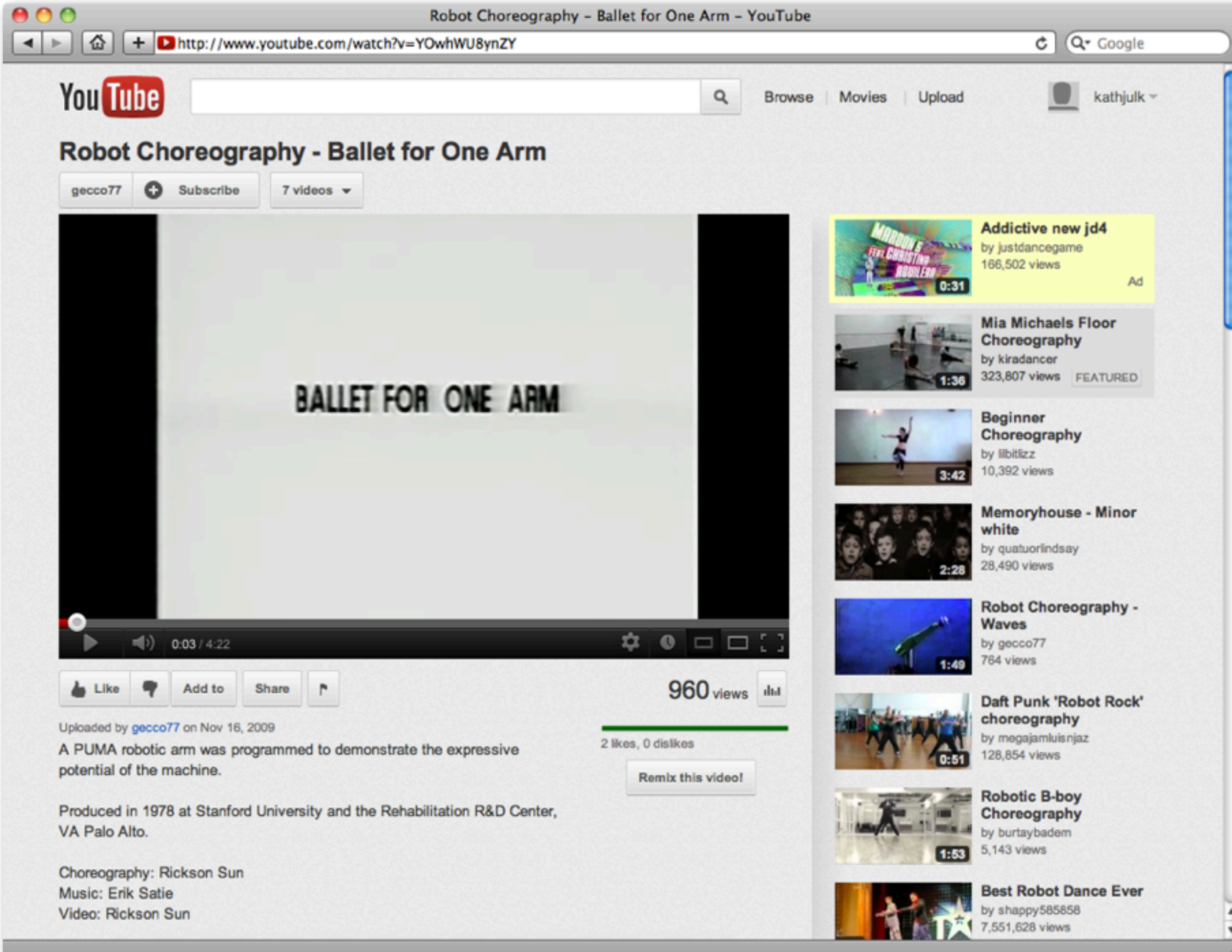
MEAM 520

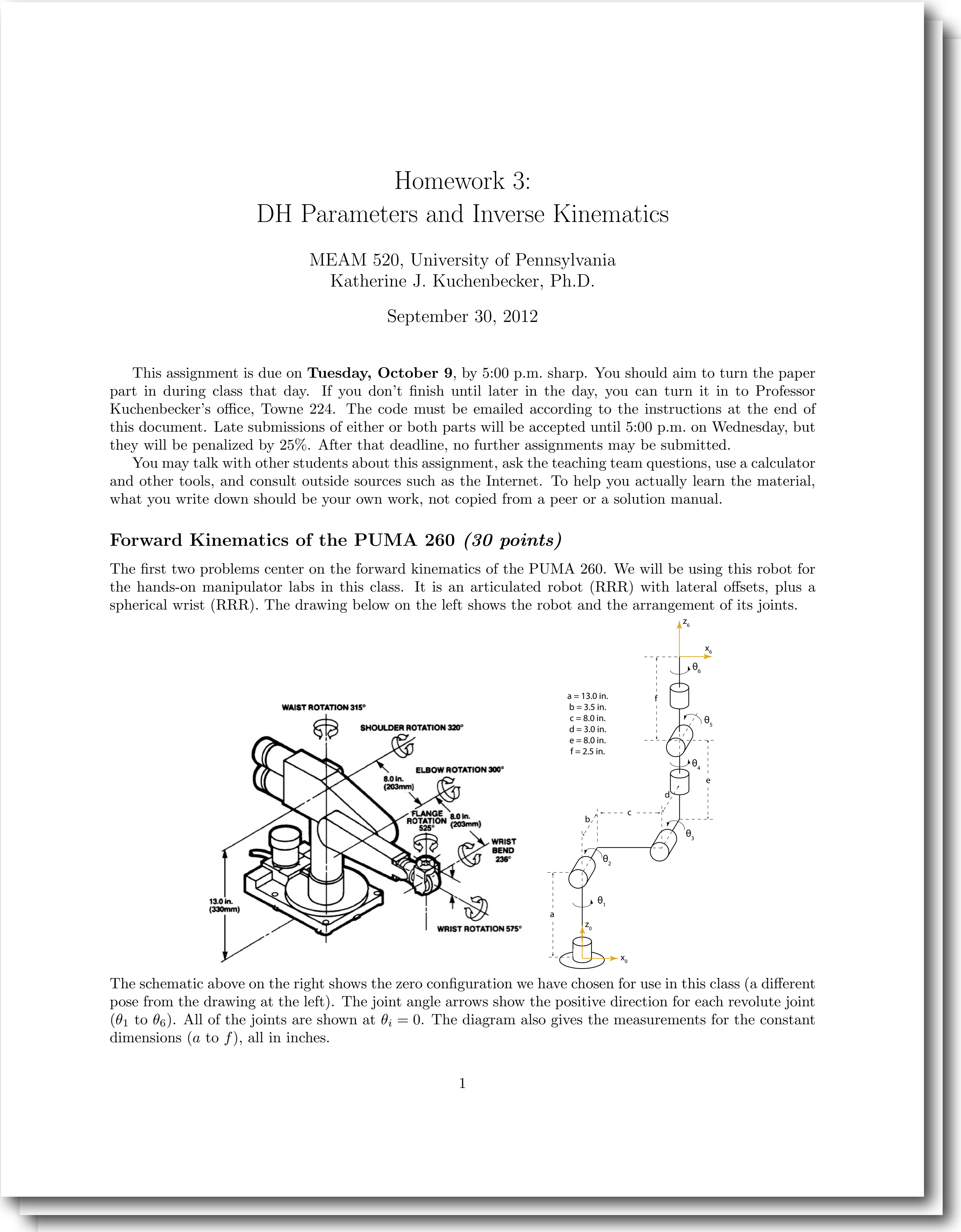
More Manipulator Kinematics

Katherine J. Kuchenbecker, Ph.D.
General Robotics, Automation, Sensing, and Perception Lab (GRASP)
MEAM Department, SEAS, University of Pennsylvania

Lecture 10: October 9, 2012


Robot Choreography - Ballet for One Arm - YouTube
http://www.youtube.com/watch?v=YOwhWU8ynZY
Google
You Tube
Browse
Movies
Upload
kathjulk
Robot Choreography - Ballet for One Arm
gecco77
Subscribe
7 videos
BALLET FOR ONE ARM
0:03 / 4:22
Like
Add to
Share
960 views
Uploaded by gecco77 on Nov 16, 2009
A PUMA robotic arm was programmed to demonstrate the expressive potential of the machine.
Produced in 1978 at Stanford University and the Rehabilitation R&D Center, VA Palo Alto.
Choreography: Rickson Sun
Music: Erik Satie
Video: Rickson Sun
2 likes, 0 dislikes
Remix this video!
Addictive new jd4
by justdancegame
166,502 views
Ad
0:31
Mia Michaels Floor Choreography
by kiradancer
323,807 views
FEATURED
1:36
Beginner Choreography
by lilbitlizz
10,392 views
3:42
Memoryhouse - Minor white
by quatuorlindsay
28,490 views
2:28
Robot Choreography - Waves
by gecco77
764 views
1:49
Daft Punk 'Robot Rock' choreography
by megajamluisnjaz
128,854 views
0:51
Robotic B-boy Choreography
by burtaybadem
5,143 views
1:53
Best Robot Dance Ever
by shappy585858
7,551,628 views

Homework 3 due today by 5:00 p.m.



Homework 3:
DH Parameters and Inverse Kinematics

MEAM 520, University of Pennsylvania
Katherine J. Kuchenbecker, Ph.D.

September 30, 2012

This assignment is due on **Tuesday, October 9**, by 5:00 p.m. sharp. You should aim to turn the paper part in during class that day. If you don't finish until later in the day, you can turn it in to Professor Kuchenbecker's office, Towne 224. The code must be emailed according to the instructions at the end of this document. Late submissions of either or both parts will be accepted until 5:00 p.m. on Wednesday, but they will be penalized by 25%. After that deadline, no further assignments may be submitted.

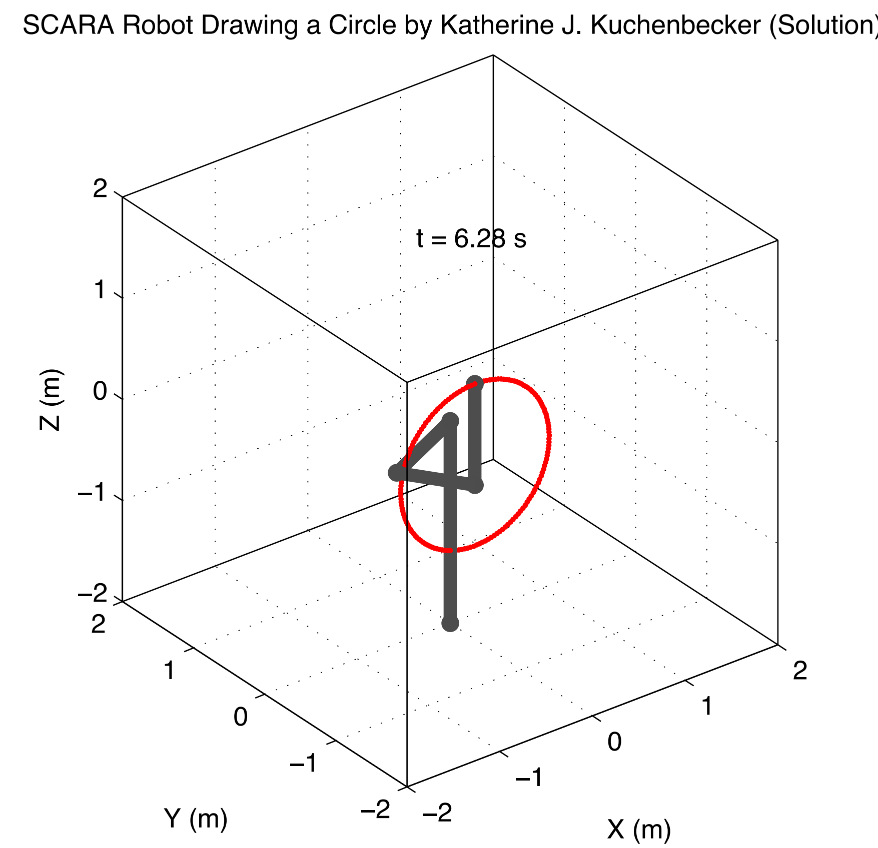
You may talk with other students about this assignment, ask the teaching team questions, use a calculator and other tools, and consult outside sources such as the Internet. To help you actually learn the material, what you write down should be your own work, not copied from a peer or a solution manual.

Forward Kinematics of the PUMA 260 ***(30 points)***

The first two problems center on the forward kinematics of the PUMA 260. We will be using this robot for the hands-on manipulator labs in this class. It is an articulated robot (RRR) with lateral offsets, plus a spherical wrist (RRR). The drawing below on the left shows the robot and the arrangement of its joints.

The schematic above on the right shows the zero configuration we have chosen for use in this class (a different pose from the drawing at the left). The joint angle arrows show the positive direction for each revolute joint (θ_1 to θ_6). All of the joints are shown at $\theta_i = 0$. The diagram also gives the measurements for the constant dimensions (a to f), all in inches.

1



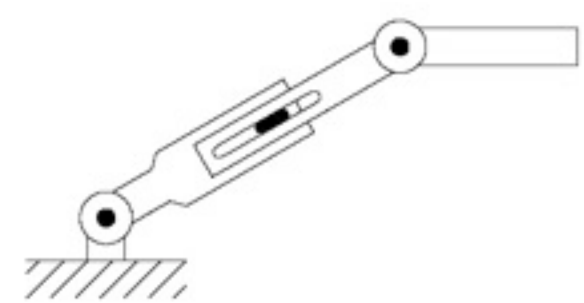


Figure 3.33: Three-link planar robot with prismatic joint.

A few more good questions about DH parameters...

