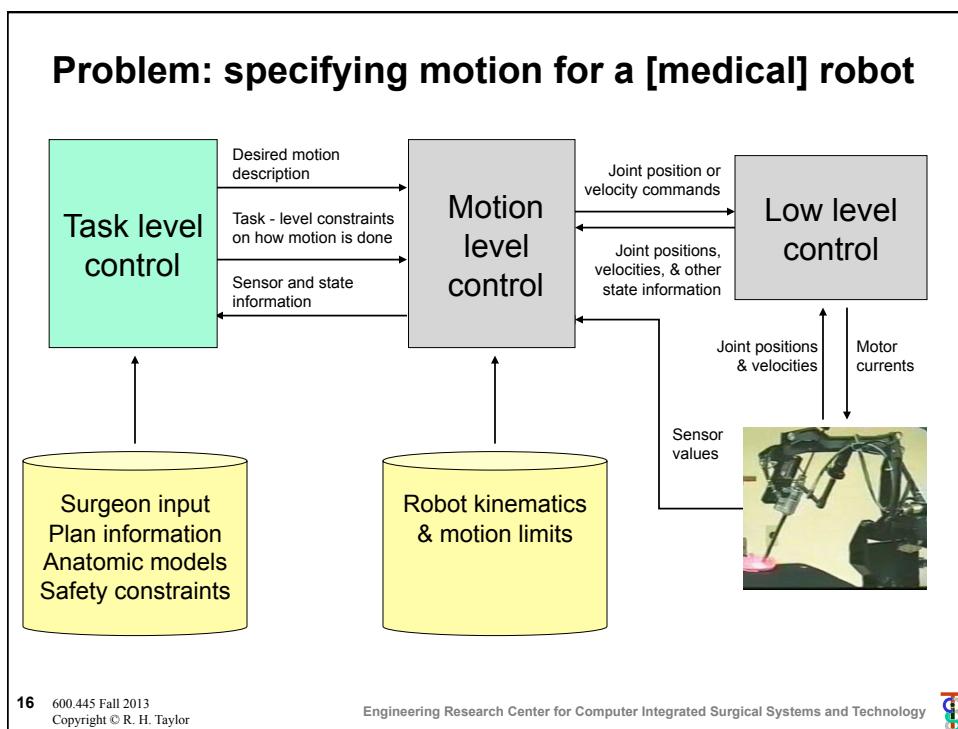
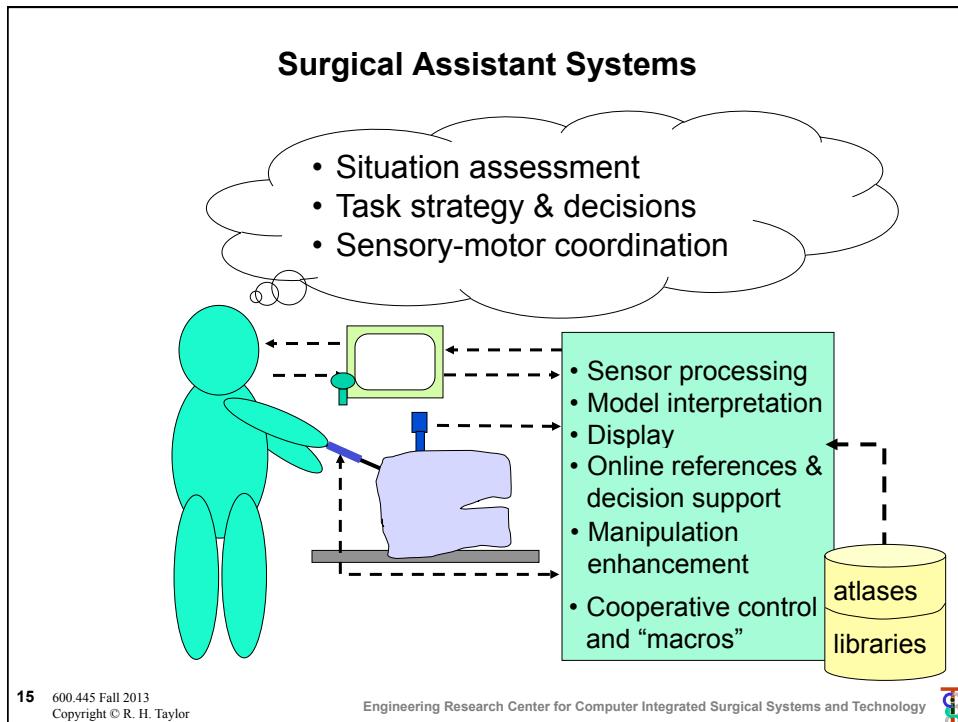
- Situation assessment
- Task strategy & decisions
- Sensory-motor coordination



14 R. Taylor  
600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology





## Background: Robot Kinematics



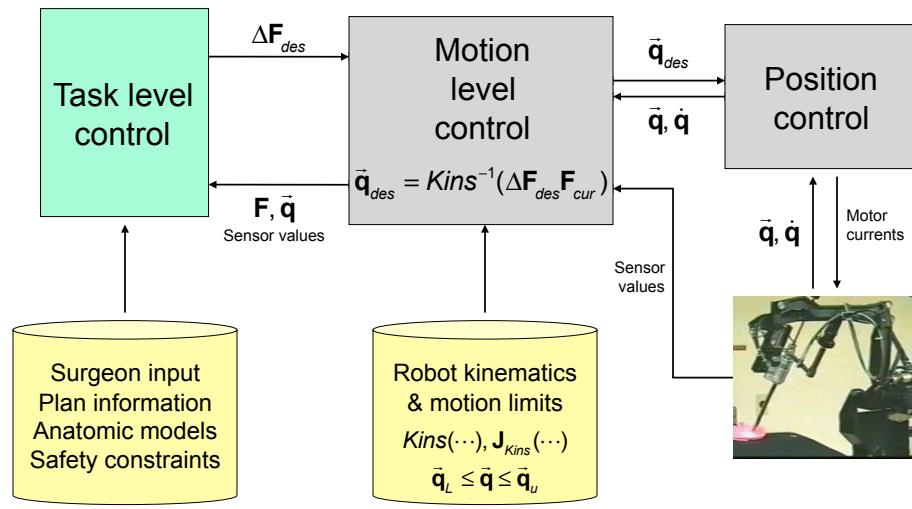
$$\begin{aligned}\text{Pose } \mathbf{F}(\vec{q} + \Delta\vec{q}) &= \text{kins}(\vec{q} + \Delta\vec{q}) \\ \Delta\mathbf{F} \cdot \mathbf{F} &= \text{kins}(\vec{q} + \Delta\vec{q}) \\ \Delta\mathbf{F} &= \text{kins}(\vec{q} + \Delta\vec{q}) \text{kins}(\vec{q})^{-1}\end{aligned}$$

17 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## One implementation



18 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Background: Jacobean Robot Motion Control

Let  $\mathbf{F} = [\mathbf{R}, \mathbf{p}]$  be the current pose of a robot end effector and  $\vec{\mathbf{q}} = [q_1, \dots, q_N]$  be the current joint position values corresponding to  $\mathbf{F}$ . I.e.,  $\mathbf{F} = Kins(\vec{\mathbf{q}})$ , where  $Kins(\dots)$  is a function computing the "forward kinematics" of the robot.



$$\begin{aligned} \text{Pose } \mathbf{F}(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) &= kins(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) \\ \Delta\mathbf{F} \bullet \mathbf{F} &= kins(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) \\ \Delta\mathbf{F} &= kins(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) kins(\vec{\mathbf{q}})^{-1} \end{aligned}$$



## Background: Jacobean Robot Motion Control

Let  $\mathbf{F} = [\mathbf{R}, \mathbf{p}]$  be the current pose of a robot end effector and  $\vec{\mathbf{q}} = [q_1, \dots, q_N]$  be the current joint position values corresponding to  $\mathbf{F}$ . I.e.,  $\mathbf{F} = Kins(\vec{\mathbf{q}})$ , where  $Kins(\dots)$  is a function computing the "forward kinematics" of the robot. Let  $\Delta\mathbf{F} \bullet \mathbf{F} = Kins(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}})$

For small  $\Delta\vec{\mathbf{q}}$ , we can write the following expression for  $\Delta\mathbf{F} = [Rot(\vec{\alpha}), \vec{\varepsilon}]$

$$\begin{aligned} \Delta\mathbf{F} &= Kins(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) Kins(\vec{\mathbf{q}})^{-1} \\ \text{which we typically linearize as} & \quad \Delta\vec{\mathbf{q}} \rightarrow \Delta\vec{\mathbf{x}} \\ \Delta\vec{\mathbf{x}} &= \begin{bmatrix} \vec{\alpha} \\ \vec{\varepsilon} \end{bmatrix} \approx \mathbf{J}_{Kins}(\vec{\mathbf{q}}) \Delta\vec{\mathbf{q}} \end{aligned}$$



Note that here we are computing  $\Delta\mathbf{F}$  in the base frame of the robot. If we want to compute  $\Delta\mathbf{F}$  in the end effector frame, so that  $\mathbf{F} \bullet \Delta\mathbf{F} = Kins(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}})$ , then we will get a slightly different expression for  $\mathbf{J}_{Kins}(\vec{\mathbf{q}})$ , though the flavor will be the same



## Background: Jacobean Robot Motion Control



$$\begin{aligned}\text{Pose } \mathbf{F}(\vec{q} + \Delta\vec{q}) &= \text{kins}(\vec{q} + \Delta\vec{q}) \\ \Delta\mathbf{F} \cdot \mathbf{F} &= \text{kins}(\vec{q} + \Delta\vec{q}) \\ \Delta\mathbf{F} &= \text{kins}(\vec{q} + \Delta\vec{q}) \text{kins}(\vec{q})^{-1}\end{aligned}$$

$$\begin{bmatrix} \vec{\alpha} \\ \varepsilon \end{bmatrix} \approx \mathbf{J}(\vec{q}) \Delta\vec{q}$$

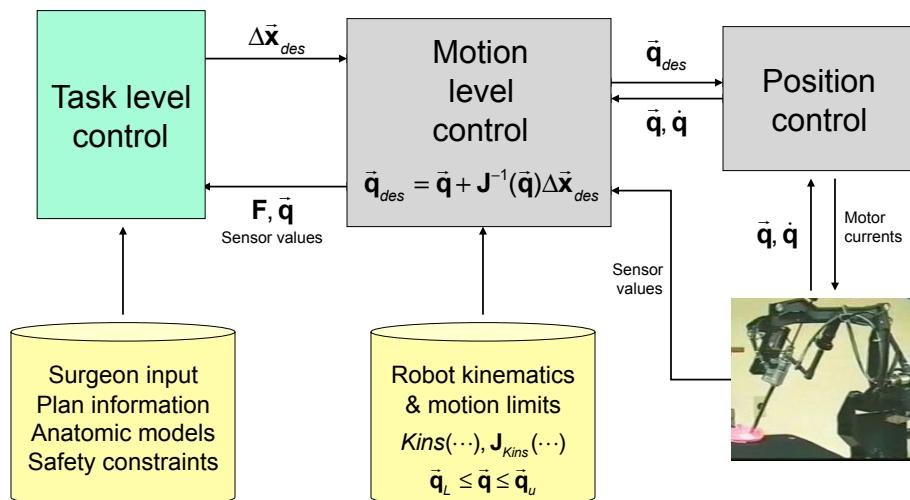
$$\Delta\vec{q} \approx \mathbf{J}(\vec{q})^{-1} \begin{bmatrix} \vec{\alpha} \\ \varepsilon \end{bmatrix}$$

