Close this window

Compiled Messages

Print Save as File

Subject: HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 5, 2011 8:29 PM

Hi everyone,

I thought I would, out of fairness, share a hint I discovered with the rest of the class on how to do the first part of HW2. IMHO, you shouldn't get hung up on the math due to the sampling differences between the C/A code for PRN22 but if you do, here's a hint:

Check out the "resample" command in MATLAB. This will allow you to resample a signal using interpolation. For example:

1) Get PRN 22 code output (should be able to do this from HW1). The output should be in the chipping rate of 1.023 MHz. In case your code doesn't do this, define your TIME vector for the CA code.
2) Upsample by the following way:

a) Define timeseries object: ts1 = timeseries(data,time);
b) Upsample (or downsample): ts2 = resample(data,new time vector,'zoh');

Note the 'zoh' option will do a zero-order hold on the interpolation, meaning it will only "match" values if the carrier, in this case the CA code, is there.

Then you can do the math. You can index into the data by using the ts2.data(:,:) method, like a structure.

IMHO, I hope this doesn't violate any rules because this first part is only the first 15% of the homework and the real meat is doing the Doppler and time delay bin searches. I haven't gotten there yet. I also am doing this in MATLAB 2009a ... I'm just resisting change to the wildly unstable and bloated 2010 ...

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 5, 2011 11:03 PM

I actually have a question ... how long is the search taking for folks?

I just attempted part 2 and it is taking a LOOOONG time to do the search through all possible delays and doppler effects, so I'm thinking perhaps I did it wrong? I did a quick calculation and my first pass took around 30 minutes. My next one with a more fine grid is predicted to take more than three hours!

Any hints in the possible range of f_D? more efficient code implementation?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Yao-cheng Lin **Date:** October 6, 2011 12:17 AM

About the range of fD, you could consider the definition of fD = fL1 * vd / c
where vd is the relative velocity

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 7, 2011 11:18 PM

So, basically after trying the circcorr function provided by the Professor, I determined that it was my own correlation function which was really inefficient. Basically, my old correlation function was:

for n=1:length(tau)
y_lag = circshift(y, [0 tau(n)]);
Rxy(n) = 1/length(x) * sum(x.y_lag);
end

So, basically we have a whole bunch of tau to search through this time, so having a loop inside a loop inside another loop was really inefficient.

Using the circcorr function was many many thousands of times faster. So I decided to use that. That is also my recommendation to others in the class.

No on to my next question ... has anyone gotten results? For some reason, I'm getting correlation peaks that are like, less than 0.1. To me, this sorta makes sense that we have a sinusoidal function with values between -1 and 1, multiplied by a function with 1's, 0's, and -1's, so the average could be really small.

However, I'd like to make sure that I'm not totally missing something here. I'm pretty confident I found the peaks because the rest of the points are like, 1e-6 ... but I'd like to compare results with others if possible...

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 9, 2011 12:11 PM

Thanks for the help Patrick. I am glad that this all makes sense in the end math-wise. Those notes were throwing me for a loop.

It sounds like you found a good way to get everything organized. I seem to be getting decent at an index in matlab, but this seems to go beyond that a lot.

So I know the basic premise. so we start with the

delayed signal (I did this by just reassembling the BPSK vector by truncating it and adding the truncated part on the end). I have this running 5714 times. since we each chip lasts about 5 times as long as before and we need to take a new delay sample every 1/5 Tc I figured this would work.

Now inside of this loop we have to run every test doppler for each delay. I decided on a doppler range from 13.1khz-17.8khz (did anyone else get this)? As for the integration time I am a little unsure because the notes said TI>1ms, but by my calculation I got TI to be (2.8e-5:3.66e-7:3.82e-5) (anyone else get this to get enough samples)

The problem I have is keeping all of the values cataloged. Going through each of these would give me a 3.2e9 values, which is going to be over my memory allocation. I was thinking of breaking the iterations if the values exceed a certain value, but I don't really have any idea how to do that in matlab.

Sorry for the crazy long post, but I can't figure out how to format this.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 9, 2011 1:22 PM

Sorry the thing with the delay is totally wrong. I changed it to testing 2558 delays and testing every 2 BPSK values. It isn't quite testing 1 per every chip but he said circcorr needed 1.5Tc for testing.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 9, 2011 5:27 PM

I just started problem 2 and I'm having a tricky time determining how long the integration time should be for delay determination. In class he seemed to say about a few ms, but I was thinking it would the entire 5ms of raw data. The reason I think that is because iaren't we cross correlating the "entire" raw data with the I signal which would mean we would integrate for the entire length of the data?

