**Homework #2**
due Wednesday 10/11/23 11:59pm

This week, I want us to explore forward and inverse kinematics, workspaces, and multiplicities "by hand" using 2 and 3 DOF mechanisms. Next week we'll generalize to kinematic algorithms for more complex robots. But hopefully seeing the underlying concepts will help ground our understanding.

We will continue to ask how much time this takes, so I don't get too carried away (or at least find out when I did). Certainly ask us for help, if you don't see how to approach a problem!

**Problem 1 (Install Ubuntu 22.04 Jammy Jellyfish, ROS Humble Hawksbill) - 10 points:**

As mentioned, I would like to start installing ROS. We will not use the software yet, but getting a head-start avoids any stresses if the install doesn't go smoothly.

As such, please work on the other problems while downloading or waiting. And please don't spend more than 20 minutes debugging unexpected errors/issues. Full credit if everything works or if you make a reasonable effort, remaining stuck even after a discussion on Piazza or with the TAs in person. Instead let me know the challenges, so we can consider next steps or backup options.

You are welcome to use a different version of Ubuntu/ROS, but be warned of subtle differences, so that our examples may not work and the TAs may not be able to provide as much support.

We will install in two steps:

(a) Install Ubuntu, likely via a virtual machine (VM). Options are:
  - (i) Commercial products (VMware or Parallels). I have not personally tested.
  - (ii) VirtualBox (see instructions on Canvas). Recommended for Windows. Also runs on older Macs with Intel CPUs. It is popular and well documented, though may be slow or have issues.
  - (iii) Multipass (see instructions on Canvas). Recommended for Macs. Also runs on Windows Pro, Enterprise, or Education (requires Hyper-V). It has worked very well on my Mac.

(b) Customize the Ubuntu setup and install ROS using the instructions and script on Canvas.

(c) Test, making Atlas dance on screen!
  - (i) Download the `atlas_description.zip` file from Canvas *into the VM*.
  - (ii) Place under `~/robotws/src` and unzip there.
  - (iii) Start a new terminal (should show "`Set up ROS2 (local) for ~/robotws`") and build using: (note, this is *NOT* in `src`!)

```
cd ~/robotws
colcon build --symlink-install
source ~/robotws/install/setup.bash
```

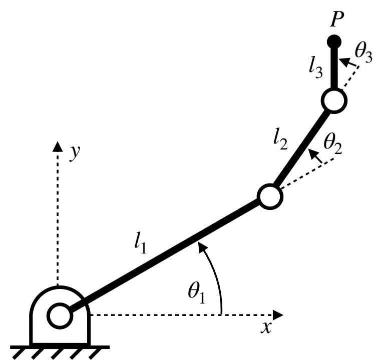
  - (iv) Run using the command

```
ros2 launch atlas_description viewatlas.launch.py
```

Please submit a screen shot of Atlas in a cool pose. And/or report any issues you encounter.

**Problem 2 (Workspace of Planar 3R Robot) - 14 points:**

We are seeking the workspace of the planar 3R robot




We will use the tip point $P$ of the last link as the output. That is, using the point $P$ in the $x, y$ plane as the task space. The link lengths are known to be

$$l_1 = 5\text{m}, \qquad l_2 = 2\text{m}, \qquad l_3 = 1\text{m}$$

Please draw (carefully, to scale) the workspace and label the dimensions for the following two cases, assuming the joints can move through the ranges:

(a) $0 \leq \theta_1 \leq \pi$ $\quad \theta_2$ unlimited $\quad \theta_3$ unlimited

(b) $0 \leq \theta_1 \leq \pi$ $\quad 0 \leq \theta_2 \leq \pi$ $\quad 0 \leq \theta_3 \leq \pi$

