Scripting Task

Here is a little task to see your ability to think out a system problem. This is something I worked on in the past.

We want to build an automated version of our SatPaq, one where there is no user to point the device to the satellites. The device will need to align itself. BUT, compasses are unreliable - if the device gets put on a metal shipping container, the compass will be affected and may report inaccurate heading. So your task is to write a compass calibration script.

## Tiny background:

The SatPaq connects with geostationary satellites, which are at 0 latitude and a fixed longitude. Given a user’s location, there is a best azimuth and elevation to point to the satellite. Here we’ll disregard the elevation, just think about the azimuth. We can use the signal strength of our SatPaq to hone in on the true satellite heading, and use this to calibrate the reported compass heading. You’ll use python or another scripting language that you’re comfortable with writing to simulate a calibration: as quickly as possible find the calibration heading.

Here’s the plan: I have attached two plots of our device measuring signal strength over 180 degrees. You’ll notice that the device has an antenna pattern of 70 degrees or so to the 3 dB points. The peak point is the azimuth angle to that satellite.

With one satellite you could use the known longitude of the satellite and the peak plot angle to determine what direction the SatPaq is looking. Since we communicate with two satellites, we might be able to get a more refined answer. These satellites are at W95 deg and [W129](https://www.satbeams.com/satellites?norad=27715) degs.

Given infinite time, we could do this easily. We would finely step the stepper motor along the 180 degrees and get a detailed plot, as we did for these plots. But we want the calibration script to run quick. Each measurement of signal strength takes 0.4 seconds. Instead of a linear search, we want a better search, perhaps a binary search or similar. That way we quickly and coarsely find the good area and then precisely find the peak.

## Tasks:

Write a script to take in the data of both satellites. Make a plot of the power vs. antenna angle for both. Then build a function to “measure” these values at K points (angles). Start with a random value (0-180) position of the sensor. Circle-shift the patterns by this angle amount to simulate the SatPaq being at an unknown position. Then ‘find’ the offset from the peak of the satellite plot to the known satellite longitude. Do these for both satellites. Come up with an estimate for the SatPaq starting position. Do this 25 times and plot the true and estimate positions.

The function should also output how fast the process is. Remember we want to quickly point the SatPaq to the satellite. Assume a measurement takes 0.4 seconds. Pick a reasonable number for stepper motor rotation speed. Clearly outline any other assumptions you make, so we can talk about them later.

You should build a function for the trivial search case: sequentially measure every two degrees, with Q measurements at each angle position. Say you take 5 measurements, then 90\*5\*.4 = 180 seconds. What kind of trade off would lowering the number of points in the average create? Decide if it is worth it. What else should we be thinking about when we go to implement this routine? How much longer would it take to use measurements of both satellites? How much more accurate could the heading estimate be? Is it worth it?

Think about what could go wrong taking this theoretical approach to an actual platform? How can you anticipate and handle it?

You might plot out the time it takes for a couple of different strategies you come up with.

When you finish, upload it to a personal github and we’ll go over it when we speak.

## Rules:

1. Don’t spend more than 5-10 hours on this.
2. You can use the internet
3. You can ask us for help, but you can also just make assumptions to keep moving forward. Oftentimes a model with clear assumptions is all we can achieve.

Data

[G12](https://drive.google.com/open?id=1LzomASuQBXYtP0_8T1VEx8KwA1b2wEmD)

[G3C](https://drive.google.com/open?id=1nNRr-P-ek3YnXNlFHI5_U8mYsgiYRsK6)