

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

(Part 1)

Russell H. Taylor  
601.455/655

1 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



1

## Disclosures & Acknowledgments

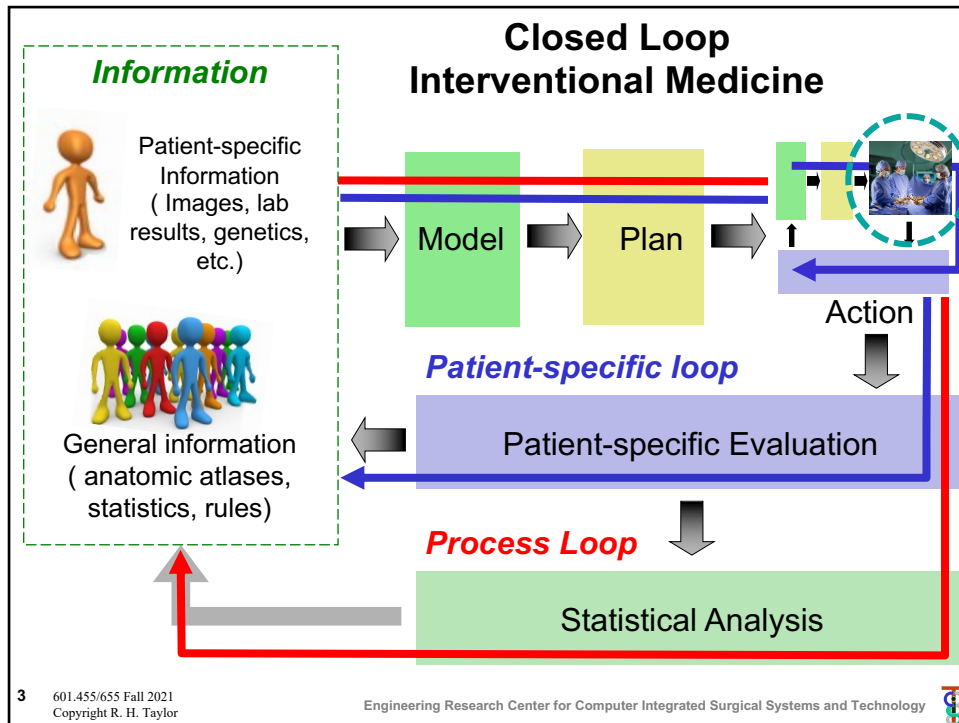
- **This is the work of many people**
- Some of the work reported in this presentation was supported by fellowship grants from Intuitive Surgical and Philips Research North America to Johns Hopkins graduate students and by equipment loans from Intuitive Surgical, Think Surgical, Philips, Kuka, and Carl Zeiss Meditec.
- Some of the work reported in this talk incorporates intellectual property that is owned by Johns Hopkins University and that has been or may be licensed to outside entities, including including Intuitive Surgical, Varian Medical Systems, Philips Nuclear Medicine, Galen Robotics and other corporate entities. Prof. Taylor has received or may receive some portion of the license fees. Also, Dr. Taylor is a paid consultant to and owns equity in Galen Robotics, Inc. These arrangements have been reviewed and approved by JHU in accordance with its conflict of interest policy.
- Much of this work has been funded by Government research grants, including NSF grants EEC9731478 and IIS0099770 and NIH grants R01-EB016703, R01-EB007969, R01-CA127144, R42-RR019159, and R21-EB0045457; by Industry Research Contracts, including from Think Surgical; by gifts to Johns Hopkins University from John C. Malone, Richard Swirnow and Paul Maritz; and by Johns Hopkins University internal funds.

2 601.455/655 Fall 2021  
Copyright R. H. Taylor

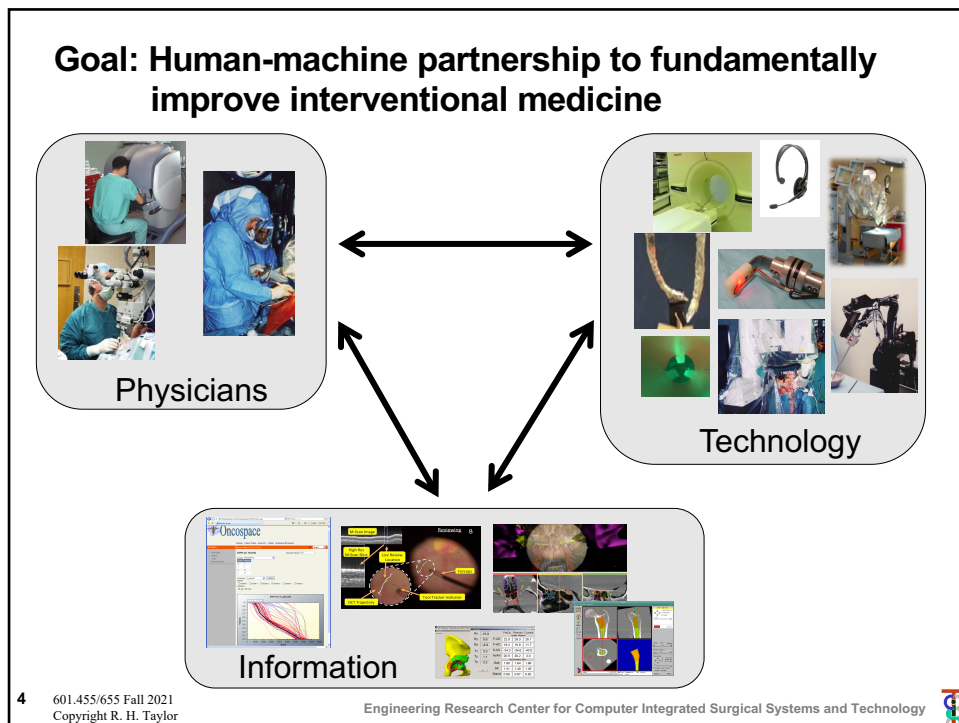
Engineering Research Center for Computer Integrated Surgical Systems and Technology



2



3



4

## 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

5

601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



5

## 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.

6

601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



6

## Surgical CAD/CAM: Orthopaedic Robots



Robodoc



PPS: device "maps out" to existing drill platforms. The PPS instrument has been designed for use in microsurgical and orthopaedic applications.

Blue Belt Technologies



D. Glzman & M. Shoham



ACROBOT surgical robot

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

7 601.455/655 Fall 2021  
Copyright R. H. Taylor

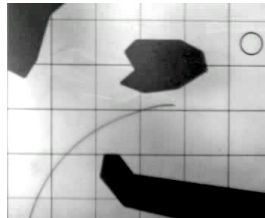
Engineering Research Center for Computer Integrated Surgical Systems and Technology

7

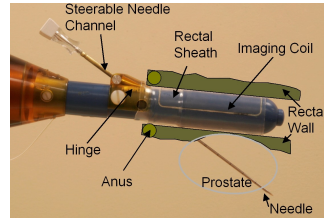
## Image-guided needle placement



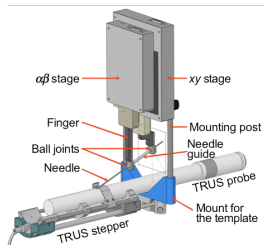
Masamune, Fichtinger, Iordachita, ...



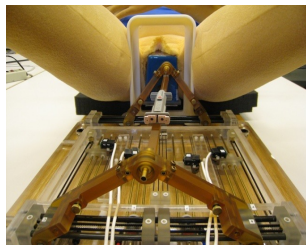
Okamura, Webster, ...



Krieger, Fichtinger, Whitcomb, ...



Fichtinger, Kazanides, Burdette, Song ...



Iordachita, Fischer, Hata...



Taylor, Masamune, Susil, Patriciu, Stoianovici, ...

8 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology

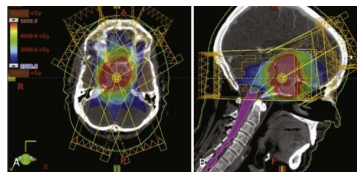
8

## 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/)

9

601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



9

## 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), mouse, or other
  - 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

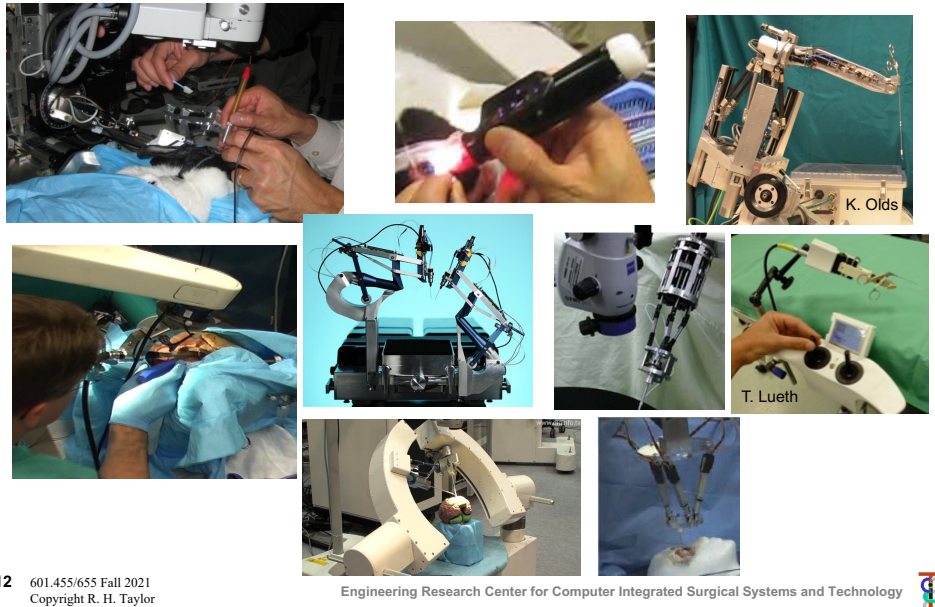
601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