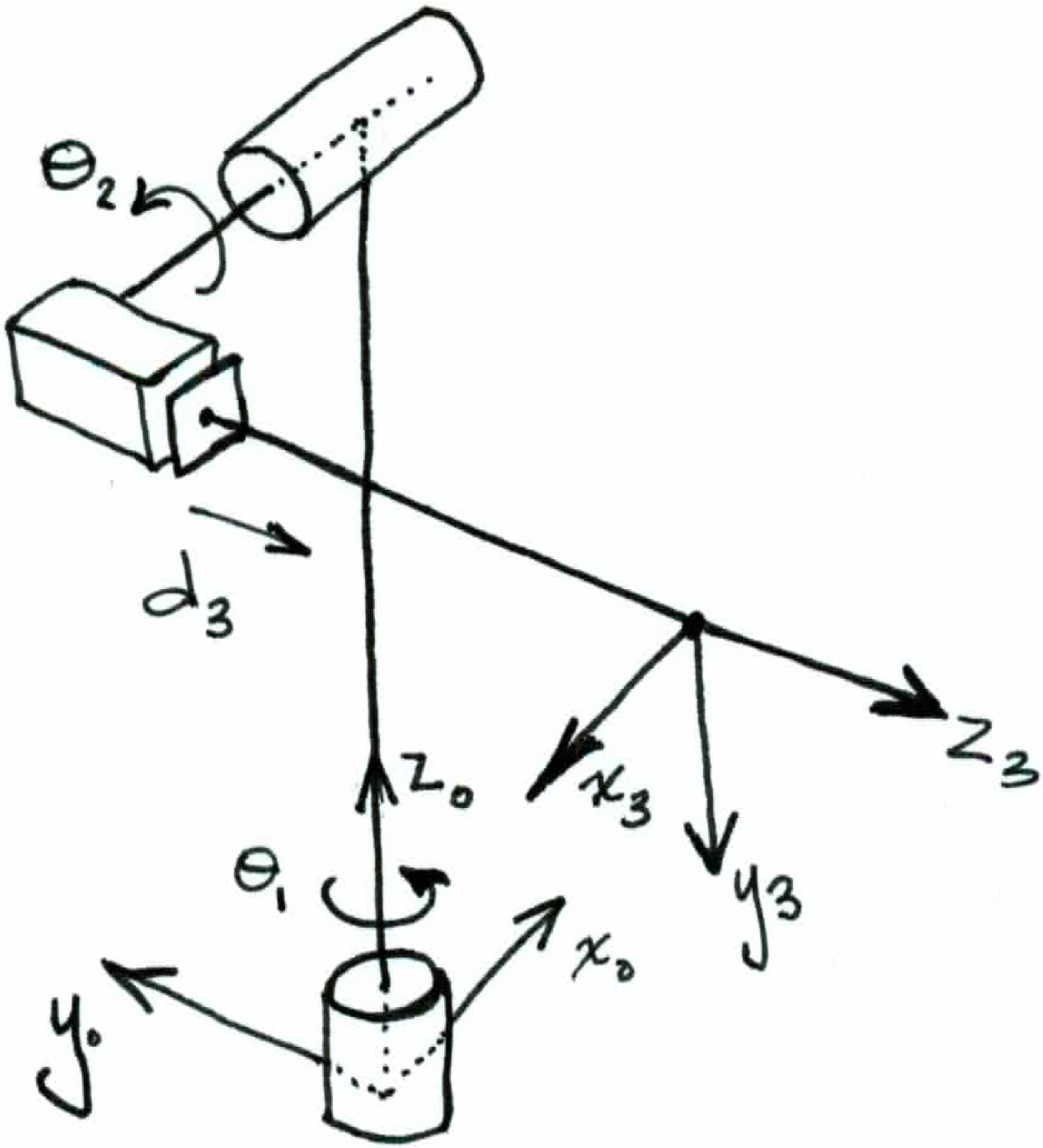
The Denavit-Hartenberg Convention

(DH1) The axis x_i is perpendicular to the axis z_{i-1}

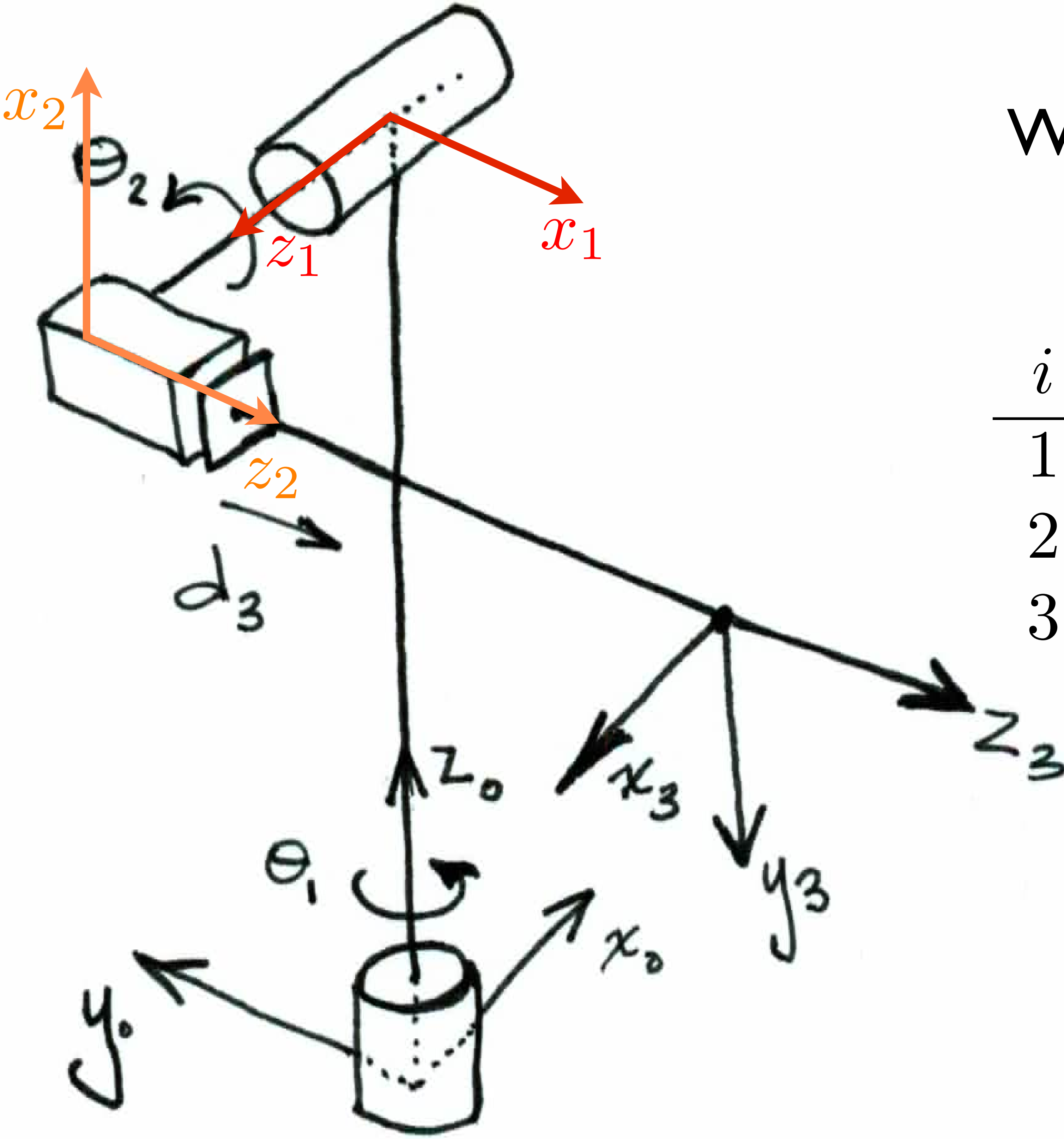
(DH2) The axis x_i intersects the axis z_{i-1}

a_i **Link Length**	the perpendicular distance between z_i and z_{i-1}, measured along x_i
α_i **Link Twist**	the angle between z_{i-1} and z_i, measured in the plane normal to x_i (right-hand rule around x_i)
d_i **Link Offset**	the distance along z_{i-1} from o_{i-1} to the intersection with x_i
θ_i **Joint Angle**	the angle between x_{i-1} and x_i, measured in the plane normal to z_{i-1} (right-hand rule around z_{i-1})

Denavit & Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," ASME Journal of Applied Mechanics, June 1955

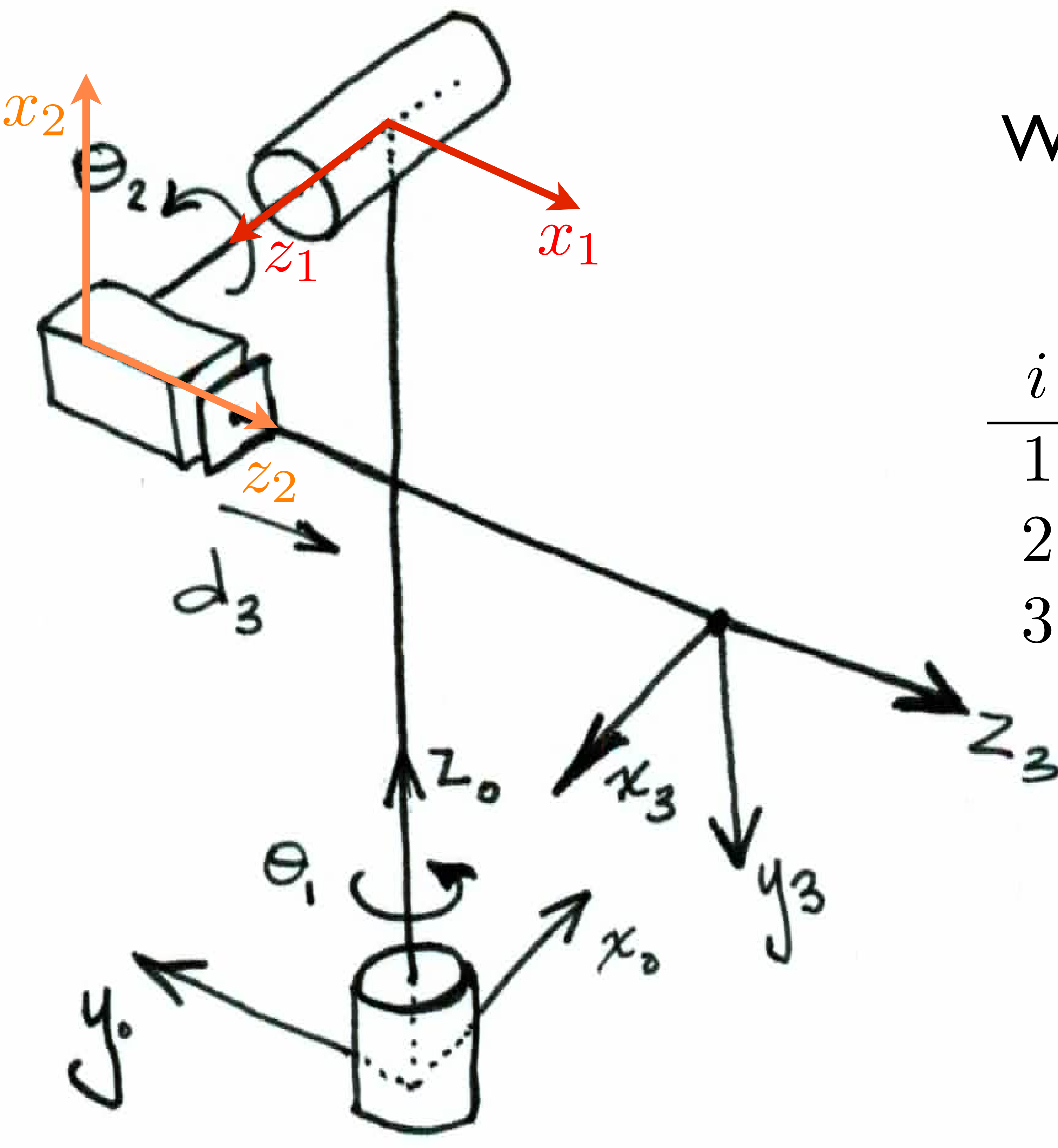


- What type of robot is this?
- Work with a partner to determine:
  - Where should all the z-axes be?
  - Where should all the frame origins be?
  - Where should all the x-axes be?



What are the DH parameters?

i	a	α	d	θ
1				
2				
3				



What are the DH parameters?

i	a	α	d	θ
1	0	$+90^\circ$	d_1	$\theta_1^* - 90^\circ$
2	0	$+90^\circ$	d_2	$\theta_2^* + 90^\circ$
3	0	0°	d_3^*	$+90^\circ$

Questions ?

Project 1 : PUMA Light Painting

Team Formation

You will work in a team of 3
(33 teams of 3, only one team of 4)

Each team must have at least one undergraduate
and at least one graduate student.
Submatriculants count as undergraduates.

(52 undergrads, 51 grads)

Try to have two MEAM students and
one non-MEAM student on each team
(71 MEAM, 33 non-MEAM)

Pick your team by midnight on Tuesday, October 9,
or ask to be assigned. Email denwong@seas.upenn.edu

Time to meet teammates at the end of class

MEAM.Design : MEAM520-12C-P01-Teams

http://medesign.seas.upenn.edu/index.php/Courses/MEAM520-12C-P01-Teams

MEAM.Design : MEAM520-12C-P01-Teams

MEAM.Design - MEAM 520 - **PUMA Light Painting: Team Selection**

Project 1 is PUMA Light Painting. The first part of this project is selecting your team. You must pick your team for PUMA Light Painting by midnight on **Tuesday, October 9**. Note that this is earlier than the deadline that was originally announced in class, to leave time for us to assign teams for individuals who do not pick a team. The first deliverable your team will need to complete together will likely be due on Tuesday, October 16.

Below are the team selection guidelines:

- Each team will include exactly three students.
- Each team must have at least one undergraduate and one graduate student. Submatriculants count as undergraduates.
- Aim to have two MEAM students and one non-MEAM student on each team. This is recommended, not required.

Looking for teammates? Try using the Project Teams & Study Groups tool on Piazza.

When you have chosen your team, please send a list of all three names to Denise Wong (MEAM 520 TA).

We will be happy to assign you to a team if you can't find two partners. Just send an email to Denise to explain the situation. Include your name, email address, and degree program to assist her in matching you with other students. Teams will be assigned on a rolling basis to give you as much time as possible to work together on the project.

If Denise does not hear from you by midnight on Tuesday, she will assign you to a team on Wednesday. The list below shows all confirmed teams.

Team - Members
01 - Jason Gui, Yu Luo, and Xiaoting Zheng
02 - Cristina Sorice, Elizabeth Fedalei, Thomas Koletschka
03 - Bo Yuan, Noam Eisen, Dalton Banks
04 - Chaoyi Huang, Jiali Sheng, Chao Liu
05 - Nick McGill, Niko Vladimirov, Jonathan Balloch
06 - Chao Qu, Michael Gosselin, Yunkai Cui
07 - Wei-Ting Lo, Robyn Schwartz, Alexandre Miranda Anon
08 - Rui Zhang, Zhongqi Yue, Alex Sher
09 - Vivian Chu, Jennifer Hui, Eza Koch
10 - Jia Lue Huang, Mike Lo, Wenbin Zhao
11 - Tianyu Dong, Yanwei Du, Eleonoor Bosch

MEAM.Design : MEAM520-12C-P01-IK

http://medesign.seas.upenn.edu/index.php/Courses/MEAM520-12C-P01-IK

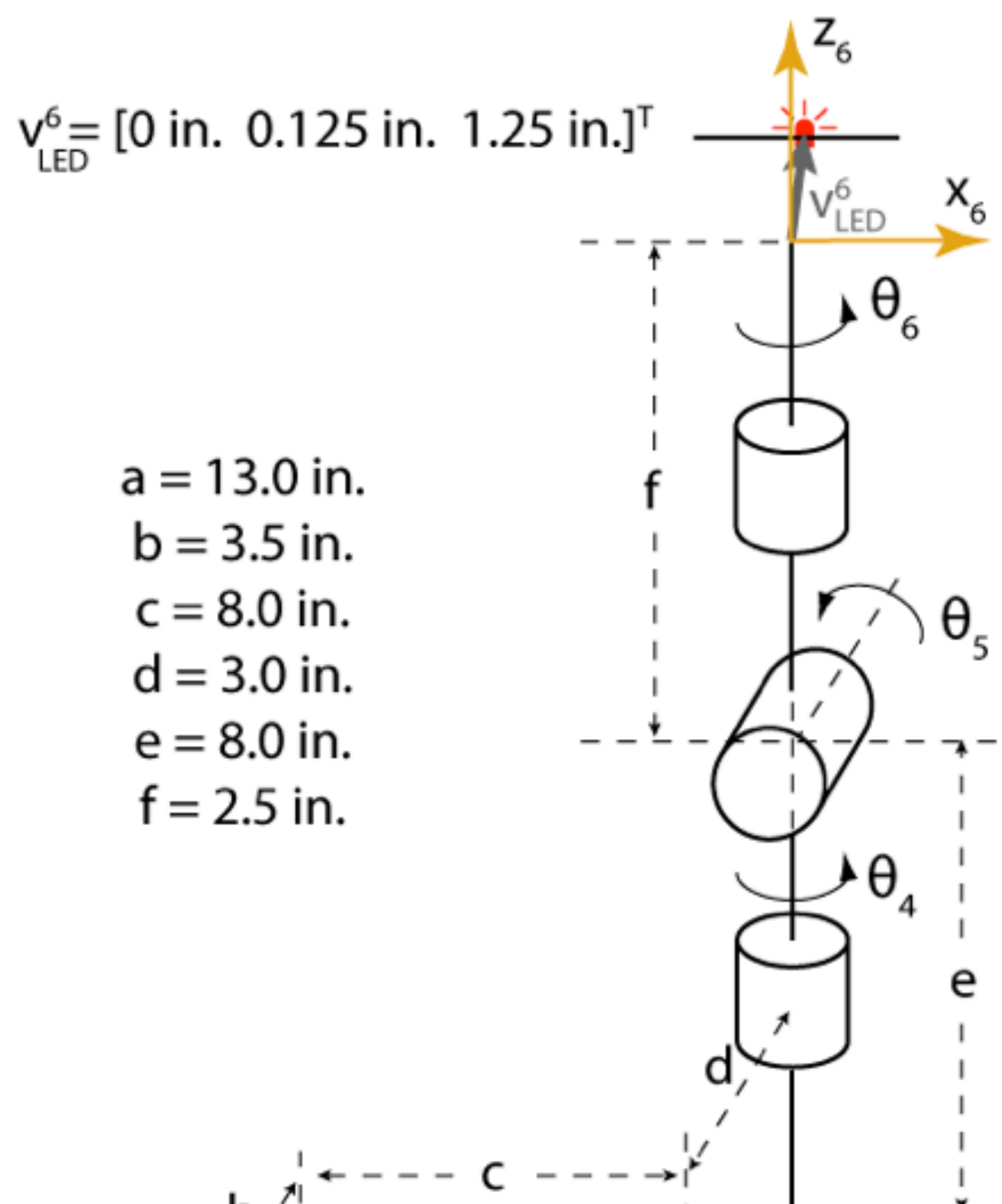
MEAM.Design : MEAM520-12C-P01-IK

View Logout
Edit Upload

MEAM.Design - MEAM 520 - **PUMA Light Painting: IK**

Now that you have your team, it's time to get to work on project 1. This assignment is due by **5:00 p.m. on Tuesday, October 16**. Your team must submit this assignment and get it to work correctly before you will be allowed to do the next part of the project. Submissions after the deadline will be penalized, but not as harshly as for individual homework assignments.

