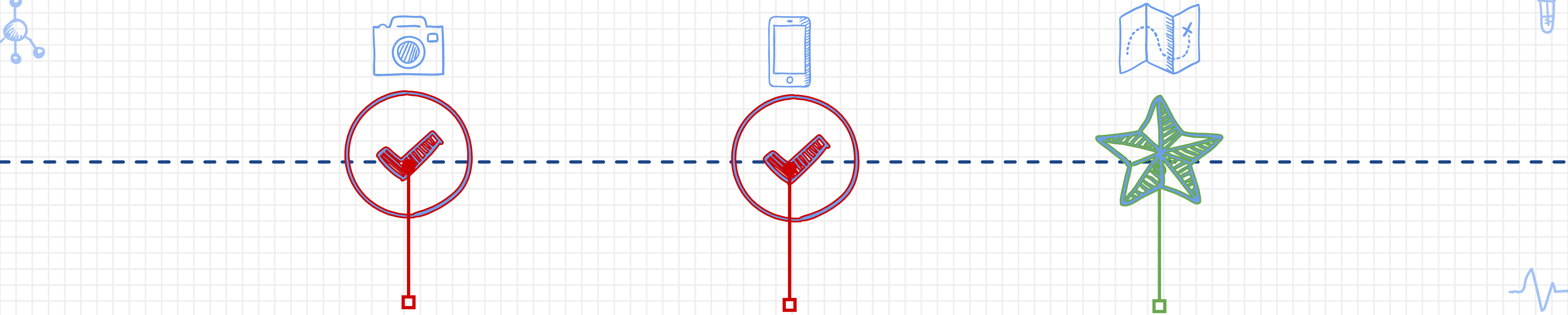
EECS16A
Acoustic Positioning System 1

TA, ASE, ASE, ASE

Where Are We Now?



