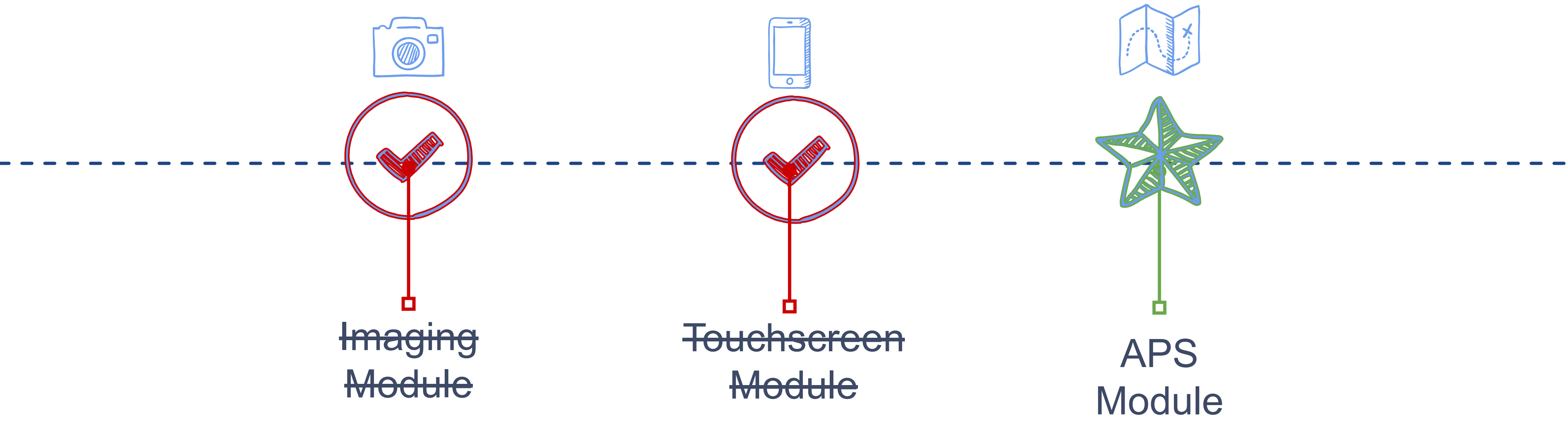
EECS16A
Acoustic Positioning System 1

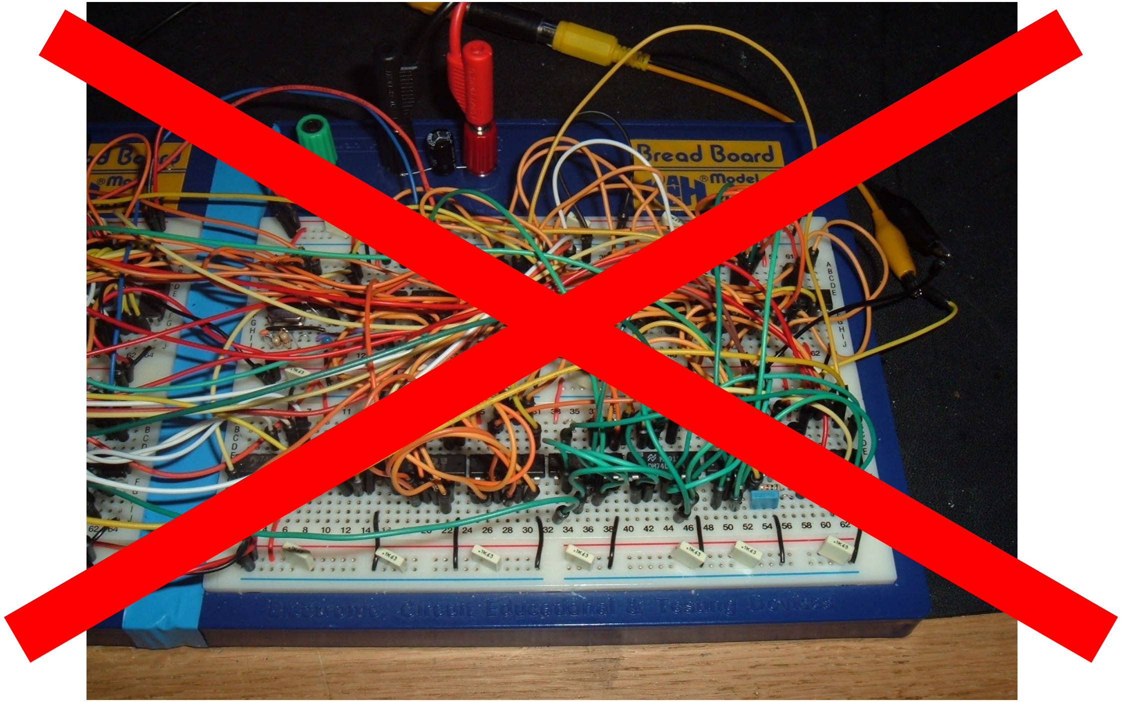
Insert your names here

Where Are We Now?



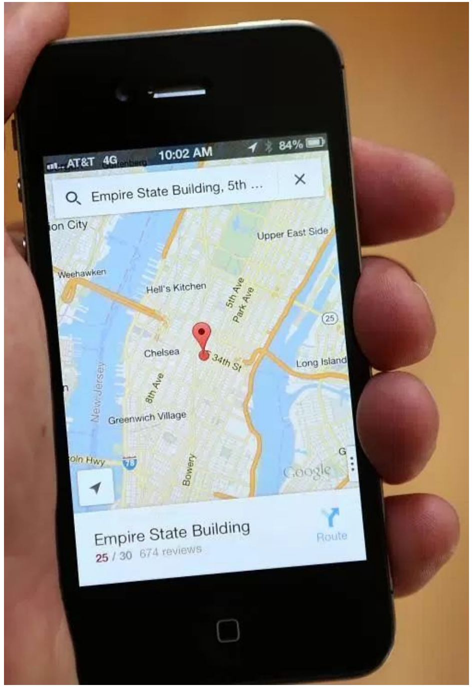
Announcements

- All software

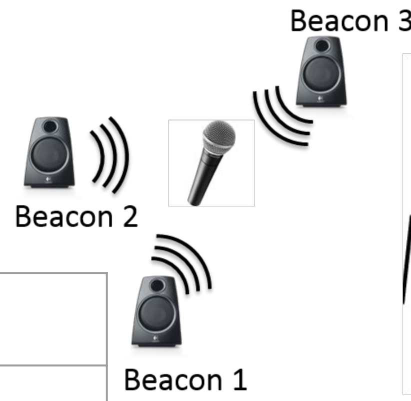


Today’s Lab: Acoustic Positioning System

- Global Positioning System (GPS)
  - Uses radio waves instead of sound waves