The final goal of this project is to create a beautiful light painting by taking a long-exposure video and photo of the PUMA moving an LED around in the air. As an intermediate step toward that goal, you and your teammates must solve the inverse kinematics of the robot, so that you can later safely move its end-effector wherever is needed for your artwork.



LED Location

The center of the LED is located at approximately [0" 0.125" 1.25"]' in frame 6. The position and orientation of frames 0 and 6 are specified in the image at right, as are the positive directions for all the joints. These conventions match what was specified in Homework 3.

Joint Angle Limits

θ1 (waist) range = 290 deg , lowerlimit = -180 deg , upperlimit = 110 deg
θ2 (shoulder) range = 315 deg , lowerlimit = -75 deg , upperlimit = 240 deg
θ3 (elbow) range = 295 deg , lowerlimit = -235 deg , upperlimit = 60 deg
θ4 (wrist) range = 620 deg , lowerlimit = -580 deg , upperlimit = 40 deg
θ5 (bend) range = 230 deg , lowerlimit = -120 deg , upperlimit = 110 deg
θ6 (flange) range = 510 deg , lowerlimit = -215 deg , upperlimit = 295 deg

PUMA 260 Simulator

At some point soon, we will publish a full forward kinematics simulator for the PUMA 260 robot. It will have the same software interface as our real PUMA robot. You may find it useful to use the simulator to verify your forward kinematics and inverse kinematics solutions. More details will be

Go to "http://medesign.seas.upenn.edu/index.php/Courses.MEAM520-12C-P01-IK"

MEAM 410/510
MEAM 520
IPD 501
SAAST

GUIDES
Materials
Laser Cutting
3D Printing
Machining
ProtoTRAK
PUMA 260
PHANToM
BeagleBoard
MAEVARM
Phidget
Tap Chart

SOFTWARE
SolidWorks
Matlab
NX
Nastran
Fluent, Gambit
SolidCAM
Eagle

OTHER
Vendor List
Design Links

Editing Tips

a long-exposure video and photo of the PUMA moving an LED around in the air. As an intermediate step toward that goal, you and your teammates must solve the inverse kinematics of the robot, so that you can later safely move its end-effector wherever is needed for your artwork.

LED Location

The center of the LED is located at approximately [0" 0.125" 1.25"]' in frame 6. The position and orientation of frames 0 and 6 are specified in the image at right, as are the positive directions for all the joints. These conventions match what was specified in Homework 3.

Joint Angle Limits

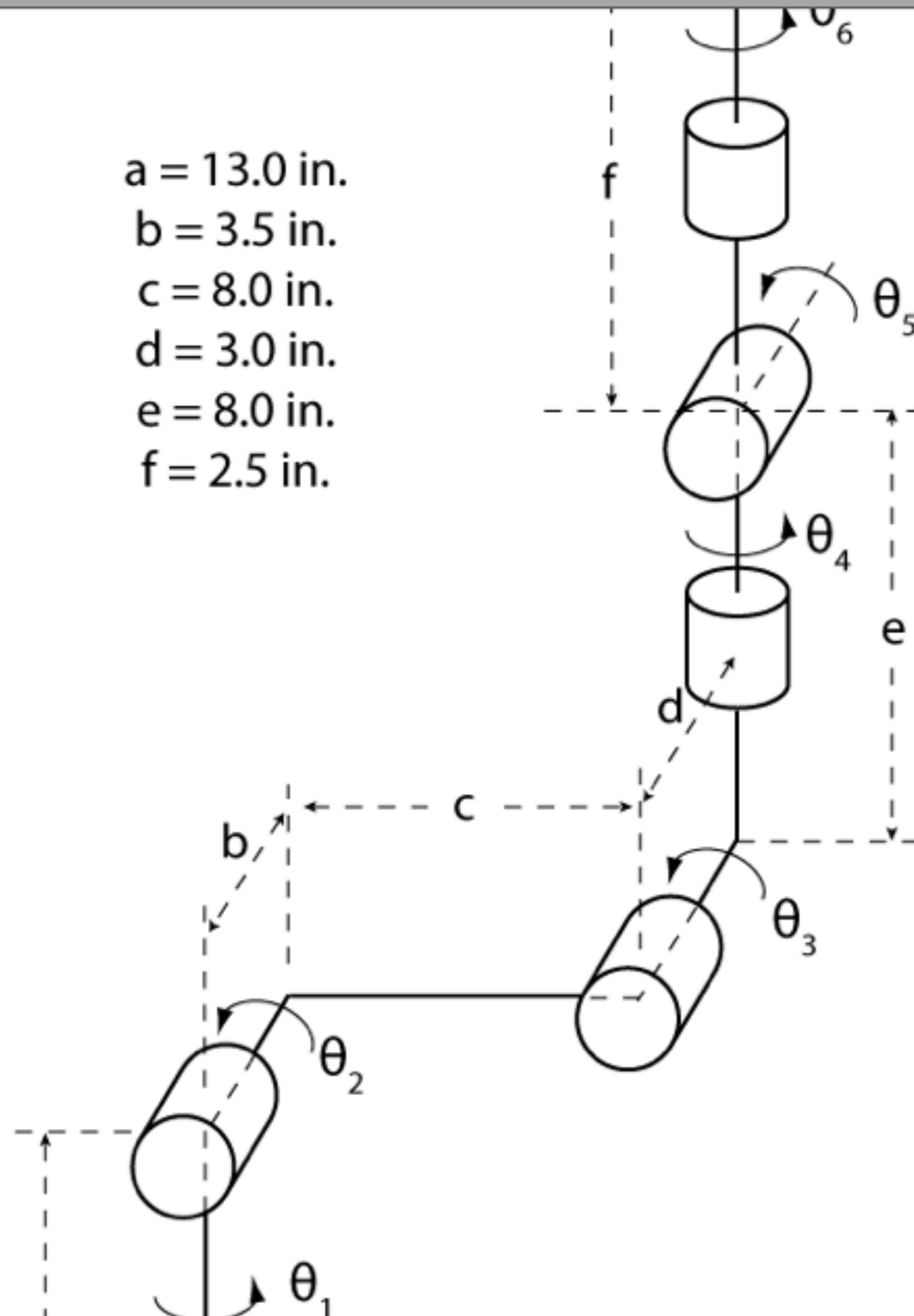
θ1 (waist) range = 290 deg , lowerlimit = -180 deg , upperlimit = 110 deg
θ2 (shoulder) range = 315 deg , lowerlimit = -75 deg , upperlimit = 240 deg
θ3 (elbow) range = 295 deg , lowerlimit = -235 deg , upperlimit = 60 deg
θ4 (wrist) range = 620 deg , lowerlimit = -580 deg , upperlimit = 40 deg
θ5 (bend) range = 230 deg , lowerlimit = -120 deg , upperlimit = 110 deg
θ6 (flange) range = 510 deg , lowerlimit = -215 deg , upperlimit = 295 deg

PUMA 260 Simulator

At some point soon, we will publish a full forward kinematics simulator for the PUMA 260 robot. It will have the same software interface as our real PUMA robot. You may find it useful to use the simulator to verify your forward kinematics and inverse kinematics solutions. More details will be posted about the simulator after everyone has turned in Homework 3.

Task 1: PUMA Forward Kinematics Function

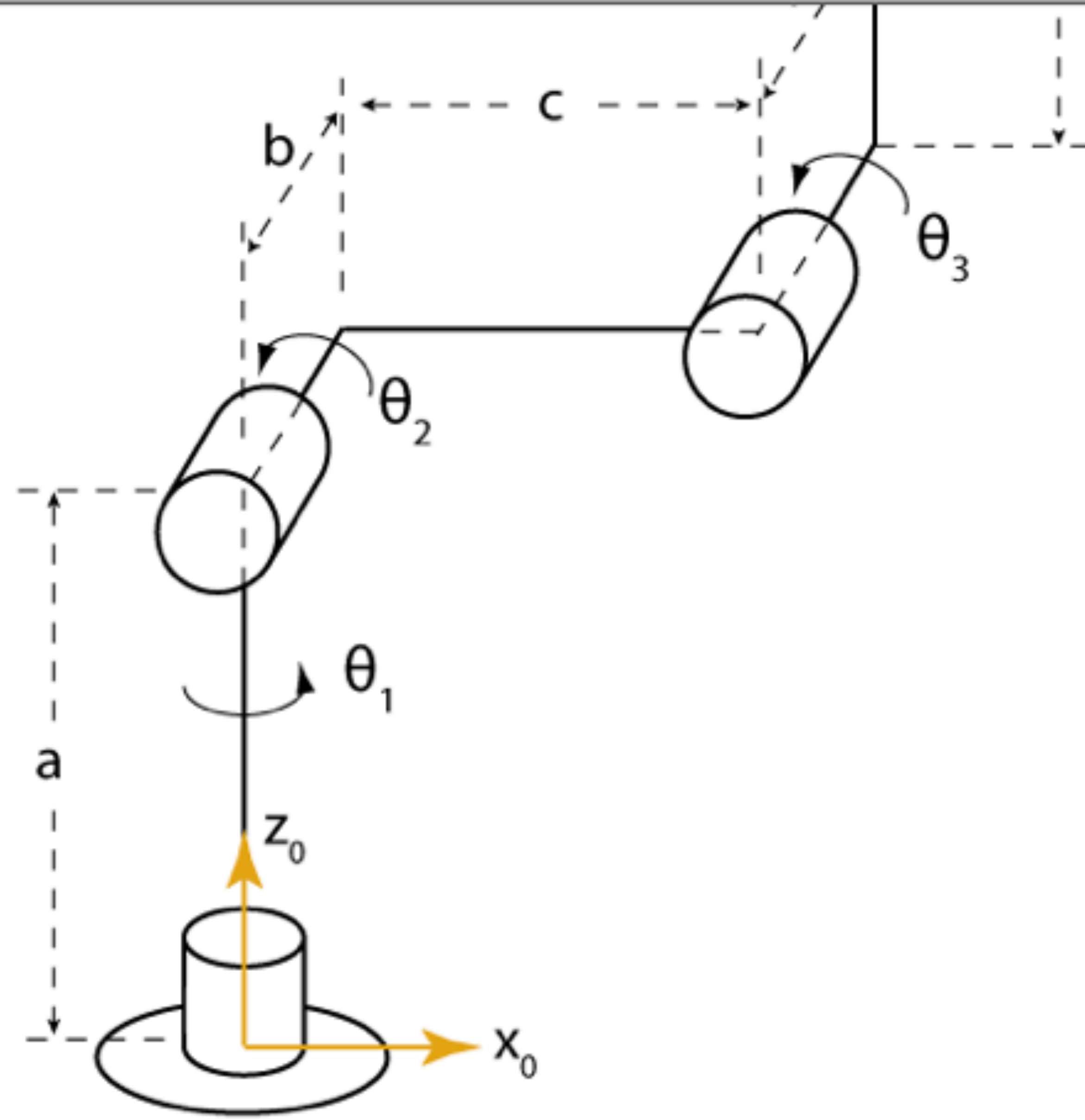
Talk about the forward kinematics of the PUMA 260 with your teammates. Compare the DH parameters and animations that you each created for Homework 3, and figure out any discrepancies between your solutions. Then work together to complete the `puma_fk_team00.m` starter function provided in the zip file of starter code. Change the name of the file and the name of the function on the file's first line to include your team's two-digit number, as defined on the teams page. This function takes the robot's six joint angles in radians and returns the four-by-four homogenous transformation representing the pose of frame 6 in frame 0. If needed, you are welcome to create other functions in separate files (following the same team naming convention) or as subfunctions within



the PUMA 260 robot. It will have the same software interface as our real PUMA robot. You may find it useful to use the simulator to verify your forward kinematics and inverse kinematics solutions. More details will be posted about the simulator after everyone has turned in Homework 3.

Task 1: PUMA Forward Kinematics Function

Talk about the forward kinematics of the PUMA 260 with your teammates. Compare the DH parameters and animations that you each created for Homework 3, and figure out any discrepancies between your solutions. Then work together to complete the `puma_fk_team00.m` starter function provided in the zip file of starter code. Change the name of the file and the name of the function on the file's first line to include your team's two-digit number, as defined on the teams page. This function takes the robot's six joint angles in radians and returns the four-by-four homogenous transformation representing the pose of frame 6 in frame 0. If needed, you are welcome to create other functions in separate files (following the same team naming convention) or as subfunctions within your FK function. You may find this function useful in the following steps.



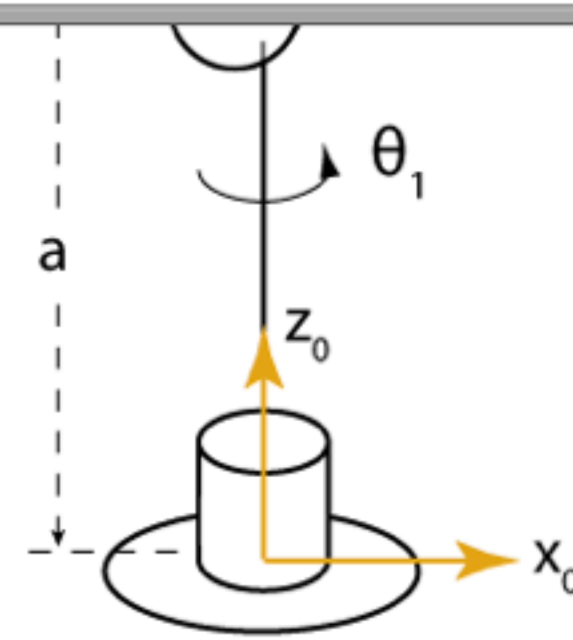
Task 2: PUMA Inverse Kinematics Function

Derive a full inverse kinematics solution for the PUMA 260, remembering the concept of kinematic decoupling. Here the goal is to find the values of the joint angles (theta1 to theta6, in radians) that put the robot's LED end-effector at a certain position and orientation that are specified relative to frame 0. Note that there are generally eight possible solutions to the PUMA's IK. To start, you should focus on finding one valid solution. Once you get that working, add in code to handle the constraint that the selected solution should be close to the robot's current configuration; the robot's current joint angles can be optionally passed into the function as th1now to th6now. If no valid solution exists, your function should return NaN (not a number) for all six joint angles. Complete the `puma_ik_team00.m` starter function provided in the zip file below, again entering your team number where needed.

