Homework 1:
MATLAB Programming and Reachable Workspace

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

September 4, 2013

This assignment is due on Tuesday, September 10, by midnight (11:59:59 p.m.) Your code should be submitted via email according to the instructions at the end of this document. Late submissions will be accepted until Thursday, September 12, by noon (11:59:59 a.m.), but they will be penalized by 10% for each partial or full day late, up to 20%. After the late 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 any other individual or team. Any submissions suspected of violating Penn's Code of Academic Integrity will be reported to the Office of Student Conduct. If you get stuck, post a question on Piazza or go to office hours!

Individual vs. Pair Programming

This class will use the programming language MATLAB to analyze and simulate robotic systems and also to control real robots. Some students in the class have never used MATLAB before, and others are quite familiar with it. The goal of this assignment is to get everyone starting to use MATLAB to improve their understanding of robotic systems.

If you have not used MATLAB much before, you should do this assignment with another student in our class. If you are already pretty comfortable with MATLAB, you should do this assignment alone. Read the assignment to decide which option is right for you.

If you do this homework with a partner, you may work with anyone you choose; the only stipulation is that they also have only a little MATLAB experience. If you are looking for a partner, consider using the "Search for Teammates!" tool on Piazza.

If you are in a pair, you should work closely with your partner throughout this assignment, following the paradigm of pair programming. You will turn in one MATLAB script for which you are both jointly responsible, and you will both receive the same grade. Please follow these pair programming guidelines, which were adapted from "All I really need to know about pair programming I learned in kindergarten," by Williams and Kessler, *Communications of the ACM*, May 2000:

- Start with a good attitude, setting aside any skepticism and expecting to jell with your partner.
- Don't start writing code alone. Arrange a meeting with your partner as soon as you can.
- Use just one computer, and sit side by side; a desktop computer with a large monitor is better for this than a laptop. Make sure both partners can see the screen.
- At each instant, one partner should be driving (using the mouse and keyboard or recording design ideas) while the other is continuously reviewing the work (thinking and making suggestions).
- Change driving/reviewing roles at least every thirty minutes, *even if one partner is much more experienced than the other.* You may want to set a timer to help you remember to switch.
- If you notice a bug in the code your partner is typing, wait until they finish the line to correct them.
- Stay focused and on-task the whole time you are working together.

- Recognize that pair programming usually takes more effort than programming alone, but it produces better code, deeper learning, and a more positive experience for the participants.
- Take a break periodically to refresh your perspective.
- Share responsibility for your project; avoid blaming either partner for challenges you run into.

MATLAB

Completing this assignment requires access to a computer that has MATLAB installed. All of the computers in the SEAS computer labs will work, or you can remotely connect to a SEAS computer lab PC by following these instructions: `http://www.seas.upenn.edu/cets/answers/virtualLab.html`

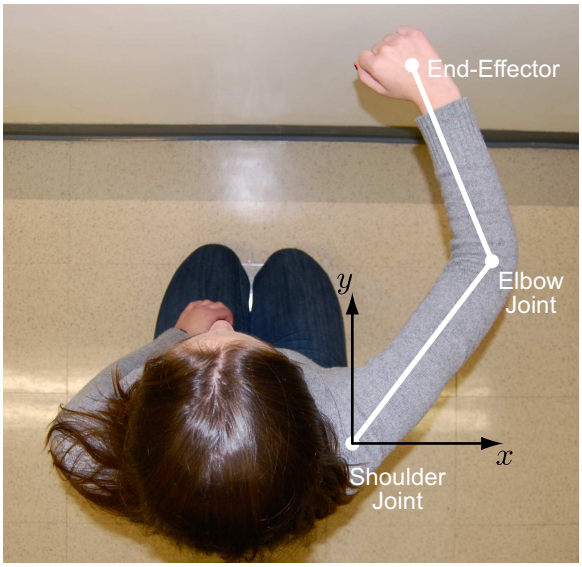
SEAS is still planning to pay for MATLAB licenses for all students in this class, but the software distribution system has not yet been set up. It will be announced on Piazza as soon as it's available.