- Understand mathematical tools used for shifting and detecting signals
  - Think about cross correlation!



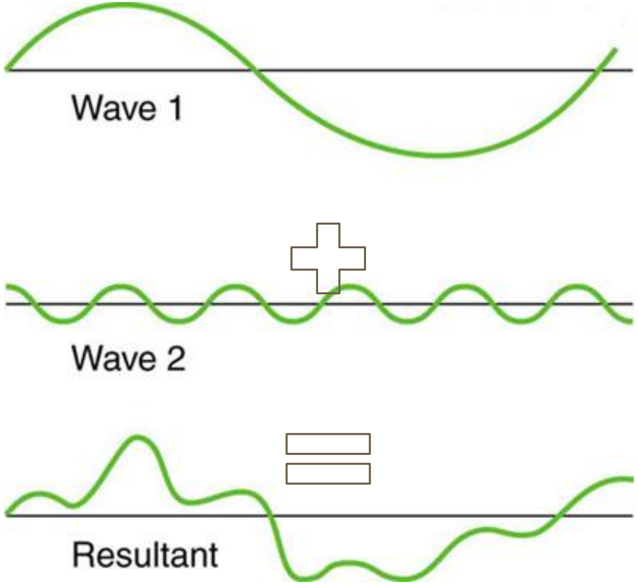
Set-up



General	Lab Specific
receiver	microphone
Satellites repeatedly transmitting specific beacon signals	Speakers repeatedly playing specific tones (beacon signals)

- Known: Location of each satellite and what beacon signal each satellite is playing
- Unknown: Location of receiver ← what we want to figure out!

