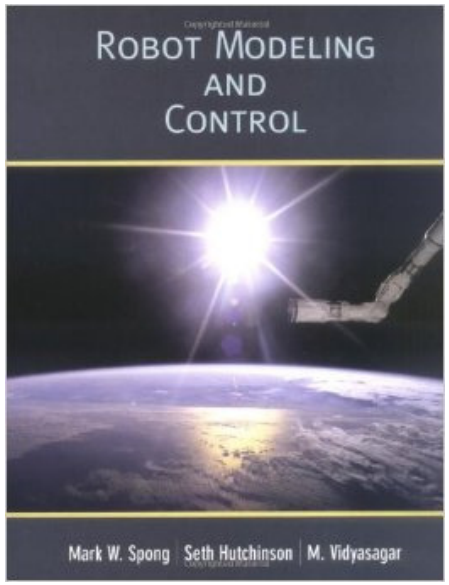
Teaching Forward and Inverse Kinematics of Robotic Manipulators via MATLAB

Philip Dames, Denise Wong, Katherine J. Kuchenbecker

ICRA
5th June 2014

MEAM 520 Overview

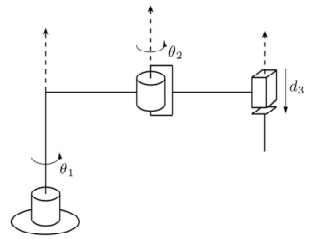
- Graduate level course
- Diverse student body
  - Mechanical Engineering, Electrical and Systems Engineering, Computer and Information Science, Embedded Systems, and Robotics
  - Enrollment of around 100 students (~40 Undergraduate, ~60 Masters, ~3 Ph.D.)
- Textbook: “Robotic Modeling and Control” by Spong, Hutchinson, and Vidyasagar



MEAM 520 Topics

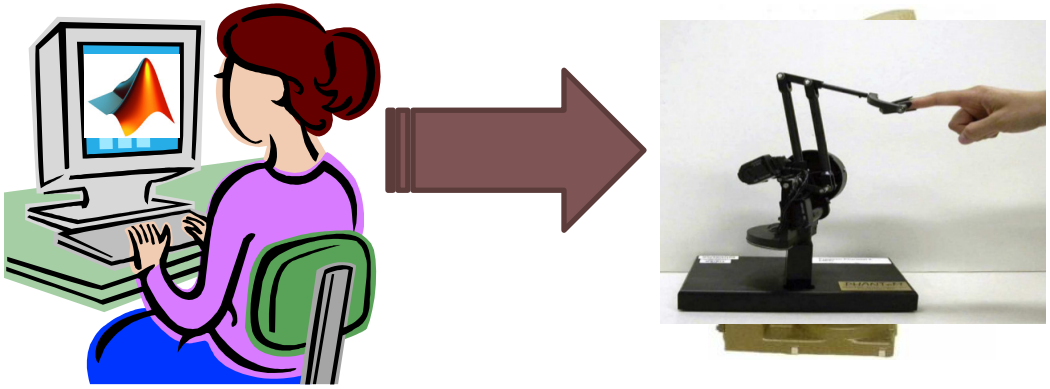
- Focus is on robotic manipulators
  - Rigid motions and homogeneous transformations
  - Forward and inverse kinematics
  - Dynamics and control
- MATLAB use in the course
  - Regular programming assignments, e.g.,
    - Graphing 2D workspace of a robot arm
    - Animating a flying box using recorded magnetic tracker data
    - Implementing Denavit-Hartenberg parameters
    - Graphing manipulability ellipsoids of a 6-DoF manipulator
  - Three projects

Selective Compliant Articulated Robot for Assembly



MEAM 520 Projects

Goal: Give students hands-on experience with real robotic hardware (under the constraint that we have 1 robot for 100 students)



