ECE4560 - Homework #2

Due: Sep. 4, 2017

Consider Figure 1 below. Suppose that the configurations associated with the two reference frames depicted were $g^{\mathcal{O}}_{\mathcal{A}} = (7, 4, 45°)$ and $g^{\mathcal{O}}_{\mathcal{B}} = (2, 7, 90°)$.

Problem 1. [10 pts] What are $g^{\mathcal{O}}_{\mathcal{A}}$ and $g^{\mathcal{O}}_{\mathcal{B}}$ in homogeneous representation?

Problem 2. [10 pts] Suppose the point q_1 in $\mathcal{O}$'s frame is given by $q^{\mathcal{O}}_1 = (2, 2)^T$. Where is it in frame $\mathcal{A}$? Where is it in frame $\mathcal{B}$?

Problem 3. [10 pts] Suppose the point q_2 in $\mathcal{B}$'s frame is given by $q^{\mathcal{B}}_2 = (1, 0)^T$. Where is it in frame $\mathcal{O}$? Where is it in frame $\mathcal{A}$?

Problem 4. [10 pts] What is $g^{\mathcal{A}}_{\mathcal{B}}$ (frame $\mathcal{B}$'s configuration relative to frame $\mathcal{A}$)?

Problem 5. [10 pts] Let coordinate frame $\mathcal{C}$ have the configuration $g^{\mathcal{A}}_{\mathcal{C}} = (1, 4, -15°)$ relative to frame $\mathcal{A}$. Then what are $g^{\mathcal{O}}_{\mathcal{C}}$ and $g^{\mathcal{B}}_{\mathcal{C}}$?

Problem 6. [20 pts] Figure 2 depicts a two-link rotational, planar manipulator. It is a standard example manipulator used to convey concepts simply and will figure in class. This problem examines the geometry associated with the manipulator's hand (or end-effector) position as a function of the free joint variables. In this case, these two variables are angular variables that rotate. The two joint angles, α_1 and α_2, are depicted in Figure 2. They are free to vary, wheras the link lengths l_1 and l_2 are fixed. This problem also explores the calculus of the hand as a function of the joint variables through the Jacobian.

Let the link lengths be $l_1 = 1$ and $l_2 = \frac{1}{2}$.

(a) Given the joint angles $\alpha = (\frac{\pi}{2}, -\frac{\pi}{3})^T$, what is the end-effector position (x_e, y_e)? Provide as a function $q_e : \mathbb{R}^2 \to \mathbb{R}^2$.

Regarding the notation: the q is interpreted to be the Euclidean position and the $\cdot_e$ subscript stands for end-effector.

(b) What is the Jacobian of this function? (i.e., Dq_e?)

(b) Using the Jacobian, the joint angles from part (a), and the joint velocities $\dot{\alpha} = (-\frac{1}{5}, \frac{1}{2})^T$, what is the end-effector velocity?

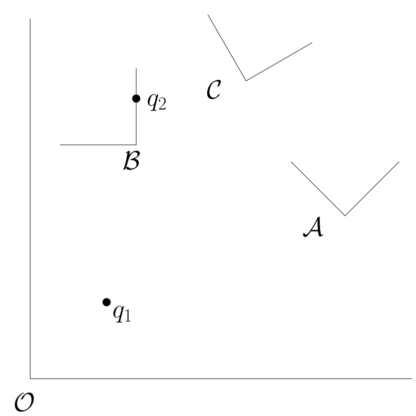

Figure 1: Coordinates frames and points.

(c) Find a joint configuration α that causes the manipulator Jacobian to no longer be invertible. Describe this joint configuration if possible.

File for Problem 6. There should be a Matlab function called `planarR2_display` in the class wiki for visualizing the manipulator, along with some instructions.

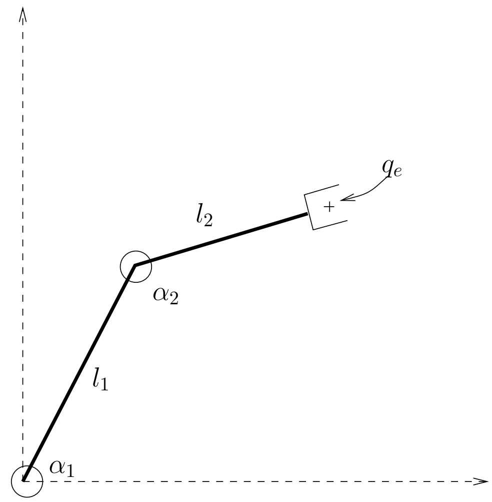

Figure 2: Planar 2R Manipulator.