10

## Precision Augmentation



12

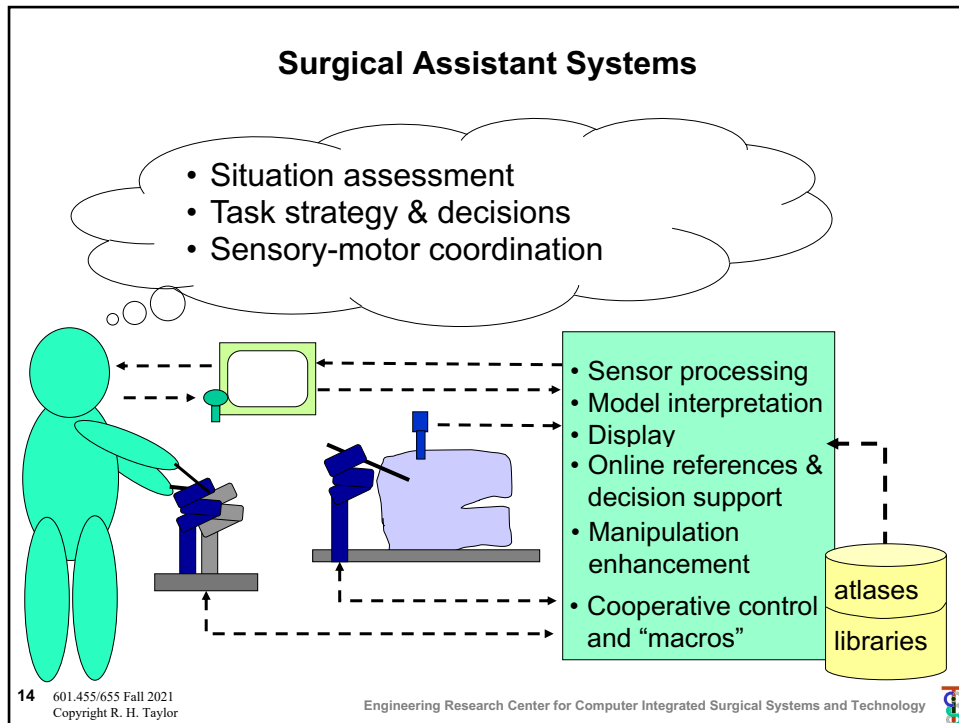
## Common classes of medical robots

- **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.

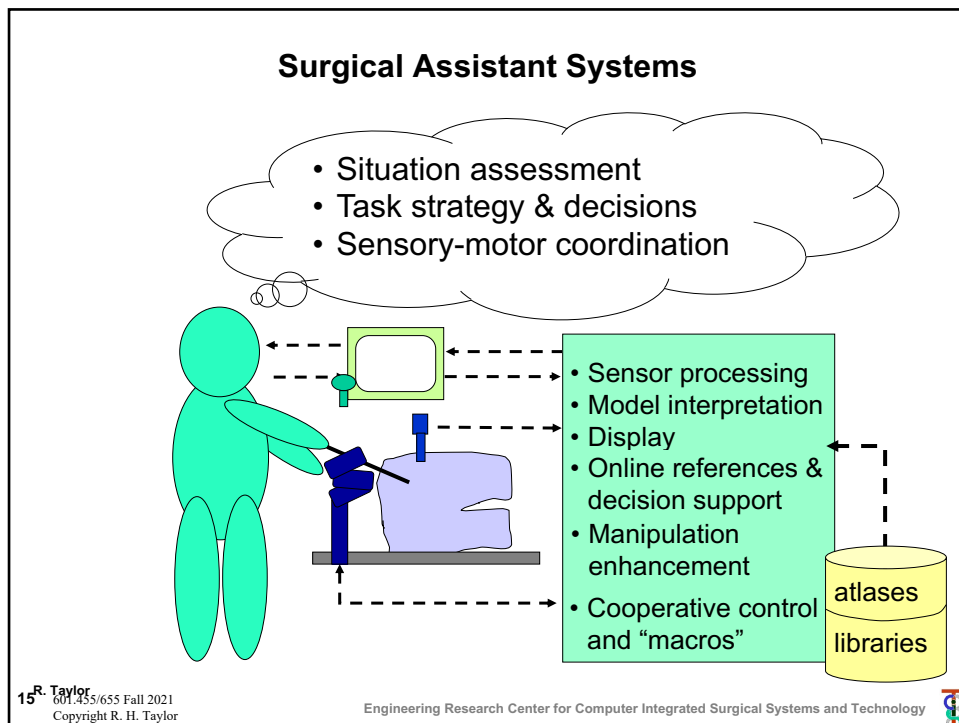
13 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology

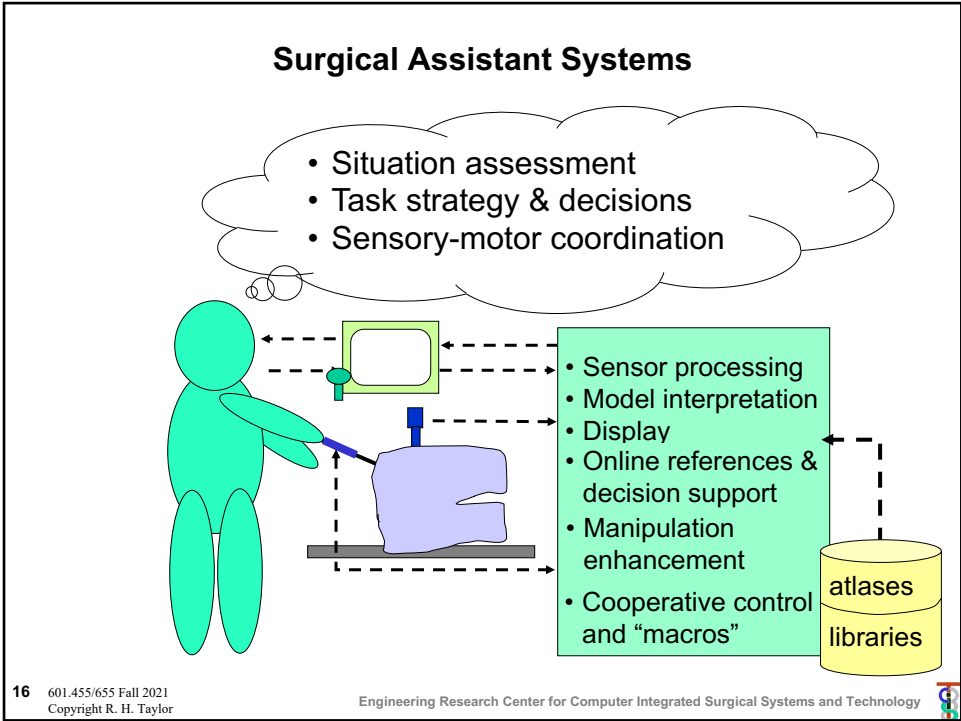
13



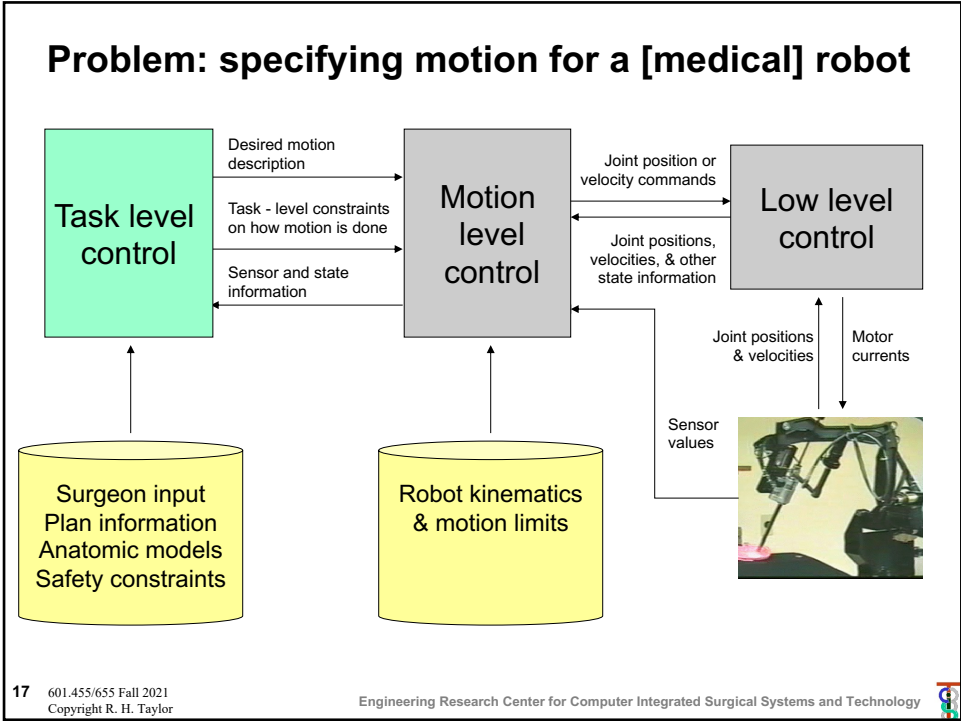
14



15



16



17



## Background: Robot Kinematics



$$\text{Pose } F(\vec{q} + \Delta\vec{q}) = kins(\vec{q} + \Delta\vec{q})$$

$$\Delta F \cdot F = kins(\vec{q} + \Delta\vec{q})$$

$$\Delta F = kins(\vec{q} + \Delta\vec{q}) kins(\vec{q})^{-1}$$

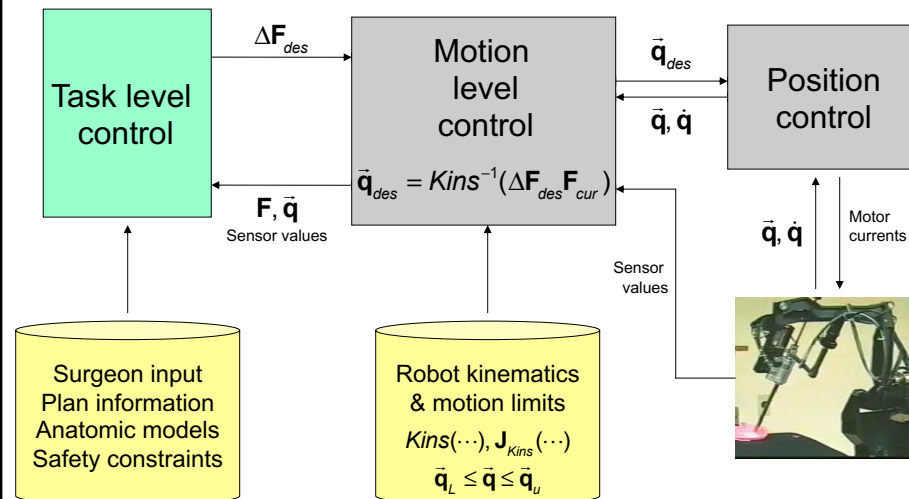
18 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



18

## One implementation



19 601.455/655 Fall 2021  
Copyright R. H. Taylor

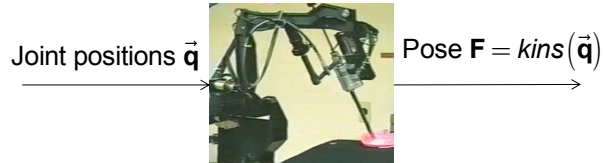
Engineering Research Center for Computer Integrated Surgical Systems and Technology



19

## Background: Jacobean Robot Motion Control

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



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

20 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



20

## Background: Jacobean Robot Motion Control

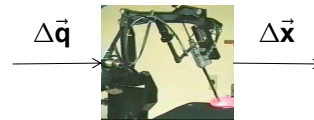
Let  $\mathbf{F}=[\mathbf{R},\bar{\mathbf{p}}]$  be the current pose of a robot end effector and  $\bar{\mathbf{q}}=[q_1,\dots,q_N]$  be the current joint position values corresponding to  $\mathbf{F}$ . I.e.,  $\mathbf{F}=\mathit{Kins}(\bar{\mathbf{q}})$ , where  $\mathit{Kins}(\dots)$  is a function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

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

$$\Delta\mathbf{F} = \mathit{Kins}(\bar{\mathbf{q}} + \Delta\bar{\mathbf{q}}) \mathit{Kins}(\bar{\mathbf{q}})^{-1}$$

which we typically linearize as

$$\Delta\bar{\mathbf{x}} = \begin{bmatrix} \bar{\alpha} \\ \bar{\varepsilon} \end{bmatrix} \approx \mathbf{J}_{\mathit{Kins}}(\bar{\mathbf{q}}) \Delta\bar{\mathbf{q}}$$



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

$\mathbf{F} \bullet \Delta\mathbf{F}=\mathit{Kins}(\bar{\mathbf{q}} + \Delta\bar{\mathbf{q}})$ , then we will get a slightly different expression for  $\mathbf{J}_{\mathit{Kins}}(\bar{\mathbf{q}})$ , though the flavor will be the same

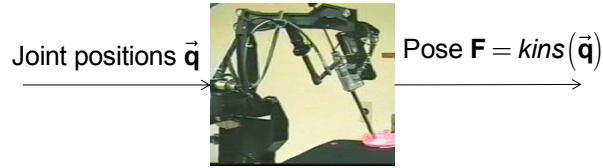
21 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



21

## Background: Jacobean Robot Motion Control



$$\text{Pose } \mathbf{F}(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) = \text{kins}(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}})$$

$$\Delta\mathbf{F} \cdot \mathbf{F} = \text{kins}(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}})$$

$$\Delta\mathbf{F} = \text{kins}(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}) \text{kins}(\vec{\mathbf{q}})^{-1}$$

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

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

22 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



22

## Background: Jacobean Robot Motion Control

### Alternative way of solving:

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

$$\Delta\vec{\mathbf{q}} \approx \underset{\Delta\vec{\mathbf{q}}}{\text{argmin}} \left\| \mathbf{J}(\vec{\mathbf{q}}) \Delta\vec{\mathbf{q}} - \begin{bmatrix} \vec{\alpha} \\ \varepsilon \end{bmatrix} \right\|^2$$

### Advantages:

- Produces solution even if kinematically redundant or kinematically deficient
- Can add auxiliary constraints or objective function terms

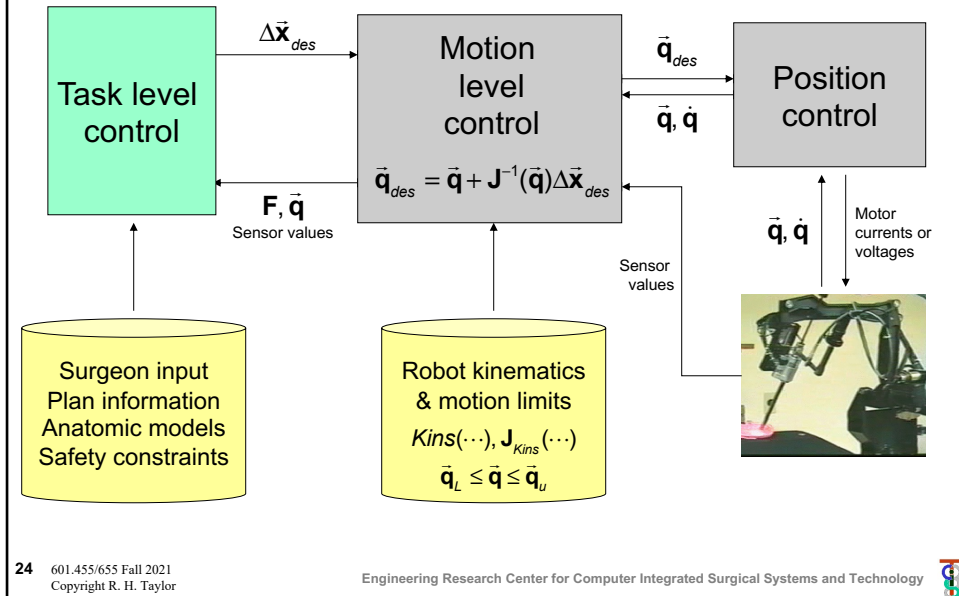
23 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



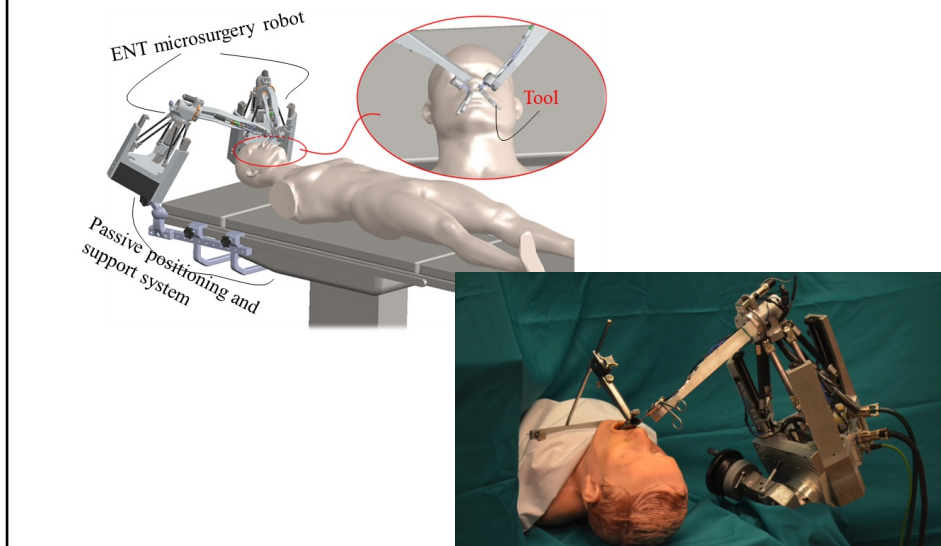
23

## Jacobian motion control implementation



24

## What about parallel-link robots?



25 601.455/655 Fall 2021 Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology

25

$\mathbf{F}_p(\vec{q})$   
 $\vec{a}_i$   
 $\mathbf{F}_{pe}(\vec{\theta})$   
 $\mathbf{F}_e(\vec{q}, \vec{\theta})$   
 $q_i$   
 $\vec{b}_i$

$$\vec{q} = [q_1, \dots, q_6]^T = \text{invk}(\mathbf{F}_p)$$

$$q_i = \|\mathbf{F}_p(\vec{q})\vec{a}_i - \vec{b}_i\|$$

$$\mathbf{F}_p(\vec{q} + \Delta\vec{q}) = \Delta\mathbf{F}_p(\vec{q}, \Delta\vec{q})\mathbf{F}_p(\vec{q})$$

$$\Delta\mathbf{F}_p \approx [\mathbf{I} + \text{sk}(\vec{\alpha}_p), \vec{\varepsilon}_p]$$

$$\vec{\gamma}_p = [\vec{\alpha}_p^T, \vec{\varepsilon}_p^T]^T$$

$$\Delta\vec{q} \approx \mathbf{J}_{\text{invk}}(\mathbf{F}_p(\vec{q}))\vec{\gamma}_p$$

$$\vec{\gamma}_p \approx \mathbf{J}_{\text{invk}}(\mathbf{F}_p(\vec{q}))^{-1}\Delta\vec{q}$$