~~Imaging Module~~

~~Touchscreen Module~~

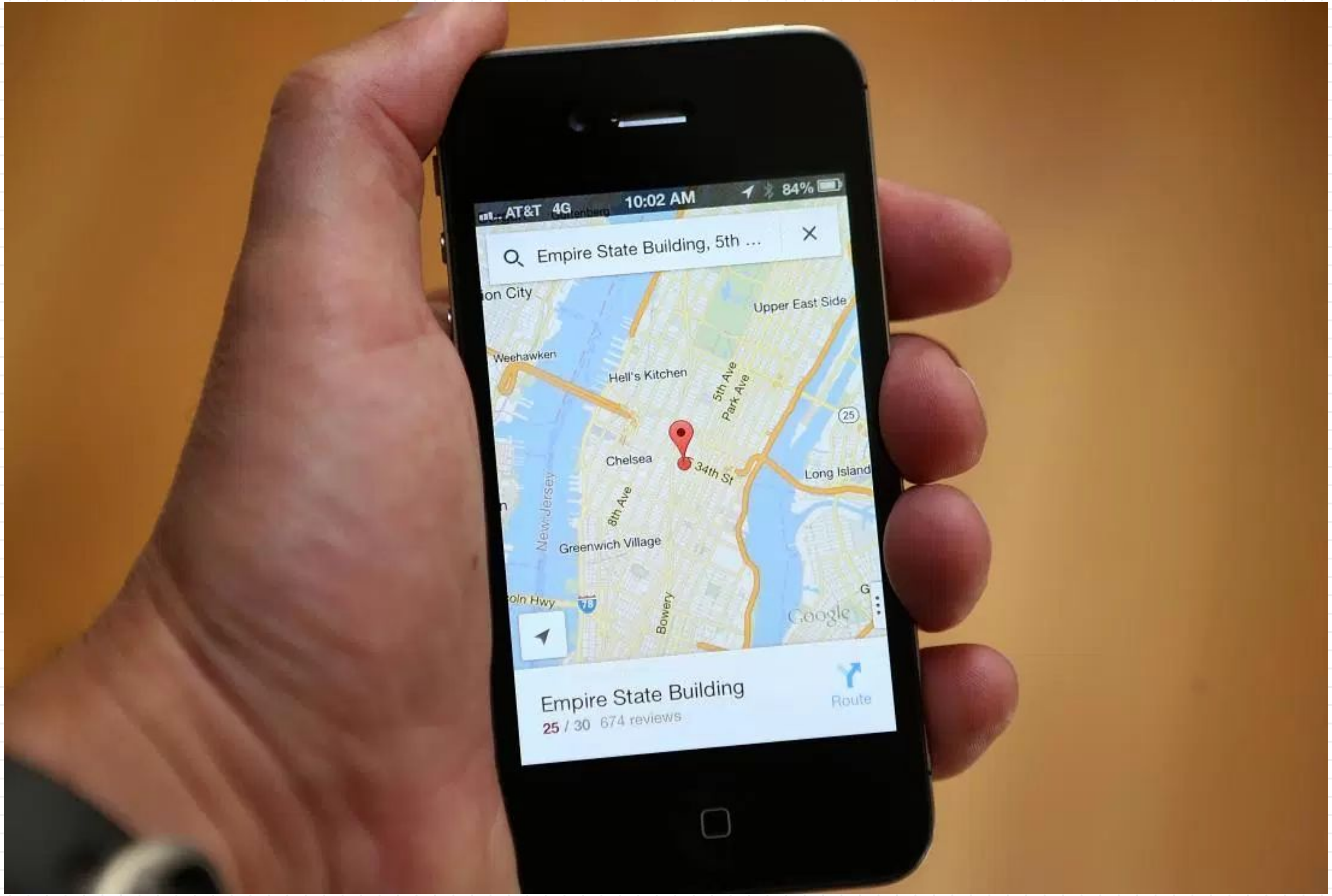
APS Module

Announcements

- All software; no hardware involved
    - Can use your own laptop

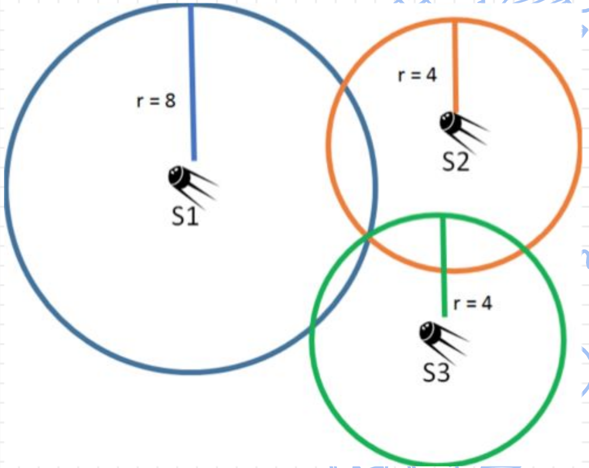
Today's lab: Acoustic positioning system

