Homework 2:
Manipulator Kinematics and DH Parameters

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

September 18, 2012

This assignment is due on **Thursday, September 27 (updated)**, 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 Friday, 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.

Written Problems *(30 points)*

The first set of problems are written, including two from the textbook, *Robot Modeling and Control* by Spong, Hutchinson, and Vidyasagar (SHV). Please follow the extra clarifications and instructions when provided. Write in pencil, show your work clearly, box your answers, and staple your pages together.

1. Custom problem – Kinematics of Baxter *(5 points)*
   Rethink Robotics recently released a new robot named Baxter. Watch YouTube videos of Baxter (e.g., http://www.youtube.com/watch?v=rjPFqkFyrOY) to learn about its kinematics. Draw a schematic of the serial kinematic chain of Baxter's left arm (the one the woman is touching in the picture above.) Use the book's conventions for how to draw revolute and prismatic joints in 3D.

2. SHV 3-7, page 113 – Three-link Cartesian Robot *(10 points)*
   Your solution should include a schematic of the manipulator with appropriately placed coordinate frames, a table of the DH parameters, and the final transformation matrix. Then answer the following question: What are the x, y, and z coordinates of the tip of the robot's end-effector in the base frame (as a function of the robot parameters and the joint coordinates)?

3. SHV 3-6, page 113 – Three-link Articulated Robot *(15 points)*
Your solution should include a schematic of the manipulator with appropriately placed coordinate frames, a table of the DH parameters, and the final transformation matrix. Then answer the following question: What are the x, y, and z components in the base frame of a unit vector pointing along the robot's last link (from the third joint to the tip, as a function of the robot parameters and the joint coordinates)?

MATLAB Programming *(30 points)*

Your task for this question is to update a provided MATLAB script so that it animates a SCARA robot moving under several conditions.

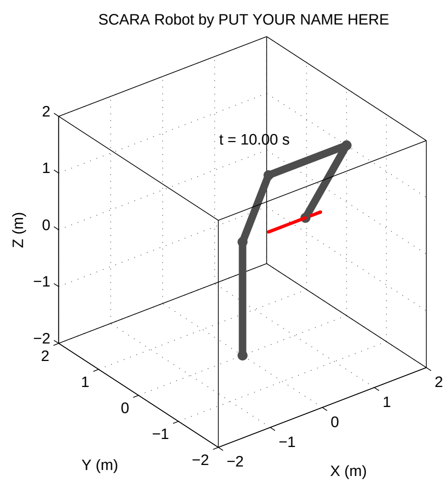
A. Read Example 3.6 in the textbook. It derives the forward kinematics of a SCARA manipulator using the DH convention. You can find this example on pages 91–93 of SHV. The provided solution includes a schematic, a table of DH parameters, the A matrices, and the full transformation T_4^0. You don't need to re-derive any of these answers, but you should understand this manipulator's structure before moving on. Note that the example shows four joints, but this problem will consider only the first three joints (ignoring the R joint at the wrist, which is parameterized by θ_4^*).

B. Download the script called `scara_robot_starter.m` from Homework 2 on the MEAM 520 wiki. This script sets up an environment to animate the movement of a SCARA robot (a particular RRP manipulator) over time. It draws a simple ball and stick model of the robot.

C. Rename the starter file to `scara_robot_yourpennkey.m` so that everyone will have a unique file name. Your PennKey is the first part of your Penn email address.

D. Put your name at the top of the file where it says `student_name = 'PUT YOUR NAME HERE';` This string is automatically added to the title of the created animation.

E. Run the code and watch what happens. As provided, the script shows an arbitrary static robot shape with a tip that moves back and forth in a straight line, leaving a red trail as it goes. The last frame of the animation should look like the plot below. Note that the elapsed virtual time is shown in a text box that floats above the robot.



inin

After the simulation runs, you can rotate the plot to better see its three-dimensional nature using the "Rotate 3D" tool. This tool is the rounded counter-clockwise arrow in the toolbar at the top of the plot window: 

F. Your first task is to construct three joint coordinate histories for the robot to follow. The starter code includes five robot motion modes, dictated by the value of the variable `motion_mode`, which you set on line 42.

- `motion_mode = 0` makes the robot sit still with all joint coordinates at zero. This joint coordinate history is programmed for you.
- `motion_mode = 1` makes only joint 1 move. This joint coordinate history is programmed for you.
- `motion_mode = 2` makes only joint 2 move. You must program this joint coordinate history.
- `motion_mode = 3` makes only joint 3 move. You must program this joint coordinate history.
- `motion_mode = 4` makes all of the joints move in an interesting pattern of your choosing. You must program this joint coordinate history.

Follow the provided examples to fill in joint coordinate histories for motion modes 2, 3, and 4. You can using the time variable `t` to make the active joints move in whatever way you want over time. For now, the position of the tip of the robot is being set so that `x = theta1`, `y = theta2`, and `z = d3` so you can check your joint coordinate calculations. Make sure your script runs, and make sure the tip moves the way it should in all five motion modes before moving on to the next step.

G. Update the code between the two lines of stars to make the animation show the links of a SCARA robot. This code is inside a `for` loop that runs over the entire time vector to animate the motion.

At each iteration, the robot's joint coordinates are available for you to use as $\theta_1 =$ `theta1`, $\theta_2 =$ `theta2`, and $d_3 =$ `d3`. Creating this animation will require you to decide the points along the robot that you want to display and then figure out how to calculate their positions given the robot's current configuration. I recommend you start by plotting just one point, such as the end-effector tip, and then add the other points of the robot.

You should put the output of your calculations in the matrix `points_to_plot`. This matrix has three rows, corresponding to x, y, and z coordinates in the base frame. Each column represents one point that you want to plot, and the order of the columns is the order in which the points are plotted. The last column should be the position of the tip of the robot; it is the only point whose trajectory is tracked using a history of red dots.

Important: Your calculations may not use any built-in or downloaded functions dealing with rotation matrices, homogeneous transformations, Denavit-Hartenberg parameters, robot animations, or related topics. Instead, you must type all your calculations yourself, using only low-level functions such as `sin`, `cos`, and vector/matrix math.

When you get the calculations correct, the stick figure robot should move as specified in the selected motion mode. Test out all five motion modes and try to get them all to work. Observe the motion that results in each case.

H. Go back and update the joint coordinate histories for motion mode 4 to make it produce a tip trajectory that you particularly like. Spend some time playing with different functions to improve your understanding of this robot's kinematics.

Submitting Your Code

Follow these instructions to submit your code:

1. Start an email to `meam520@seas.upenn.edu`
2. Make the subject *Homework 2: Your Name*, replacing *Your Name* with your name.
3. Attach your correctly named MATLAB script to the email.
4. Optionally include any comments you have about this assignment.
5. Send the email.