Set-up

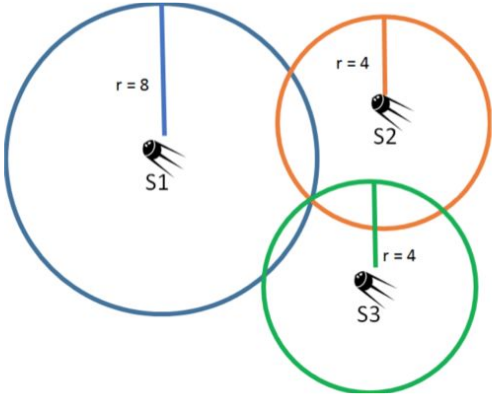


- Satellite:
  - Known, periodic waveforms
  - Know satellite location
- Receiver:
  - Will record the waveform
    - Sum of all shifted beacons
  - Waveform will be shifted from known satellite waveform based on how far it is from satellite (sound takes time to travel)

Let’s go backwards

Assume we know the **distance** between the receiver and every satellite

- Use **lateration** and the satellites’ locations to locate the receiver!
- How many satellites do we need in a 2D world?



How do we get those distances?

Assume we know the **time-delay** (in secs) of every beacon

- Use the **speed of sound** through air to get exactly how far our receiver is from every satellite
  - $d = v_s \cdot t$
  - $v_s \approx 343$ m/s

How do we get those time-delays?

Assume we know how many **samples** it takes for each beacon signal to arrive at the receiver

- Use the **sampling frequency** of receiver to get the **time-delay**
  - Sampling frequency [samples/sec] - rate at which microphone records samples

$$\frac{\text{samples}}{f_s} = \frac{\text{samples}}{\frac{samples}{second}} = \text{seconds}$$

Poll Time!

Let the sampling frequency be 1000 Hz and the speed of sound be 343 m/s. If I detect a signal with a delay of 100 samples, what is the distance between the speaker and the mic?

- 3430 m
- 34.3 m
- 343 m
- 3.43 m

Poll Time!

Let the sampling frequency be 1000 Hz and the speed of sound be 343 m/s. If I detect a signal with a delay of 100 samples, what is the distance between the speaker and the mic?

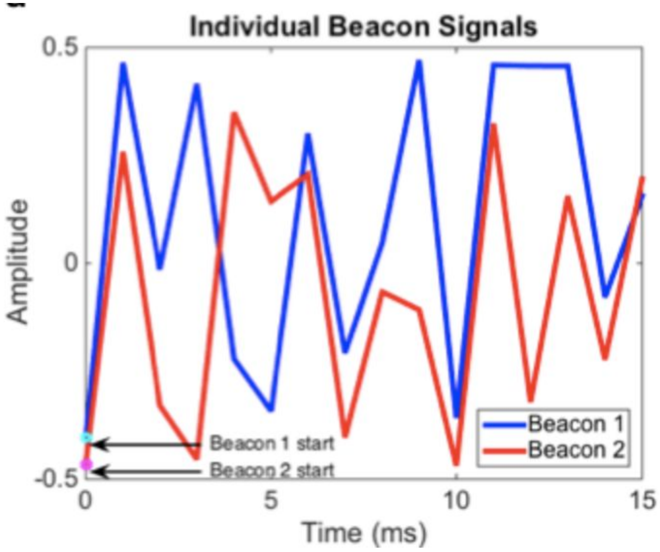
- 3430 m
- **34.3 m** →
- 343 m
- 3.43 m

$$\text{time delay} = \frac{\text{samples}}{\text{sampling frequency}} = \frac{100\ \text{samples}}{1000\ \text{Hz}} = 0.1\ \text{s}$$