21 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



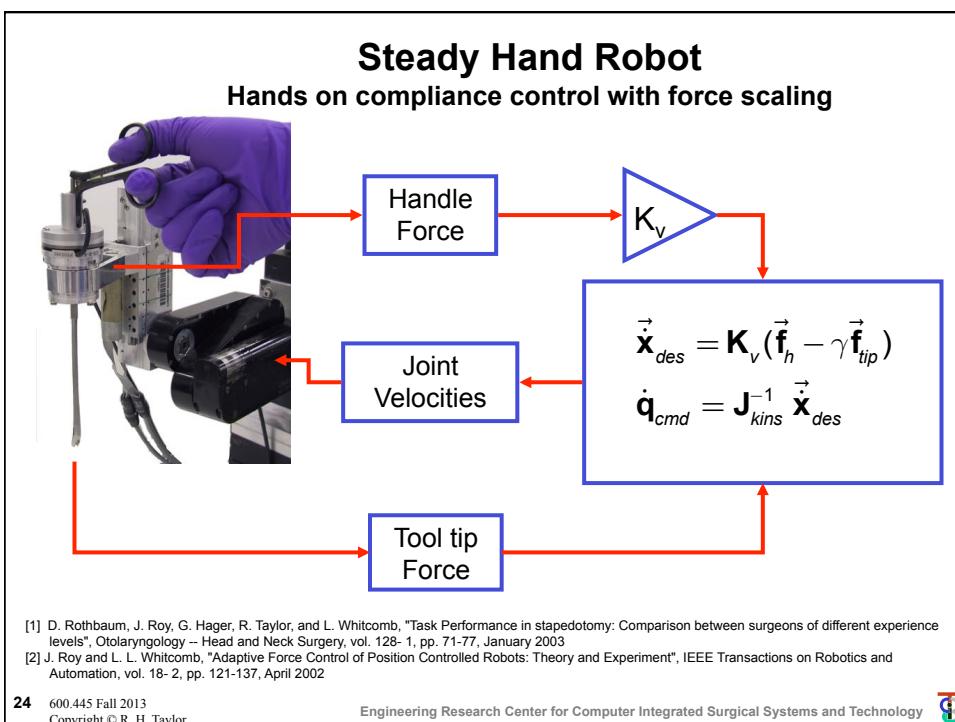
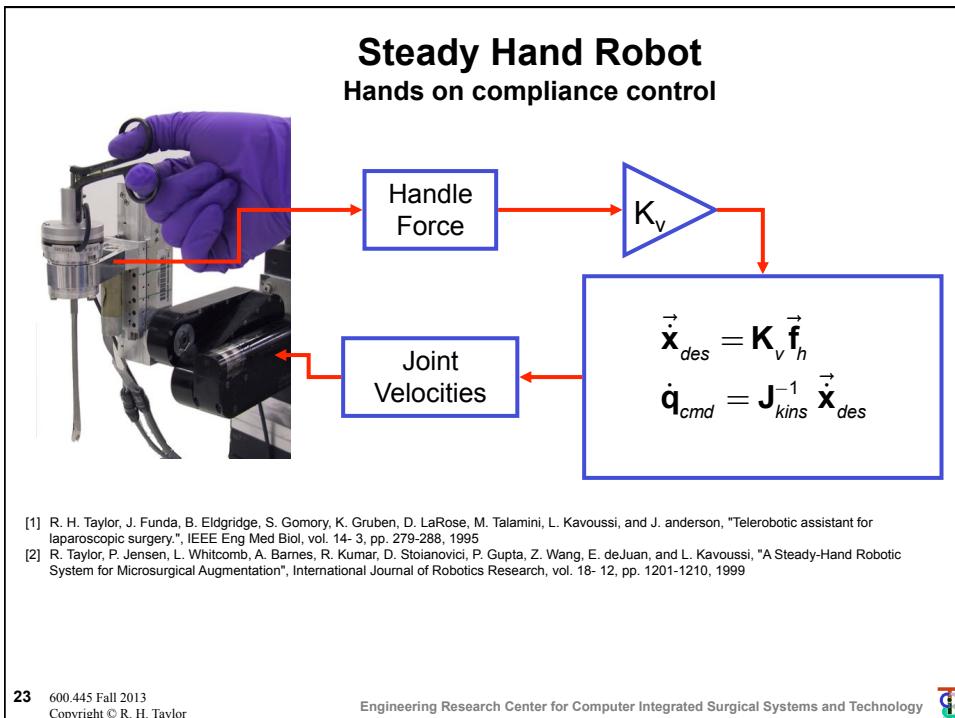
## Jacobian motion control implementation



22 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology





## Example: Fenestratration of Stapes Footplate

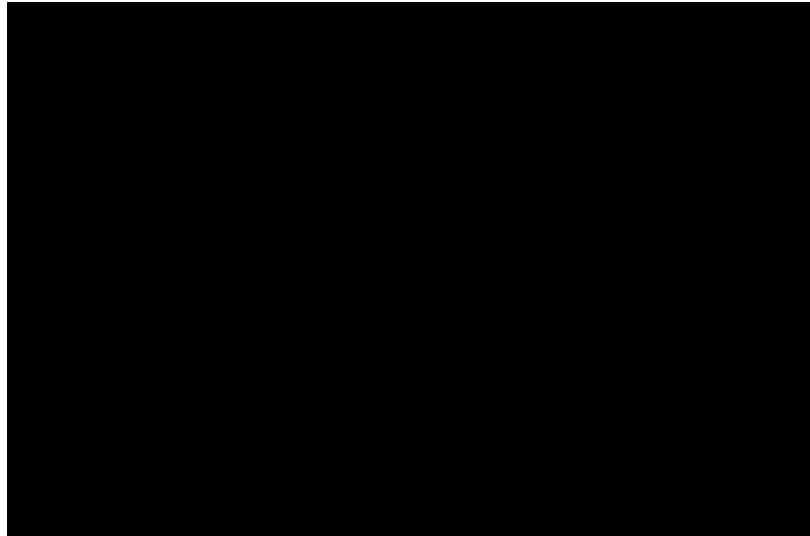


25 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Example: Fenestratration of Stapes Footplate



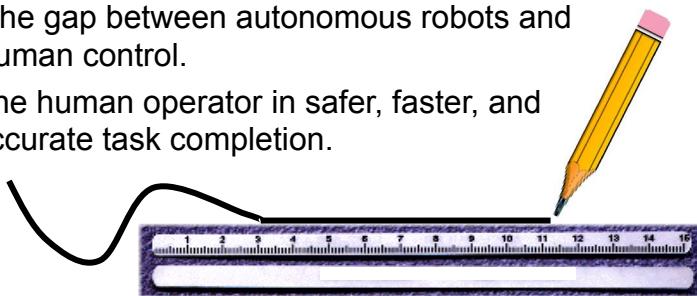
26 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Virtual Fixtures

- Bridge the gap between autonomous robots and direct human control.
- Assist the human operator in safer, faster, and more accurate task completion.



- Broadly Categorized
  - Guidance VF
  - Forbidden Region VF
- Different implementation
  - Tele-manipulation
  - Cooperative Control



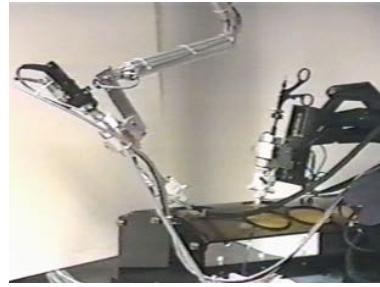
## Background: Virtual Fixtures

- First proposed for complex telerobotic tasks, but draw upon rich prior research in robot assembly and other manufacturing automation applications
- Many authors, e.g.,
  - L. B. Rosenberg, "Virtual Fixtures: Perceptual Tools for Telerobotic Manipulation," *Proc. IEEE Virtual Reality International Symposium*, 1993.
  - B. Davies, S. Harris, M. Jakopac, K. Fan, and J. Cobb, "Intraoperative application of a robotic knee surgery system," *MICCAI* 1999.
  - S. Park, R. D. Howe, and D. F. Torchiana, "Virtual Fixtures for Robotic Cardiac Surgery", *MICCAI* 2001.
  - S. Payandeh and Z. Stanisic, "On Application of Virtual Fixtures as an Aid for Telemanipulation and Training," *Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2002.
- Discussion that follows draws upon work at IBM Research and within the CISST ERC at JHU. E.g.,
  - Funda, R. Taylor, B. Eldridge, S. Gomory, and K. Gruben, "Constrained Cartesian motion control for teleoperated surgical robots," *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 453-466, 1996.
  - R. Kumar, An Augmented Steady Hand System for Precise Micromanipulation, Ph.D thesis in Computer Science, The Johns Hopkins University, Baltimore, 2001.
  - M. Li, M. Ishii, and R. H. Taylor, "Spatial Motion Constraints in Medical Robot Using Virtual Fixtures Generated by Anatomy," *IEEE Transactions on Robotics*, vol. 2, pp. 1270-1275, 2006.
  - A. Kapoor, M. Li, and R. H. Taylor "Constrained Control for Surgical Assistant Robots," in *IEEE Int. Conference on Robotics and Automation*, Orlando, 2006, pp. 231-236.
  - A. Kapoor and R. Taylor, "A Constrained Optimization Approach to Virtual Fixtures for Multi-Handed Tasks," in *IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, 2008, pp. 3401-3406.
  - M. Li, *Intelligent Robotic Surgical Assistance for Sinus Surgery*, PhD Thesis in Computer Science Baltimore, Maryland: The Johns Hopkins University, 2005.
  - Ankur Kapoor, *Motion Constrained Control of Robots for Dexterous Surgical Tasks*, Ph.D. Thesis in Computer Science, The Johns Hopkins University, Baltimore, September 2007

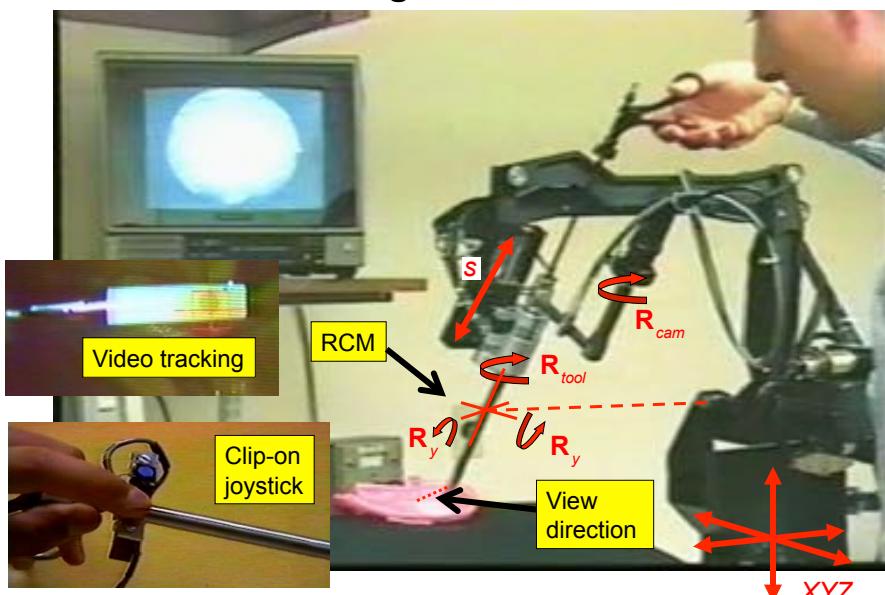


## Original Motivation for IBM Work

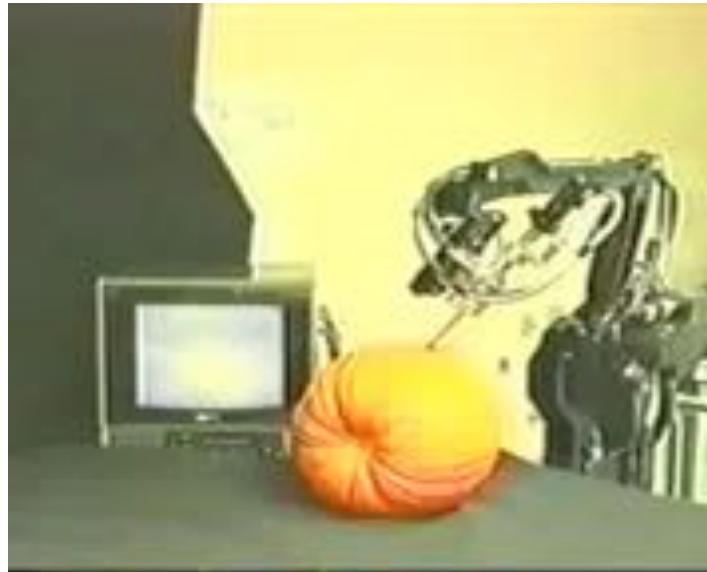
- Kinematic control of robots for MIS
- E.g., LARS and HISAR robots
- LARS and other IBM robots were kinematically redundant
  - Typically 7-9 actuated joints
- But tasks often imposed kinematic constraints
  - E.g., no lateral motion at trocar
- Some robots (e.g., IBM/JHU HISAR and CMI's AESOP) had passive joints
- General goals
  - Exploit redundancy in best way possible
  - Come as close as possible to providing desired motion subject to robot and task limits
- **Our approach:** view this as a constrained optimization problem



## LARS degrees of freedom



## LARS Video

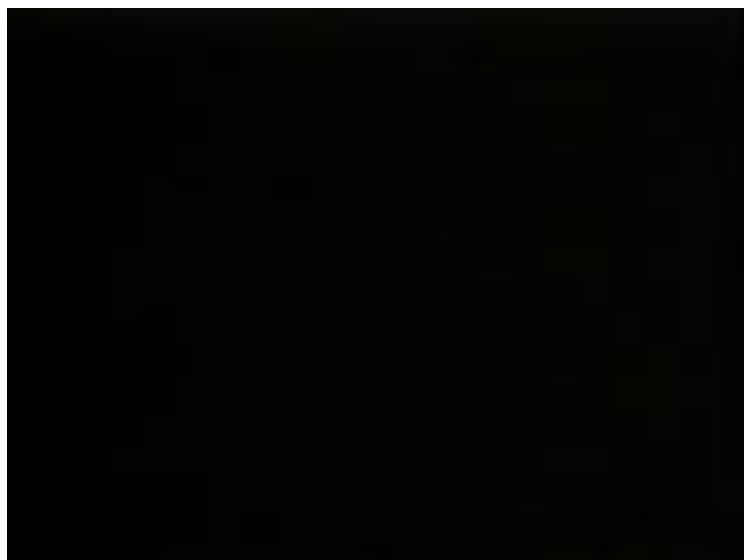


**32** 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## LARS Video



**33** 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Motion Specification Problem

- Requirements

- The tool shaft must pass within a specified distance of the entry port into the patient's body
- The individual joint limits may not be exceeded