26 601.455/655 Fall 2021  
 Copyright R. H. Taylor  
 Engineering Research Center for Computer Integrated Surgical Systems and Technology

26

$\mathbf{F}_p(\vec{q})$   
 $\vec{a}_i$   
 $\mathbf{F}_{pe}(\vec{\theta})$   
 $\mathbf{F}_e(\vec{q}, \vec{\theta})$   
 $q_i$   
 $\vec{b}_i$

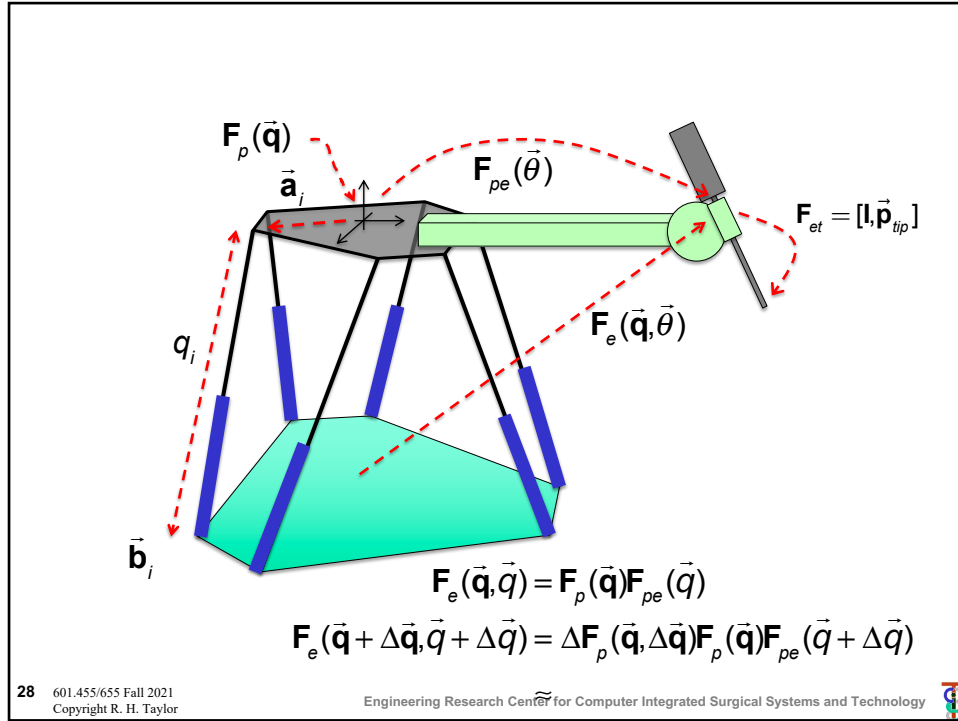
$$\mathbf{F}_{pe}(\vec{\theta} + \Delta\vec{\theta}) = \mathbf{F}_{pe}(\vec{\theta})\Delta\mathbf{F}_{pe}^{\text{(right)}}(\vec{\theta}, \Delta\vec{\theta})$$

$$\approx \mathbf{F}_{pe}(\vec{\theta}) \cdot [\mathbf{I} + \text{sk}(\vec{\alpha}_{pe}), \vec{\varepsilon}_{pe}]$$

$$\begin{bmatrix} \vec{\alpha}_{pe} \\ \vec{\varepsilon}_{pe} \end{bmatrix} = \vec{\gamma}_{pe} = \mathbf{J}_{pe}(\vec{\theta})\Delta\vec{\theta} = \begin{bmatrix} \mathbf{J}_{pe}^{\text{R}}(\vec{\theta}) \\ \mathbf{J}_{pe}^{\text{P}}(\vec{\theta}) \end{bmatrix} \Delta\vec{\theta}$$

27 601.455/655 Fall 2021  
 Copyright R. H. Taylor  
 Engineering Research Center for Computer Integrated Surgical Systems and Technology

27



28

$$\mathbf{F}_e(\vec{q}, \vec{\theta}) = \mathbf{F}_p(\vec{q})\mathbf{F}_{pe}(\vec{\theta})$$

$$\mathbf{F}_e(\vec{q} + \Delta\vec{q}, \vec{\theta} + \Delta\vec{\theta}) = \Delta\mathbf{F}_p(\vec{q}, \Delta\vec{q})\mathbf{F}_p(\vec{q})\mathbf{F}_{pe}(\vec{\theta} + \Delta\vec{\theta})$$

$$\Delta\mathbf{F}_e(\vec{q}, \Delta\vec{q}, \vec{\theta}, \Delta\vec{\theta})\mathbf{F}_e(\vec{q}, \vec{\theta}) = \Delta\mathbf{F}_p(\vec{q}, \Delta\vec{q})\mathbf{F}_p(\vec{q})\mathbf{F}_{pe}(\vec{\theta})\Delta\mathbf{F}_{pe}(\vec{\theta}, \Delta\vec{\theta})$$

$$\Delta\mathbf{F}_e(\vec{q}, \Delta\vec{q}, \vec{\theta}, \Delta\vec{\theta}) = \Delta\mathbf{F}_p(\vec{q}, \Delta\vec{q})\mathbf{F}_p(\vec{q})\mathbf{F}_{pe}(\vec{\theta})\Delta\mathbf{F}_{pe}(\vec{\theta}, \Delta\vec{\theta})\mathbf{F}_e(\vec{q}, \vec{\theta})^{-1}$$

$$= \Delta\mathbf{F}_p(\vec{q}, \Delta\vec{q})\mathbf{F}_p(\vec{q})\mathbf{F}_{pe}(\vec{\theta})\Delta\mathbf{F}_{pe}(\vec{\theta}, \Delta\vec{\theta})\mathbf{F}_e(\vec{q}, \vec{\theta})^{-1}$$

$$\Delta\mathbf{R}_e(\vec{q}, \Delta\vec{q}, \vec{\theta}, \Delta\vec{\theta}) = \Delta\mathbf{R}_p(\vec{q}, \Delta\vec{q})\mathbf{R}_p(\vec{q})\mathbf{R}_{pe}(\vec{\theta})\Delta\mathbf{R}_{pe}(\vec{\theta}, \Delta\vec{\theta})\Delta\mathbf{R}_{pe}(\vec{q}, \vec{\theta})^{-1}\mathbf{R}_{pe}(\vec{q}, \vec{\theta})^{-1}$$

$$\mathbf{I} + sk(\vec{\alpha}_e) \approx (\mathbf{I} + sk(\vec{\alpha}_p))\mathbf{R}_e(\vec{q}, \vec{\theta})(\mathbf{I} - sk(\vec{\alpha}_{pe}))\mathbf{R}_e(\vec{q}, \vec{\theta})^{-1}$$

$$\mathbf{I} + sk(\vec{\alpha}_e) \approx (\mathbf{I} + sk(\vec{\alpha}_p))(\mathbf{I} - \mathbf{R}_e(\vec{q}, \vec{\theta})sk(\vec{\alpha}_{pe})\mathbf{R}_e(\vec{q}, \vec{\theta})^{-1})$$

$$\mathbf{I} + sk(\vec{\alpha}_e) \approx (\mathbf{I} + sk(\vec{\alpha}_p))(\mathbf{I} - sk(\mathbf{R}_e(\vec{q}, \vec{\theta})\vec{\alpha}_{pe}))$$

$$\vec{\alpha}_e \approx \vec{\alpha}_p - \mathbf{R}_e(\vec{q}, \vec{\theta})\vec{\alpha}_{pe}$$

$$\vec{\alpha}_e \approx \mathbf{J}_{mvk}^{\vec{\alpha}}(\mathbf{F}_p(\vec{q}))^{-1}\Delta\vec{q} - \mathbf{R}_e(\vec{q}, \vec{\theta})\vec{\alpha}_{pe}$$