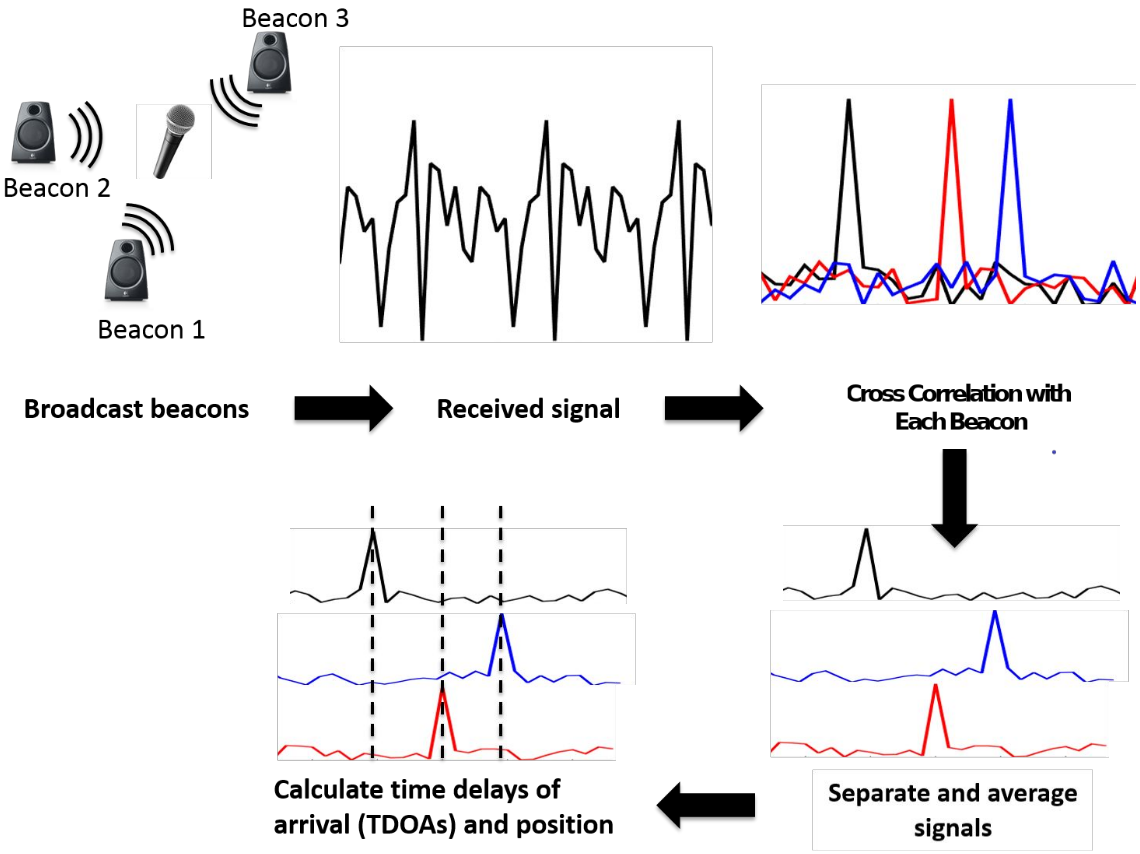
$$\text{d} = \text{v} \cdot \text{t} = 343\ \text{m/s} \cdot 0.1\ \text{s} = \boxed{34.3\ \text{m}}$$

How do we get sample delays?

- Receiver's recorded signal is the sum of all the beacon signals
- Need to separate the recorded signal into the individual beacon signals to see how many samples each signal is delayed by



Overview


Beacon 3
Beacon 2
Beacon 1
Broadcast beacons
Received signal
Cross Correlation with
Each Beacon
Separate and average
signals
Calculate time delays of
arrival (TDOAs) and position

Recall: Inner (Dot) product

- Computes how similar two vectors are

$$\langle \vec{x}, \vec{y} \rangle \equiv \vec{x} \cdot \vec{y} \equiv \vec{x}^T \vec{y}$$

$$= \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$= x_1 y_1 + x_2 y_2 + \cdots + x_n y_n$$

$$= \sum_{i=1}^{n} x_i y_i$$

Recall: Inner (Dot) product

$$\langle \vec{x}, \vec{y} \rangle = \|x\| \, \|y\| \cos \theta$$

An alternate form of the dot product

- **Given this expression, with ||x|| = ||y||, when is this expression maximized?**
    - θ = 0
    - vectors point in the SAME DIRECTION, so they are the SAME SIGNAL