Problem 7. [20 pts] Consider again the two-link manipulator of Figure 2. At some point, it will be necessary to generate a trajectory for manipulators to follow. Let's start with a simple problem. Using the code stubs on t-square, integrate the differential equation

$$\begin{aligned}\dot{\alpha}_1 &= \frac{1}{3}\cos(t) \\ \dot{\alpha}_2 &= -\frac{1}{4}\sin(t)\end{aligned}$$

given the initial joint angles $\alpha(0) = (-\frac{3\pi}{2}, \frac{\pi}{6})^T$.

(a) Plot the joint angles versus time, e.g., α_i versus t.

(b) Using the forward kinematics, plot the end-effector configuration versus time, e.g., x versus t and y versus t. You might want to write a function called `fkin2R` that computes this for you so that it can be easy to do (takes in the α and spits out the q_e). Even better would be to vectorize the function so that a matrix of multiple joint angles can be passed in, and a matrix of end-effector positions gets returned.

(c) Do a parametric plot of the end-effector position, e.g., $x(t)$ and $y(t)$ together.

You can use the `planar2R_display` function to visualize the results as an animation (the code loop is very similar to the animation code I sent for the Hilare robot in the last homework). Use the same link lengths as the earlier homework problem.

Problem 8. [10 pts] This problem relates to the earlier hammer questions (previous homework). Using the Matlab SE2 class provided on t-square, plot a visual representation of the frames $\mathcal{O}$, $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$, and the three points $q_{\mathcal{A}}$, $q_{\mathcal{B}}$, and $q_{\mathcal{C}}$ all with respect to the observer frame $\mathcal{O}$. Also add the nudged hammer position to the figure (should have been Problem 7 in the previous homework), but label it differently than the other points (see `help plot`) to figure out how to change the point label.

If you have any problems with the code, please try to make sense of the Matlab error(s) and ask your fellow classmates prior to asking me. This is my weird way of encouraging dialog between students. Also, it is always best to see me in person for code questions, as e-mail is not the best media for resolving those kinds of issues. Trust me, I can't even help people over the phone, so e-mail is even worse.

The `SE2.m` file provides the class definition you need for the plotting. Use of the `SE2` class is very similar to classes in most other languages. We will explore this code and add to it over the course of the semester. An example is:

```
g = SE2();
figure(1);
g.plot();
```

I leave the rest up to you to figure out. There should be enough documentation in the code to do so. You'll have to make use of Matlabs `hold` function to plot all of the frames and points.

File for Problem 8. Should be found on class wiki at the "Matlab Class Stubs" link/page.

Problem 9. [5 pts] Decide if you are going to

1. Follow the traditional manipulator lab track;
2. Follow the mobile robotics lab track using the Turtlebot;
3. Follow the bipedal locomotion lab track;
4. Follow a research-inspired lab track; or
5. Follow your dreams!

For the last option, please make sure to discuss with me the topic so that we can figure out how appropriate it is to the course, and that it is reasonably scoped for the semester.

For both the traditional manipulator, Turtlebot, and biped tracks, you may pair up. No triplets allowed. If you know who you'd like to work with, then please be sure to include this information in the submission. If not, then I will let everyone know what the deal is after processing the class choices. This way, it is possible to still pair up if desired.

For the alternative tracks, again it is best if you submit as part of a group.

The Jacobian. The Jacobian from multi-variable calculus is just the differential of a vector valued function. Given a function that takes in a vector of dimension n and returns a vector of dimensioon m, the Jacobian is:

$$Df = \begin{bmatrix} \frac{\partial f^1}{\partial x^1} & \cdots & \frac{\partial f^1}{\partial x^n} \\ \vdots & & \vdots \\ \frac{\partial f^m}{\partial x^1} & \cdots & \frac{\partial f^m}{\partial x^n} \end{bmatrix}$$

Recall that it acts like a directional derivative and gives the rate of change of the output given the current input value and the rate of change of the input value. Converted to English, we have that the function input and ouput,

$$\vec{y} = f(\vec{x})$$

changes under a change in $\vec{x}$ as per:

$$\Delta y = Df(\vec{x}) \cdot \Delta x$$

which is usually written more like:

$$\dot{\vec{y}} = Df(\vec{x}) \cdot \dot{\vec{x}}$$

where the dot represents a time rate of change.

Manipulator Display Functions. The manipulator display functions take in the $\vec{\alpha}$ vector of manipulator joint variables and plot the manipulator at that joint configuration. An optional second variable, specified as a row-vector, takes the link lengths. If not specified, then the link lengths assume default values.

To invoke the RR planar manipulator display function:

```
>> planarR2_display([pi/10; pi/12],  [1, 1]);
```

Without the default values the link lengths are `[1 , 1/2]`. Continuing, to invoke the RRR planar manipulator display function:

```
>> planarR3_display([pi/10; pi/12; -pi/3],  [1, 1, 1/2]);
```

Without the default values the link lengths are `[1 , 1/2, 1/4]`.