- Goals

- Aim the camera as close as possible at a target
  - or move view in direction indicated by clip-on pointing device
  - or move to track a video target on an instrument
  - or aim the working channel of the endoscope at a target
  - or something else (maybe a combination of goals)
- Keep the view as "upright" as possible
- Tool should pass as close as possible to entry port center
- Keep joints far away from their limits, to preserve options for future motion
- Minimize motion of XYZ joints
- *Etc.*



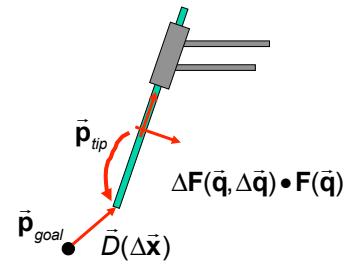
## Our approach: view as an optimization problem

- Currently formulate problem as constrained least squares problem
- Express goals in the objective function
- If multiple goals, objective function is a weighted sum of individual elements
- Add constraints for requirements
- Express constraints and objective function terms in whatever coordinate system is convenient
- Use Jacobean formulation to transform to joint space
- Solve for joint motion



## Example: keep tool tip near a point

$$\begin{aligned}\vec{D}(\Delta\vec{x}) &= \Delta\mathbf{F}(\vec{q}, \Delta\vec{q}) \cdot \mathbf{F} \cdot \vec{p}_{tip} - \vec{p}_{goal} \\ &= \vec{\alpha} \times \vec{t} + \vec{\varepsilon} + \vec{t} - \vec{p}_{goal} \quad \text{where } \vec{t} = \mathbf{F} \cdot \vec{p}_{tip} \\ \vec{\alpha} &= \mathbf{J}_{\vec{\alpha}}(\vec{q}) \Delta\vec{q} \\ \vec{\varepsilon} &= \mathbf{J}_{\vec{\varepsilon}}(\vec{q}) \Delta\vec{q}\end{aligned}$$



Suppose we want to stay as close as possible while never going beyond 3mm from goal and also obeying joint limits

$$\Delta\mathbf{q}_{des} = \arg \min_{\Delta\vec{q}} \left\| \vec{D}(\Delta\vec{x}) \right\|^2 = \left\| \vec{\alpha} \times \vec{t} + \vec{\varepsilon} + \vec{t} - \vec{p}_{goal} \right\|^2$$

Subject to

$$\begin{aligned}\vec{\alpha} &= \mathbf{J}_{\vec{\alpha}}(\vec{q}) \Delta\vec{q} \\ \vec{\varepsilon} &= \mathbf{J}_{\vec{\varepsilon}}(\vec{q}) \Delta\vec{q} \\ \left\| \vec{\alpha} \times \vec{t} + \vec{\varepsilon} + \vec{t} - \vec{p}_{goal} \right\| &\leq 3 \\ \vec{q}_L - \vec{q} &\leq \Delta\vec{q} \leq \vec{q}_U - \vec{q}\end{aligned}$$



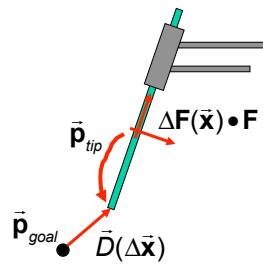
## Example: keep tool tip near a point

Suppose we want to stay as close as possible while never going beyond 3mm from goal and also obeying joint limits, but we also want to minimize the change in direction of the tool shaft

$$\Delta\mathbf{q}_{des} = \arg \min_{\Delta\vec{q}} \zeta \left\| \vec{D}(\Delta\vec{x}) \right\|^2 + \eta \left\| \vec{\alpha} \times \mathbf{R} \cdot \vec{z} \right\|^2$$

Subject to

$$\begin{aligned}\vec{x} &= \mathbf{F} \cdot \vec{p}_{tip} \\ \vec{D}(\Delta\vec{x}) &= \vec{\alpha} \times \vec{t} + \vec{\varepsilon} + \vec{x} - \vec{p}_{goal} \\ \vec{\alpha} &= \mathbf{J}_{\vec{\alpha}}(\vec{q}) \Delta\vec{q}; \quad \vec{\varepsilon} = \mathbf{J}_{\vec{\varepsilon}}(\vec{q}) \Delta\vec{q} \\ \left\| \vec{D}(\Delta\vec{x}) \right\| &\leq 3 \\ \vec{q}_L - \vec{q} &\leq \Delta\vec{q} \leq \vec{q}_U - \vec{q}\end{aligned}$$



## Solving the optimization problem

- **Constrained linear least squares**
  - Combine constraints and goals from task and robot control
  - Linearize and constrained least squares problem

$$\Delta \vec{q}_{des} = \underset{\Delta q}{\operatorname{argmin}} \left\| \mathbf{E}_{task} \Delta \vec{x} - \vec{f}_{task} \right\|^2 + \left\| \mathbf{E}_q \Delta \vec{x} - \vec{f}_q \right\|^2$$

subject to

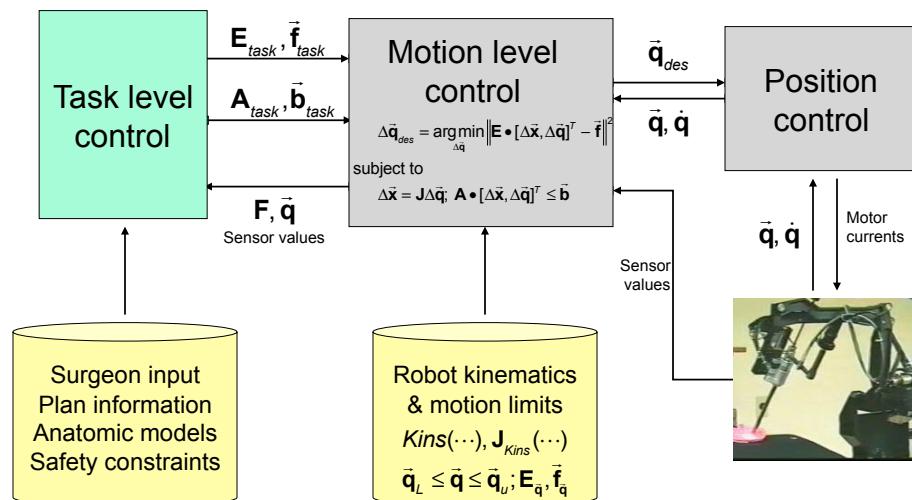
$$\Delta \vec{x} = \mathbf{J} \Delta \vec{q}; \quad \mathbf{A}_{task} \Delta \vec{x} \leq \vec{b}_{task}; \quad \mathbf{A}_q \Delta \vec{q} \leq \vec{b}_q$$

- E.g., using “non-negative least squares” methods developed by Lawson and Hanson
- Approach used in our IBM work and in Kumar, Li, Kapoor theses

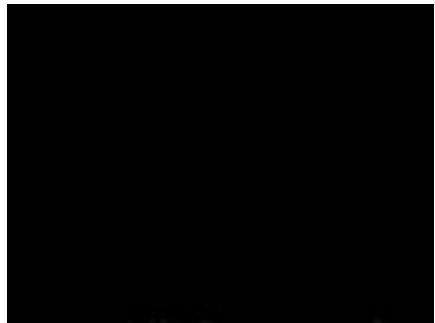
- **Constrained nonlinear least squares**
  - Approach explored by Kapoor (discuss later)



## Linear least squares implementation



## Some IBM Movies



Early Constrained Motion System (LapSYS)



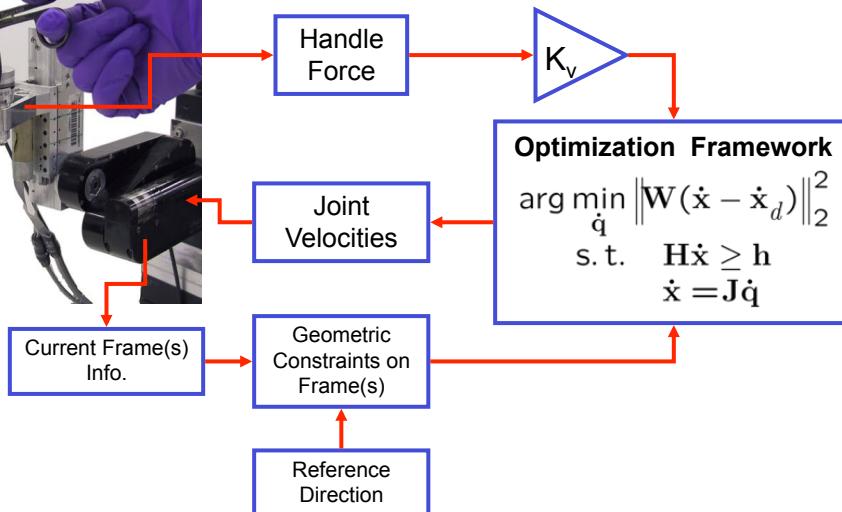
Vision-guided targeting

40 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Steady Hand Robot High Level Constrained Control

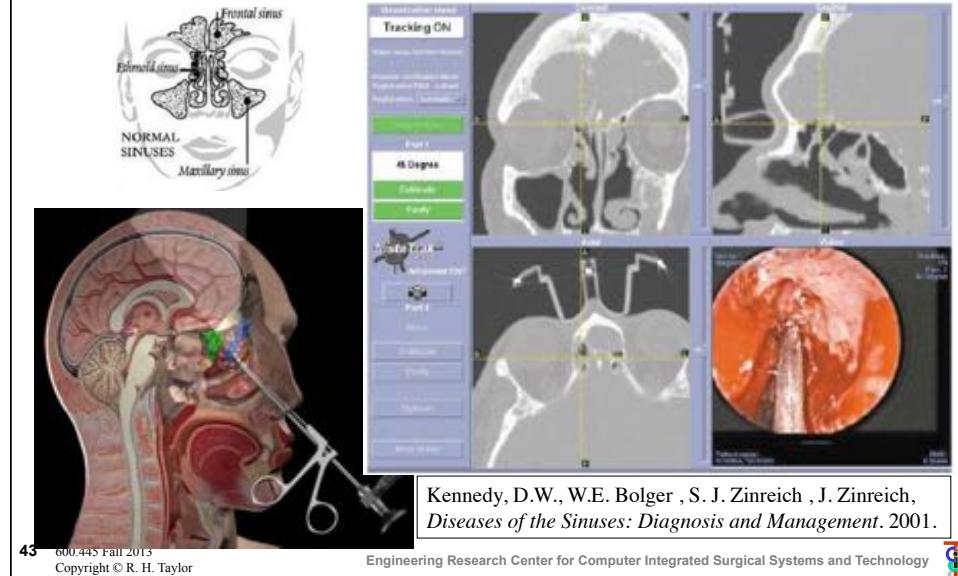


42 600.445 Fall 2013  
Copyright © R. H. Taylor

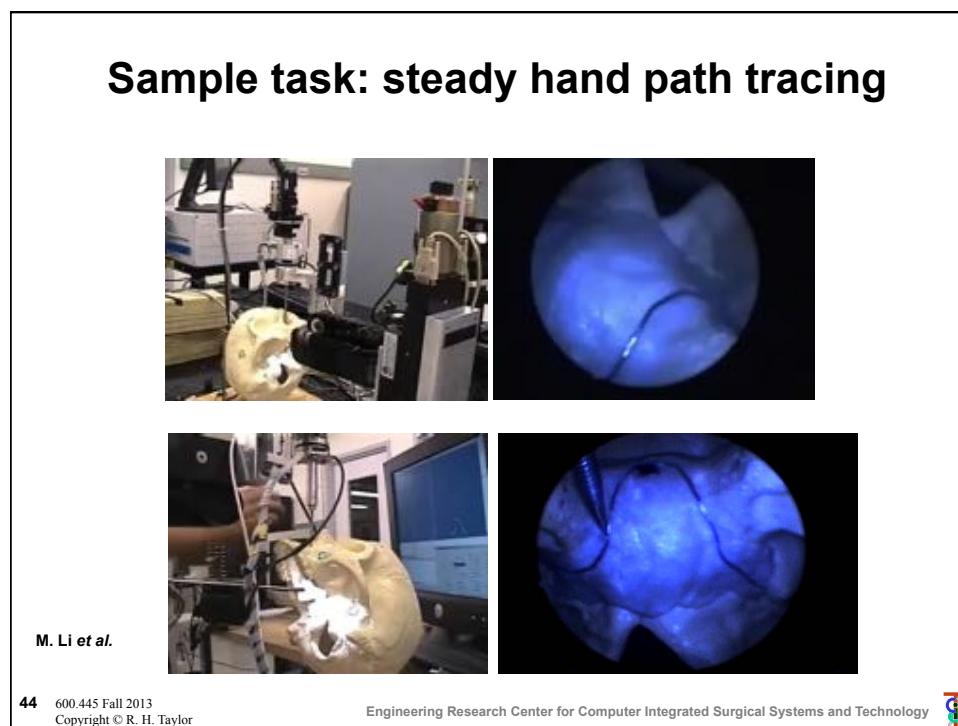
Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Typical application domain: endoscopic sinus surgery