The bigger the dot product magnitude, the more “similar” the two vectors are

Tool: Cross-correlation

$$corr_r(B_A)[k] = \sum_{i=-\infty}^{\infty} r[i]B_A[i-k] \Leftrightarrow$$

In Python:
`cross_correlation(r, B_A)[k]`

- Mathematical tool for finding similarities between signals
- **Idea**: Computes dot product between r and signal B_A shifted by k samples

- From the previous slide, the peak of the cross-correlation vector tells us which shift amount makes B_A "most similar" to r

Poll Time!

Given the inner product expression, with ||x|| = ||y||, when is the magnitude of this expression maximized?

- theta = 0
- theta = 90
- theta = 180
- theta = -90

Poll Time!

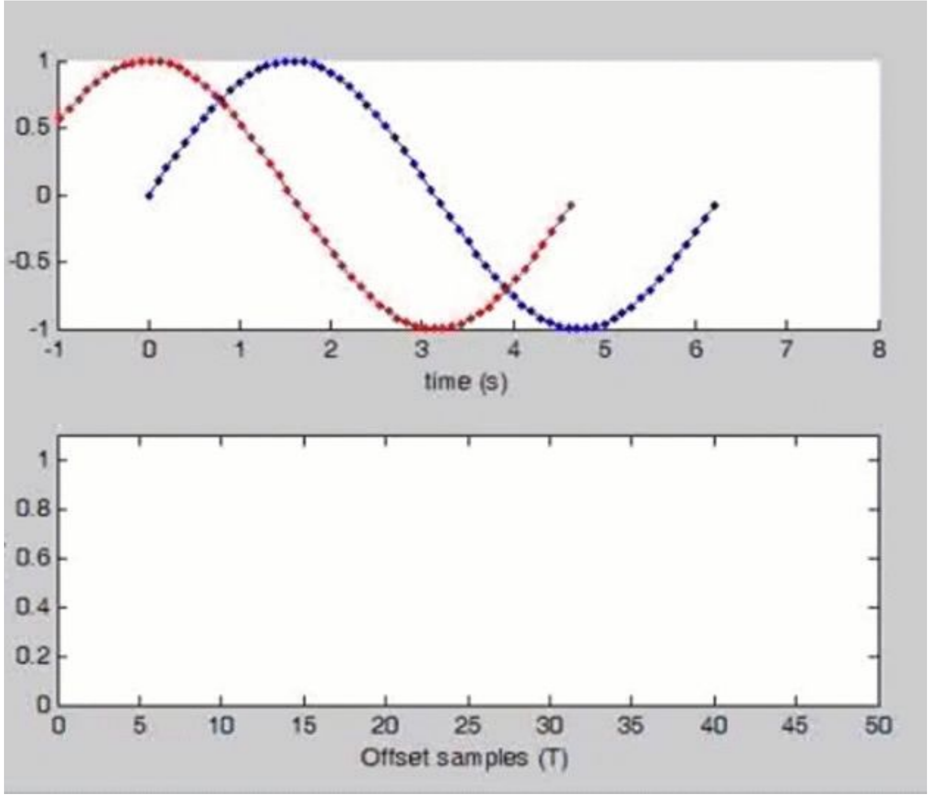
Given the inner product expression, with ||x|| = ||y||, when is this expression maximized?