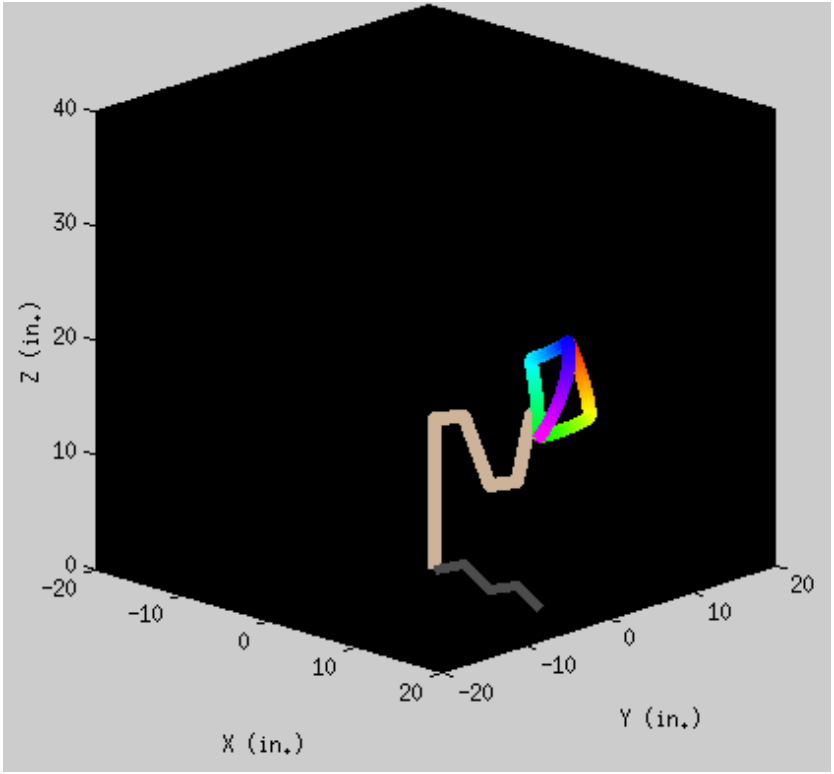
1. PUMA Dance
2. PUMA Light Painting
3. Phantom Premium Virtual Haptic Environment

PUMA 260 Manipulator

- 6-DoF serial manipulator
  - RRR with offsets
  - Spherical wrist
- Connected to a Linux computer
  - Servo2Go ISA card
  - Optical encoders and DC brushed motors
- Controlled from MATLAB
  - Custom mex API in C++
  - Tightly integrated with simluation environment

PUMA 260 Simulator

- Uses Peter Corke's Robotics Toolbox
- Kinematic simulator
  - Specify joint angles as a function of time
  - Joint angles/angular velocities limited
  - Allows for real-time execution of trajectories
- Allows students to test code *before* using the real robot
  - `pumaStart('Hardware','on')`



Forward and Inverse Kinematics

Joint coordinates

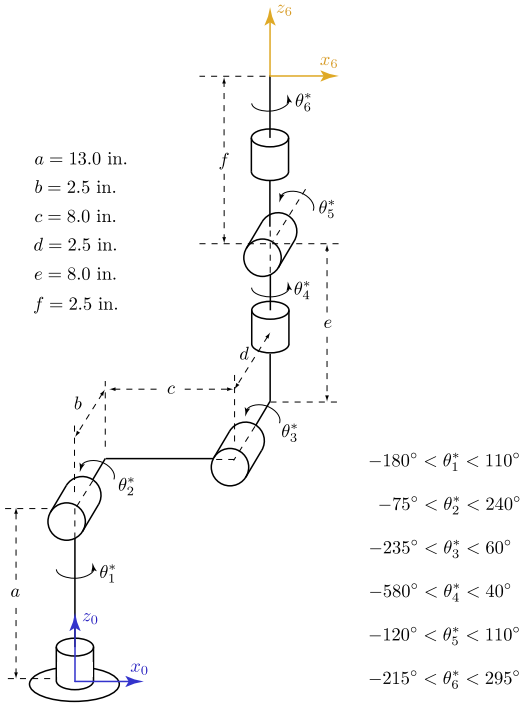
PUMA 260:
$\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$

Forward Kinematics

Inverse Kinematics

Pose of end-effector

PUMA 260:
$x, y, z, \varphi, \theta, \psi$



PUMA Dance

- Teams of 2 students solve the forward kinematics of PUMA
  - Specify waypoints in joint space with a time stamp
  - Implemented multiple interpolation methods
- Dances were played in time with music and lights

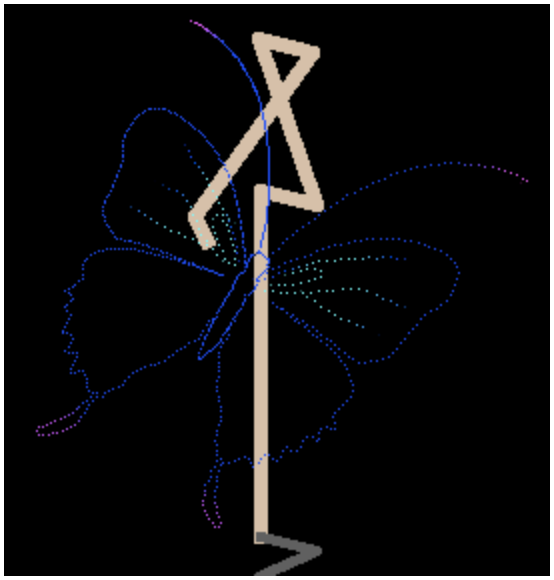
PUMA Dance Video

PUMA Light Painting

- Teams of 3 students solve the inverse kinematics of PUMA
  - All 8 solutions
- PUMA arm fitted with tri-color LED
  - Students control the 3D position, orientation, and RGB color of the end-effector
- Take long exposure image using webcam or DSLR camera