29 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology

29

$$\mathbf{F}_e(\vec{\mathbf{q}}, \vec{\theta}) = \mathbf{F}_p(\vec{\mathbf{q}}) \mathbf{F}_{pe}(\vec{\theta})$$

$$\mathbf{F}_e(\vec{\mathbf{q}} + \Delta\vec{\mathbf{q}}, \vec{\theta} + \Delta\vec{\theta}) = \Delta\mathbf{F}_p(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}) \mathbf{F}_p(\vec{\mathbf{q}}) \mathbf{F}_{pe}(\vec{\theta} + \Delta\vec{\theta})$$

$$\Delta\mathbf{F}_e(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}, \vec{\theta}, \Delta\vec{\theta}) \mathbf{F}_e(\vec{\mathbf{q}}, \vec{\theta}) = \Delta\mathbf{F}_p(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}) \mathbf{F}_p(\vec{\mathbf{q}}) \mathbf{F}_{pe}(\vec{\theta}) \Delta\mathbf{F}_{pe}(\vec{\theta}, \Delta\vec{\theta})$$

$$\Delta\mathbf{F}_e(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}, \vec{\theta}, \Delta\vec{\theta}) \vec{\mathbf{p}}_e = \Delta\mathbf{F}_p(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}) \mathbf{F}_e \Delta\vec{\mathbf{p}}_{pe}$$

$$\Delta\mathbf{R}_e(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}, \vec{\theta}, \Delta\vec{\theta}) \vec{\mathbf{p}}_e + \Delta\vec{\mathbf{p}}_e = \Delta\mathbf{R}_p(\vec{\mathbf{q}}, \Delta\vec{\mathbf{q}}) (\mathbf{R}_e \Delta\vec{\mathbf{p}}_{pe} + \vec{\mathbf{p}}_e) + \Delta\vec{\mathbf{p}}_p$$

$$(\mathbf{I} + sk(\vec{\alpha}_e)) \vec{\mathbf{p}}_e + \vec{\varepsilon}_e \approx (\mathbf{I} + sk(\vec{\alpha}_p)) (\mathbf{R}_e \vec{\varepsilon}_{pe} + \vec{\mathbf{p}}_e) + \vec{\varepsilon}_p$$

$$\vec{\mathbf{p}}_e + sk(\vec{\alpha}_e) \vec{\mathbf{p}}_e + \vec{\varepsilon}_e \approx \vec{\mathbf{p}}_e + sk(\vec{\alpha}_p) \vec{\mathbf{p}}_e + \mathbf{R}_e \vec{\varepsilon}_{pe} + \vec{\varepsilon}_p$$

$$\vec{\varepsilon}_e \approx -sk(\vec{\mathbf{p}}_e) \vec{\alpha}_p + \mathbf{R}_e \vec{\varepsilon}_{pe} + \vec{\varepsilon}_p$$

30 601.455/655 Fall 2021 Copyright R. H. Taylor Engineering Research Center for Computer Integrated Surgical Systems and Technology

30

### Steady Hand Robot

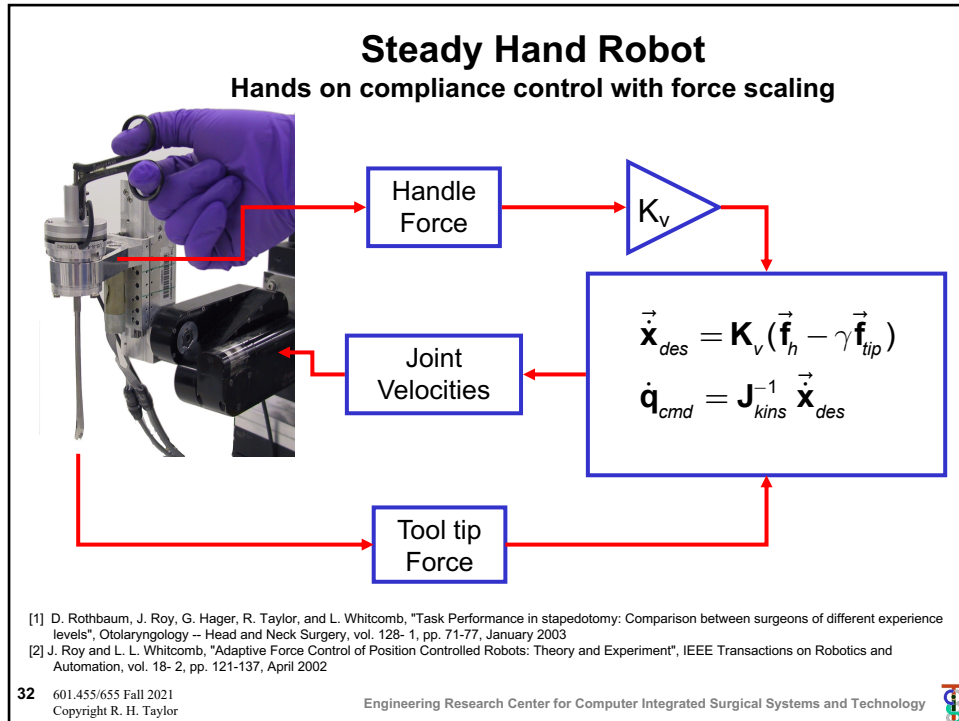
Hands on compliance control

[1] R. H. Taylor, J. Funda, B. Eldridge, S. Gomory, K. Gruben, D. LaRose, M. Talamini, L. Kavoussi, and J. anderson, "Telerobotic assistant for laparoscopic surgery.", IEEE Eng Med Biol, vol. 14- 3, pp. 279-288, 1995

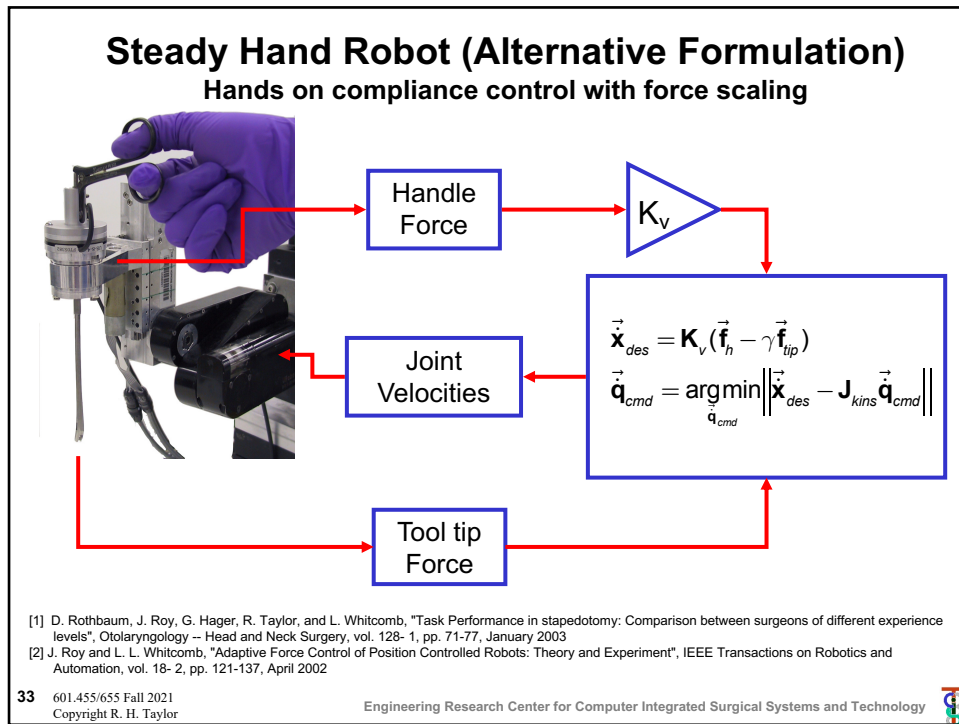
[2] R. Taylor, P. Jensen, L. Whitcomb, A. Barnes, R. Kumar, D. Stoianovici, P. Gupta, Z. Wang, E. deJuan, and L. Kavoussi, "A Steady-Hand Robotic System for Microsurgical Augmentation", International Journal of Robotics Research, vol. 18- 12, pp. 1201-1210, 1999

31 601.455/655 Fall 2021 Copyright R. H. Taylor Engineering Research Center for Computer Integrated Surgical Systems and Technology

31



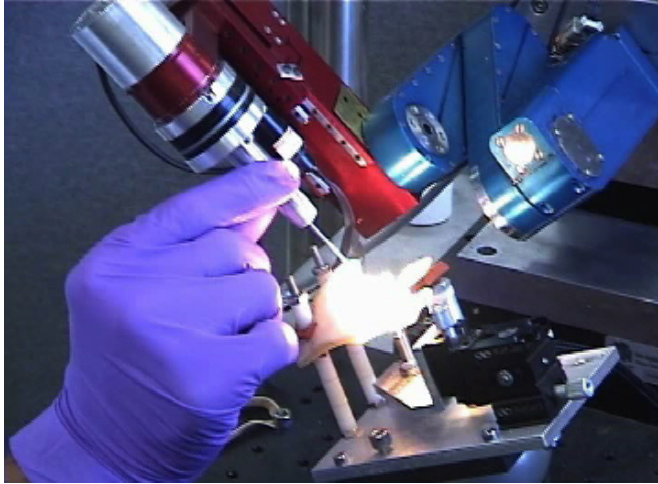
32



33



## Example: Fenestratration of Stapes Footplate



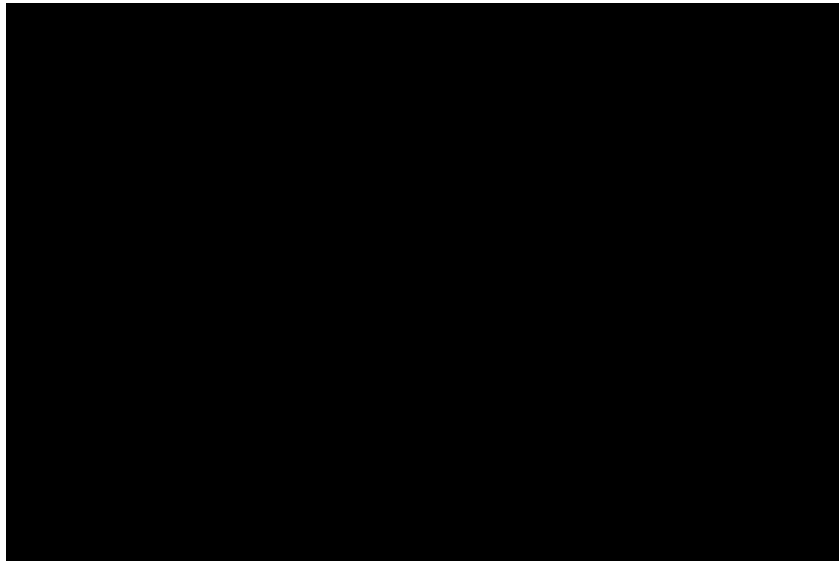
34 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