- **theta = 0**
- theta = 90
- **theta = 180**
- theta = -90

Tool: Cross-correlation

- **At ~ how many offset samples are the signals most similar?**

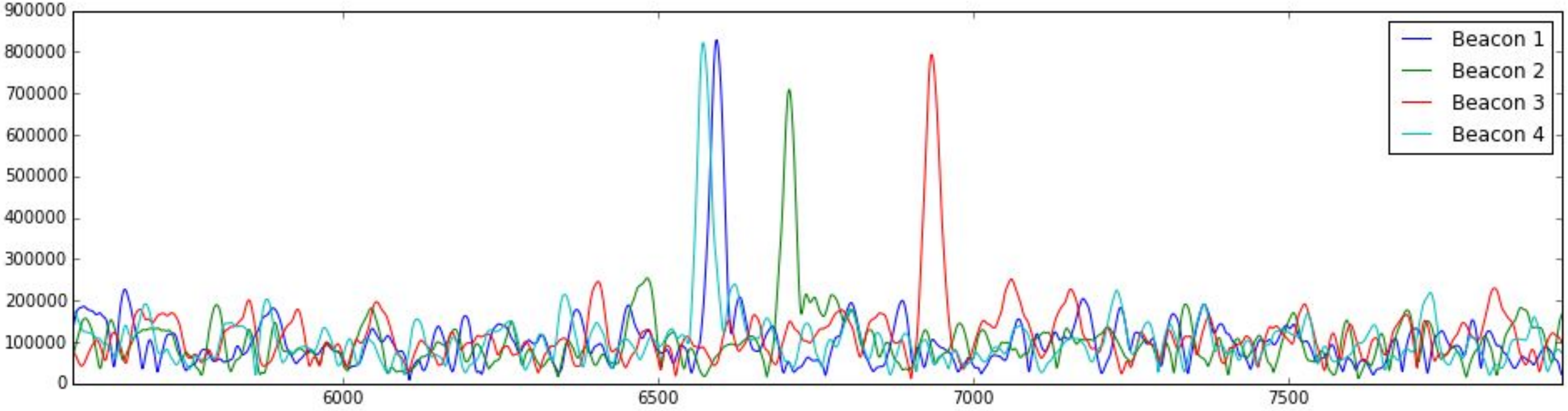
blue = r
red = B_A



Note: zero pad signals to match length

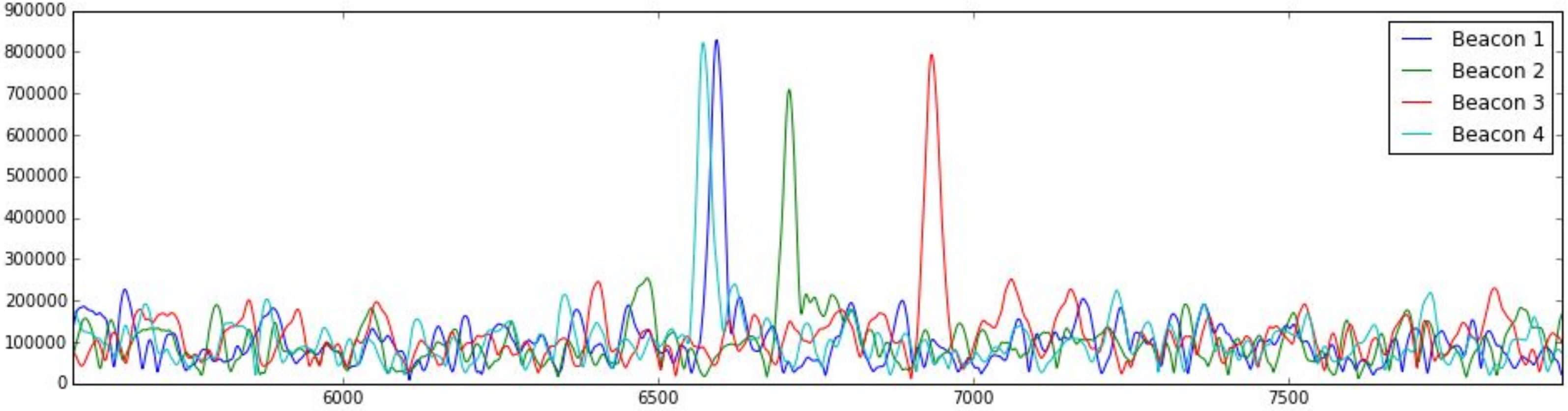
How to use?