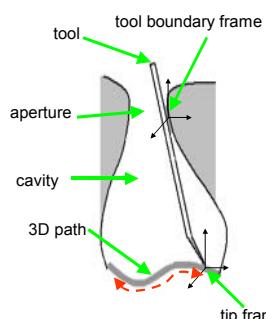


## Sample task: steady hand path tracing



## Goal: robotically-assisted sinus surgery

- **Difficulties with conventional approach**
  - Complicated geometry
  - Safety-critical structures
  - Limited work space
  - Awkward tools
- **Our approach**
  - Cooperatively controlled “Steady hand” robot
  - Registered to CT models
  - “Virtual fixtures” automatically derived from models



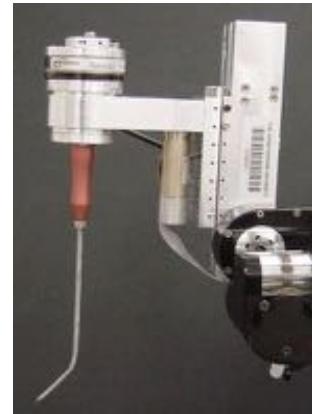
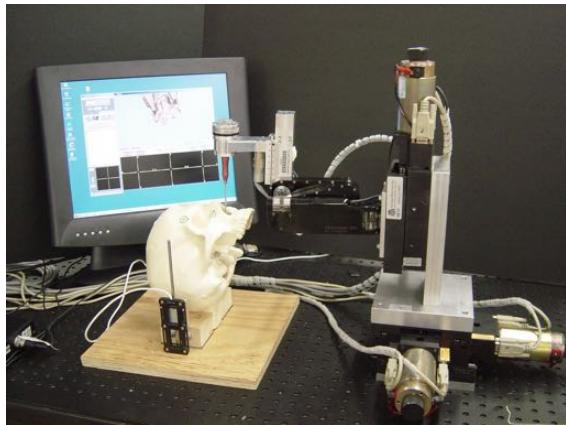
M. Li et al.

45 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Experiment Setup



M. Li et al.

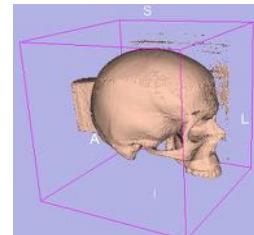
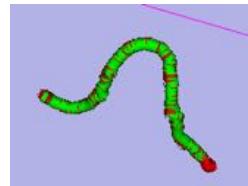
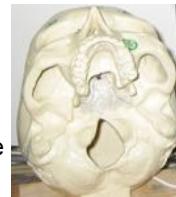
46 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Experimental setup

- Plastic Skull Phantom
  - Target path defined by embedded wire
  - Radioopaque fiducials implanted on skull for registration
- Computer model
  - Extracted from CT scan using standard software (Slicer)
- 3D tracking of tools, etc. using Northern Digital Optotrak®
- Co-register model, robot, and optical tracker using standard techniques



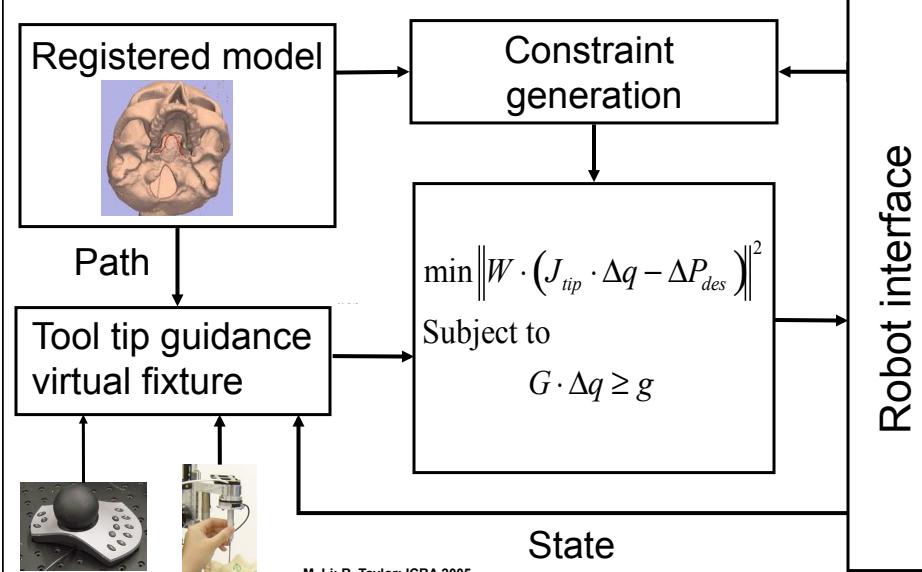
M. Li et al.

47 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Virtual Fixture Online Implementation



48 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Boundary Constraints Generation

- Anatomy – triangulated surface models
  - Patient-specific model of nose & sinus derived from CT
  - High complexity: 182,000 triangles & 99,000 vertices
- Tool shaft -- cylinder
- The boundary constraint generation requires us to find close-point pairs between boundary surface model & tool shaft



M. Li et al.

49 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Boundary Constraints Generation

- Anatomy – triangulated surface models
  - Patient-specific model of nose & sinus derived from CT
  - High complexity: 182,000 triangles & 99,000 vertices
- Tool shaft -- cylinder
- The boundary constraint generation requires us to find close-point pairs between boundary surface model & tool shaft
- **Problem: How can we generate the right constraints in real time???**



M. Li et al.

50 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Our solution: efficient search method using covariance tree representation of model

### Covariance trees:

- Williams & Taylor, 1998; other authors
- Variation of k-d trees
- Basic idea:
  - Hierarchically split 3D model into sub-volumes
  - Realign coordinate system of each sub-volume to align with moments of inertia
- Produces bounding boxes that closely approximate boundaries & fast searches

M. Li et al.

51 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Our solution: efficient search method using covariance tree representation of model

### Covariance trees:

- Williams & Taylor, 1998; other authors
- Variation of k-d trees
- Basic idea:
  - Hierarchically split 3D model into sub-volumes
  - Realign coordinate system of each sub-volume to align with moments of inertia
- Produces bounding boxes that closely approximate boundaries & fast searches

M. Li et al.

52 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Our solution: efficient search method using covariance tree representation of model

### Covariance trees:

- Williams & Taylor, 1998; other authors
- Variation of k-d trees
- Basic idea:
  - Hierarchically split 3D model into sub-volumes
  - Realign coordinate system of each sub-volume to align with moments of inertia
- Produces bounding boxes that closely approximate boundaries & fast searches

M. Li et al.

53 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## One difference from ICP problem

### One difference from ICP problem:

- Here we in principle need to identify all anatomy that can interfere with tool shaft
- Consequently modify search to find all triangle edges that are closer than some threshold to tool shaft
- Further modify to prune search to eliminate redundant constraints

54 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Control Implementation

- Formulate constrained least squares problem
- Constraints & objective function include terms for desired tip motion, joint limits, boundary constraints

$$\zeta = \min_{\Delta q} \left\| \begin{bmatrix} W_{tip} & & \\ & W_k & \\ & & W_{joints} \end{bmatrix} \cdot \begin{bmatrix} J_{tip}(q) \\ J_k(q) \\ I \end{bmatrix} \Delta q - \begin{bmatrix} \Delta P_{tip-des} \\ 0 \\ 0 \end{bmatrix} \right\|$$

subject to

$$\begin{bmatrix} H_{tip} & & \\ & H_k & \\ & & H_{joints} \end{bmatrix} \cdot \begin{bmatrix} J_{tip}(q) \\ J_k(q) \\ I \end{bmatrix} (\Delta q) \geq \begin{bmatrix} h_{tip} \\ h_k \\ h_{joints} \end{bmatrix}$$

M. Li et al.



## Control Implementation

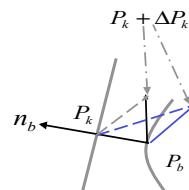
- Tip frame  $\Delta P_{tip} = J_{tip}(q) \cdot \Delta q$

$$\begin{array}{ll} \|\Delta P_{tip} - \Delta P_{tip-des}\| \\ \Delta P_{tip-d}^T \cdot \Delta P_{tip} \geq THD \end{array} \quad \begin{array}{ll} \min \quad \zeta_{tip} = \|W_{tip} \cdot (J_{tip}(q) \Delta q - \Delta P_{tip-des})\| \\ \text{subject to } H_{tip-des} J_{tip}(q) \Delta q \geq h_{tip} \end{array}$$

- Boundary constraint  $\Delta P_k = J_k(q) \cdot \Delta q$

$$\begin{array}{l} \|W_k \cdot \Delta P_k\| \\ n_b^T \cdot (P_k + \Delta P_k - P_b) \geq d \end{array}$$

$$\begin{array}{ll} \min \quad \zeta_k = \|W_k J_k(q) \Delta q\| \\ \text{subject to } H_k J_k(q) \Delta q \geq h_k \end{array}$$



- Joints limitation

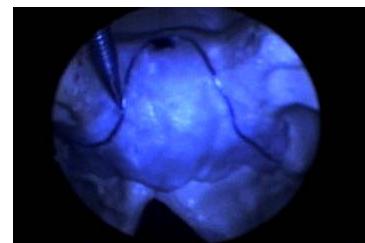
$$\begin{array}{l} \|W_{joints} \cdot \Delta q\| \\ q_{\min} - q \leq \Delta q \leq q_{\max} - q \end{array}$$

$$\begin{array}{ll} \min \quad \zeta_{joints} = \|W_{joints} \Delta q\| \\ \text{subject to } H_{joints} \Delta q \geq h_{joints} \end{array}$$



## Control implementation

- Solve problem numerically with standard methods (Lawson & Hanson, 1974)
- Performance:
  - 6 ms/iteration on 2GHz Pentium 4 PC
  - Typically 20 to 39 constraints



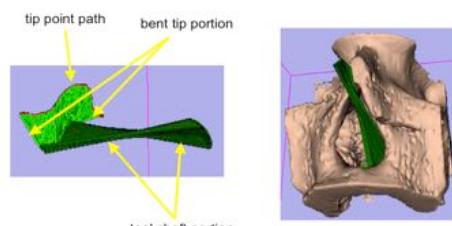
M. Li et al.

57 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Results



The average time in each control loop for the boundary searching is ~6ms



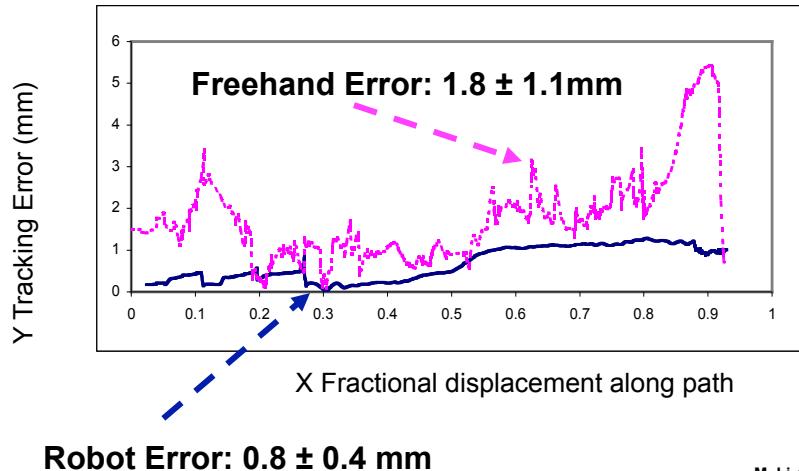
M. Li et al.

58 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Results: Robot vs Freehand



M. Li et al.

59 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Results: Robot vs Freehand

Trial#	Free hand		Robot Guidance	
	Average	Average	Average	Average
	Error	Time	Error	Time
	(mm)	(s)	(mm)	(s)
1	1.785	26.354	0.736	18.972
2	1.632	29.358	0.757	15.275
3	1.796	27.372	0.765	16.29
4	2.061	25.436	0.779	19.439
5	2.119	24.533	0.777	16.209
avg	1.819	26.611	0.763	17.237
std	1.126	1.863	0.395	1.848

Dashed red arrows point from the circled 'avg' values in the table to the corresponding lines in the graph above.