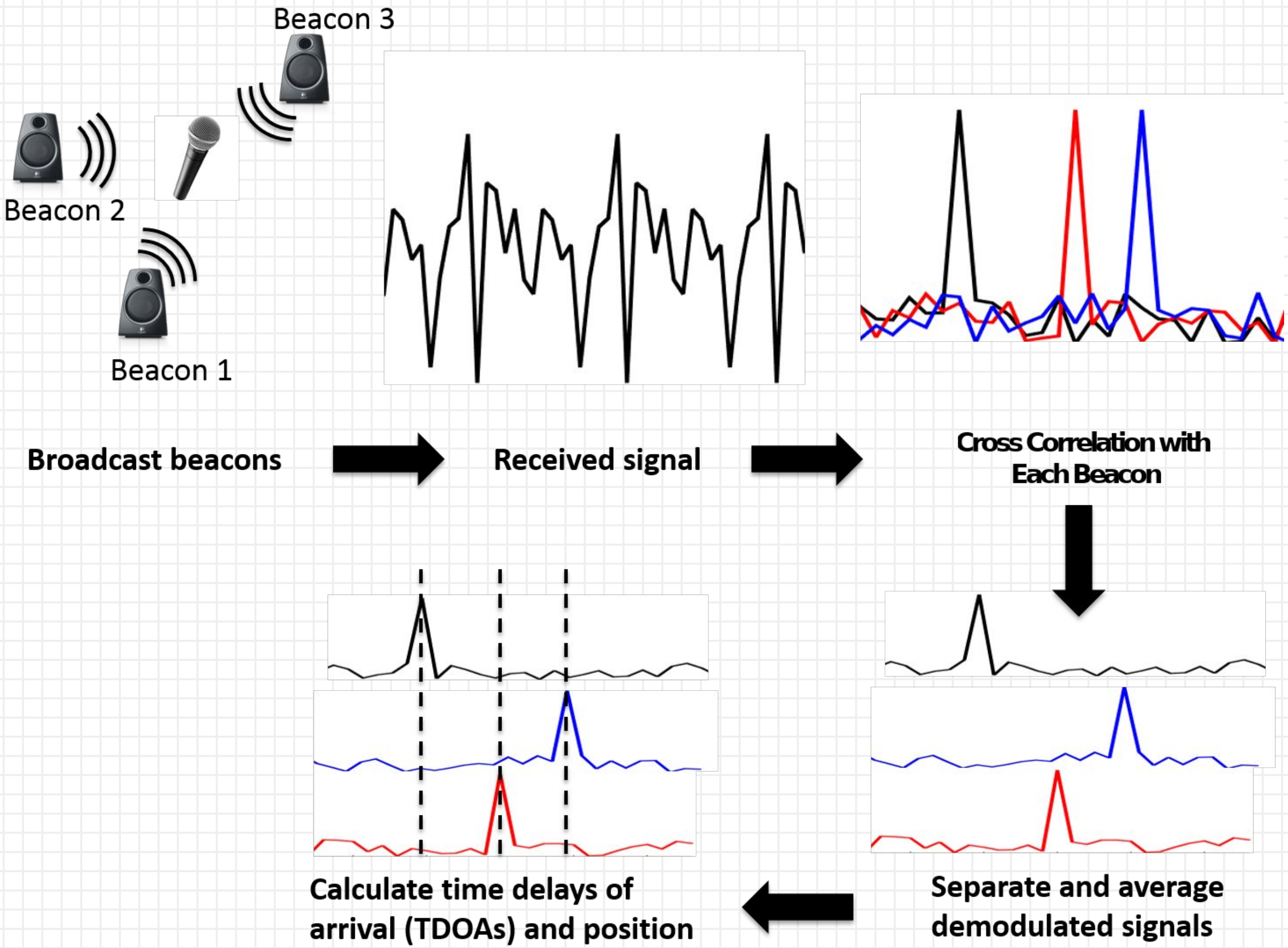
- Global Positioning System (GPS)
    - Basically the same thing
    - Uses radio waves instead of sound waves
- Understand mathematical tools used for sifting and detecting signals
    - Think about cross correlation!


AT&T 4G
10:02 AM
84%
Empire State Building, 5th ...
Upper East Side
Weehawken
Hell's Kitchen
5th Ave
Park Ave
Chelsea
34th St
Long Island
8th Ave
New Jersey
Greenwich Village
Bowery
Google
Empire State Building
25 / 30 674 reviews
Route

GPS?

- Satellites send signals at known times (beacons are synchronized)
  - But we aren't synchronized to the beacons
- Receiver (us) gets these signals
- From **time-delay** of a beacon signal, receiver calculates **distance** to the beacon
- From **distances** to satellites, **position** is determined by **lateration**
- **How many beacons do you need to determine your location in 2D?**




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

Time of flight

- Receiver gets signals from multiple satellites at the same time
  - Each is a known beacon/waveform
  - Periodic
  - We also know where the satellites are
- The receiver then determines when each beacon is received, with reference to when other beacons are received
  - Harder than it seems! Why?

Problem

- Our antenna receives all the signals at once
  - We have to separate out the useful information
- We have no clue when the satellites sent their signals
  - Signals repeat every 230ms
  - Because of this we can’t use the start of the recording as a reference
- Even if we can separate the info, we can’t just wait until we receive something, because we don’t know when it was sent
  - This week we will cheat a little bit, next time we’ll see how to really handle it

Recall: Inner (Dot) product

- A mathematical operation for vectors
- One way to think about it is that it computes how similar two vectors are

$$
\begin{aligned}
\langle \vec{x}, \vec{y} \rangle \equiv \vec{x} \cdot \vec{y} \equiv \vec{x}^T \vec{y} \\
= \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \\
= x_1 y_1 + x_2 y_2 + \cdots + x_n y_n \\
= \sum_{i=1}^{n} x_i y_i
\end{aligned}
$$

Recall: Inner (Dot) product

- A mathematical operation for vectors
- One way to think about it is that it computes how similar two vectors are

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

An alternate form of the dot product

- **Given this expression, and assuming ||x|| = ||y|| = 1, when is this expression maximum?**

The value is maximized when theta = 0
This is when the vectors point in the **SAME DIRECTION,** which is to say, the vectors are the **SAME SIGNAL**