Also my code is taking FOREVER to run, and I'm using a recommended Tc/2 delay interval which is about 4.8e-7 seconds. Were you guys getting really slow results too?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 9, 2011 6:48 PM

I see what you're saying about TI, but I guess I'm thinking about it wrong. I think 1ms would be right because that is 1 period of the code. I am doing something wrong with TI because I'm using it to find my step in fd and that's wrong?
Using some geometry that I was figuring I got that the range in fd should be about 4.7khz, but using TI=1 that means the spread is 500hz. Can anyone explain how this is supposed to go? I got 13.1khz to 17.8 khz. I know we can tell exactly how we did the geometry for this, but I am confused about this discrepancy.

I finally got my delayed signals all sorted out. I did it by creating a 28570x2558 matrix and recording all of the delayed signals at once. It is taking about 5 seconds at this point. Did anyone else approach it this way?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 9, 2011 8:27 PM

I started by cross-correlating 1ms with various delays and see if I can have a strong correlation.

My plots don't look nearly as good as the notes (though I don't think they are going to look like those). And I was getting a time delay of 4.41e-5 seconds for satellite 22 and 6.3e-6 for satellite 27. Also my magnitudes were roughly 0.0262 and 0.0274 which seems kinda low but they are a lot larger than the mean value of roughly 1e-6.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 11, 2011 9:14 PM

Okay, I was totally screwing this up, so my previous comment is totally obsolete.

I "think" I am getting a bit closer, but I am not totally sure if it is working right. So I've been cross-correlating my local signals with multiple delays to generate and I and Q signal. From there I take I^2+Q^2 and compare the sum of the squares to the ambiguity function squared. From this I have gotten a delay of roughly 12 microseconds for satellite 22 and my ration of the largest to the mean value is about 10 which isn't great, but I guess with the only 1.5 bits of quantization and the noise in the input that that result might not be so far fetched. Anyone got anything similar?

Thanks,

Vikas

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 11, 2011 10:44 PM

hmm. Maybe I am doing this wrong still.

are you saying you are just doing cross correlations of the non-delayed signal with a test delay, then finding that z?

then you cross the delays with the hw2_data and then find that z?

then you compare the two?

Also, could you give some advice on the noise? I am totally lost with that.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 11, 2011 11:36 PM

The correlation should be done using circcorr, and all you have to do, is define your integration time, restrict the signals to just that time, and then do one FOR loop for each f_D bin with the correlation between the locally generated I and Q (with test dopply) and NO delay. circcorr takes care of that for you.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic

Author: Griffin Rowell **Date:** October 11, 2011 11:57 PM

ok I am having a matlab crisis. Manipulating my code is now a monumental task and I am pleading for help to fix it. right now I have a bunch of huge arrays (1,2585,28750) that I am multiplying. Each iteration is:

```
load X22_89.mat
load hw2_data.mat
hw2_data=transpose(hw2_data);
hw2=zeros(1,1,28570);
hw2(:,1,:)=hw2_data;
I2289=zeros(1,2585,28570);
Q2289=zeros(1,2585,28570);


for k=1:2585
I2289(1,k,:)=(circcorr(hw2(:,1,:),XI(:,k,:),.001))./1023;
Q2289(1,k,:)=(circcorr(hw2(:,1,:),XQ(:,k,:),.001))./1023;
end
```

Does anyone now a way to get it so this runs through each iteration and defines a new variable?

Alternatively, does anyone know an easier way? Patrick, are you saying you just did circcorr for each delayed signal with the sample signal? Are you saying you didn't calculate any of the delayed signals at all? You just circcorr the BPSK signal multiplied by the sin and cos functions and then cross those with the sample signal?

Also I am not familiar with the term bin. I think I am missing something huge here in the programming. It's just not feasible for this to work. In the example I gave above each of those XI and XQ come from other huge arrays I had to calculate. I am just wanting to start over at this point. Any help would be greatly appreciated.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Nissa Smith **Date:** October 12, 2011 12:05 AM

I think, at least what I did is this:

Used circcorr for (I and hw2_data) and (Q and hw2_data) which automatically does the tau delays (homework 1 made us do the delays for like -10 to 10 chips remember?)

so you just need a for loop to cycle through different fd values then.

like:

```
fd=range
for d=1:however many
Ians(d,:)=circcorr(I,hw2)
Qans(d,:)=circcor(Q,hw2)
end
```

hopefully I didn't do it wrong O.O

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 12, 2011 12:13 AM

oh so you're saying that out BPSK already has all of the delays built into it and a plot of this stuff will reveal that naturally. I have been superficially truncating the code and reattaching it to the end. Thank you Patrick. I really appreciate it. That makes a lot of sens. Hopefully tomorrow I can redo this properly without these crazy arrays.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 12, 2011 12:13 AM

Oh that was you, Nissa. That you.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 12, 2011 12:36 PM

Nissa,

You are neglecting to insert the period of the data in the circcorr function. The format of the function is:

R_I = circcorr(I, hw2_data,1/f_S) where f_S is your frequency of the sampled signal as defined in the paramters in the homework.