**Approx 1.5:1 improvement in time!**

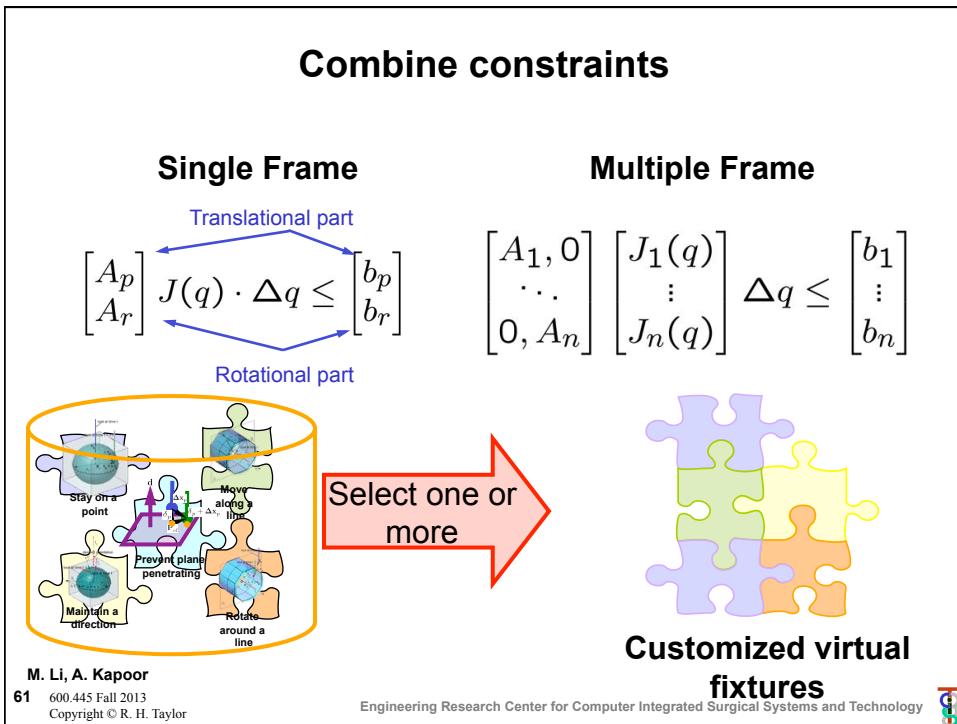
M. Li et al.

60 600.445 Fall 2013  
Copyright © R. H. Taylor

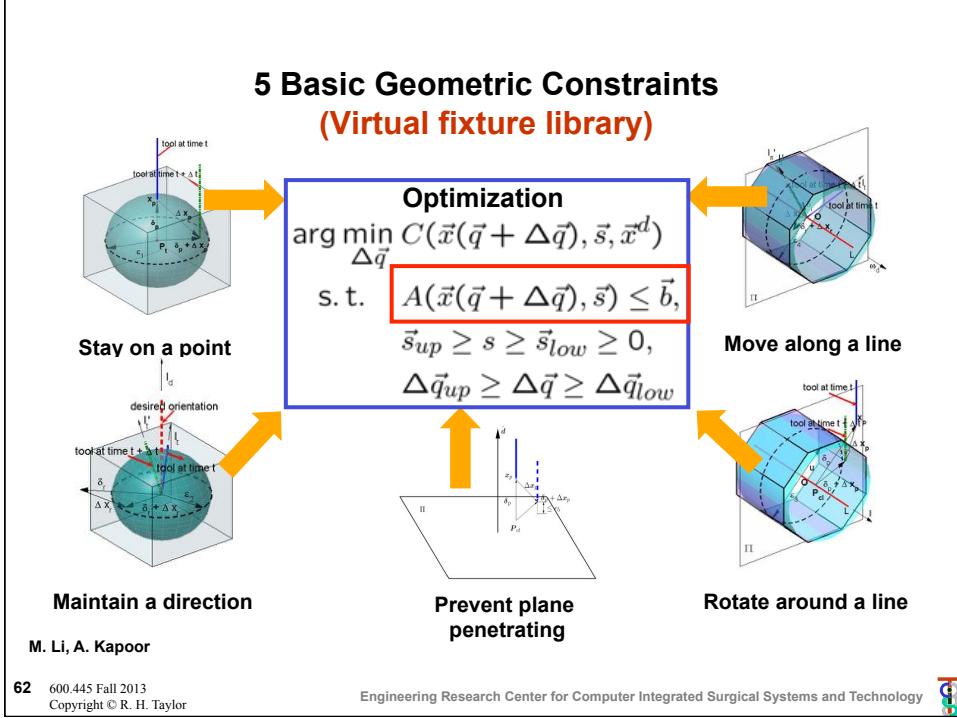
Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Combine constraints



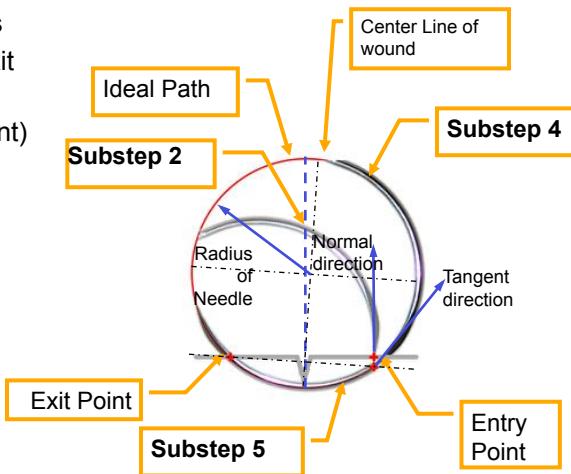
## 5 Basic Geometric Constraints (Virtual fixture library)



## Example: Suturing

The suturing task involves

- Select entry and exit points
- Align (Move & Orient) Needle
- Bite: Pass Needle
- Loop
- Knot



M. Li, A. Kapoor

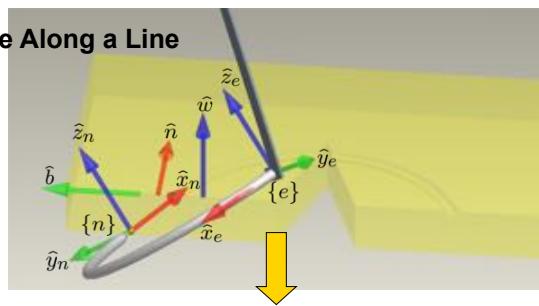
63 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology

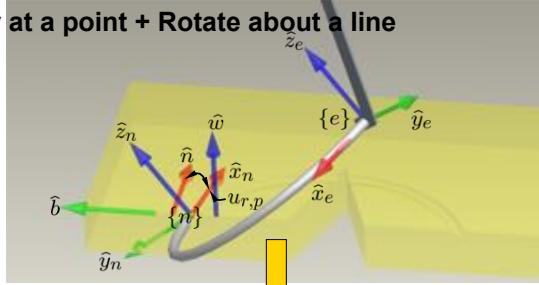


## Suturing: Align Step

### 0. Move Along a Line



### 1. Stay at a point + Rotate about a line



M. Li, A. Kapoor

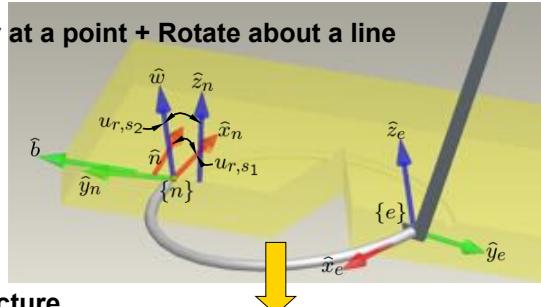
64 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



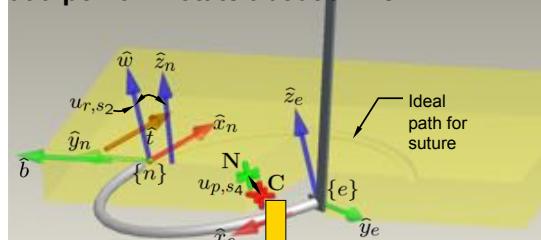
## Suturing: Align Step

### 2. Stay at a point + Rotate about a line



### 3. Puncture

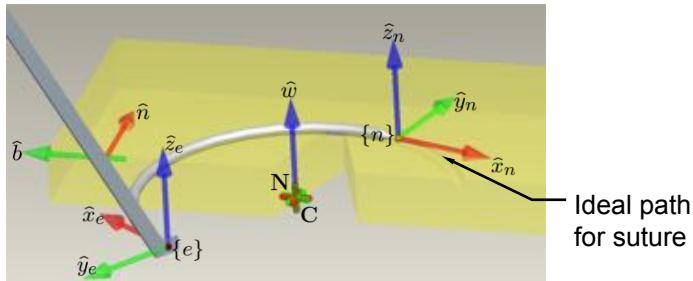
### 4. Stay at a point + Rotate about a line



65 600.445 Fall 2013  
Copyright © R. H. Taylor



## Suturing: Bite Step



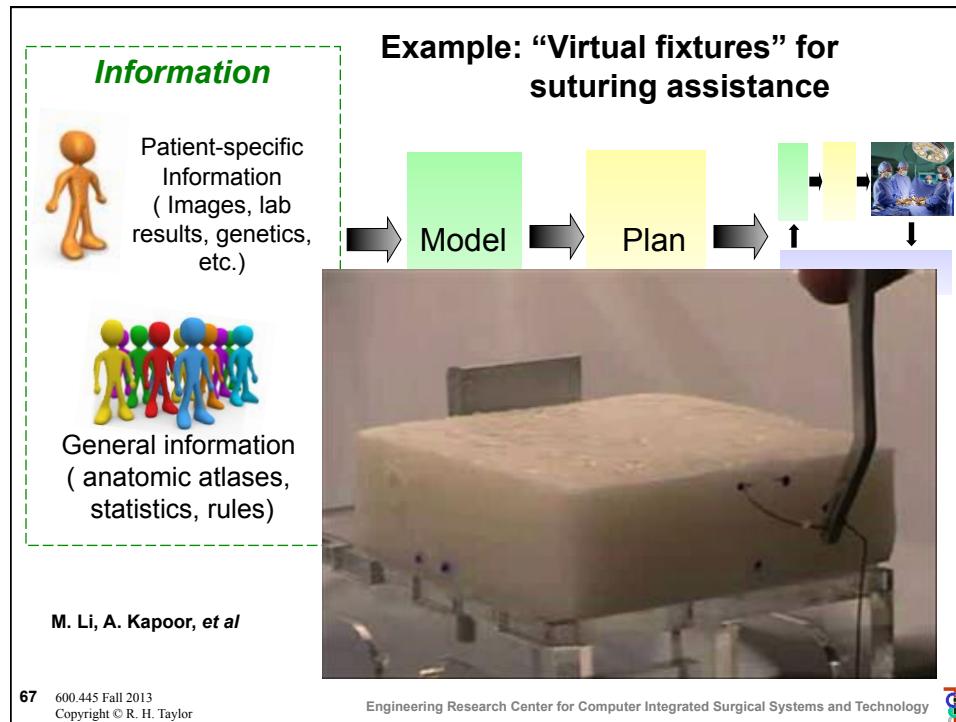
- Ideal trajectory is a circle with radius equal to needle radius.
- Needle plane is parallel to entry and exit points and surface normal.

M. Li, A. Kapoor

66 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology





## Suturing: Results

The average error (mm) in ideal and actual points as measured by OptoTrak®  
Preliminary data collected from 4 users 5 trials each.

Error	Entry (mm)	Exit (mm)
Robot	0.6375; $\sigma = 0.12$	0.7742; $\sigma = 0.37$
Manual	--	2.1; $\sigma = 1.2$

- Suturing task using VF showed significant improvement in performance over freehand.
  - Can be performed at awkward angles
  - Avoids multiple trials and large undesirable movements inside tissue.

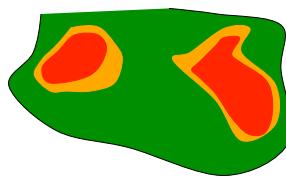
M. Li, A. Kapoor

68 600.445 Fall 2013  
Copyright © R. H. Taylor

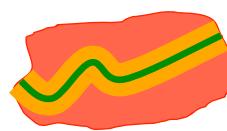
Engineering Research Center for Computer Integrated Surgical Systems and Technology

## Hard and soft constraints

■ Preferred region  
■ Safety region  
■ Forbidden region



Avoidance



Line following

- Constraints on the task can be “hard” or “soft”
- The relative sizes depend on the procedure, ranging from microns to tenths of millimeter.
- Soft constraints allow the controller to accommodate uncertainties inherent in surgical procedures.