- Cross correlating should tell us where each beacon signal arrived in our recorded signal
- Let's cross-correlate each of the known beacon signals with what we recorded and plot the result



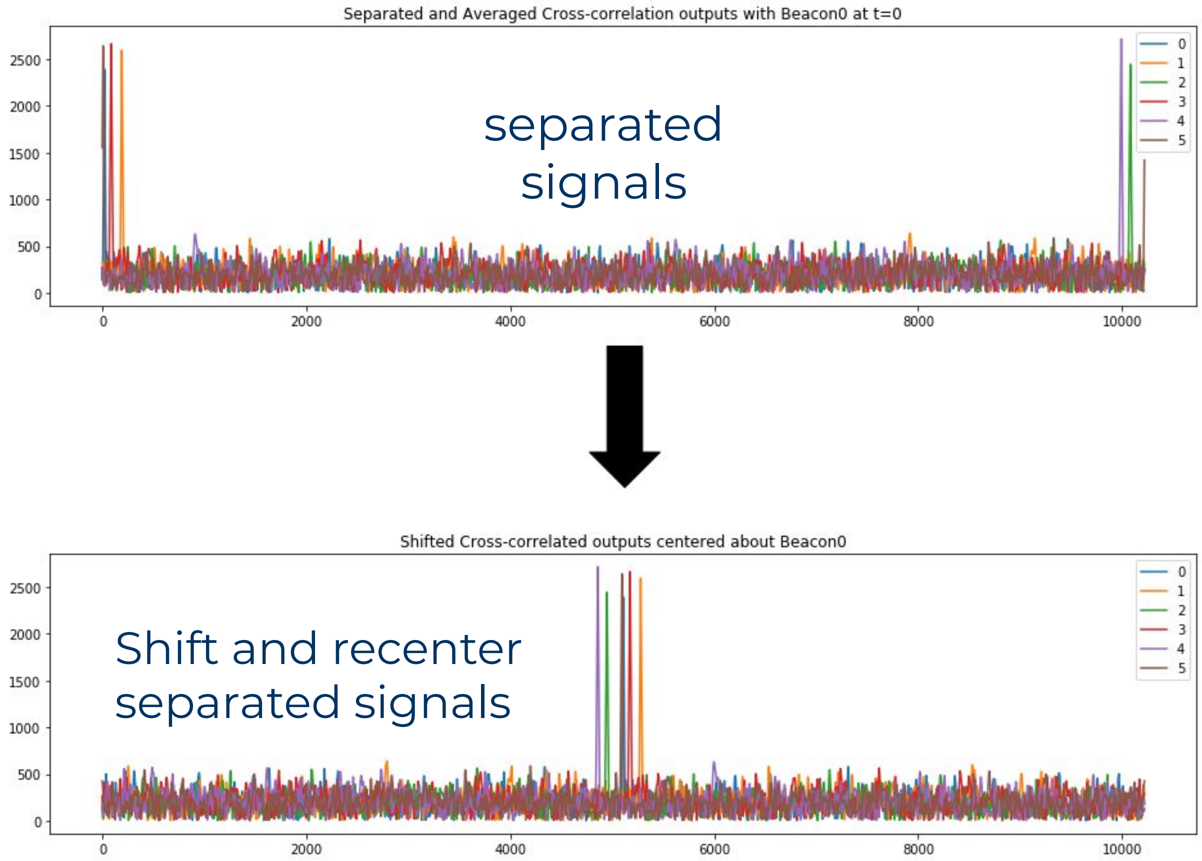
Absolute or relative sample delays?

- We can see peaks where each beacon signal arrived!
- But notice it only gives us **relative** sample delays
  - Still can't tell how many absolute samples each beacon is delayed, we don't know when it was supposed to arrive
- Arbitrarily pick a beacon to be the reference point