34

## Example: Fenestratration of Stapes Footplate



35 601.455/655 Fall 2021  
Copyright R. H. Taylor

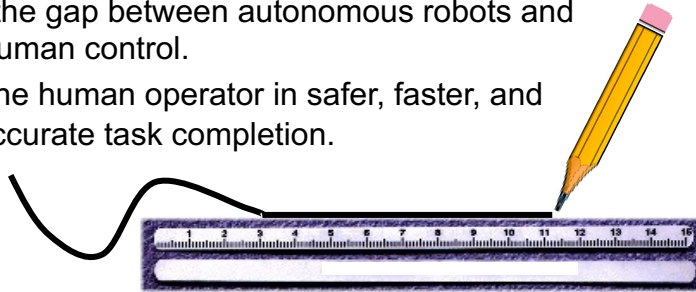
Engineering Research Center for Computer Integrated Surgical Systems and Technology



35

## 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

36 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



36

## 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. Jakopec, 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

37 601.455/655 Fall 2021  
Copyright R. H. Taylor

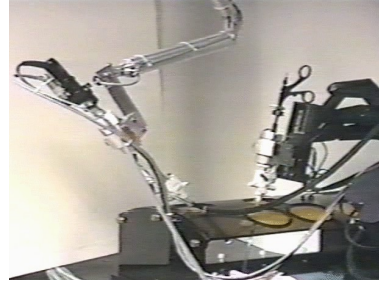
Engineering Research Center for Computer Integrated Surgical Systems and Technology



37

## 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



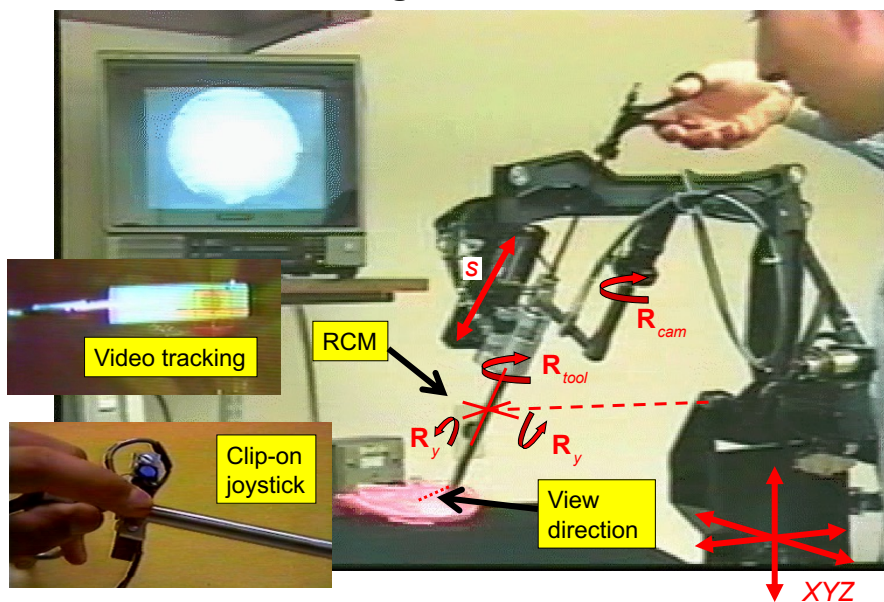
38 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



38

## LARS degrees of freedom



39 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



39

## LARS Video



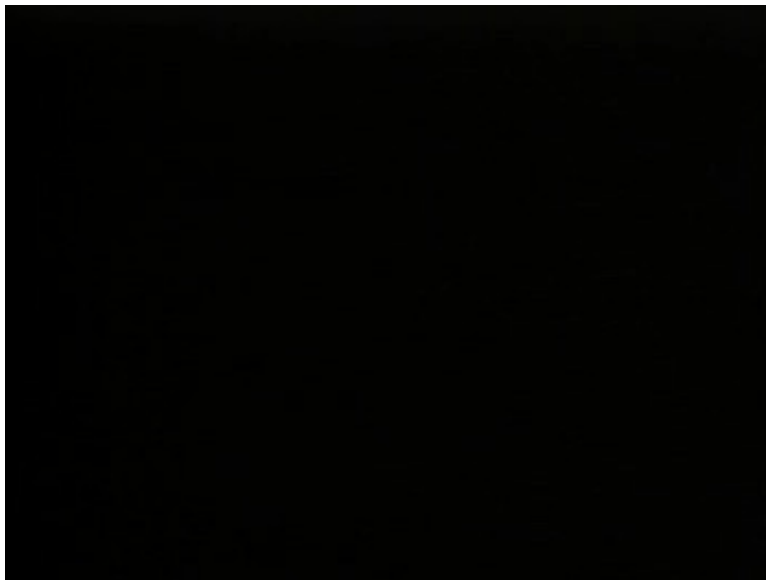
40 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



40

## LARS Video



41 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



41

## 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.*

42 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



42

## 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

43 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



43

### Example: keep tool tip near a point

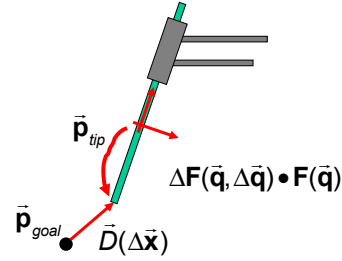
$$\begin{aligned}\bar{D}(\Delta\bar{\mathbf{x}}) &= \Delta\mathbf{F}(\bar{\mathbf{q}}, \Delta\bar{\mathbf{q}}) \cdot \mathbf{F} \cdot \bar{\mathbf{p}}_{tip} - \bar{\mathbf{p}}_{goal} \\ &= \bar{\alpha} \times \bar{\mathbf{t}} + \bar{\varepsilon} + \bar{\mathbf{t}} - \bar{\mathbf{p}}_{goal} \quad \text{where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

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

$$\Delta\mathbf{q}_{des} = \underset{\Delta\bar{\mathbf{q}}}{\text{argmin}} \|\bar{D}(\Delta\bar{\mathbf{x}})\|^2 = \|\bar{\alpha} \times \bar{\mathbf{t}} + \bar{\varepsilon} + \bar{\mathbf{t}} - \bar{\mathbf{p}}_{goal}\|^2$$

Subject to

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



44 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



44

### Example: keep tool tip near a point

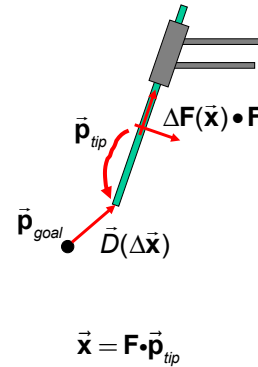
Suppose we want to stay as close as possible while never going beyond 3mm from goal and also obeying joint limits, but we also want to minimize the change in direction of the tool shaft

$$\Delta\mathbf{q}_{des} = \underset{\Delta\bar{\mathbf{q}}}{\text{argmin}} \zeta \|\bar{D}(\Delta\bar{\mathbf{x}})\|^2 + \eta \|\alpha \times \mathbf{R} \cdot \bar{\mathbf{z}}\|^2$$

Subject to

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

Note weighting factors



45 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



45

## Solving the optimization problem

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

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

subject to

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

- 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)
- **Can also minimize other objective functions**
  - E.g., minimize an L1 norm (linear programming problem)

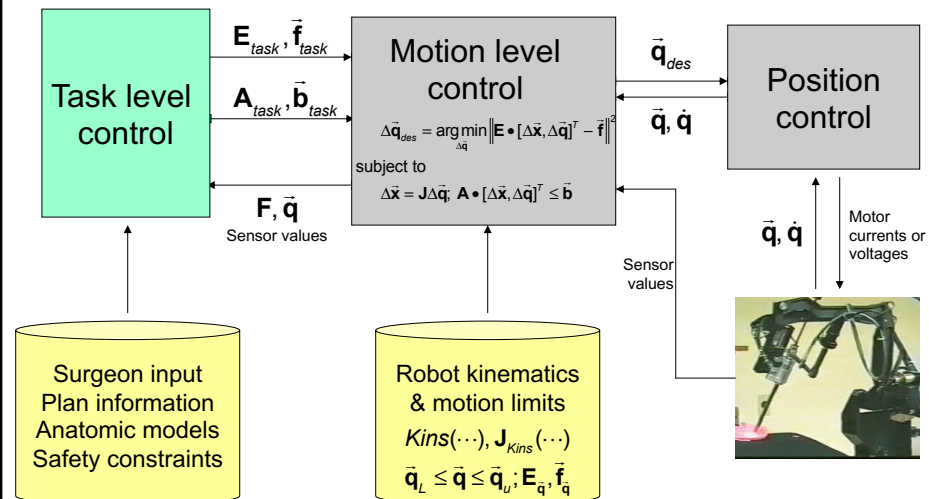
46 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



46

## Linear least squares implementation



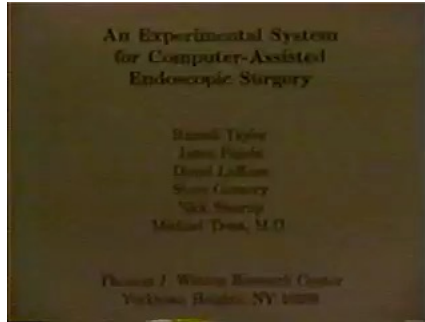
47 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



47

## Some IBM Movies



Early Constrained Motion System (LapSYS)