Thanks: A. Kapoor

69 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



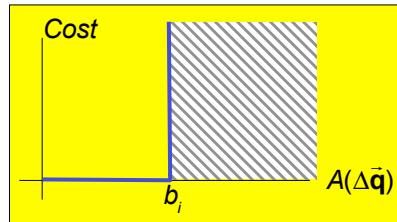
## “Soft” constraint implementation

Suppose that we have a problem of the form

$$\Delta \vec{q}_{\text{des}} = \arg \min \|E(\Delta \vec{q})\|^2$$

subject to a constraint of the form

$$A_i(\Delta \vec{q}) \leq b_i$$



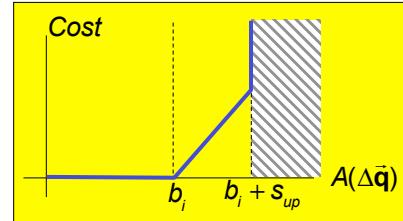
70 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## “Soft” constraint implementation

But suppose we want to make the barrier “soft”. I.e., allow the robot to go beyond the barrier at increasing cost until it hits a harder barrier later



Add an explicit slack  $s_i$  and add a penalty term to the objective function

$$\Delta\vec{q}_{\text{des}} = \arg \min \|\mathbf{E}(\Delta\vec{q})\|^2 + \eta_i s_i^2$$

subject to a constraint of the form

$$A_i(\Delta\vec{q}) - s_i \leq b_i$$

$$0 \leq s_i \leq s_{up,i}$$

This process can be repeated several times to produce progressively steeper costs



## Example: Stay near a point

Target Position:  $\vec{x}_0$

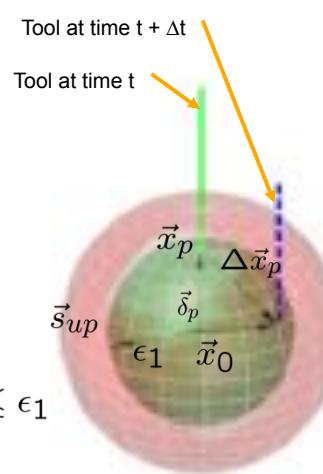
After incremental motion

$\vec{x}_p + \Delta\vec{x}_p$  close to  $\vec{x}_0$

We want...

$$A(\vec{x}, s) = \|\vec{\delta}_p + \Delta\vec{x}_p\|^2 - s \leq \epsilon_1$$

$$\text{where } \vec{\delta}_p = \vec{x}_p - \vec{x}_0$$



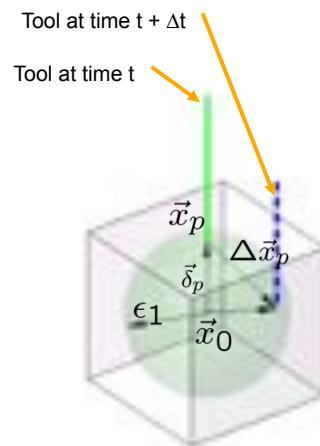
## Using Linear Constrained Quadratic Optimization

### Matrix representation

$$A \cdot \Delta \vec{x} - s \leq b$$

Use Constrained Least Squares to solve

$$\begin{aligned} \arg \min_{\Delta \vec{q}} \quad & \| \Delta \vec{x} - \Delta \vec{x}^d \|^2 \\ \text{s.t.} \quad & A \cdot \Delta \vec{x} - s \leq b \end{aligned}$$



A. Kapoor, et al.

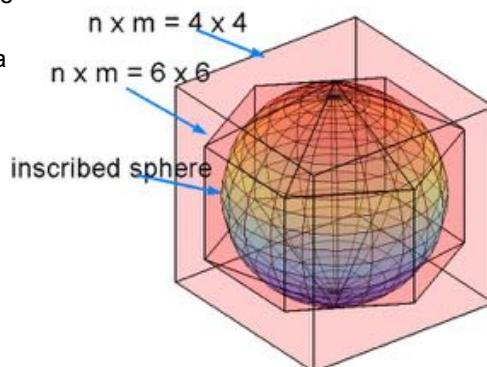
73 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Linear approximation for constraints

- $n \times m$  increase
  - Polyhedron approaches the inscribed sphere
  - Linearized conditions are a better approximation
  - More constraints require more time to solve the optimization problem
- Symmetrical polyhedron
  - $n \times m = 4 \times 4$
- Bounded polyhedron
  - $n \times m = 3 \times 3$



A. Kapoor, et al.

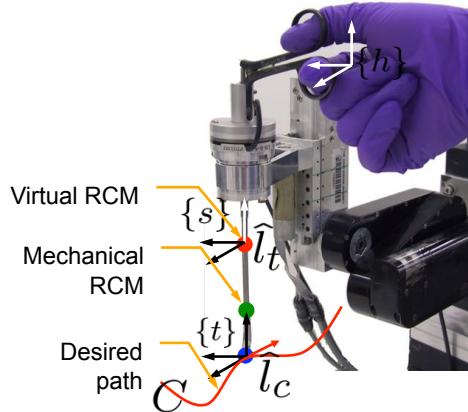
74 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Example Task

- Constraint 1: Tip to move along curve C
- Constraint 2: Origin of  $\{s\}$  to move along
- Objective: Handle to follow user input



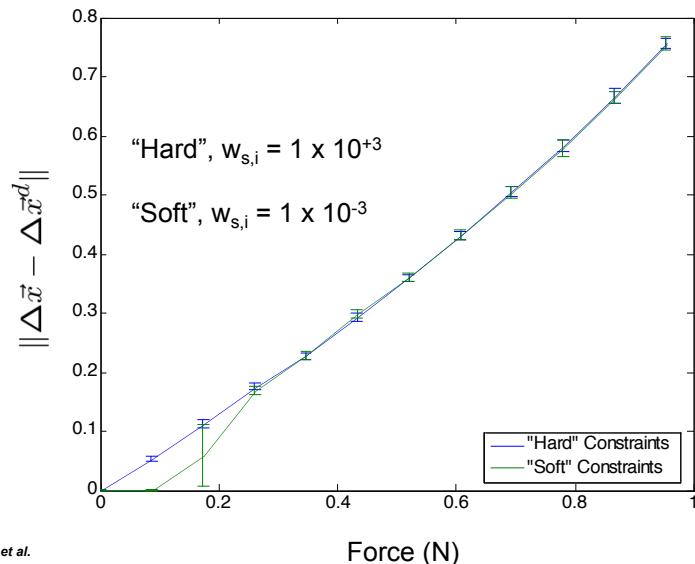
A. Kapoor, et al.

75 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Results for Example Task



A. Kapoor, et al.

76 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Nonlinear Optimization

- One problem with linearized least squares is the proliferation of constraints to approximate the real constraints
- Consequently, it is worth considering alternatives that can handle more general formulas “directly”

$$\Delta \vec{q}_{des} = \arg \min_{\Delta \vec{q}} C(\Delta \vec{x}, \Delta \vec{q}, \vec{s})$$

subject to

$$\Delta \vec{x} = \mathbf{J} \Delta \vec{q}$$

$$\mathbf{A}(\Delta \vec{x}, \Delta \vec{q}, \vec{s}) \leq \vec{b}$$



## Using Non-Linear Constrained Optimization

- Use Sequential Quadratic Program\* method
- SQP solves the following problem iteratively

$$\begin{aligned} & \arg \min_{d^k} \nabla C(\vec{x}(\vec{q} + \Delta \vec{q}^k), \vec{s}^k, \vec{x}^d)^t \vec{d} + \frac{1}{2} \vec{d}^t B_k \vec{d} \\ & s.t. \quad \nabla A_{\mathcal{A}_k}(\vec{x}_i(\vec{q} + \Delta \vec{q}^k), \vec{s}^k)^t \vec{d} \leq \vec{b}_{\mathcal{A}_k}, \end{aligned}$$

- Start with a solution  $[\Delta q^k, s^k]^t$
- Descent direction along with step size determine next solution  $[\Delta q^{k+1}, s^{k+1}]^t$

\*P. Spellucci, *Math. Prog.*, '98



## Remarks: Non-Linear Constraints

- Current incremental motion can be used as starting guess for next motion
- Worst case number of constraints  $n$  times  $m$ ,  $n = \#$  variables,  $m = \#$  nonlinear constraints
- Analytical gradient increases speed

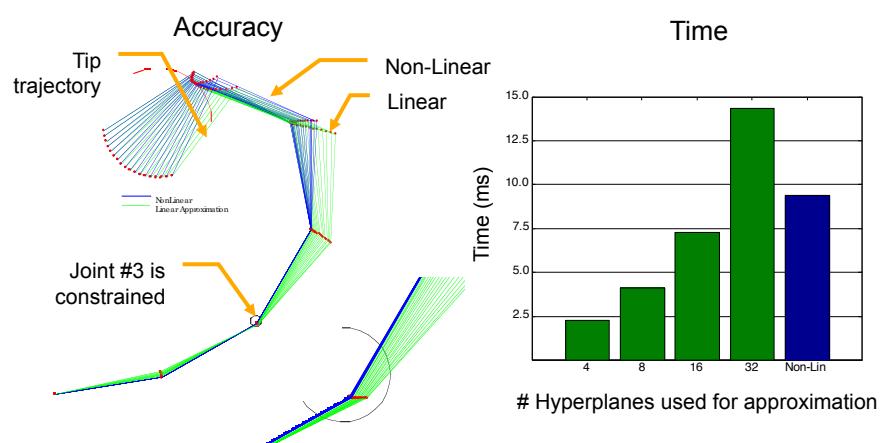
A. Kapoor, et al.

79 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Linear v. Non-Linear Constraints



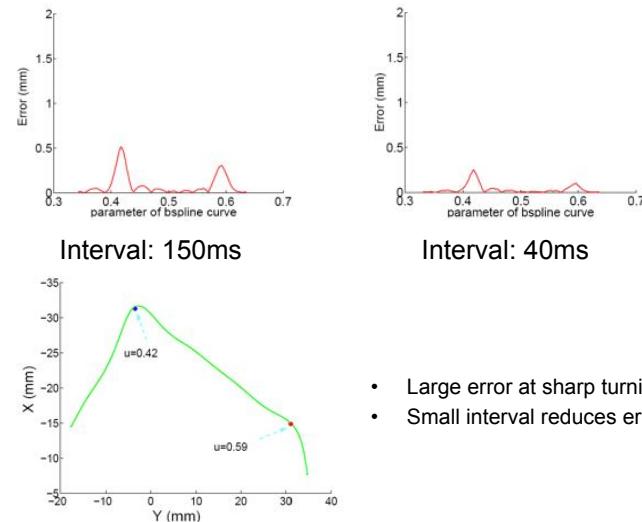
A. Kapoor, et al.

80 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Effect of increasing control-loop time



Ming Li et al., IROS '05

81 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Example: Two-handed virtual fixture for centering knot with visual feedback

### Information



Patient-specific Information  
( Images, lab results, genetics, etc.)



General information  
( anatomic atlases, statistics, rules)



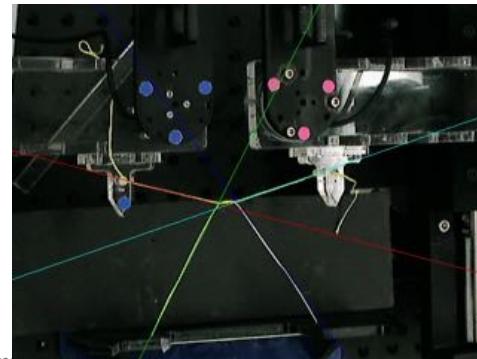
Model



Plan



Action



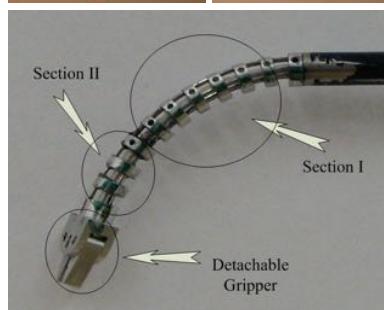
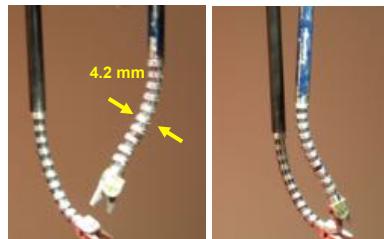
Ankur Kapoor

82 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Scalable Robot for Dexterous Surgery in Small Spaces (aka Snake Like Robot)