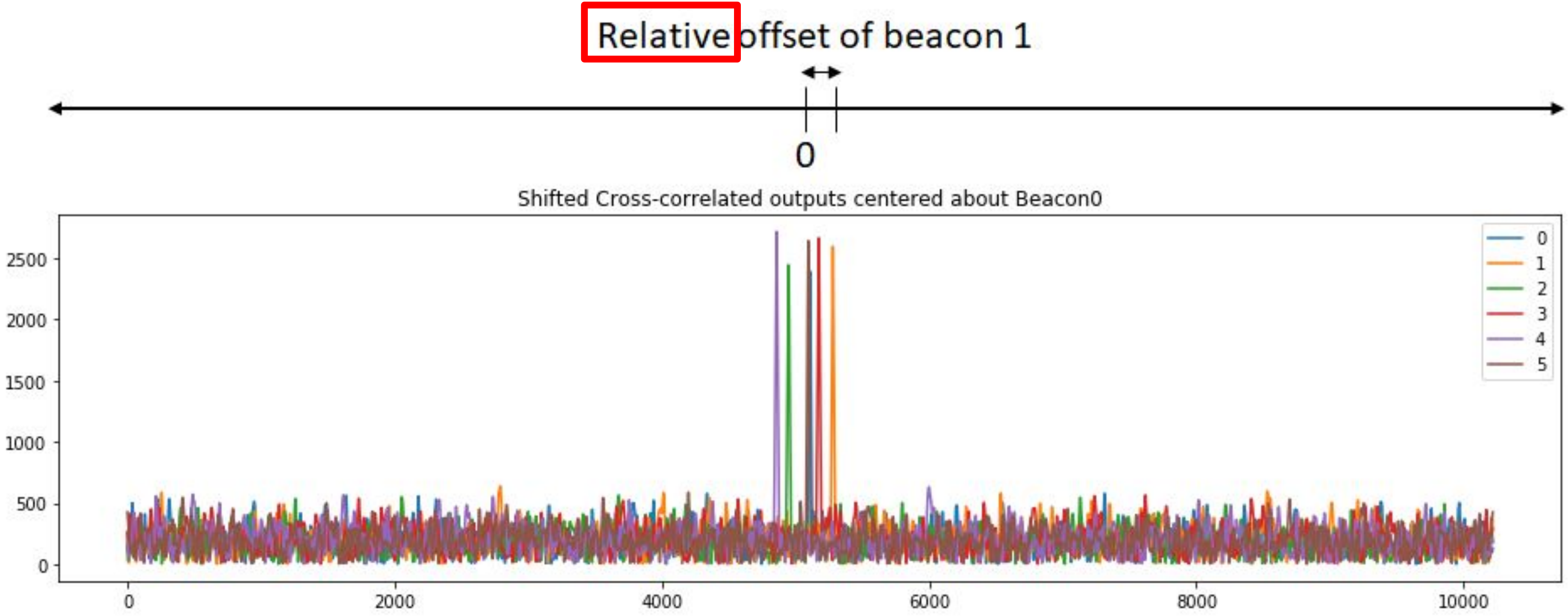
“Sacrificing” a beacon

- Let’s shift our axis so beacon 0 has a delay of 0
- We could pick any beacon to be the center
    - 0 is arbitrary



“Sacrificing” a beacon

Now beacon 0 is at our new “origin” and all computations are relative to the new “0”



Relative Measurements

- Now, we are able to compute **relative** sample delays, then **relative** time delays
- How do we get from **relative** time delays to **absolute** distances?
    - With the current set-up, we can't :(

Additional assumption for APS 1

- What if we knew the absolute sample delay of beacon 0?
  - Now we can adjust all our relative measurements to absolute ones!
  - Assume delay_0 is given, then

$$\text{delay}_i = \text{delay}_{i \text{ relative to } 0} + \text{delay}_0$$

- Then we can use absolute time-delays to get distances, then location!

Notes + Next Lab:

- If we know the absolute sample delay of beacon 0, we can locate the receiver
    - Note that this the same as telling you exactly how far the receiver is from satellite 0
- This week, this value will be given to you
- Find out how to get around this assumption in APS 2!
- Some plots are laggy; don't overwhelm them by interacting too much at once. Please wait for them to render before changing inputs.

FOR TODAY, FOLLOW THESE INSTRUCTIONS IF IN-PERSON

- Download and extract the zip file
- Open Anaconda Powershell Prompt and run "jupyter notebook"
- Navigate to the lab folder and proceed as usual