Task 3: Script to Test PUMA Inverse Kinematics

To help you assess the quality of your inverse kinematics function, you must also create a script that does thorough IK testing. At a minimum, it should compute two scores, both from 0 (worst performance) to 100 (perfect performance). One score is for finding inverse kinematics solutions without worrying about the robot's current configuration, and the second score is for finding inverse kinematics solutions close to the current joint angles of the robot. It is up to you to figure out what should be in this file to make it a reasonable test of the code. Make sure to test end-effector poses that are not reachable due to joint limits or other constraints. Complete the `test_puma_ik_team00.m` starter script provided in the zip file below, again entering your team number where needed.

takes the robot's six joint angles in radians and returns the four-by-four homogenous transformation representing the pose of frame 6 in frame 0. If needed, you are welcome to create other functions in separate files (following the same team naming convention) or as subfunctions within your FK function. You may find this function useful in the following steps.



Task 2: PUMA Inverse Kinematics Function

Derive a full inverse kinematics solution for the PUMA 260, remembering the concept of kinematic decoupling. Here the goal is to find the values of the joint angles (theta1 to theta6, in radians) that put the robot's LED end-effector at a certain position and orientation that are specified relative to frame 0. Note that there are generally eight possible solutions to the PUMA's IK. To start, you should focus on finding one valid solution. Once you get that working, add in code to handle the constraint that the selected solution should be close to the robot's current configuration; the robot's current joint angles can be optionally passed into the function as th1now to th6now. If no valid solution exists, your function should return NaN (not a number) for all six joint angles. Complete the `puma_ik_team00.m` starter function provided in the zip file below, again entering your team number where needed.

Task 3: Script to Test PUMA Inverse Kinematics

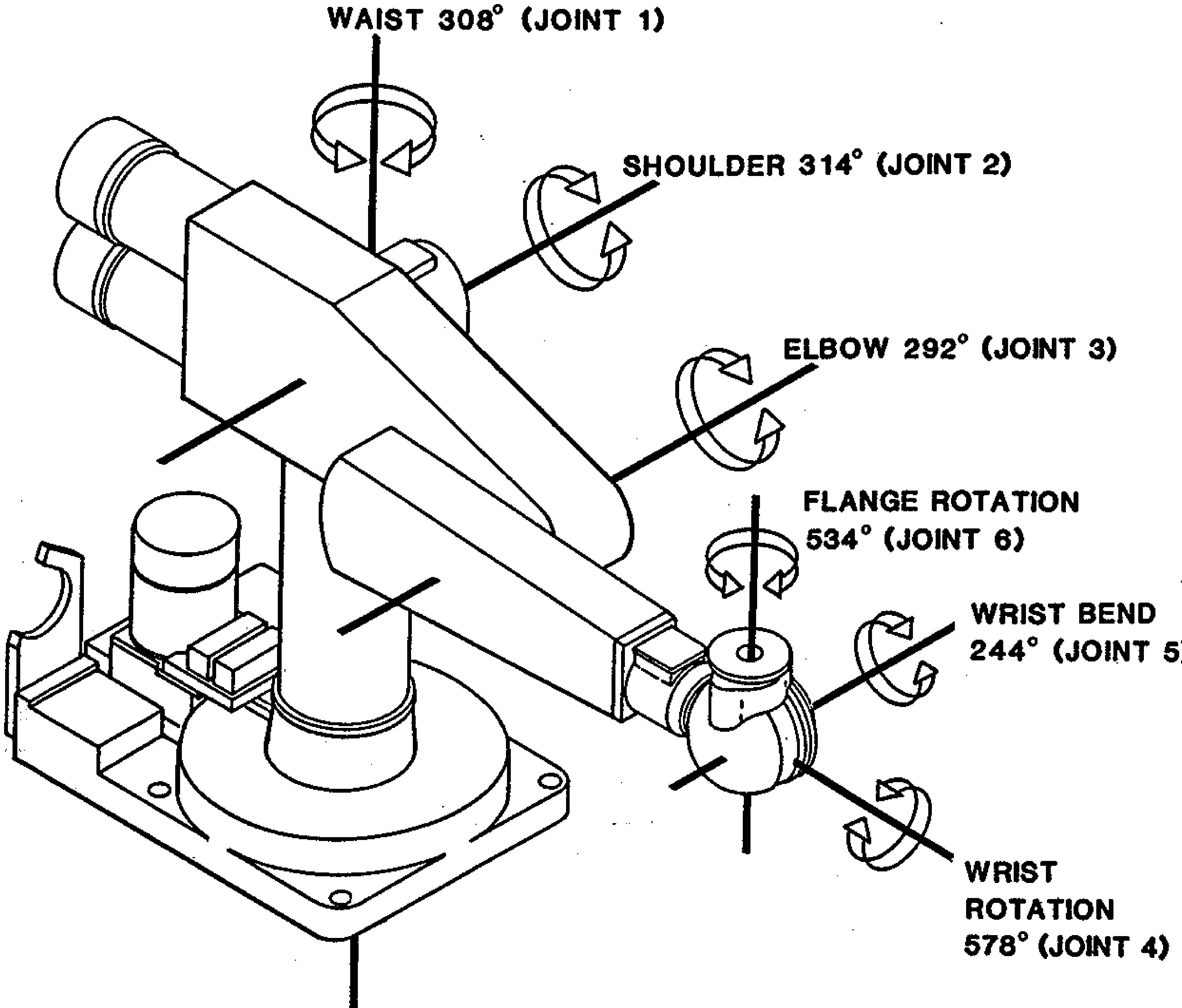
To help you assess the quality of your inverse kinematics function, you must also create a script that does thorough IK testing. At a minimum, it should compute two scores, both from 0 (worst performance) to 100 (perfect performance). One score is for finding inverse kinematics solutions without worrying about the robot's current configuration, and the second score is for finding inverse kinematics solutions close to the current joint angles of the robot. It is up to you to figure out what should be in this file to make it a reasonable test of the code. Make sure to test end-effector poses that are not reachable due to joint limits or other constraints. Complete the `test_puma_ik_team00.m` starter script provided in the zip file below, again entering your team number where needed.

Submission

1. Start an email to meam520@seas.upenn.edu
2. Make the subject *PUMA IK: Team 00*, replacing 00 with your team number.
3. Attach all of your correctly named MATLAB files to the email. They should be `puma_fk_teamXX.m`, `puma_ik_teamXX.m`, and `test_puma_ik_teamXX.m`, where XX is your team number, plus any additional files you may have created, also named according to this convention.
4. Optionally include any comments you have about this assignment.
5. Send the email.

Please come talk to the teaching team or post questions on Piazza if you get stuck on any part of this assignment.

po1-ik.zip (starter files)


WAIST 308° (JOINT 1)
SHOULDER 314° (JOINT 2)
ELBOW 292° (JOINT 3)
FLANGE ROTATION 534° (JOINT 6)
WRIST BEND 244° (JOINT 5)
WRIST ROTATION 578° (JOINT 4)