Thus the bigger the dot product, the more "similar" the two vectors are

Tool: Cross-correlation

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

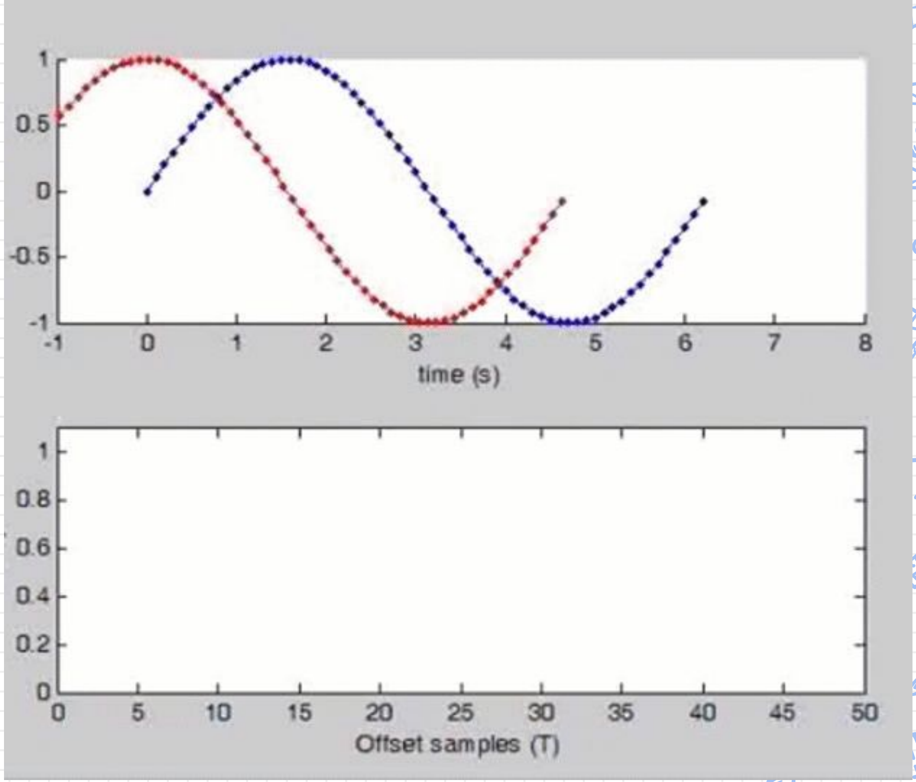
In Python:

`cross_correlation(r, B_A)`

- Mathematical tool for finding similarities between signals
- **Idea**: Take B_A and slide over *r*, compute dot product, slide again
  - Gets plotted with the <u>shift amount</u>
- From the previous slide, <u>peak</u> of cross-correlation tells us which shift amount makes B_A “most similar” to *r*

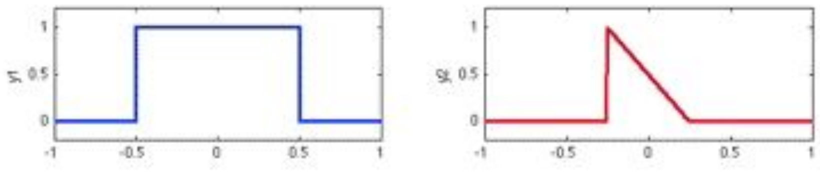
Tool: Cross-correlation

- "Sliding Dot Product"?
- Helps us find a specific signal midst a mix of many signals
  - Dot product computes similarity
  - Sliding dot product tells us how similar two signals are for a given shift amount (see gif)
- Use it to decode ambiguous texts from your crush
- **At how many offset samples is the signal most similar?**