**Problem 3 (Tip Motion of Planar 3R Robot) - 14 points:**

Consider again a planar 3R robot, but this time with more "extreme" link lengths (with link 1 being much longer than links 2 and 3):

$$l_1 = 6\text{m}, \qquad l_2 = 1\text{m}, \qquad l_3 = 0.5\text{m}$$

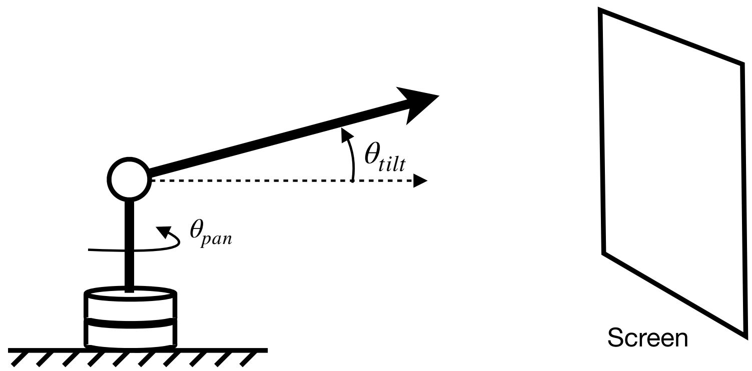
The joint angles are programmed to vary as a function of time, with

$$\begin{aligned}
\theta_1(t) &= \omega_1 t && \text{with} \quad \omega_1 = 1\text{rad/sec} \\
\theta_2(t) &= \omega_2 t + \pi && \text{with} \quad \omega_2 = -4\text{rad/sec} \\
\theta_3(t) &= \omega_3 t + \pi && \text{with} \quad \omega_3 = 12\text{rad/sec}
\end{aligned}$$

Please graph the path $y(t)$ versus $x(t)$ that the tip point $P$ of the last link traces in the $x, y$ plane. You will likely want to compute this numerically, e.g. using Desmos, Matlab, or Mathematica.

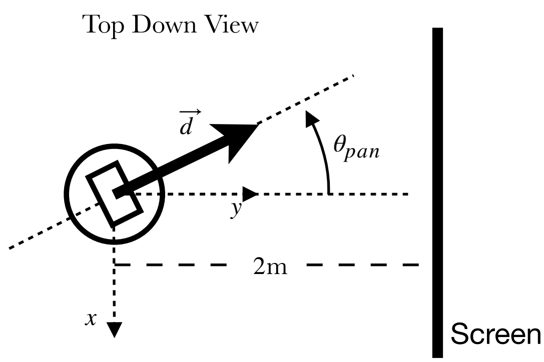
**Problem 4 (Laser Pointer on 2DOF Pan/Tilt Gimbal) - 20 points:**

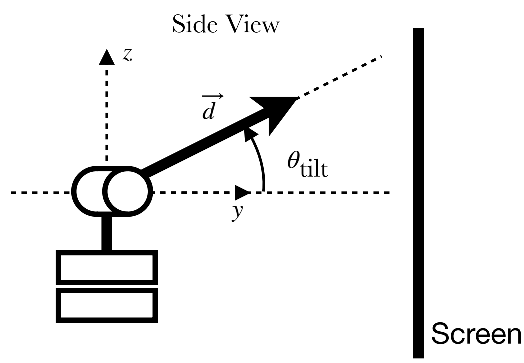
You find a laser pointer mounted on a pan/tilt gimbal, in front of and facing a flat vertical screen.




The pan $\theta_{\text{pan}}$ and tilt $\theta_{\text{tilt}}$ angles are defined such that for zero values the laser aims horizontally and perpendicularly at the origin/center of the screen. The screen is 2m in front of the gimbal.

If you prefer, here are a top-down and side view, keeping in mind that in the side view, the laser has panned into the plane.



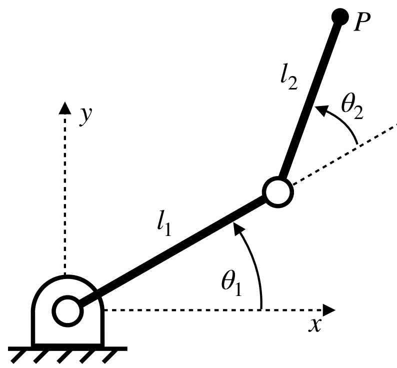

When you start the gimbal mechanism, the angles move back and forth over time as

$$\theta_{\text{pan}}(t) = \frac{\pi}{3}\sin(2t)$$
$$\theta_{\text{tilt}}(t) = \frac{\pi}{3}\sin(1t) - \frac{\pi}{9}\cos(6t)$$

Please graph the picture that the laser pointer draws on the flat screen. You will again most likely want to compute this numerically, e.g. using Desmos, Matlab, or Mathematica.