[Shomin & Fiene, ASME IDETC, 2011]

Simulation to Hardware



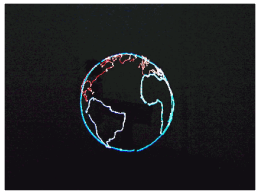
Simulation

Hardware

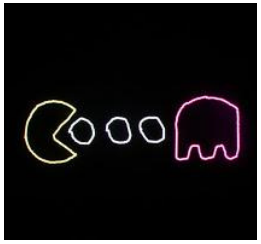
Light Painting Results



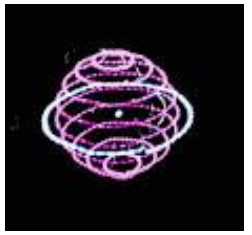






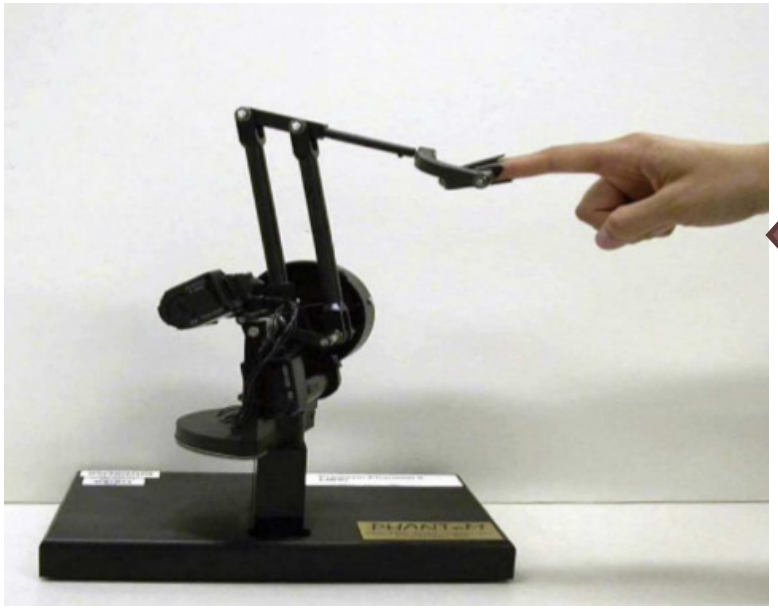





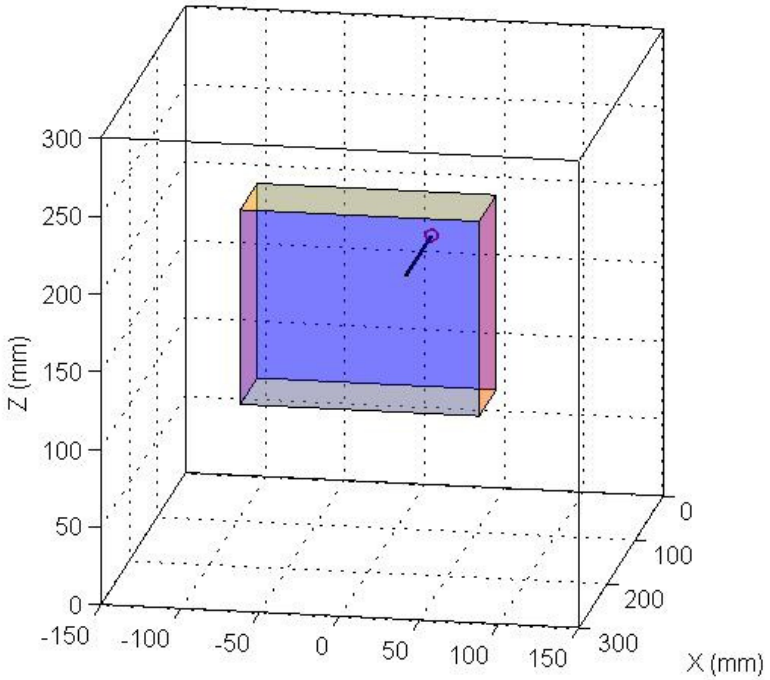


Virtual Haptic Environment

Phantom Premium 1.0



Legacy SensAble ISA card



Conclusion

- MATLAB provides an easy-to-use environment for students to explore robotics
- Simple transition between simulation and hardware
- Student testimonials:
  - "The mapping between theory and application through the problems and projects is very **helpful in learning**."
  - "Simulations obviously take a lot of time but it is **great to have videos and graphics to go along with the concepts**. Makes everything way more clear."
  - "The assignments are a lot of **fun** especially the coding assignments. I **seriously look forward** to working on them."

Questions?

Katherine J. Kuchenbecker
Associate Professor
kuchenbe@seas.upenn.edu