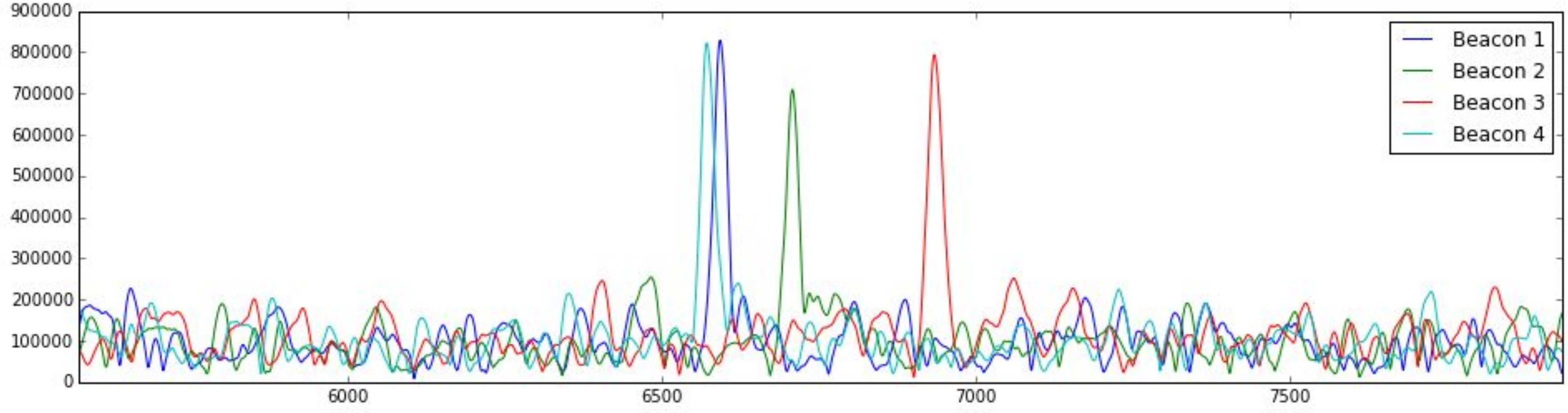
How to use?

- Cross correlating should tell us where our beacons arrived in our signal
- From there we can try to find a way to compute the time delays
    - Then we can find the distances!



Solution attempt

- Let's cross-correlate each of the known beacon signals with what we recorded and plot the result
  - What do you expect to see?

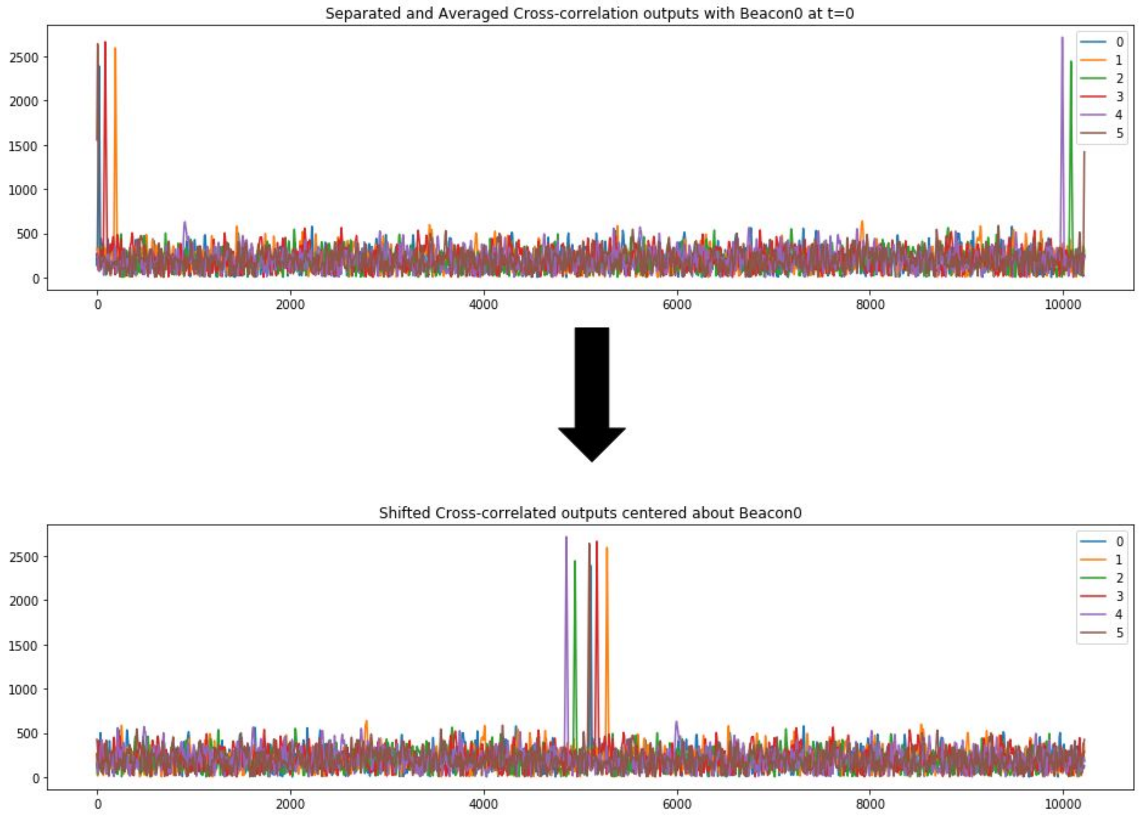


Ok, what now?