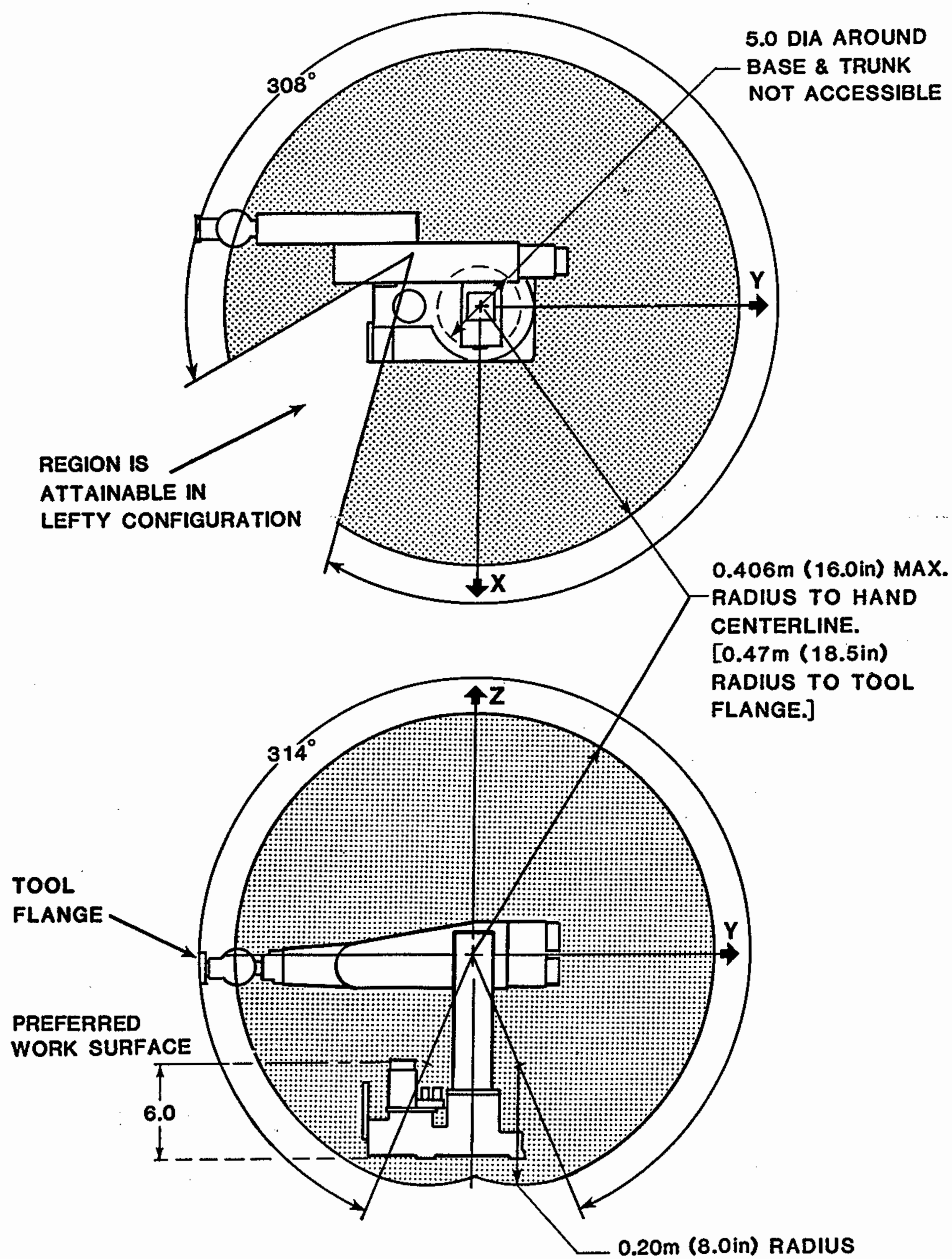


Figure 2-2. Robot Arm: Operating Envelope

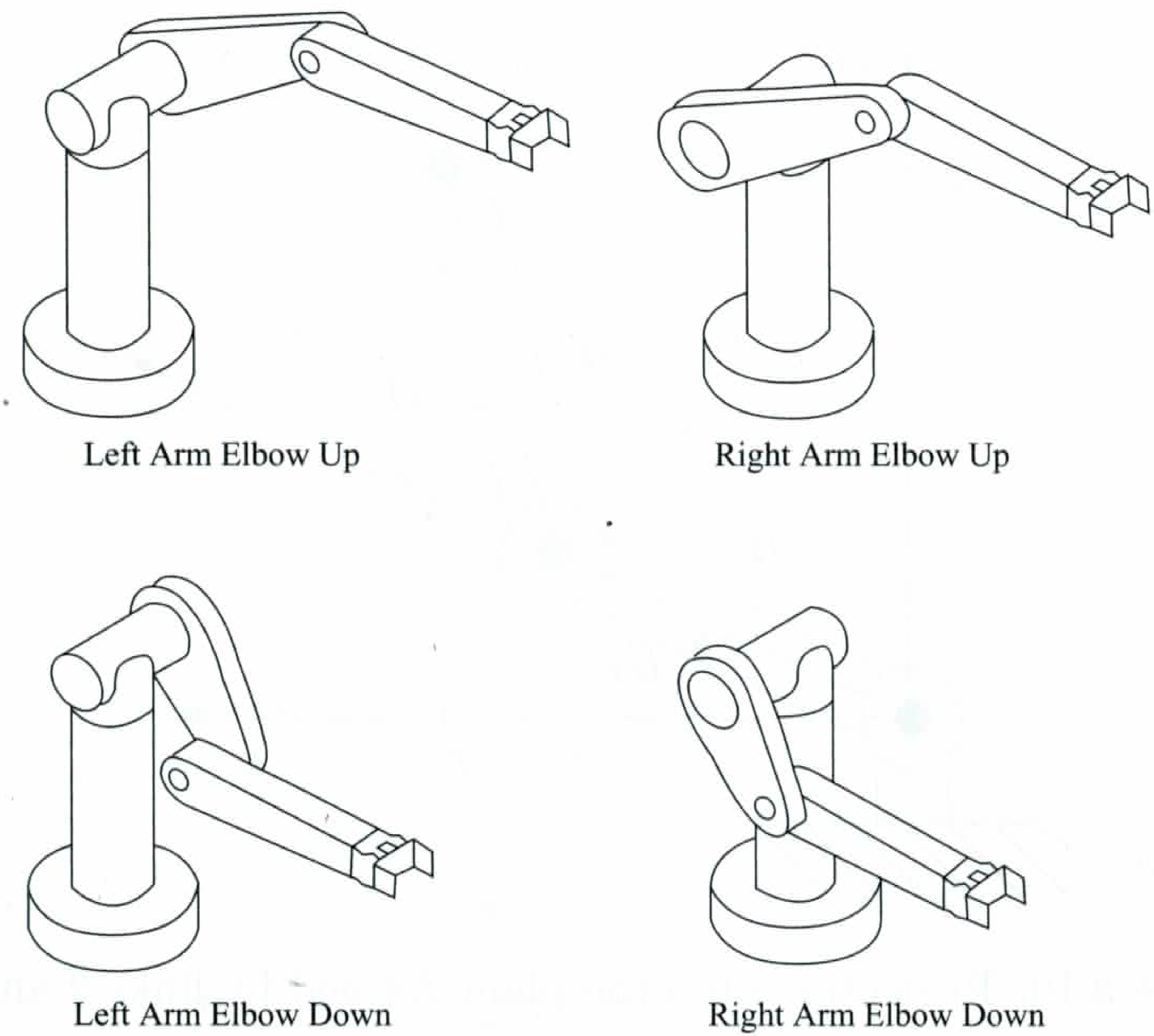
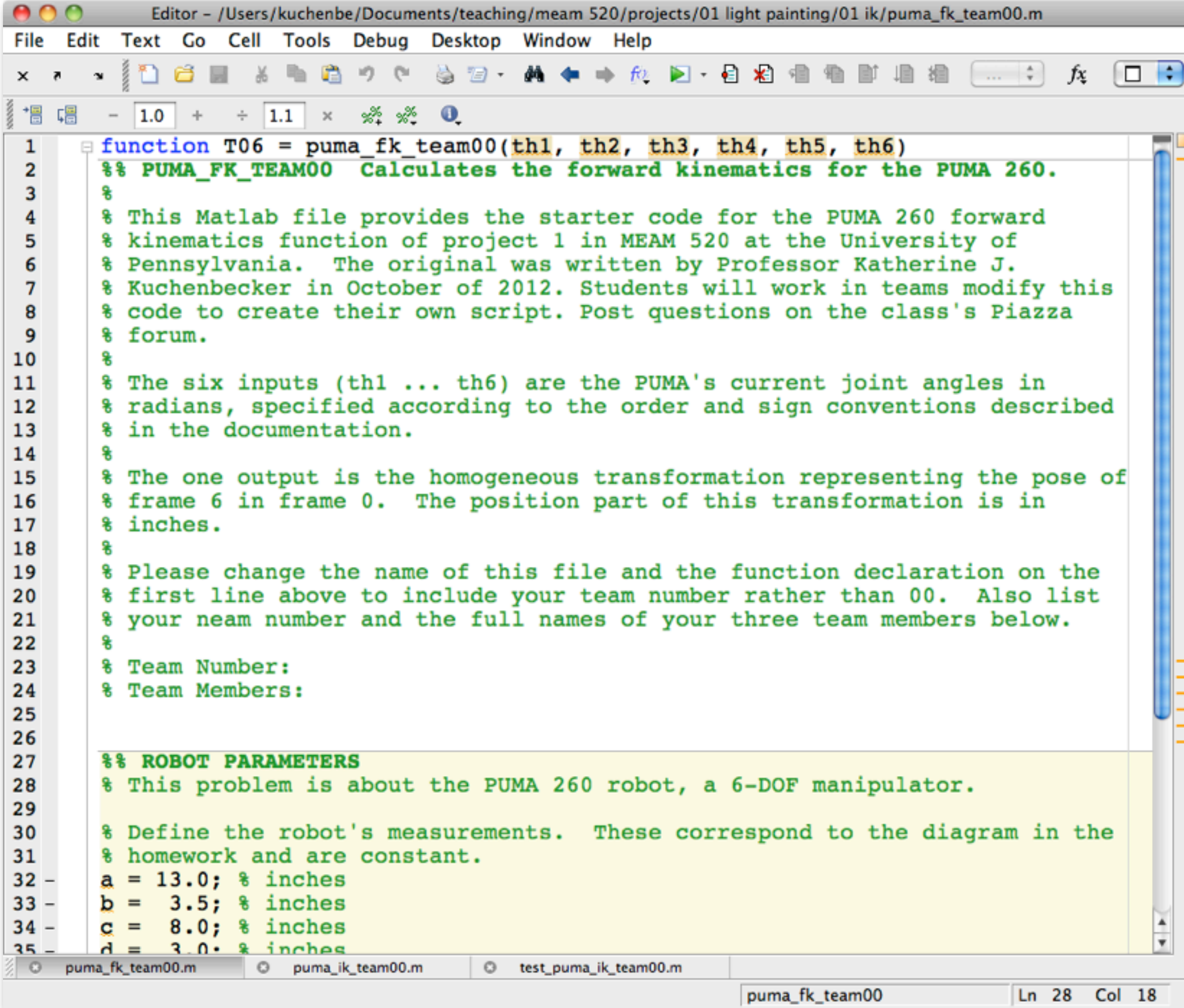
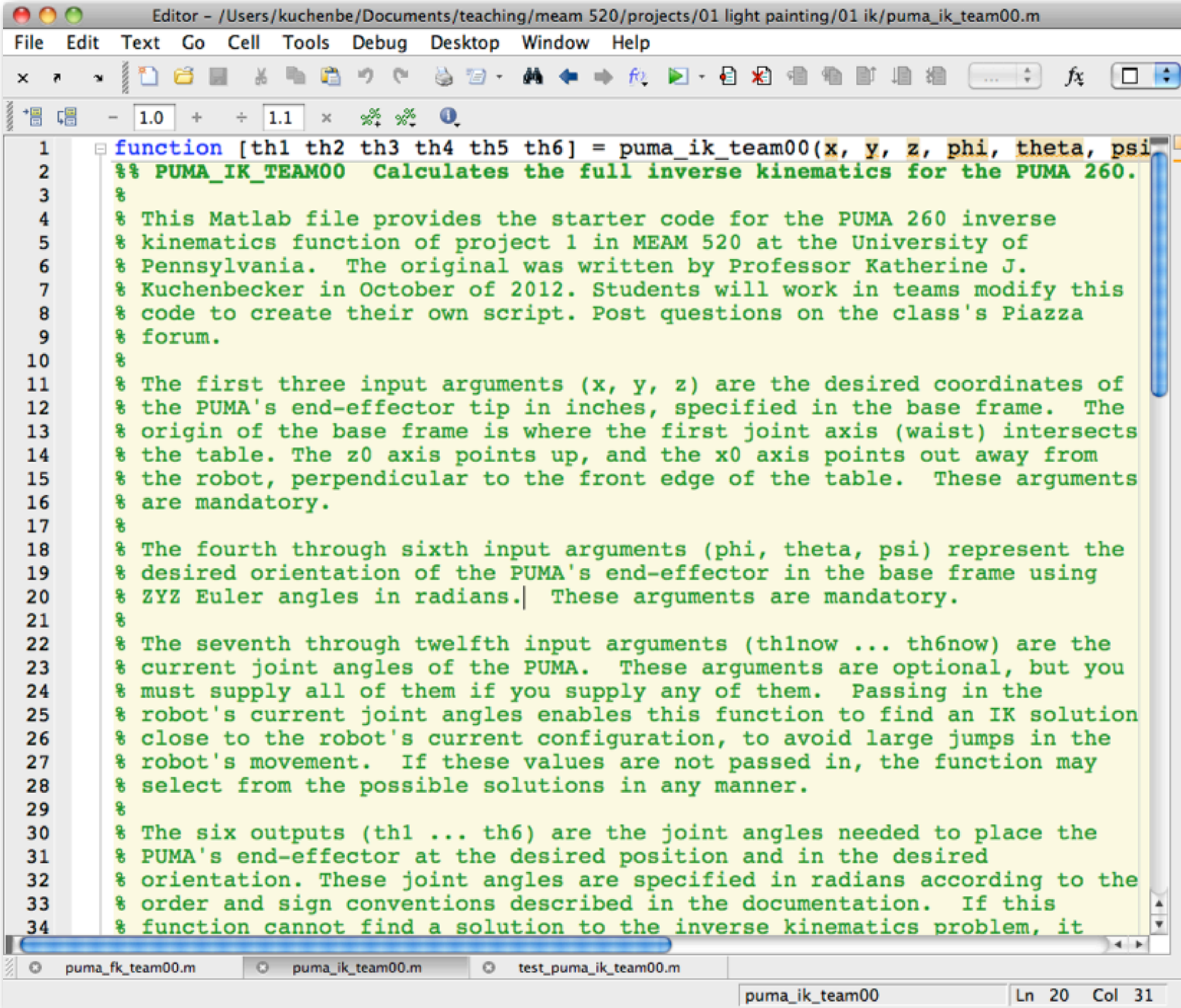
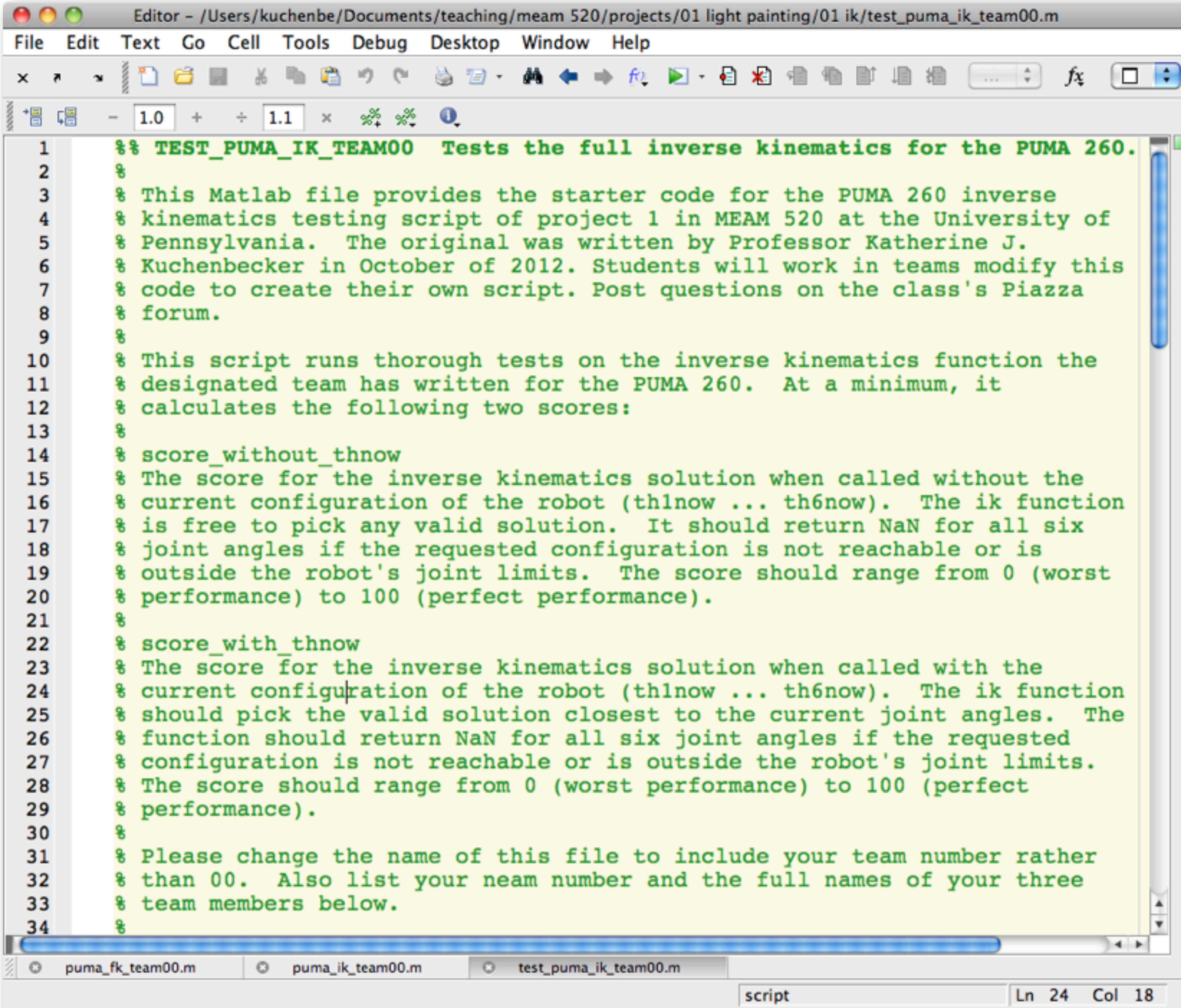


Figure 3.20: Four solutions of the inverse position kinematics for the PUMA manipulator.


Editor – /Users/kuchenbe/Documents/teaching/meam 520/projects/01 light painting/01 ik/puma_fk_team00.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1.0
1.1
function T06 = puma_fk_team00(th1, th2, th3, th4, th5, th6)
%% PUMA_FK_TEAM00 Calculates the forward kinematics for the PUMA 260.
%
% This Matlab file provides the starter code for the PUMA 260 forward
% kinematics function of project 1 in MEAM 520 at the University of
% Pennsylvania. The original was written by Professor Katherine J.
% Kuchenbecker in October of 2012. Students will work in teams modify this
% code to create their own script. Post questions on the class's Piazza
% forum.
%
% The six inputs (th1 ... th6) are the PUMA's current joint angles in
% radians, specified according to the order and sign conventions described
% in the documentation.
%
% The one output is the homogeneous transformation representing the pose of
% frame 6 in frame 0. The position part of this transformation is in
% inches.
%
% Please change the name of this file and the function declaration on the
% first line above to include your team number rather than 00. Also list
% your neam number and the full names of your three team members below.
%
% Team Number:
% Team Members:
%% ROBOT PARAMETERS
% This problem is about the PUMA 260 robot, a 6-DOF manipulator.
% Define the robot's measurements. These correspond to the diagram in the
% homework and are constant.
a = 13.0; % inches
b = 3.5; % inches
c = 8.0; % inches
d = 3.0; % inches
puma_fk_team00.m
puma_ik_team00.m
test_puma_ik_team00.m
puma_fk_team00
Ln 28 Col 18



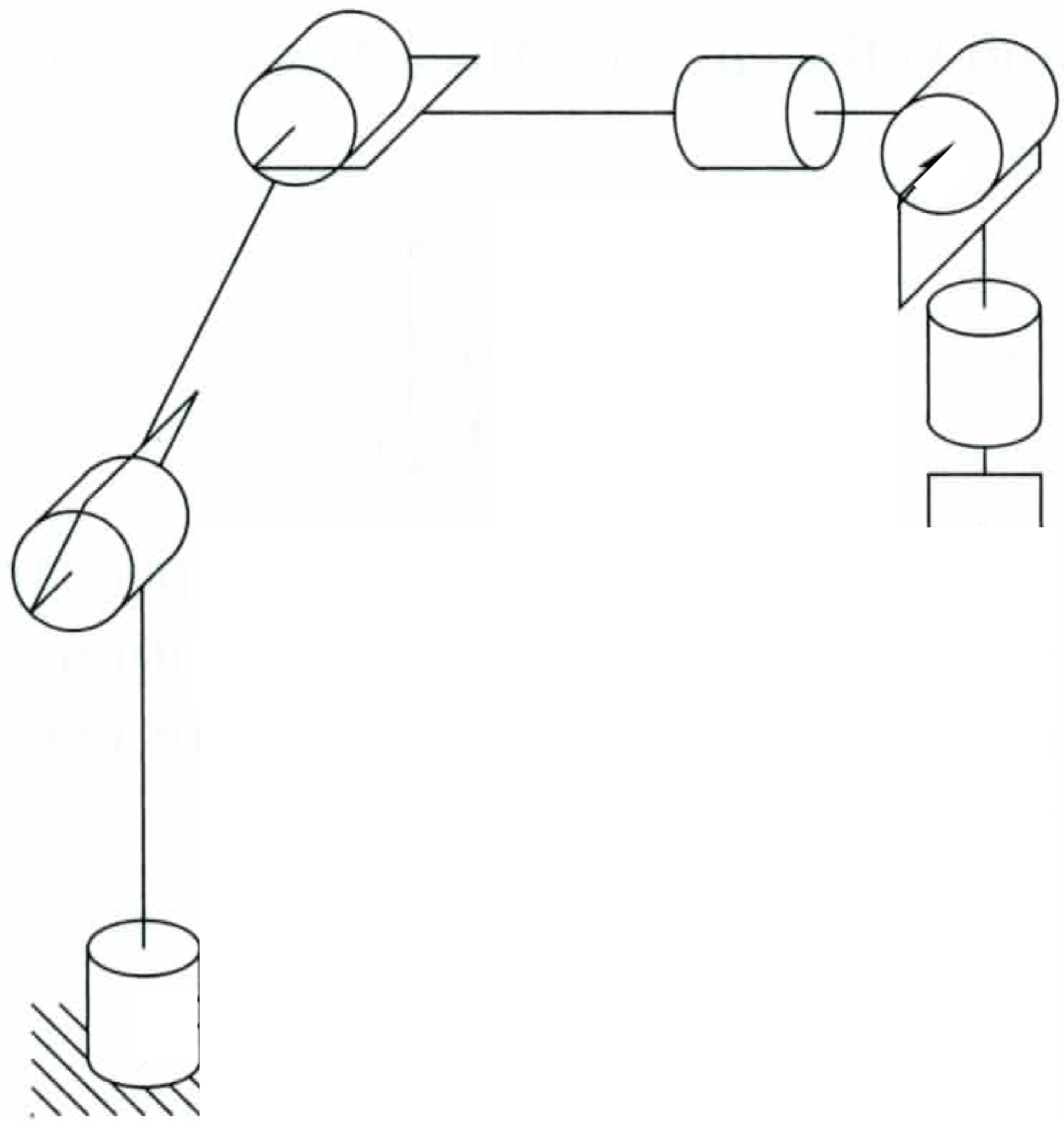

```
function [th1 th2 th3 th4 th5 th6] = puma_ik_team00(x, y, z, phi, theta, psi
%% PUMA_IK_TEAM00  Calculates the full inverse kinematics for the PUMA 260.
%
% This Matlab file provides the starter code for the PUMA 260 inverse
% kinematics function of project 1 in MEAM 520 at the University of
% Pennsylvania.  The original was written by Professor Katherine J.
% Kuchenbecker in October of 2012. Students will work in teams modify this
% code to create their own script. Post questions on the class's Piazza
% forum.
%
% The first three input arguments (x, y, z) are the desired coordinates of
% the PUMA's end-effector tip in inches, specified in the base frame.  The
% origin of the base frame is where the first joint axis (waist) intersects
% the table. The z0 axis points up, and the x0 axis points out away from
% the robot, perpendicular to the front edge of the table.  These arguments
% are mandatory.
%
% The fourth through sixth input arguments (phi, theta, psi) represent the
% desired orientation of the PUMA's end-effector in the base frame using
% ZYZ Euler angles in radians.  These arguments are mandatory.
%
% The seventh through twelfth input arguments (th1now ... th6now) are the
% current joint angles of the PUMA.  These arguments are optional, but you
% must supply all of them if you supply any of them.  Passing in the
% robot's current joint angles enables this function to find an IK solution
% close to the robot's current configuration, to avoid large jumps in the
% robot's movement.  If these values are not passed in, the function may
% select from the possible solutions in any manner.
%
% The six outputs (th1 ... th6) are the joint angles needed to place the
% PUMA's end-effector at the desired position and in the desired
% orientation. These joint angles are specified in radians according to the
% order and sign conventions described in the documentation.  If this
% function cannot find a solution to the inverse kinematics problem, it
```