**Problem 5 (Joint Velocities near Singularities) - 16 points:**

Recall the basic planar 2R robot from class




We will use the standard output space, being the tip position $P$ of the second link as the output. That is, using the point $P$ in the $(x, y)$ plane as the task space. The link lengths are equal and known to be

$$l_1 = l_2 = 1\text{m}$$

Further, assume the tip is moving only up/down ($x$ being constant at $d$), so that P has the following motion:

$$x(t) = d \qquad \dot{x}(t) = 0\text{m/s}$$
$$y(t) = \cos(t) \cdot 1\text{m} \qquad \dot{y}(t) = -\sin(t) \cdot 1\text{m/s}$$

With $d = 0.05$m, for one cycle ($t = 0 \ldots 2\pi$ seconds), numerically compute and plot versus time:

(a) The joint positions (angles) $\theta_1(t)$ and $\theta_2(t)$ for only the "elbow up" solution, i.e. with $\theta_2 < 0$. This should use the inverse kinematic formulas from class.

(b) The joint velocities $\dot{\theta}_1(t)$ and $\dot{\theta}_2(t)$, numerically differentiating the above $\theta_1(t), \theta_2(t)$.

Consider two specific/special configurations that occur along this motion:

(c) What is the maximal value for $\theta_1$? What is the corresponding $\theta_2$ that occurs at the same moment in time?

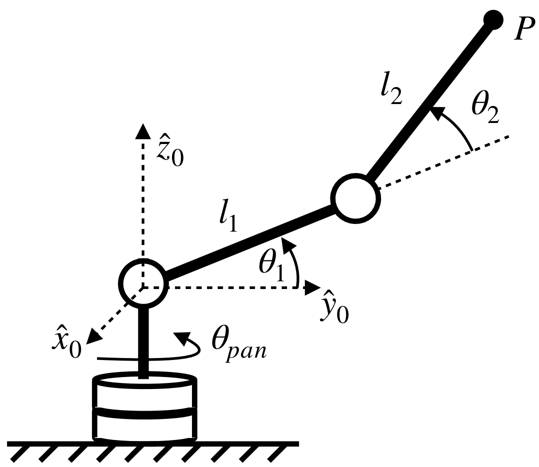
Please draw the robot in this configuration. Geometrically, what characterizes this configuration? Understanding this geometry, please give the exact formulas for $\theta_1$ and $\theta_2$ in this configuration, dependent only on $d$.

(d) What is the value of $\theta_1$ and $\theta_2$ when $y = 0$?

Please draw the robot in this configuration. And, again from geometry, please give the exact formulas for $\theta_1$ and $\theta_2$ in this configuration, again dependent only on $d$.

**Problem 6 (Tip Position of 3DOF Robot) - 22 points:**

Let us consider the classic 3DOF robot configuration (I admit, I've programmed a robot like this a few times, with or without a wrist at the end for additional orientation)




We are interested in the $x, y, z$ coordinates of the tip point $P$ with respect to the fixed frame $O$. Assume the lengths $l_1$ and $l_2$ are known and a zero pan angle $\theta_{pan}$ matches the arm in the $y, z$ plane.

(a) Please derive the forward kinematics and give the tip position $x, y, z$ as a function of the joint angles $\theta_{pan}, \theta_1, \theta_2$.

(b) How many and which multiplicities do you find? I.e. how many different ways can the arm reach any particular position.

(c) Determine the inverse kinematics, that is please give the three joint angles as a function of the tip position.

Hint: Not surprisingly, this combines ideas/steps from the two 2DOF mechanisms (planar and gimbal) in the handout. Indeed, I'm using the rather unorthodox joint names $\theta_{pan}, \theta_1, \theta_2$ (rather than $\theta_1, \theta_2, \theta_3$) to spotlight that.

**Problem 7 (Time Spent) - 4 points:**

To continue to help us gauge things, please tell us approximately how much time you spent on this homework. Any particular bottlenecks?