- Great! We can clearly see where each signal is in our received waveform
- Unfortunately we're still not quite there… This doesn't tell us much
- Idea: we don't know when the beacons arrived, but based off of the offsets we know how much longer it took for beacon 1 to arrive RELATIVE to beacon 0!
- Let's shift our axis so beacon 0 is at 0
    - We could pick any beacon to be the center. 0 is arbitrary

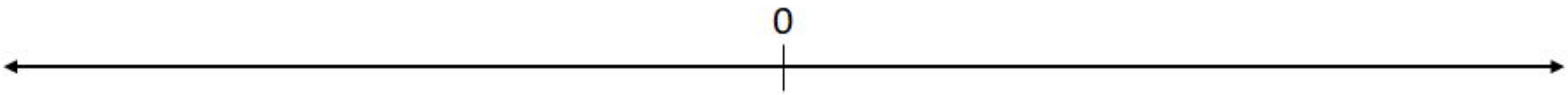
New axis

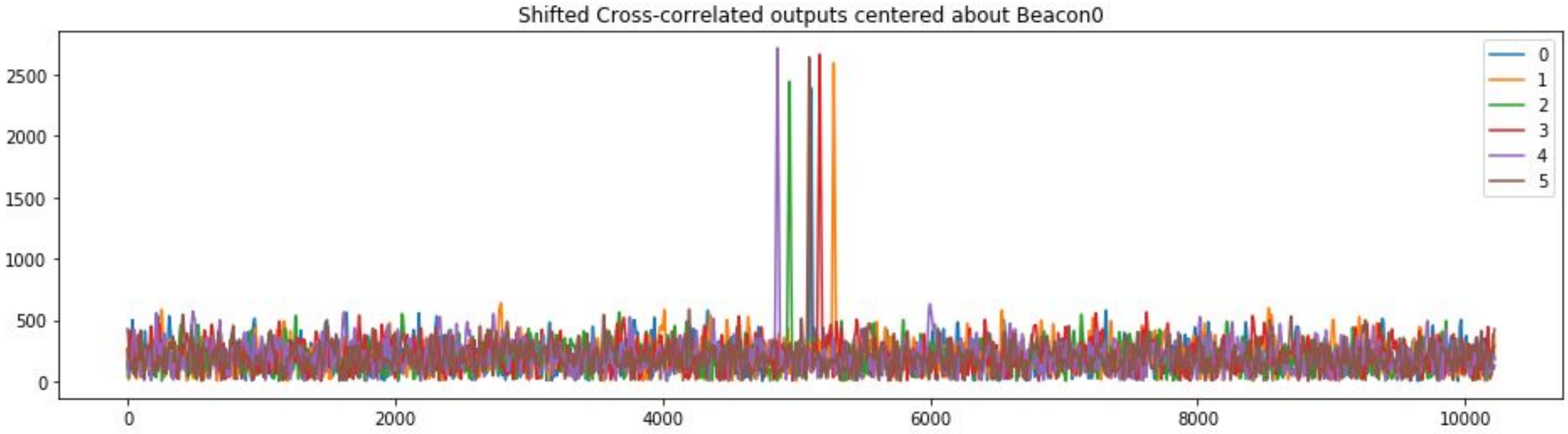
First we separate the signals



Then we shift

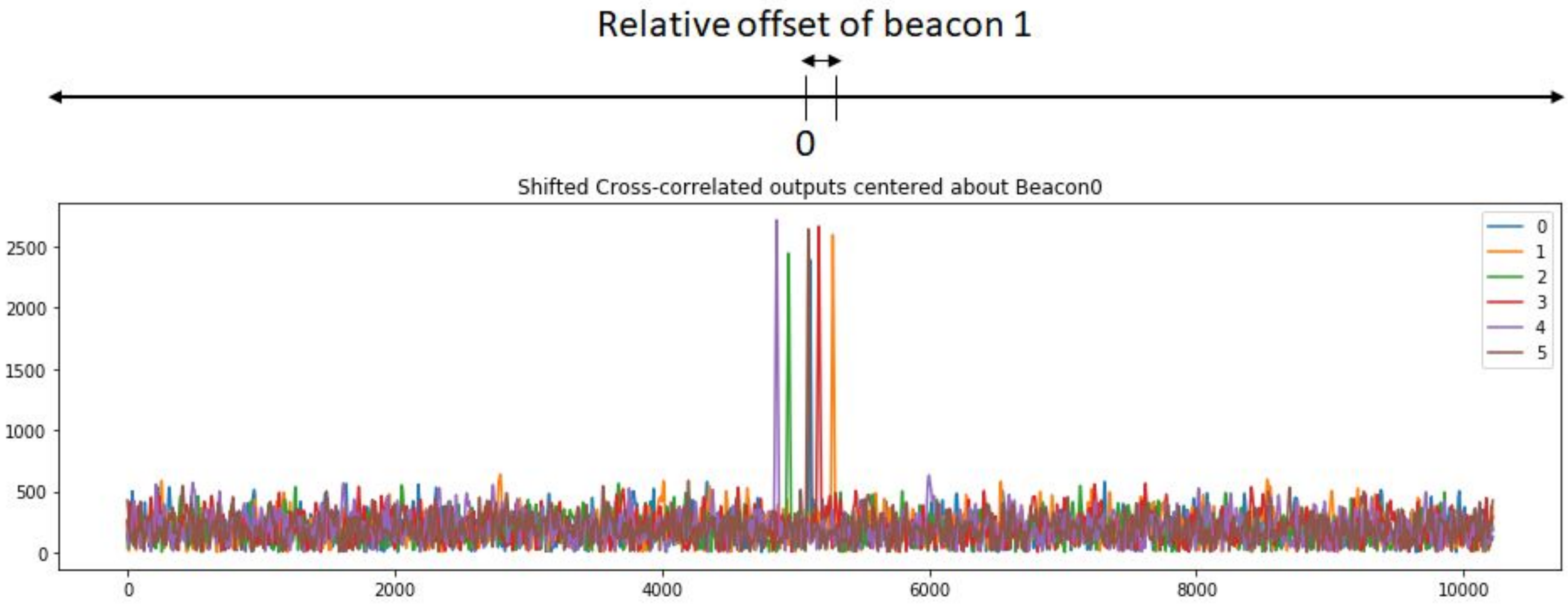
New axis





Now beacon 0 is our “origin” and all computations
can be done relative to the new “0”

New axis



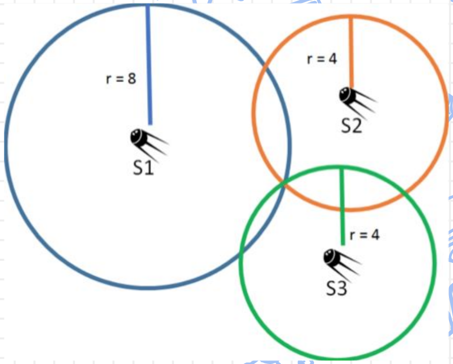
Relative offset of beacon 1
0
Shifted Cross-correlated outputs centered about Beacon0
2500
2000
1500
1000
500
0
0
2000
4000
6000
8000
10000
0
1
2
3
4
5

Shifted beacons

- We know the rate at which we recorded samples, and we know how many samples each beacon is from beacon 0
- Since sampling frequency is samples/second, then

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

- We know how long <u>relative</u> to beacon 0 it took for every other beacon to arrive
- We know where the satellites are, so we can use the distances to find our location!



- **Or can we..?**

Finally, computing distances?

- distance = rate x time
  - For beacons 1 through N, we know the time it took to travel
  - We know how fast various types of waves travel in air (AKA rate)
  - We can directly compute distance!
    - RELATIVE to beacon 0, not what we want
    - Oh, I guess we haven't quite solved it yet

Actually wait, one more problem

- We know how long it took for beacon 1 to arrive AFTER beacon 0.
- **If we magically knew beacon 0 arrived 4s into our recording, and beacon 1 arrived 3s after that, how long did it take for beacon 1 to arrive?**
    - Knowing the time beacon 0 arrived (t0) we can fully compute our distance
- But in general, we don't know when beacon 0 arrived. You'll be given it for today.

Notes + next lab:

- If we knew distance / time of flight for beacon 0, finding location is easy
    - Today this value will be given to you for testing purposes
    - Find out how to deal with this in APS2!
- It's a longer lab
    - If needed, you may finish at home and get checked off during the first 15 minutes of APS2
- Note: Sliders in the notebook should but may not work; not essential so you can move on