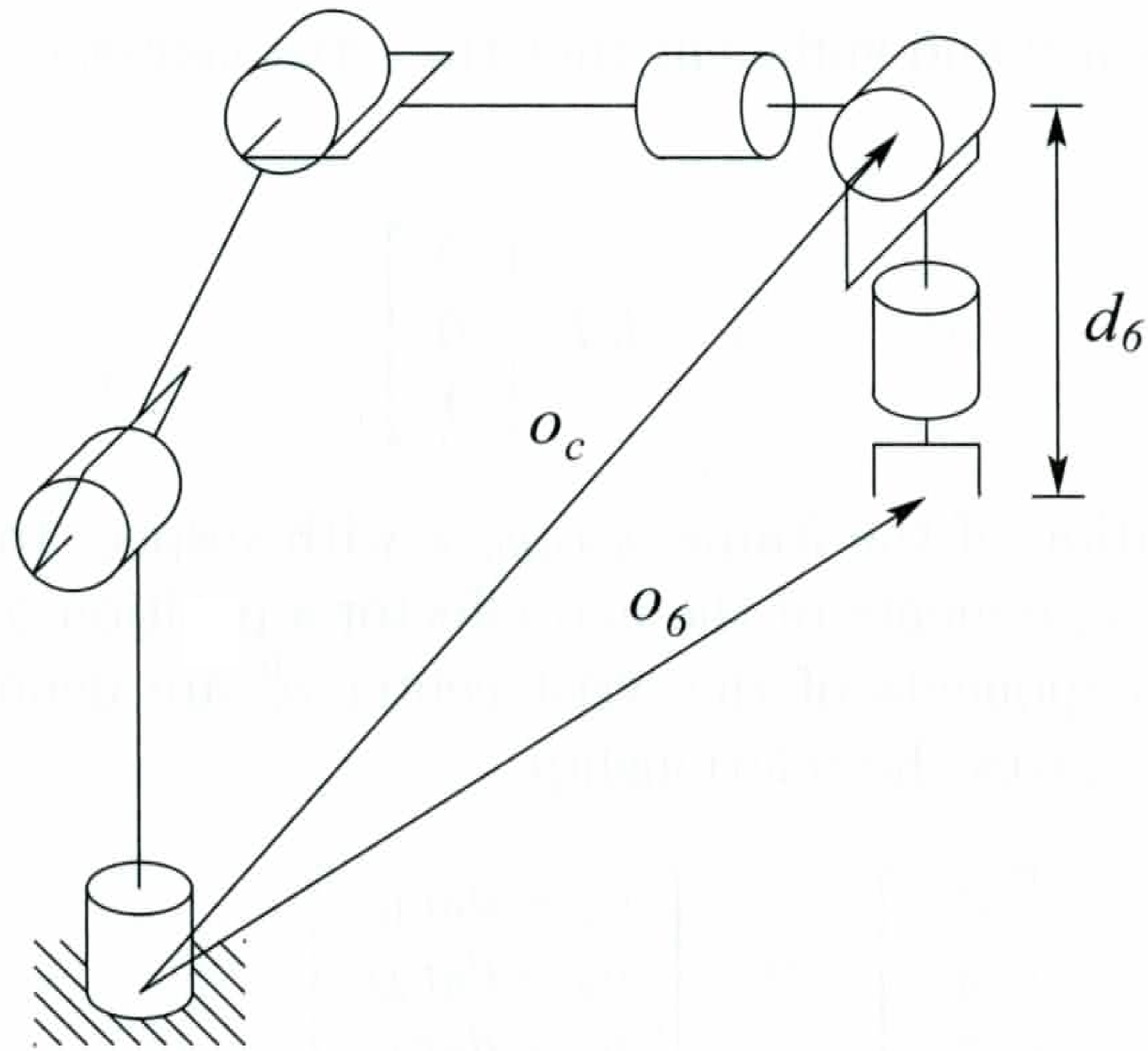



```
%% TEST_PUMA_IK_TEAM00  Tests the full inverse kinematics for the PUMA 260.
%
% This Matlab file provides the starter code for the PUMA 260 inverse
% kinematics testing script of project 1 in MEAM 520 at the University of
% Pennsylvania.  The original was written by Professor Katherine J.
% Kuchenbecker in October of 2012. Students will work in teams modify this
% code to create their own script. Post questions on the class's Piazza
% forum.
%
% This script runs thorough tests on the inverse kinematics function the
% designated team has written for the PUMA 260.  At a minimum, it
% calculates the following two scores:
%
% score_without_thnow
% The score for the inverse kinematics solution when called without the
% current configuration of the robot (th1now ... th6now).  The ik function
% is free to pick any valid solution.  It should return NaN for all six
% joint angles if the requested configuration is not reachable or is
% outside the robot's joint limits.  The score should range from 0 (worst
% performance) to 100 (perfect performance).
%
% score_with_thnow
% The score for the inverse kinematics solution when called with the
% current configuration of the robot (th1now ... th6now).  The ik function
% should pick the valid solution closest to the current joint angles.  The
% function should return NaN for all six joint angles if the requested
% configuration is not reachable or is outside the robot's joint limits.
% The score should range from 0 (worst performance) to 100 (perfect
% performance).
%
% Please change the name of this file to include your team number rather
% than 00.  Also list your neam number and the full names of your three
% team members below.
%
```

Questions ?

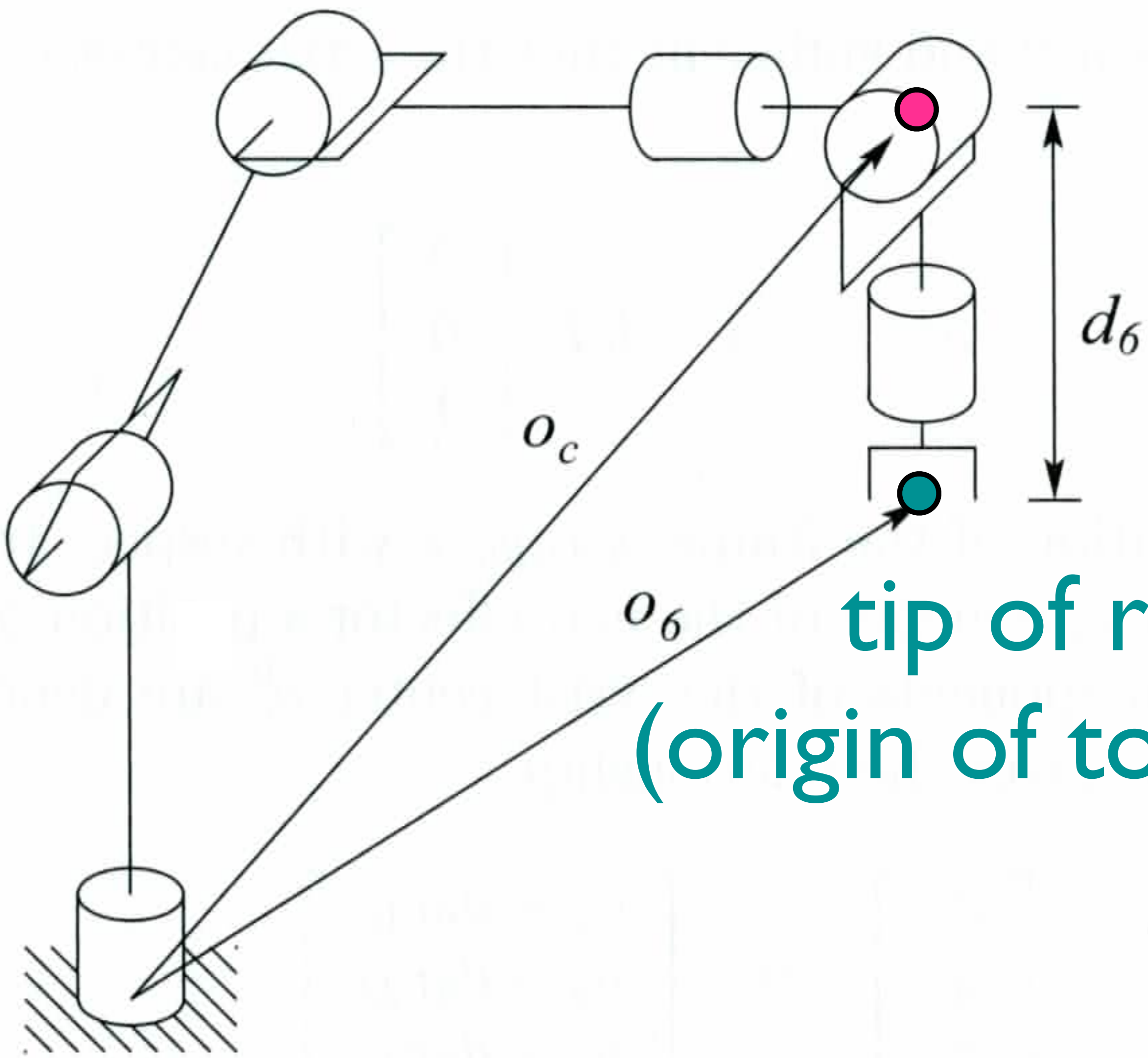
A 6-DOF Inverse Kinematics Example: Articulated/Elbow Manipulator with Spherical Wrist (pages 98-100, 103, 107-108 in SHV)





d_6
o_c
o_6

A helpful approach for 6-DOF robots: Kinematic Decoupling



$$o = o_c^0 + d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

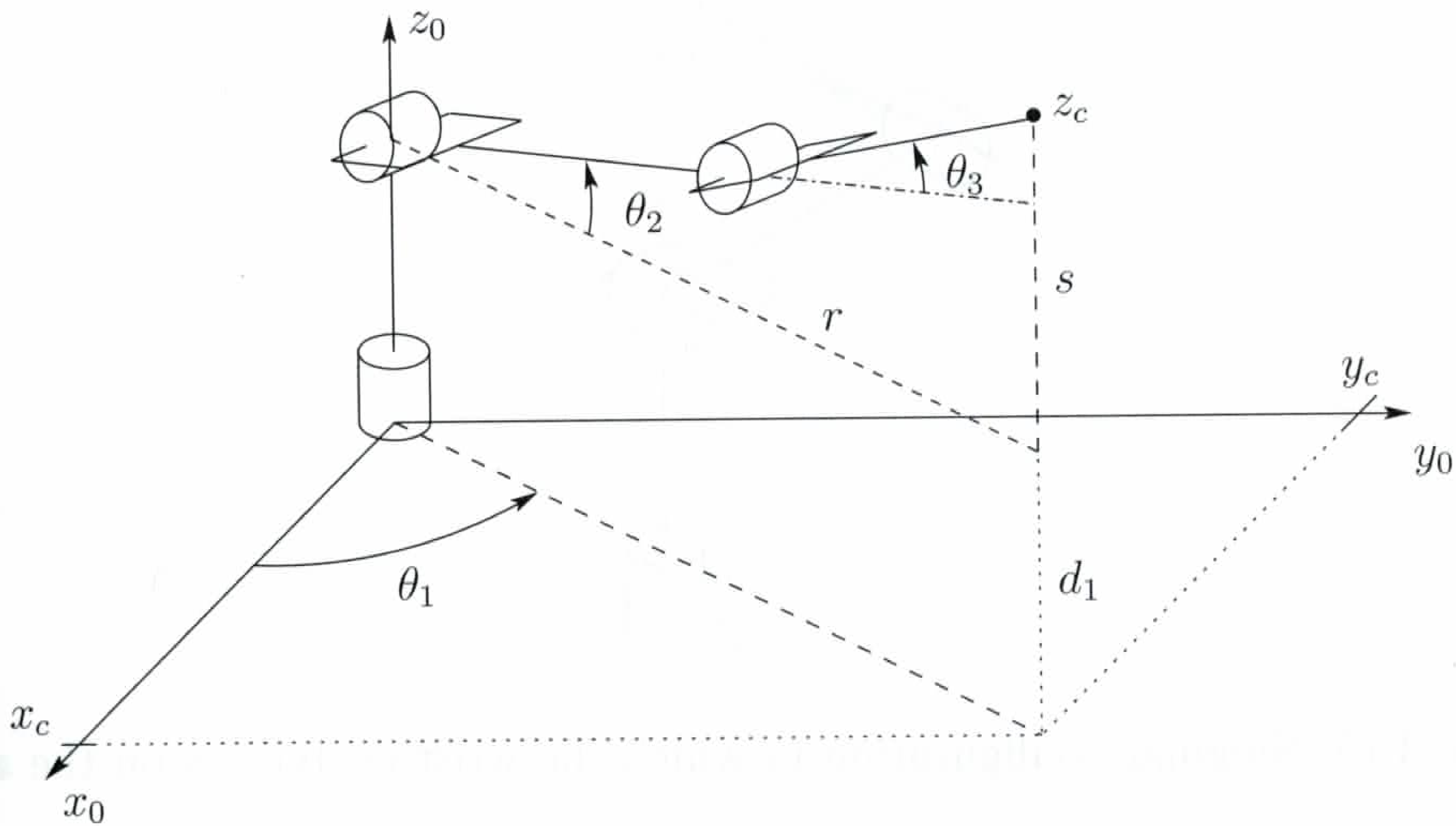
$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix}$$