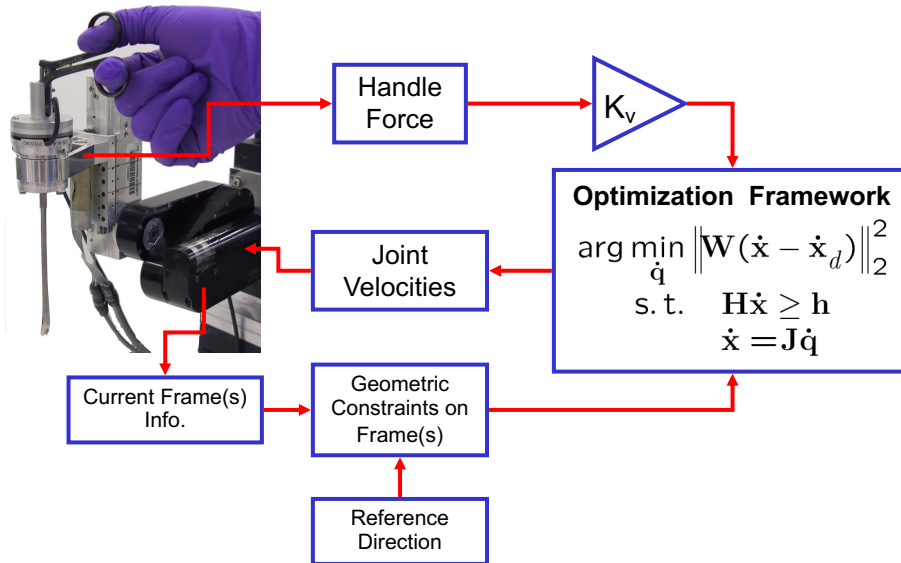


Vision-guided targeting



48

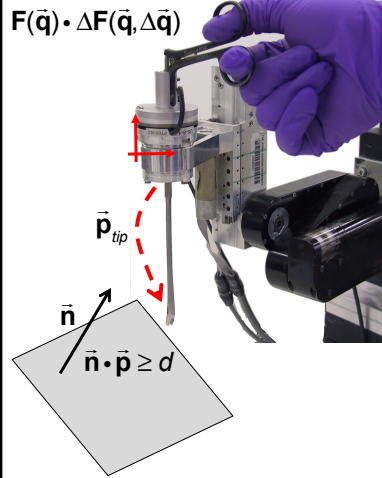
## Steady Hand Robot High Level Constrained Control



49



### Example: Hands-on Guiding with Forbidden Half Space



$$\Delta \vec{q} = \operatorname{argmin}_{\Delta \vec{q}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|$$

Such that

$$d \leq \vec{n} \cdot (\mathbf{F}(\vec{q}) \Delta \mathbf{F}_{rhs}(\vec{q}, \Delta \vec{q}) \cdot \vec{p}_{tip})$$

Note here we are using the right hand side Jacobean, since the force sensor is associated with the tool attachment point, and it is more natural for the motions to comply to pushes on the tool handle.

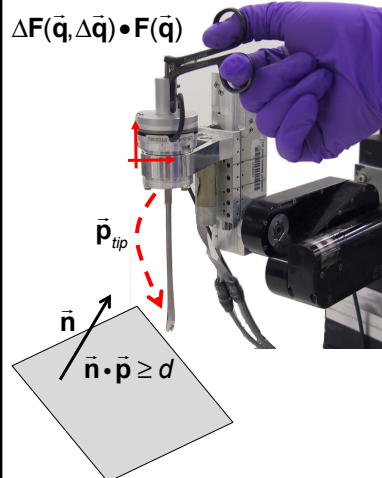
50 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



50

### Example: Hands-on Guiding with Forbidden Half Space



$$\Delta \vec{q} = \operatorname{argmin}_{\Delta \vec{q}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|$$

Such that

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

$$d \leq \vec{n} \cdot (\mathbf{F}(\vec{q}) \cdot (\vec{\alpha} \times \vec{p}_{tip} + \vec{\varepsilon} + \vec{p}_{tip}))$$

i.e.,

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

$$d \leq \vec{n} \cdot (\mathbf{R}(\vec{q}) \cdot (\vec{\alpha} \times \vec{p}_{tip} + \vec{\varepsilon} + \vec{p}_{tip}) + \vec{p}_{kins})$$

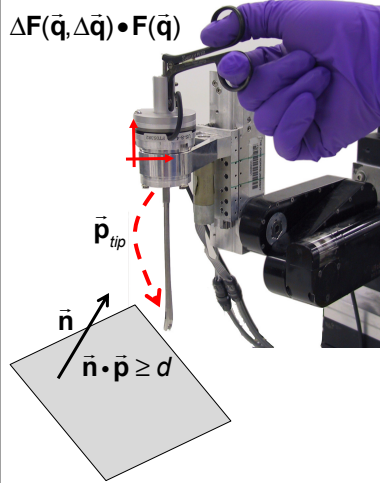
51 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



51

### Example: Hands-on Guiding with Forbidden Half Space



$$\Delta \vec{q} = \arg \min_{\Delta \vec{q}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|^2$$

Such that

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

$$d - \vec{n} \cdot \vec{x} \leq \vec{n} \cdot (\mathbf{R}(\vec{q}) \cdot (\vec{\alpha} \times \vec{p}_{tip} + \vec{\varepsilon}))$$

$$\vec{x} = \mathbf{F}(\vec{q}) \vec{p}_{tip}$$

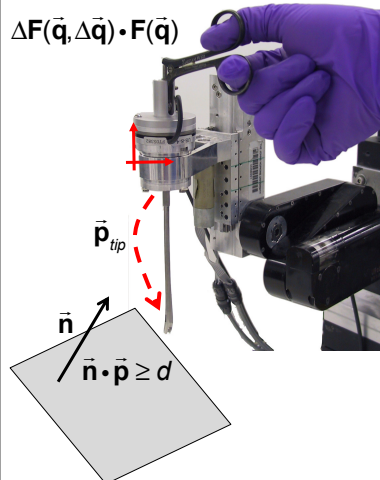
52 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



52

### Example: Hands-on Guiding with Forbidden Half Space



$$\Delta \vec{q} = \arg \min_{\Delta \vec{q}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{kins}(\vec{q}) \Delta \vec{q} \right\|^2$$

Such that

$$d \leq \vec{n} \cdot (\Delta \mathbf{F}(\vec{q}, \Delta \vec{q}) \cdot \mathbf{F}(\vec{q}) \cdot \vec{p}_{tip})$$

If we use the LHS Jacobean, we get something similar. Note however that in this case the gain matrix will likely be pose dependent, since the it is more natural for the surgeon's hand to follow the tool. So it is useful to be able to make the conversion ...

53 601.455/655 Fall 2021  
Copyright R. H. Taylor

Engineering Research Center for Computer Integrated Surgical Systems and Technology



53

## LHS versus RHS Jacobians

$$\Delta \mathbf{F}(\mathbf{J}_{kins}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}}) \cdot \mathbf{F}(\vec{\mathbf{q}}) = \mathbf{F}(\vec{\mathbf{q}}) \cdot \Delta \mathbf{F}(\mathbf{J}_{rhs}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}})$$

$$\Delta \mathbf{R}(\mathbf{J}_{kins}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}}) \cdot \mathbf{R}(\vec{\mathbf{q}}) = \mathbf{R}(\vec{\mathbf{q}}) \cdot \Delta \mathbf{R}(\mathbf{J}_{rhs}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}})$$

Define

$$\mathbf{J}_{kins} = \begin{bmatrix} \mathbf{J}_{kins}^\alpha \\ \mathbf{J}_{kins}^\varepsilon \end{bmatrix} \quad \vec{\alpha}_{kins} = \mathbf{J}_{kins}^\alpha \Delta\vec{\mathbf{q}} \quad \mathbf{J}_{rhs} = \begin{bmatrix} \mathbf{J}_{rhs}^\alpha \\ \mathbf{J}_{rhs}^\varepsilon \end{bmatrix} \quad \vec{\alpha}_{rhs} = \mathbf{J}_{rhs}^\alpha \Delta\vec{\mathbf{q}}$$

so

$$\Delta \mathbf{R}(\mathbf{J}_{rhs}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}}) = \mathbf{R}(\vec{\mathbf{q}})^{-1} \Delta \mathbf{R}(\mathbf{J}_{kins}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}}) \cdot \mathbf{R}(\vec{\mathbf{q}})$$

$$\mathbf{I} + sk(\vec{\alpha}_{rhs}) = \mathbf{I} + \mathbf{R}^{-1} sk(\vec{\alpha}_{kins}) \mathbf{R}$$

$$sk(\vec{\alpha}_{rhs}) = sk(\mathbf{R}^{-1} \vec{\alpha}_{kins})$$

$$\mathbf{J}_{rhs}^\alpha = \mathbf{R}^{-1} \mathbf{J}_{kins}^\alpha$$

and one can do something similar for the  $\Delta\vec{\mathbf{p}}$  parts.