If you haven't used MATLAB much before, you should consider attending one of the MATLAB workshops that were announced in lecture. We'll announce any additional ones once they are organized.

If you're not yet confident in your MATLAB skills, you might also want to complete the *Interactive MATLAB Tutorial* provided online by The MathWorks:
`http://www.mathworks.com/academia/student_center/tutorials/mltutorial_launchpad.html`

Plotting the Horizontal Human Arm's Reachable Workspace

Your task for this homework is to write a MATLAB script that plots the **reachable workspace of your right arm**, as seen from overhead. Recall that the reachable workspace is the entire set of points that a manipulator's end-effector can reach.



As shown in the illustration above, you should treat the human arm as a planar RR manipulator. This decision models the shoulder and the elbow both as revolute joints whose rotational axes are vertical. Treat the center of the hand as the end-effector location.

Put the origin of the coordinate frame at the shoulder joint. The x axis points out to your right, and the y axis points straight forward.

How far can your elbow flex and extend? What about your shoulder's range of motion in the horizontal plane? Examine your own arm to estimate the mechanical limits of each joint's movement. Also measure your arm to obtain the appropriate link lengths in centimeters. If you're in a team, you may use either partner's arm for this step. Write down the values you find; you'll need them in your code.

Look on pages 20 to 21 of the textbook to find equations for the x and y coordinates of the end-effector of a two-link planar RR robot as functions of the joint angles θ_1 and θ_2. Specifically, you should be looking at equations (1.1) and (1.2) and the associated diagram in Figure 1.20.

Combine all of these pieces to write a MATLAB script to plot the reachable workspace of your right arm. Here are a few additional tips and requirements:

- Start by downloading the **starter code** from Piazza. The filename is `arm_workspace_starter.m`
- Rename your file `arm_workspace_yourpennkey.m` or `arm_workspace_pennkey1_pennkey2.m` (if you're working with a partner). Your PennKey is the first part of your SEAS email address. Naming your file in this way ensures that the submissions all have unique, identifiable file names.
- Inside your script, create **sensibly named variables** to hold the values you measured above, instead of typing the numbers directly into equations. For example, the variable `elbowmin_deg` could hold the minimum angle your elbow can reach in degrees, and the variable `forearmlength_cm` could hold the length of your forearm in centimeters. Then use the name of the variable every time you need that value. This practice makes your code easier to read, debug, and update later.
- Plot a **black circle** at the location of the shoulder joint. This is already done in the starter code.
- Instead of plotting continuous regions, we recommend that you just plot a **finite set of points** to show the reachable workspace of the human arm. The starter code plots a simple set of points to show you how to do this.
- Make the plot marker **green** at locations where the end-effector can physically go and **red** at locations where the end-effector encounters a collision with your body. Use reasonable measurements for the location of your body in the workspace.
- Use units of **centimeters** in both directions.
- Note that the MATLAB functions `sin` and `cos` expect arguments in **radians**. If you want to use **degrees**, try the functions `sind` and `cosd`.
- Ensure that the view of the workspace is not distorted by calling the MATLAB function `axis equal` after you've plotted at least one thing in the figure window; this step is already done in the starter code. This function makes one unit in the x-direction be displayed at the same visual scale as one unit in the y-direction.
- For the **title** of your plot, use "Reachable Workspace by Your Name" or "Reachable Workspace by Teammate 1 and Teammate 2", inserting your full given name(s).

Submitting Your Code

Follow these instructions to submit your code:

1. Start an email to `meam520@seas.upenn.edu`
2. Make the subject *Homework 1: Your Name* or *Homework 1: Your Name and Your Teammate's Name*, replacing *Your Name* and *Your Teammate's Name* with the appropriate full names.
3. Attach your correctly named MATLAB script to the email. Please do not put it in a zip file or include any other attachments.
4. Optionally include any comments you have about this assignment and the experience of pair programming if you worked with a teammate.
5. Send the email.