position

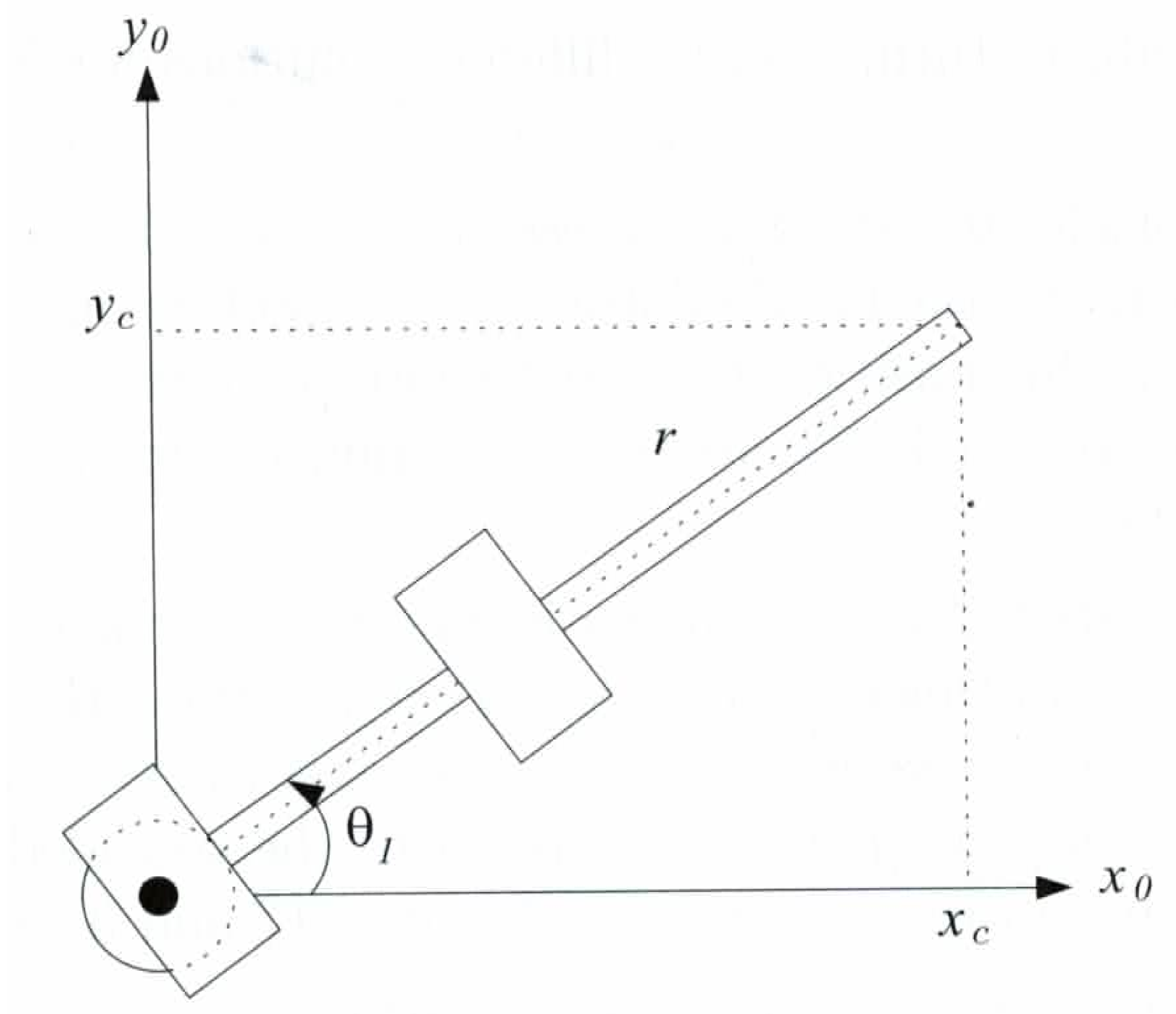
$$R = R_3^0 R_6^3$$

$$R_6^3 = (R_3^0)^{-1} R = (R_3^0)^T R$$

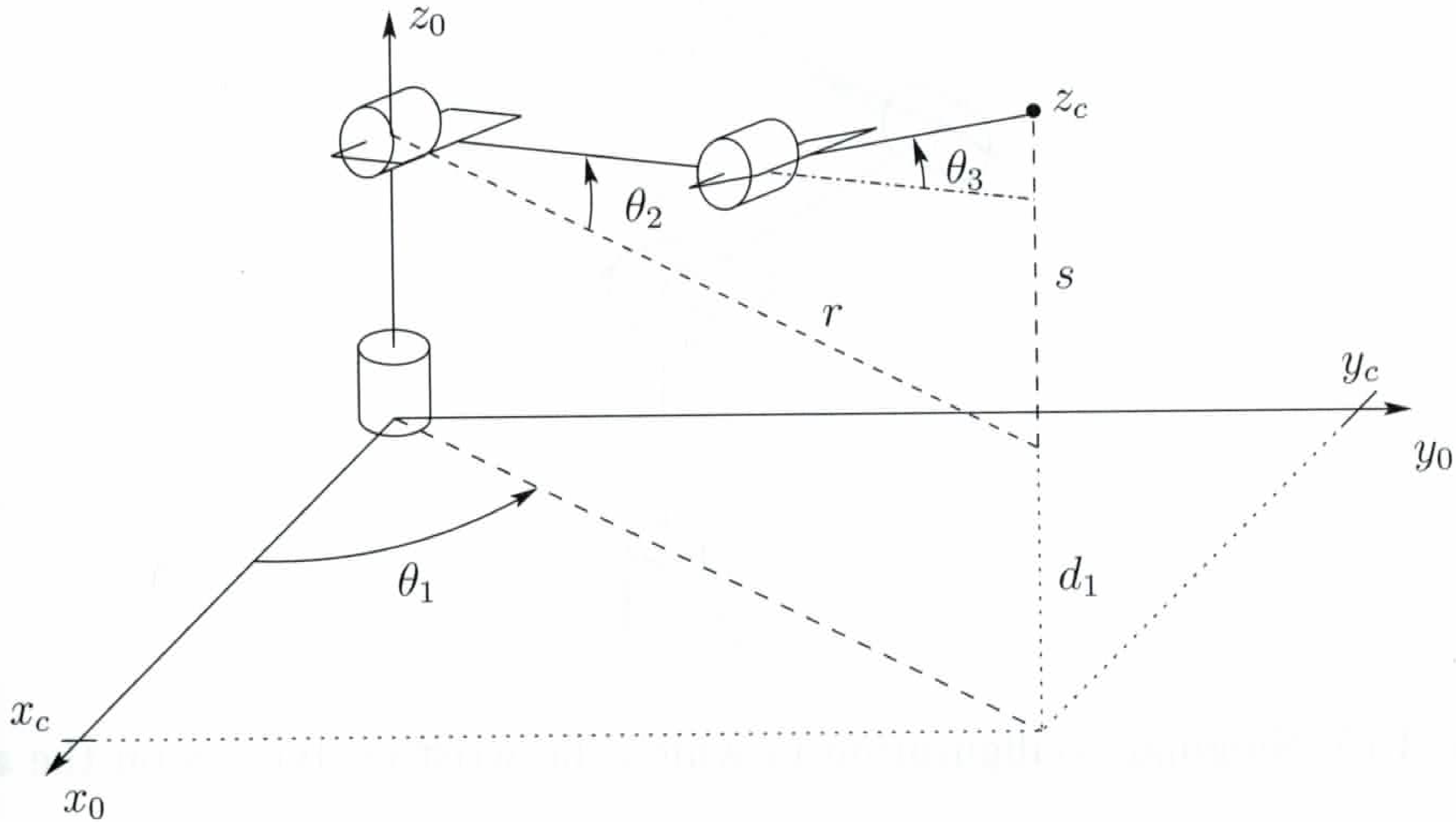
orientation

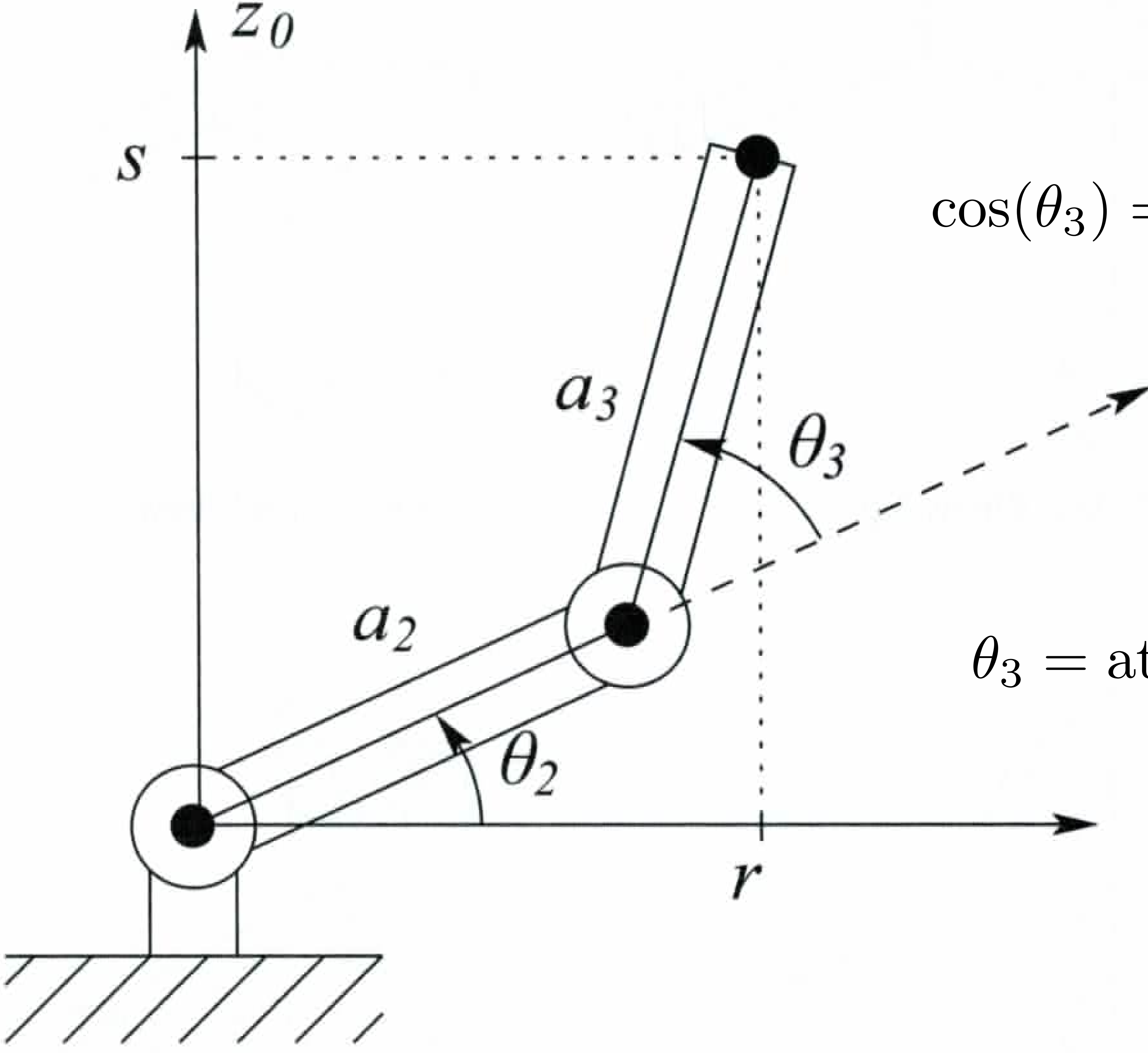


z_0
z_c
θ_2
θ_3
s
r
y_c
y_0
θ_1
d_1
x_c
x_0



$$\theta_1 = \operatorname{atan2}\left(\frac{y_c}{x_c}\right)$$


z_0
z_c
θ_2
θ_3
s
r
y_c
y_0
θ_1
d_1
x_c
x_0

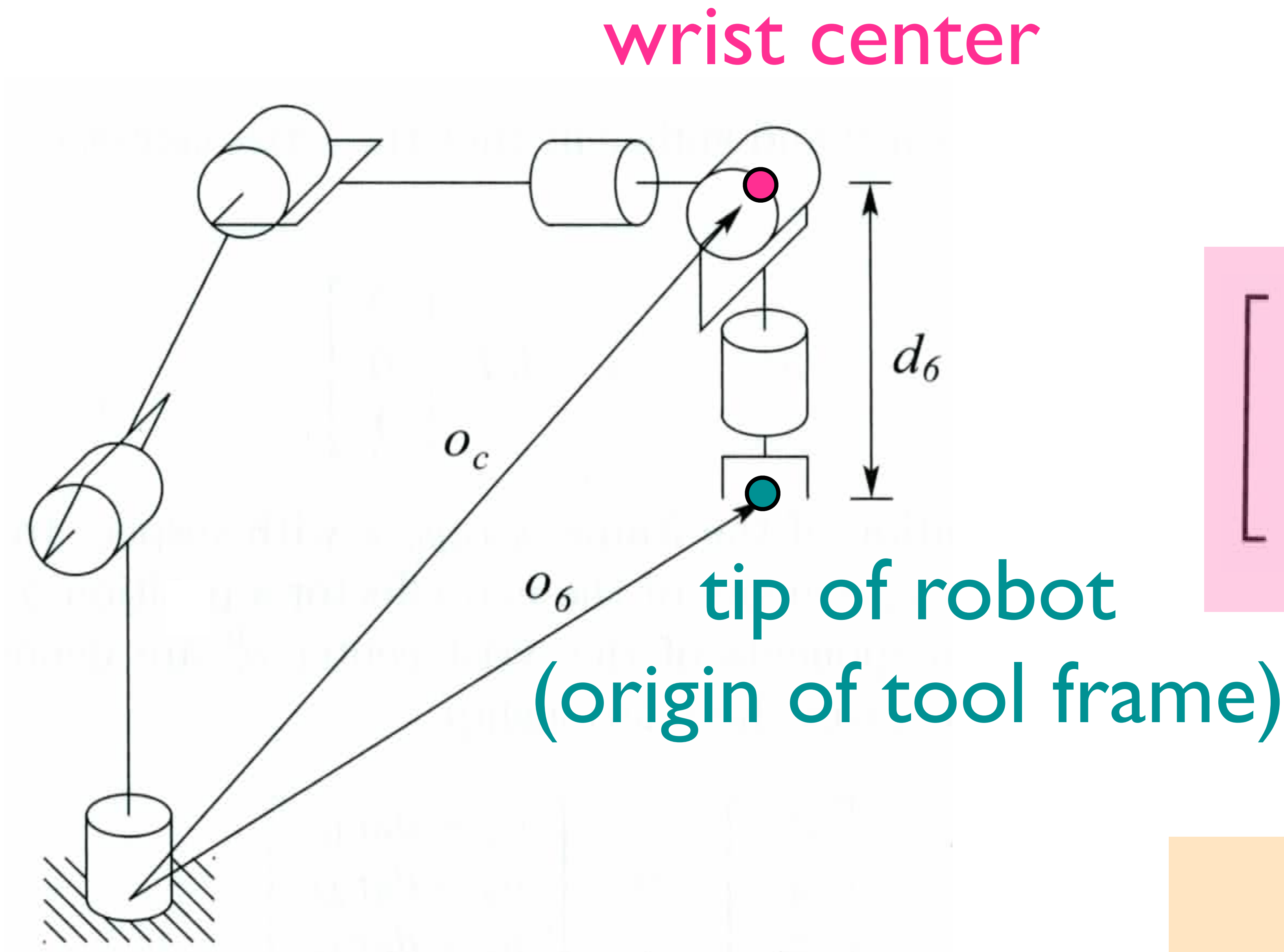


$$\cos(\theta_3) = \frac{r^2 + s^2 - a_2^2 - a_3^2}{2a_2a_3}$$

$$\cos(\theta_3) := D$$

$$\theta_3 = \text{atan2}\left(\frac{\pm\sqrt{1 - D^2}}{D}\right)$$

$$\theta_2 = ...$$



$$o = o_c^0 + d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix}$$

position

$$R = R_3^0 R_6^3$$

$$R_6^3 = (R_3^0)^{-1} R = (R_3^0)^T R$$

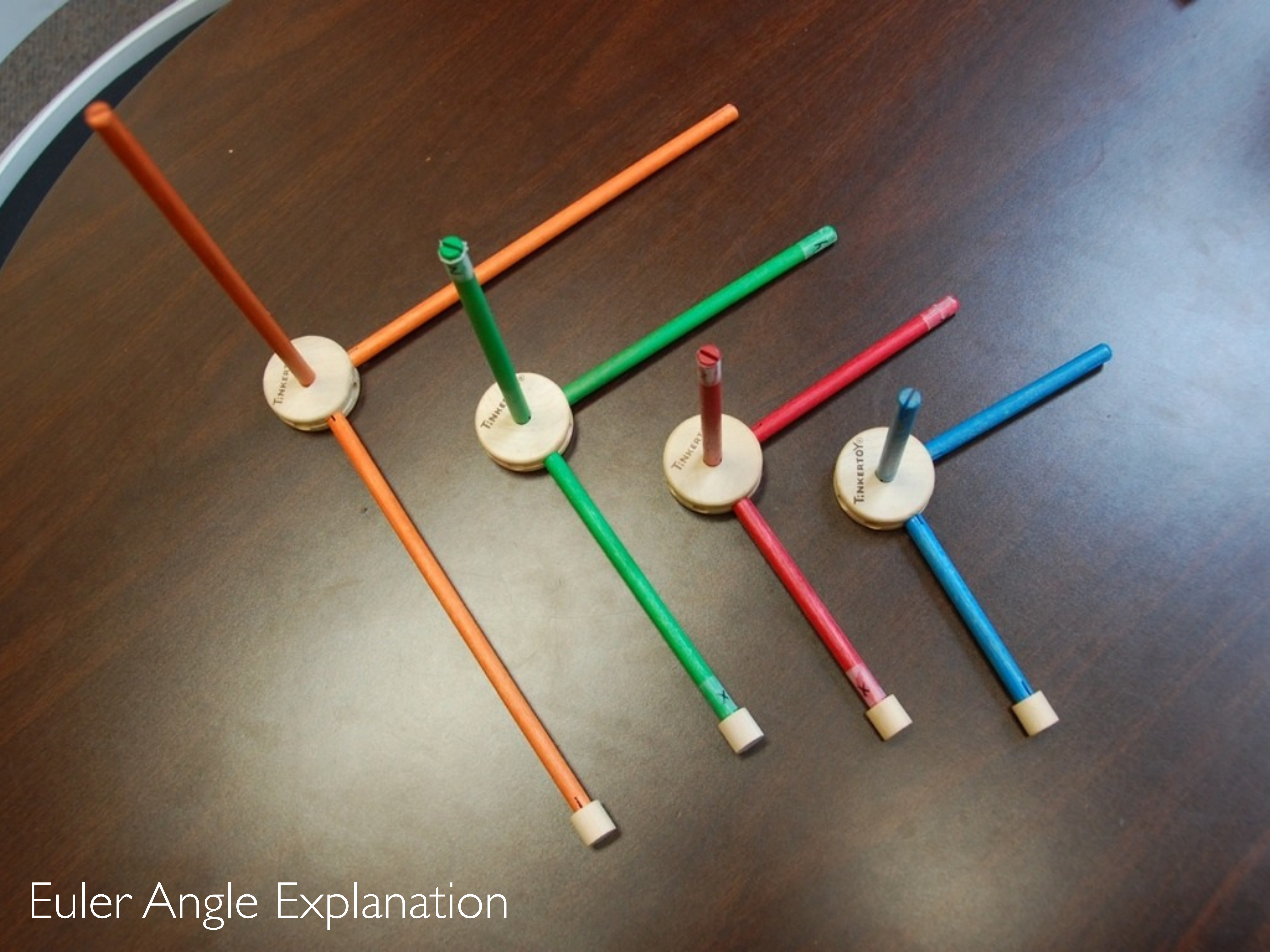
orientation

$$
\begin{aligned}
T_6^3 &= A_4 A_5 A_6 \\
&= \begin{bmatrix} R_6^3 & o_6^3 \\ 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5s_6 - s_4c_6 & c_4s_5 & c_4s_5d_6 \\ s_4c_5c_6 + c_4s_6 & -s_4c_5s_6 + c_4c_6 & s_4s_5 & s_4s_5d_6 \\ -s_5c_6 & s_5s_6 & c_5 & c_5d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

$$
\theta_4 = \phi \qquad \theta_5 = \theta \qquad \theta_6 = \psi
$$

$$
= \begin{bmatrix} c_\phi c_\theta c_\psi - s_\phi s_\psi & -c_\phi c_\theta s_\psi - s_\phi c_\psi & c_\phi s_\theta \\ s_\phi c_\theta c_\psi + c_\phi s_\psi & -s_\phi c_\theta s_\psi + c_\phi c_\psi & s_\phi s_\theta \\ -s_\theta c_\psi & s_\theta s_\psi & c_\theta \end{bmatrix}
$$

The book explains how to calculate the three angles given **R**: see SHV pages 55-56



Euler Angle Explanation

a set of DH joint variables is given by

$$\theta_1 = \text{Atan2}(x_c, y_c) \tag{3.64}$$

$$\begin{aligned}\theta_2 = {} & \text{Atan2}\left(\sqrt{x_c^2 + y_c^2 - d^2}, z_c - d_1\right) \\ & -\text{Atan2}(a_2 + a_3 c_3, a_3 s_3)\end{aligned} \tag{3.65}$$

$$\begin{aligned}\theta_3 = {} & \text{Atan2}\left(D, \pm\sqrt{1 - D^2}\right), \\ & \textit{with } D = \frac{x_c^2 + y_c^2 - d^2 + (z_c - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3}\end{aligned} \tag{3.66}$$

$$\begin{aligned}\theta_4 = {} & \text{Atan2}(c_1 c_{23} r_{13} + s_1 c_{23} r_{23} + s_{23} r_{33}, \\ & -c_1 s_{23} r_{13} - s_1 s_{23} r_{23} + c_{23} r_{33})\end{aligned} \tag{3.67}$$

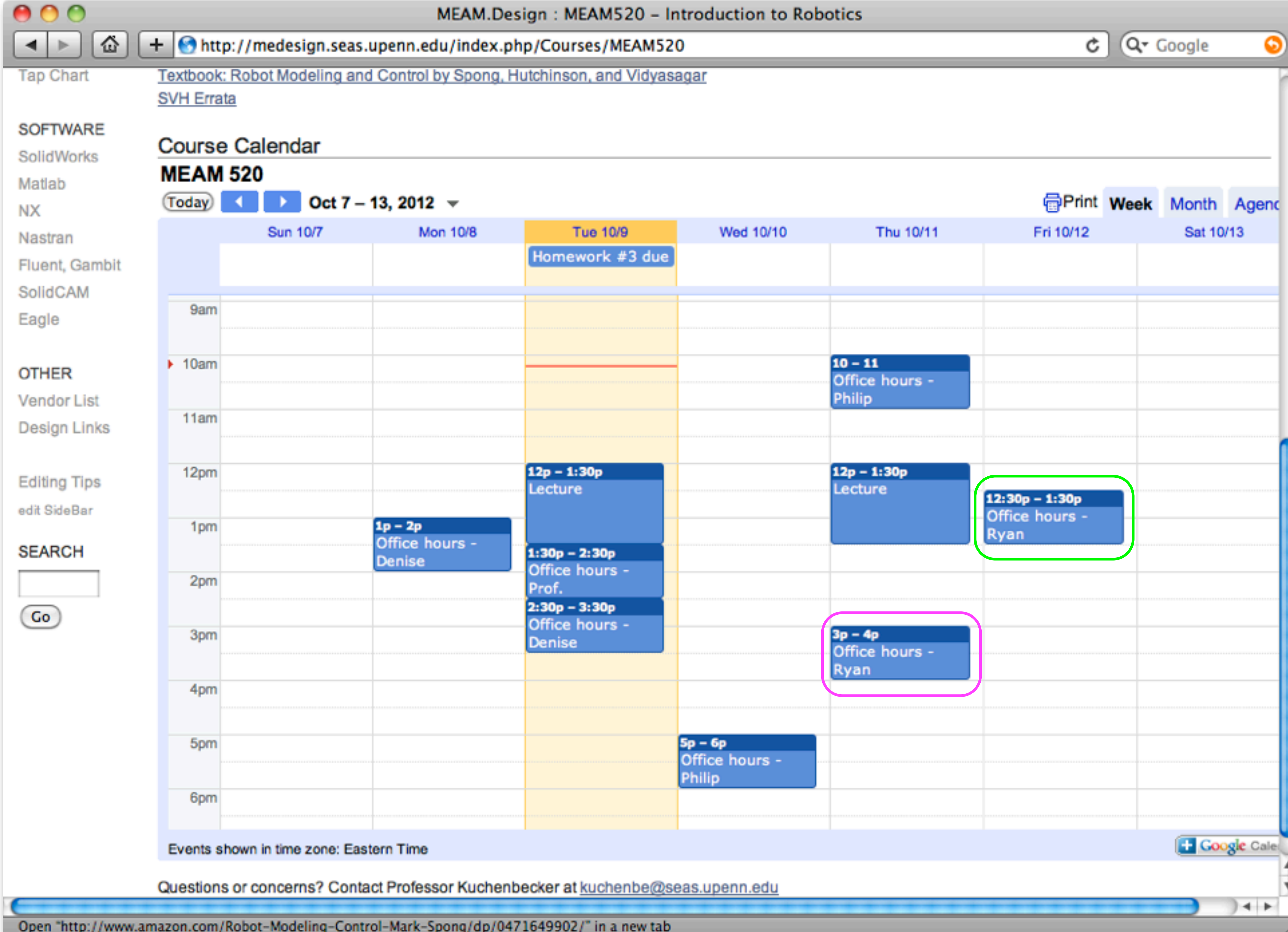
$$\theta_5 = \text{Atan2}\left(s_1 r_{13} - c_1 r_{23}, \pm\sqrt{1 - (s_1 r_{13} - c_1 r_{23})^2}\right) \tag{3.68}$$

$$\theta_6 = \text{Atan2}(-s_1 r_{11} + c_1 r_{21}, s_1 r_{12} - c_1 r_{22}) \tag{3.69}$$

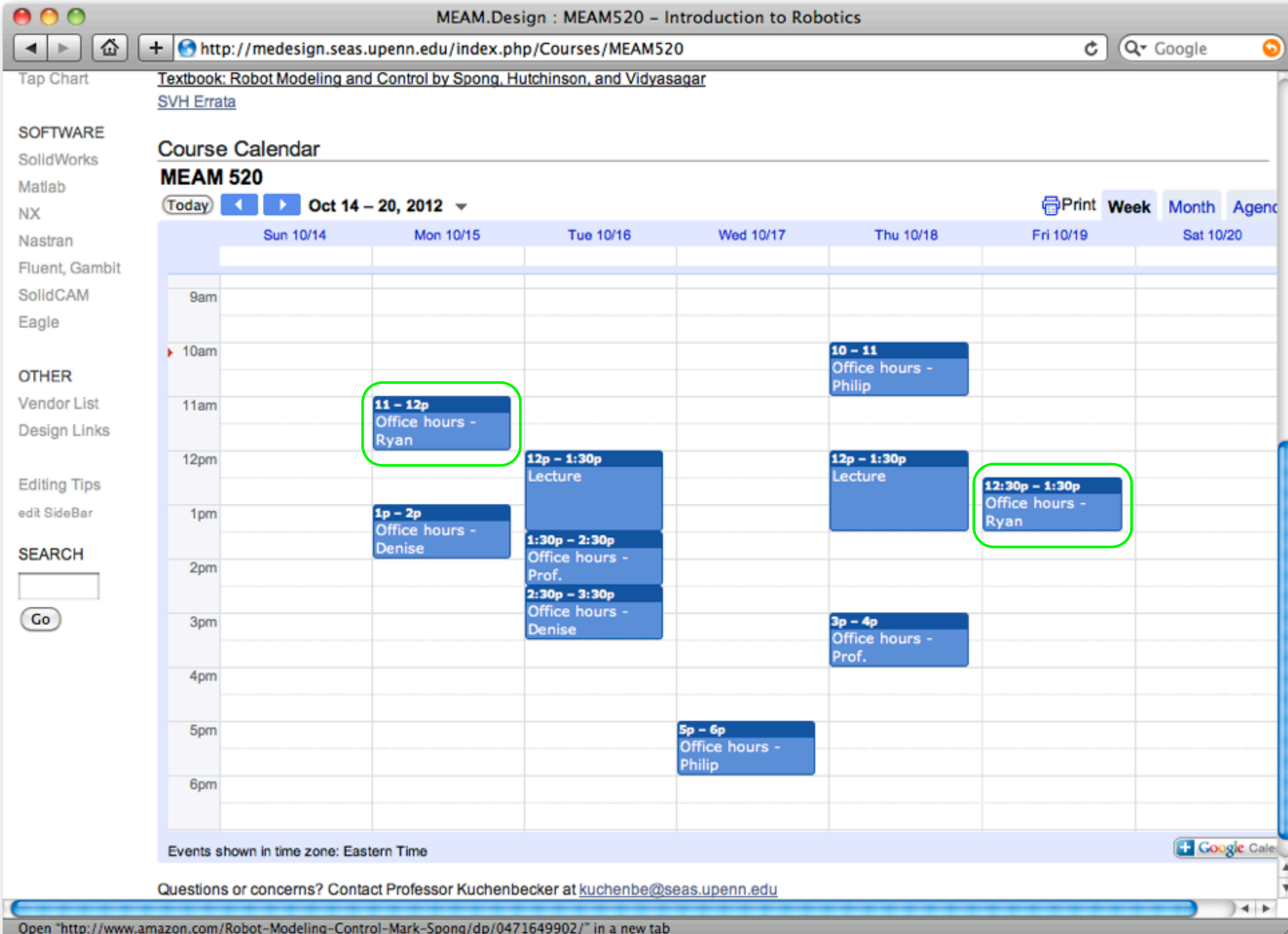
The other possible solutions are left as an exercise (Problem 3-20).