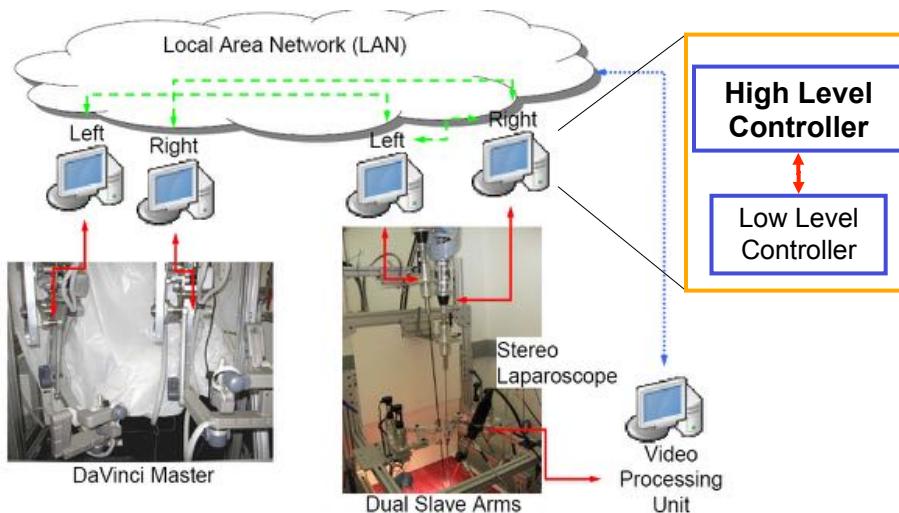
Team: A. Kapoor, Kai Xu, Wei Wei, N. Simaan,  
P. Kazanzides, R. H. Taylor  
Collaborator: P. Flint, MD

83 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



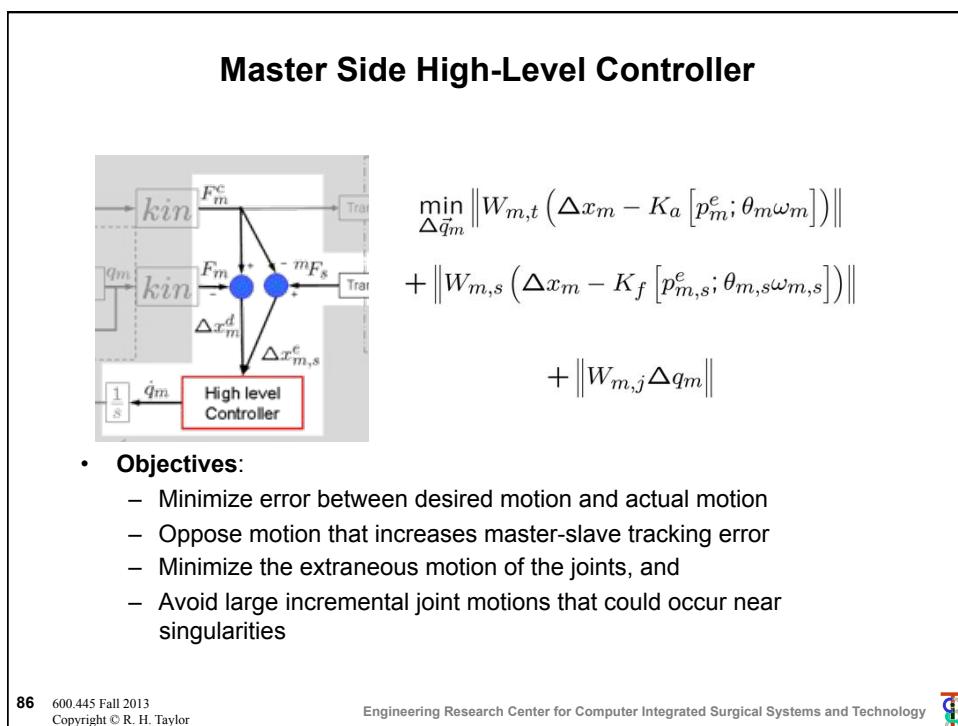
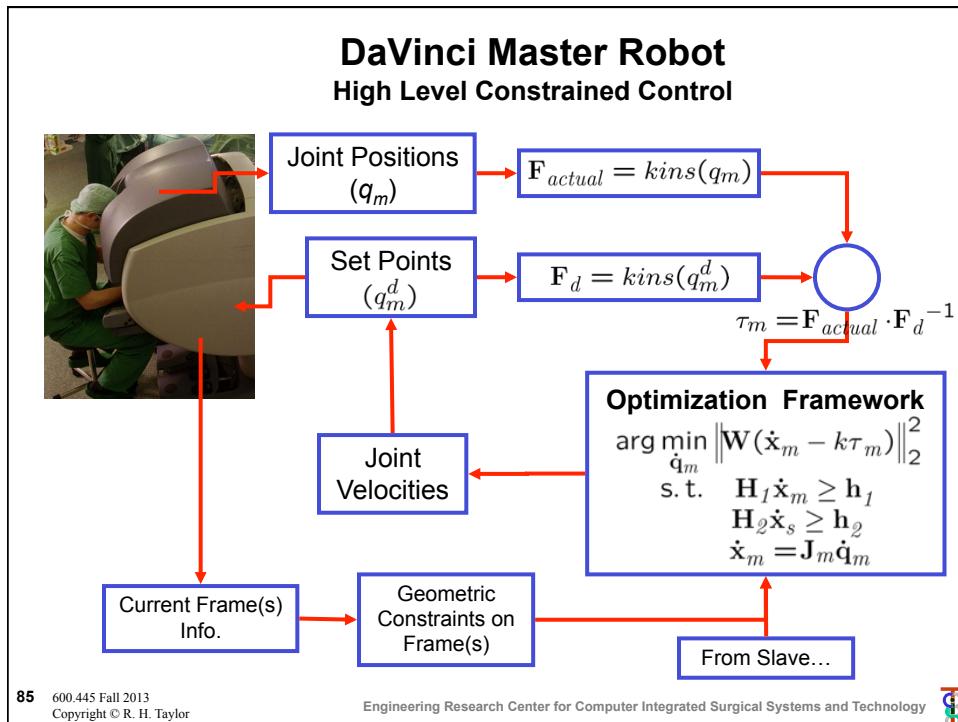
## Snake Like Robot System Architecture



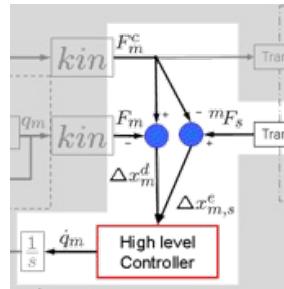
84 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology





## Master Side High-Level Controller



$$H_m \Delta q_m \geq h_m$$

that is

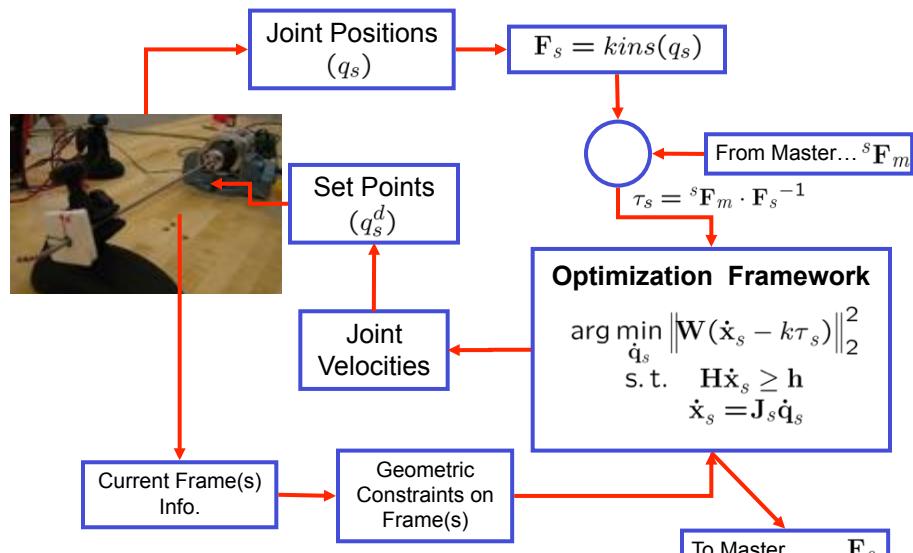
$$\begin{bmatrix} I \\ -I \\ I \\ -I \end{bmatrix} \Delta q_m \geq \begin{bmatrix} q_{m,L} - q_m \\ q_m - q_{m,U} \\ \dot{q}_{m,U} \cdot \Delta t \\ \dot{q}_{m,U} \cdot \Delta t \end{bmatrix}$$

- Constraints:**

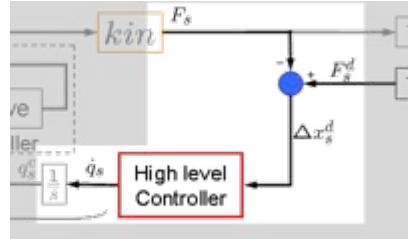
- General form:  $H_{m,j} + \Delta q_m \geq h_{m,j}$
- Not allow motion outside joint range
- Not allow motion that exceeds joint velocity limits
- Additional constraints can be added from the VF Library



## DaVinci Slave Robot High Level Constrained Control



## Slave Side High-Level Controller



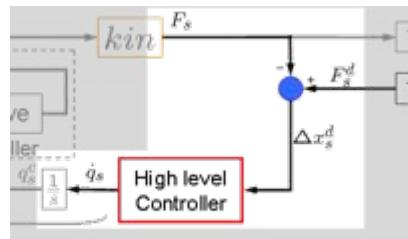
$$\begin{aligned} & \min_{\Delta q_s} \|W_{s,t}(\Delta x_s - K_a [p_s^e; \theta_s \omega_s])\| \\ & + \|W_{s,j}(q) \Delta q_s\| \\ & + \|W_{s,ss}\| \end{aligned}$$

- **Objectives:**

- Minimize error between desired motion and actual motion
- Minimize the extraneous motion of the joints, and
- Avoid large incremental joint motions that could occur near singularities



## Slave Side High-Level Controller



$$\text{such that } \begin{bmatrix} I \\ -I \\ I \\ -I \end{bmatrix} \Delta q_s \geq \begin{bmatrix} q_{s,L} - q_s \\ q_s - q_{s,U} \\ \dot{q}_{s,U} \cdot \Delta t \\ \dot{q}_{s,U} \cdot \Delta t \end{bmatrix}$$

and  $\|\vec{d}\| + \Delta x_b \cdot \hat{d}$   
 $+ \vec{v} \cdot \hat{d} + s \geq d_{safe}$   
 $0 \leq s \leq s_{lim}$

- **Constraints:**

- Not allow motion outside joint range
- Not allow motion that exceeds joint velocity limits
- **Collision avoidance between slaves**
- More constraints can be added from the VF Library



## **μForce Scaling Cooperative Control**

### **Cooperative Control**

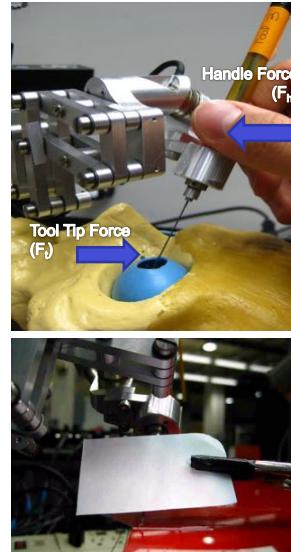
Velocity at the tool ( $V$ ) is proportional to ( $\alpha$  gain) the user's input force at the handle ( $F_h$ )

$$\dot{x} = \alpha F_h$$

### **μForce Scaling**

Amplifies ( $\gamma$  gain) the human-imperceptible forces sensed at the tool tip ( $F_t$ ) to handle interaction forces ( $F_h$ ) by modulating robot velocity.

$$\dot{x} = \alpha(F_h - \gamma F_t), \text{ e.g., } \gamma = 500$$



Kumar et al (ICRA'00); Balicki et al. (MICCAI'10); Uneri et al., BioRob 2010

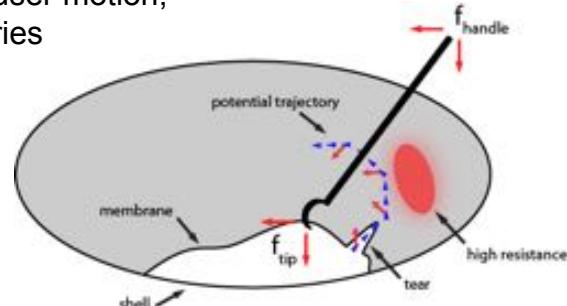
91 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## **μForce Guided Cooperative Control**

- ☛ User fights against ever increasing resistance
- ✓ Ensure safety tip force limits
- ☛ User interaction is limited at high-resistance regions
- ✓ Try to avoid those regions for later peeling
- ☛ User gets “stuck”, gives up, tries re-approach
- ✓ Ensure continuous user motion, even at the boundaries



Uneri et al., BioRob 2010

92 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## $\mu$ Force Guided Cooperative Control

- Global Limiting
  - Task-specific tip force limit
  - User controlled limit distribution
- Continuous motion at the constraint boundaries
- Virtual spring construct to ensure stability

$$f_{lim} = f_{max} \frac{|f_h|}{\|f_h\|}$$

$$\dot{x}_{lim} = \dot{x} \left( \frac{f_{lim} - |f_t|}{f_{spring}} \right)$$

### Local Force Minimization

- Guiding user towards direction of minimum resistance
- Sensitivity variable allows user override
- Haptically intuitive response
- Avoids / postpones reaching limits