## LHS versus RHS Jacobians

$$\Delta \mathbf{F}(\mathbf{J}_{kins}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}}) \cdot \mathbf{F}(\vec{\mathbf{q}}) = \mathbf{F}(\vec{\mathbf{q}}) \cdot \Delta \mathbf{F}(\mathbf{J}_{rhs}(\vec{\mathbf{q}})\Delta\vec{\mathbf{q}})$$

Define

$$\mathbf{J}_{kins} = \begin{bmatrix} \mathbf{J}_{kins}^\alpha \\ \mathbf{J}_{kins}^\varepsilon \end{bmatrix} \quad \vec{\alpha}_{kins} = \mathbf{J}_{kins}^\alpha \Delta\vec{\mathbf{q}} \quad \mathbf{J}_{rhs} = \begin{bmatrix} \mathbf{J}_{rhs}^\alpha \\ \mathbf{J}_{rhs}^\varepsilon \end{bmatrix} \quad \vec{\alpha}_{rhs} = \mathbf{J}_{rhs}^\alpha \Delta\vec{\mathbf{q}}$$

so

$$\mathbf{R}(\vec{\mathbf{q}})\vec{\varepsilon}_{rhs} + \vec{\mathbf{p}}(\vec{\mathbf{q}}) = \Delta \mathbf{R}(\vec{\alpha}_{kins})\vec{\mathbf{p}}(\vec{\mathbf{q}}) + \vec{\varepsilon}_{kins}$$

$$\mathbf{R}(\vec{\mathbf{q}})\vec{\varepsilon}_{rhs} = \vec{\mathbf{p}}(\vec{\mathbf{q}}) + \vec{\alpha}_{kins} \times \vec{\mathbf{p}}(\vec{\mathbf{q}}) + \vec{\varepsilon}_{kins} - \vec{\mathbf{p}}(\vec{\mathbf{q}})$$

$$\vec{\varepsilon}_{rhs} = \mathbf{R}(\vec{\mathbf{q}})^{-1} (\vec{\alpha}_{kins} \times \vec{\mathbf{p}}(\vec{\mathbf{q}}) + \vec{\varepsilon}_{kins})$$

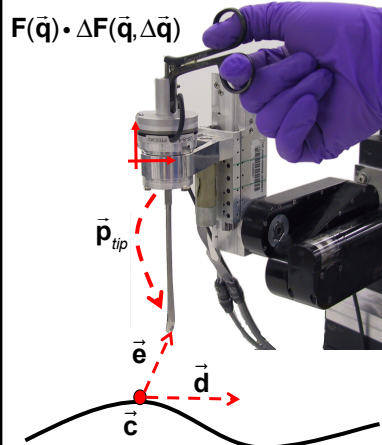
$$= \mathbf{R}(\vec{\mathbf{q}})^{-1} (\vec{\varepsilon}_{kins} - sk(\vec{\mathbf{p}}(\vec{\mathbf{q}}))\vec{\alpha}_{kins})$$

$$= \mathbf{R}(\vec{\mathbf{q}})^{-1} \vec{\varepsilon}_{kins} - \mathbf{R}(\vec{\mathbf{q}})^{-1} sk(\vec{\mathbf{p}}(\vec{\mathbf{q}}))\vec{\alpha}_{kins}$$

$$\mathbf{J}_{rhs} = \begin{bmatrix} \mathbf{R}^{-1} & \mathbf{0} \\ -\mathbf{R}^{-1} sk(\vec{\mathbf{p}}) & \mathbf{R}^{-1} \end{bmatrix} \mathbf{J}_{kins}$$



### Example: Hands-on Guiding to Follow a Path



$$\mathbf{F}(\vec{q}) \cdot \Delta \mathbf{F}(\vec{q}, \Delta \vec{q})$$

$$\Delta \vec{q} = \underset{\Delta \vec{q}}{\operatorname{argmin}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|$$

Such that

$$\vec{e} = (\mathbf{F}(\vec{q}) \Delta \mathbf{F}_{rhs}(\vec{q}, \Delta \vec{q}) \cdot \vec{p}_{tip}) - \vec{c}$$

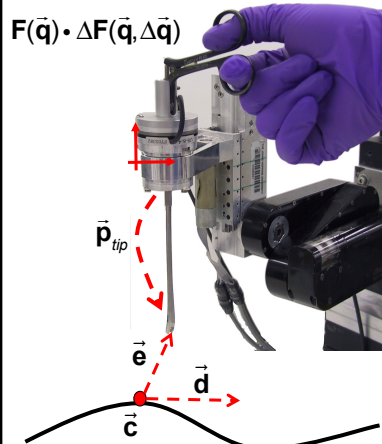
$$\delta \geq \left\| \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right\|$$

**Note:**  $\delta$  is the maximum deviation allowed from the path



56

### Example: Hands-on Guiding to Follow a Path



$$\mathbf{F}(\vec{q}) \cdot \Delta \mathbf{F}(\vec{q}, \Delta \vec{q})$$

$$\Delta \vec{q} = \underset{\Delta \vec{q}}{\operatorname{argmin}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|^2$$

Such that

$$\vec{e} = (\mathbf{F}(\vec{q}) (\vec{\alpha} \times \vec{p}_{tip} + \vec{e} + \vec{p}_{tip})) - \vec{c}$$

$$\begin{bmatrix} \vec{\alpha} \\ \vec{e} \\ \vec{e} \end{bmatrix} = \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q}$$

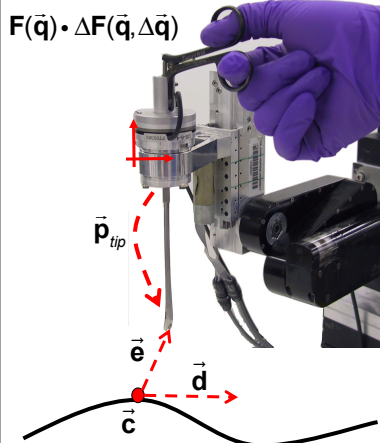
$$\delta \geq \left\| \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right\|$$

**Note:**  $\delta$  is the maximum deviation allowed from the path



57

### Example: Hands-on Guiding to Follow a Path



$$F(\vec{q}) \cdot \Delta F(\vec{q}, \Delta \vec{q})$$

$$\Delta \vec{q} = \underset{\Delta \vec{q}}{\operatorname{argmin}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|^2$$

Such that

$$\vec{e} = \mathbf{R}(\vec{q}) (\vec{\alpha} \times \vec{p}_{tip} + \vec{\varepsilon} + \vec{p}_{tip}) + \vec{p}(\vec{q}) - \vec{c}$$

$$\begin{bmatrix} \vec{\alpha} \\ \omega \end{bmatrix} = \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q}$$

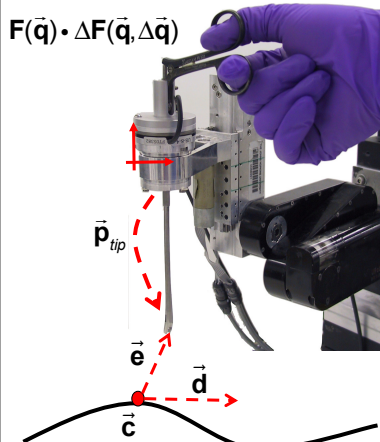
$$\delta \geq \left\| \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right\|$$

**Note:**  $\delta$  is the maximum deviation allowed from the path



58

### Example: Hands-on Guiding to Follow a Path



$$F(\vec{q}) \cdot \Delta F(\vec{q}, \Delta \vec{q})$$

$$\Delta \vec{q} = \underset{\Delta \vec{q}}{\operatorname{argmin}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q} \right\|^2$$

Such that

$$\vec{e} = \vec{p}(\vec{q}) + \mathbf{R}(\vec{q}) \vec{p}_{tip} + \mathbf{R}(\vec{q}) \vec{\varepsilon} - \mathbf{R}(\vec{q}) sk(\vec{p}_{tip}) \vec{\alpha}$$

$$\begin{bmatrix} \vec{\alpha} \\ \omega \end{bmatrix} = \mathbf{J}_{rhs}(\vec{q}) \Delta \vec{q}$$

$$\delta \geq \left\| \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right\|$$

Approximate this by

$$-\delta \leq \left( \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right) \cdot \left( \operatorname{Rot}(\vec{d}, k\pi / N) \vec{g} \right) \leq \delta$$

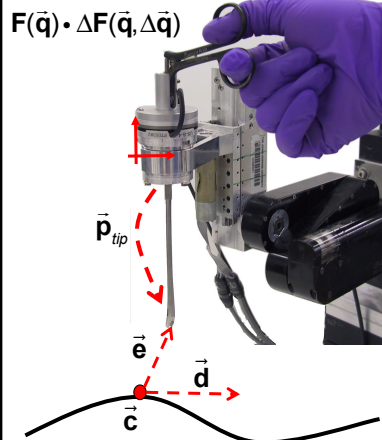
for  $0 \leq k \leq N-1$

and some  $\vec{g}$  perpendicular to  $\vec{d}$



59

### Example: Hands-on Guiding to Follow a Path



$$F(\vec{q}) \cdot \Delta F(\vec{q}, \Delta \vec{q})$$

$$\Delta \vec{q} = \underset{\Delta \vec{q}}{\operatorname{argmin}} \left\| \mathbf{K}_v \vec{f} - \mathbf{J}_{rns}(\vec{q}) \Delta \vec{q} \right\|^2 + \eta \left\| \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right\|^2$$

Such that

$$\vec{e} = \vec{p}(\vec{q}) + \mathbf{R}(\vec{q}) \vec{p}_{tip} + \mathbf{R}(\vec{q}) \vec{\varepsilon} - \mathbf{R}(\vec{q}) s k (\vec{p}_{tip}) \vec{\alpha}$$

$$\begin{bmatrix} \vec{\alpha} \\ \omega \end{bmatrix} = \mathbf{J}_{rns}(\vec{q}) \Delta \vec{q}$$

$$\delta \geq \left\| \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right\|$$

Approximate this by

$$-\delta \leq \left( \vec{e} - (\vec{d} \cdot \vec{e}) \vec{d} \right) \cdot \left( \operatorname{Rot}(\vec{d}, k\pi / N) \vec{g} \right) \leq \delta$$

for  $0 \leq k \leq N-1$

and some  $\vec{g}$  perpendicular to  $\vec{d}$