Questions ❓

New Office Hours



MEAM.Design : MEAM520 – Introduction to Robotics
http://medesign.seas.upenn.edu/index.php/Courses/MEAM520
Google
Tap Chart
Textbook: Robot Modeling and Control by Spong, Hutchinson, and Vidyasagar
SVH Errata
SOFTWARE
SolidWorks
Matlab
NX
Nastran
Fluent, Gambit
SolidCAM
Eagle
OTHER
Vendor List
Design Links
Editing Tips
edit SideBar
SEARCH
Go
Course Calendar
MEAM 520
Today
Oct 7 – 13, 2012
Print
Week
Month
Agend
Sun 10/7
Mon 10/8
Tue 10/9
Wed 10/10
Thu 10/11
Fri 10/12
Sat 10/13
Homework #3 due
9am
10am
11am
12pm
1pm
2pm
3pm
4pm
5pm
6pm
10 – 11
Office hours - Philip
12p – 1:30p
Lecture
12p – 1:30p
Lecture
12:30p – 1:30p
Office hours - Ryan
1p – 2p
Office hours - Denise
1:30p – 2:30p
Office hours - Prof.
2:30p – 3:30p
Office hours - Denise
3p – 4p
Office hours - Ryan
5p – 6p
Office hours - Philip
Events shown in time zone: Eastern Time
Google Cale
Questions or concerns? Contact Professor Kuchenbecker at kuchenbe@seas.upenn.edu
Open "http://www.amazon.com/Robot-Modeling-Control-Mark-Spong/dp/0471649902/" in a new tab


MEAM.Design : MEAM520 – Introduction to Robotics
http://medesign.seas.upenn.edu/index.php/Courses/MEAM520
Google
Tap Chart
Textbook: Robot Modeling and Control by Spong, Hutchinson, and Vidyasagar
SVH Errata
SOFTWARE
SolidWorks
Matlab
NX
Nastran
Fluent, Gambit
SolidCAM
Eagle
OTHER
Vendor List
Design Links
Editing Tips
edit SideBar
SEARCH
Go
Course Calendar
MEAM 520
Today
Oct 14 – 20, 2012
Print
Week
Month
Agenc
Sun 10/14
Mon 10/15
Tue 10/16
Wed 10/17
Thu 10/18
Fri 10/19
Sat 10/20
9am
10am
11am
12pm
1pm
2pm
3pm
4pm
5pm
6pm
10 – 11 Office hours - Philip
11 – 12p Office hours - Ryan
12p – 1:30p Lecture
12p – 1:30p Lecture
12:30p – 1:30p Office hours - Ryan
1p – 2p Office hours - Denise
1:30p – 2:30p Office hours - Prof.
2:30p – 3:30p Office hours - Denise
3p – 4p Office hours - Prof.
5p – 6p Office hours - Philip
Events shown in time zone: Eastern Time
Google Cale
Questions or concerns? Contact Professor Kuchenbecker at kuchenbe@seas.upenn.edu
Open "http://www.amazon.com/Robot-Modeling-Control-Mark-Spong/dp/0471649902/" in a new tab

Team Formation

You will work in a team of 3
(33 teams of 3, only one team of 4)

Each team must have at least one undergraduate
and at least one graduate student.
Submatriculants count as undergraduates.

(52 undergrads, 51 grads)

Try to have two MEAM students and
one non-MEAM student on each team
(71 MEAM, 33 non-MEAM)

Pick your team by midnight on Tuesday, October 9,
or ask to be assigned. Email denwong@seas.upenn.edu

Time to meet teammates at the end of class