Uneri et al., BioRob 2010

93 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



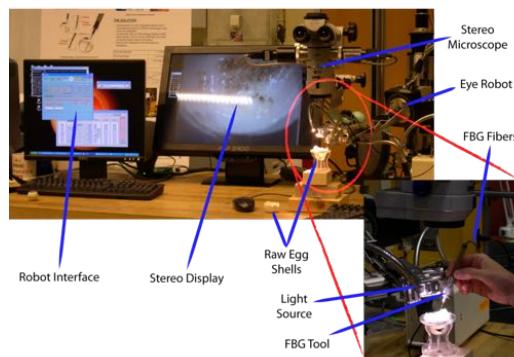
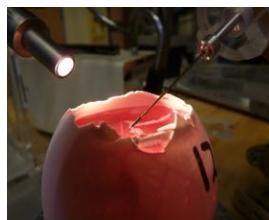
## Experimental Platform

### Focusing on:

- Properties of the tissue we interact with
- The method of interaction, i.e. performance of our algorithms

### Performed on:

- Inner shell membrane of raw eggs
- Surrogate tissue for epiretinal membrane peeling



Uneri et al., BioRob 2010

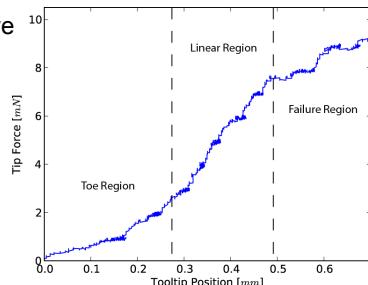
94 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Experiment: Tissue Force Characterization

- A corrected position allows us to observe tissue strain
- Controlled constant force application
  - Incremented by 1mN, with 10s delay, over a range of 1-10mN
- Characteristic curve obtained reveals a similar pattern to those seen in fibrous tissue tearing
  - Toe region: Safe
  - Linear region: Predictive
  - Failure region: Peeling



Uneri et al., BioRob 2010

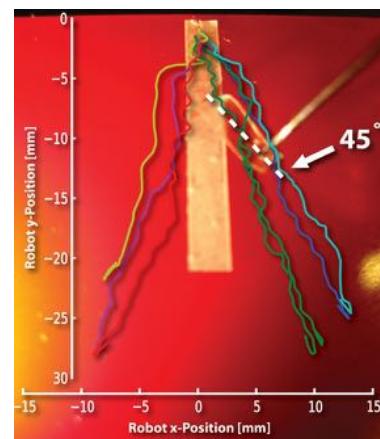
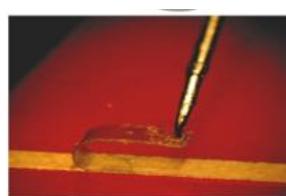
95 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Experiment: μForce Guided Cooperative Control

- Task: delaminate PVC strip with acrylic adhesive from a wax surface.
- Strip is peeled at an average of 45°
- User was guided away from the centerline in the direction of lowest resistance



Uneri et al., BioRob 2010

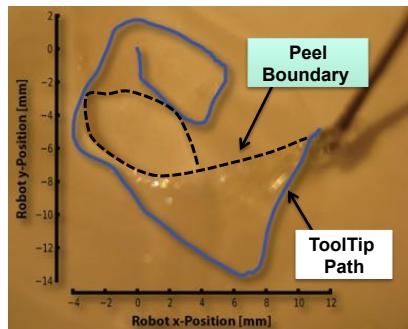
96 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Experiment: μForce Guided Cooperative Control

- Goal: Remove a section of egg inner shell membrane
- Circular trajectory consistent with the results from the strip peeling experiment
- Magnify the perception of tip forces lateral to direction of desired motion
- Results in a peel pattern seen Capsulorhexis maneuver



Peeling Inner Egg Shell Membrane

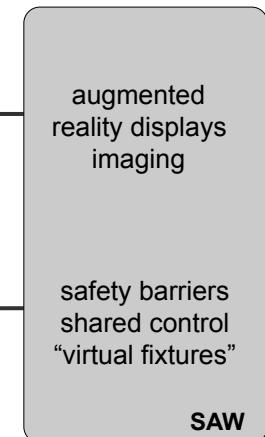
Uneri et al., BioRob 2010

97 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Information-enhanced robotic surgery

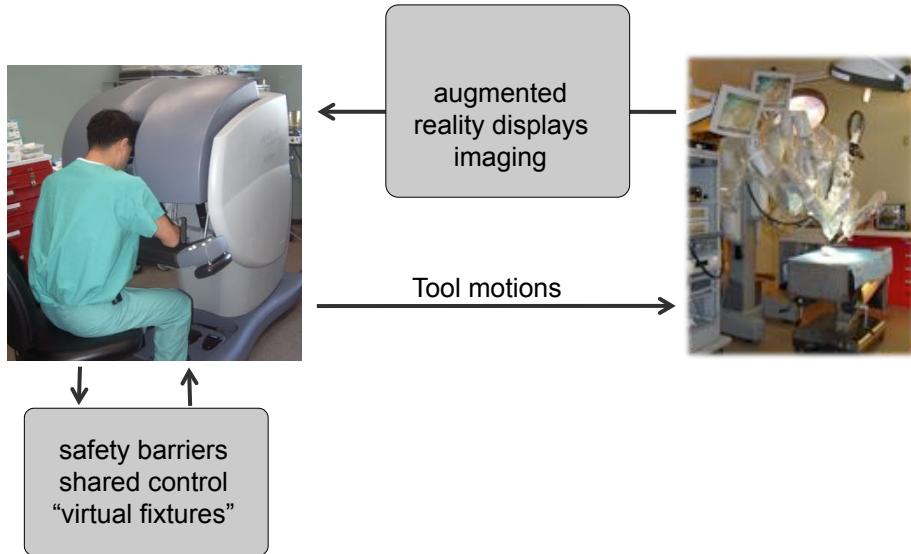


99 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Information-enhanced robotic surgery

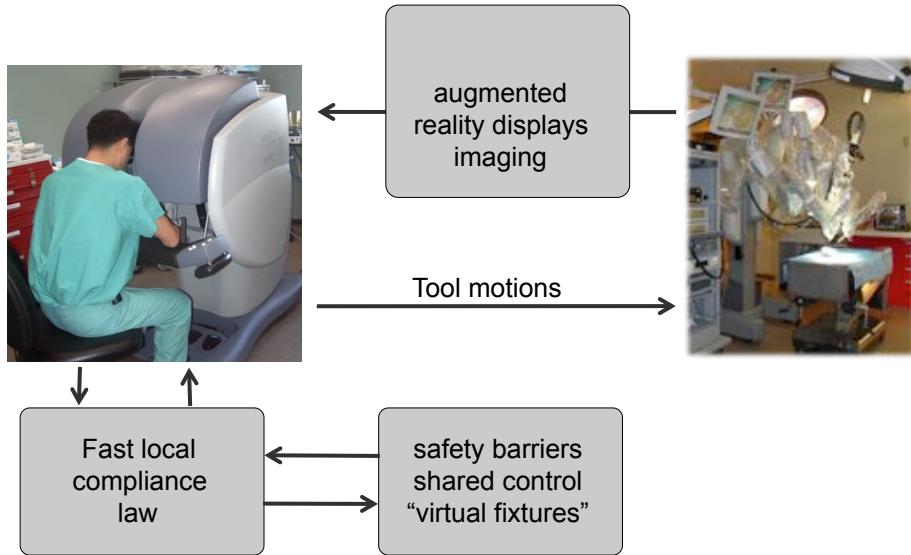


100 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Information-enhanced robotic surgery



101 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Virtual Fixture “Hook” in DaVinci API

- Experimental interface not in any clinical or commercial product.
- Specification developed jointly by JHU and Intuitive to support research
- Prototyped at JHU by Tian Xia and Russ Taylor
- Current version implemented in DaVinci “S” model by Lawton Verner at ISI, with “hooks” in a proprietary ISI Application Program Interface
- Accessed through cisst/SAW libraries

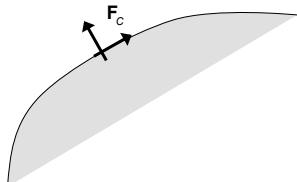


102 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Compliance virtual fixtures



$\mathbf{F} = [\mathbf{R}, \vec{\mathbf{p}}]$  = current pose;  $\dot{\mathbf{p}}$  = current velocity

$\mathbf{F}_c = [\mathbf{R}_c, \vec{\mathbf{p}}_c]$  = position compliance frame

$\vec{\mathbf{k}}^{(+)}, \vec{\mathbf{k}}^{(-)}$  = position stiffness factors

$\vec{\mathbf{b}}^{(+)}, \vec{\mathbf{b}}^{(-)}$  = damping factors

$\vec{\mathbf{g}}^{(+)}, \vec{\mathbf{g}}^{(-)}$  = force bias terms

$\mathbf{R}_o$  = orientation compliance frame

$\vec{\mathbf{k}}_o^{(+)}, \vec{\mathbf{k}}_o^{(-)}$  = orientation stiffness factors

$\vec{\tau}^{(+)}, \vec{\tau}^{(-)}$  = torque bias terms

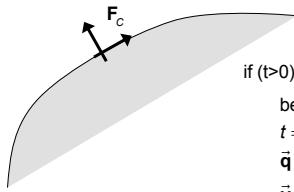
$t$  = time remaining on timeout counter

103 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Compliance virtual fixtures



```

if ( $t > 0$ ) then
begin
 $t = t - 1$ 
 $\vec{q} = \mathbf{F}_c^{-1}\vec{p} = \mathbf{R}_c^{-1}(\vec{p} - \vec{p}_c)$ 
 $\vec{v} = \mathbf{R}_c^{-1}\dot{\vec{p}}$ 
 $\vec{h} = \vec{0}; \vec{\psi} = \vec{0}$ 
for  $i \in \{x, y, z\}$  do
     $\left\{ \text{if } \vec{q}_i \leq 0 \text{ then } \vec{h}_i = \vec{g}_i^{(-)} + \vec{k}_i^{(-)}\vec{q}_i + \vec{b}_i^{(-)}\vec{v}_i \text{ else } \vec{h}_i = \vec{g}_i^{(+)} + \vec{k}_i^{(+)}\vec{q}_i + \vec{b}_i^{(+)}\vec{v}_i \right\};$ 
 $\vec{f} = \mathbf{R}_c \vec{h}; \text{ add } \vec{f} \text{ to the forces exerted on the master}$ 
 $\vec{\theta} = \text{Rodrigues vector corresponding to } \Delta \mathbf{R} = \mathbf{R}_o^{-1}\mathbf{R}$ 
for  $i \in \{x, y, z\}$  do
     $\left\{ \text{if } \vec{\theta}_i \leq 0 \text{ then } \vec{\psi}_i = \vec{\tau}_i^{(-)} + \vec{k}_i^{(-)}\vec{\theta}_i \text{ else } \vec{\psi}_i = \vec{\tau}_i^{(+)} + \vec{k}_i^{(+)}\vec{\theta}_i \right\};$ 
add  $\mathbf{R}_o \vec{\psi}$  to the torques exerted on the master
end

```

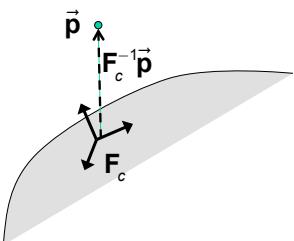
104 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Surface following virtual fixture

**Goal:** Stay on a surface; bias force drawing toward the surface; spring force resisting penetration



$\vec{p}_c$  = closest point on surface  
 $\mathbf{R}_c \vec{z}$  = surface normal at  $\vec{p}_c$

$\vec{k}^{(-)} = [0, 0, -stiffness]$

$\vec{g}^{(+)} = [0, 0, -bias]$

Others = 0

105 600.445 Fall 2013  
Copyright © R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



## Limitation and Extensions

- The specific abstraction just presented has some limitations. In particular, it separates the position and orientation compliance in a way that makes coupling of orientations and translations non-trivial.
- This can be gotten around to some extent by continually updating the virtual fixture compliance parameters.
- There are several obvious extensions that may be tried. For example, one can provide fuller matrices for virtual fixture force/torque generation. E.g.:

Compute  $\vec{q}, \vec{v}, \vec{\theta}, \vec{\phi}$  from  $\mathbf{F}_c$  and  $\mathbf{R}_o$ , where ( $\vec{\phi} = d\vec{\theta} / dt$ )

Compute a region  $i$  of local configuration space from  $\vec{q}$  and  $\vec{\theta}$

$$\begin{bmatrix} \vec{h} \\ \vec{\phi} \end{bmatrix} = \mathbf{K}_i \cdot \begin{bmatrix} \vec{q} \\ \vec{\theta} \end{bmatrix} + \mathbf{B}_i \cdot \begin{bmatrix} \vec{v} \\ \vec{\phi} \end{bmatrix} + \begin{bmatrix} \vec{g}_i \\ \vec{\tau}_i \end{bmatrix}$$