Then you do your Q signal

R_Q = circcorr(Q, hw2_ddata,1/f_S)

What you get for the R's is a vector where each element is the correlation value between the two signals delayed from -n/2 to n/2 "chips" (I put in quotes because we resampled and it is not chips anymore but more like 1/5 a chip ... anyway ...)

So what you do is loop from -f_D_range:f_D_bin:f_D_range and store each vector in the R_I and R_Q and what you get is a matrix with delays along the columns and f_D frequencies along the rows.

From there you do Z = sqrt(R_I.^2+R_Q.^2) (you need the dot in .^ for element-wise squaring). Or you can do the squaring of each vector element and store the results in a row in Z ... either way works.

Then you simply do a search for the max value ans that's your peak.

Getting into the details of how circcorr works ... essentially, the IFFT of a signal is:

x(k) = 1/N * SUM(from 1 to N) (X*omegaN^ (-(j-1)(k-1)))

Look familiar? It should. It's remarkably similar to the correlation! The only problem is X is just a vector. The solution is to find the FFT of Q and I, then multiply one by the conjugate of the other, then do the IFFT.

That's why you can get away with NOT generating a Q and I signal with a delay, since by virtue of the FFT, it takes care of the delay for you.

If you didn't use circcorr ... that is If you were to do the sum of the elements multiplied by each other, you WOULD have to generate I and Q with both f_D and tau each time. That's why it was taking forever with my original implementation. Literally like, an hour. Using circcorr is about 1000x faster, literally.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Nissa Smith **Date:** October 12, 2011 3:34 PM

Sorry I didn't mean to confuse anyone. I was just being sloppy in writing the proper format of the function. I just meant in general terms, cross correlate (I,hw2) etc.

But yes, the circccorr function with a for loop for the different fd's is the fastest thing ever! so glad that works well. ^^

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 12, 2011 5:23 PM

Thanks Nissa, Patrick, and Griffin for all the help it has really cleared up a bunch of the misconceptions I had regarding this problem and got me on the right track by using just one for loop for the the doppler frequency.

One issue that I'm having is that the HW wants us to know which satellites are available at which time. Unfortunately this sounds like it is adding huge complexity to the HW for two reasons. One it seems like we have to keep adjusting the integration time and the other involves adjusting the window in which we correlate. For example if our integration time is 1 ms we have to keep sliding which input data we correlate this with like from 1ms to 2ms or 1.5ms to 2.5 ms. It seems like more loops are necessary to do that. Am I completely missing something here?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 12, 2011 8:05 PM

Hmm...

I think you're making this more complicated than it's worth Vikas. If your integration time is 1 ms, then you WILL catch a peak. It's up to you to search for the "best" peak. If you think you want to scan between different windows in time you may as well increase your integration time.

The prof. in this homework is asking for a LAB REPORT type of ANALYSIS and INTERPRETATION of the problem. Note how he says to JUSTIFY what you chose.

Like all nonlinear problems and implementation styles, there's more than one right answer so long as the justification behind them hold water.

For example, in this problem, I could easily say "well just integrate over ALL 5 msec" or I could easily say "you just need 1 ms" but I have to justify why. That's up to you to determine.

I take an application practical approach. if I were designing a GPS acquisition algorithm, what would I do? I'd minimize the search time to just what was necessary and long enough to get a good crisp correlation if there.

HOW to do that is up to the designer. As I said, different strokes for different folks. I'm sure as long as your individual justification makes sense and you've looked at multiple different angles and chosen one that makes sense to you and follows the guidelines given in class/notes/book, the prof. and grader will be happy.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 12, 2011 10:07 PM

You make a good point Patrick. I should let my peak be a good enough peak :-P

Thanks for taking the time to respond!

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 13, 2011 3:30 AM

Patrick Thanks for this key info. It makes the analysis easier. I needed some of ur input when u say the following:

i.e. that we have to define our f_D_bin:
-f_D_range:f_D_bin:f_D_range

I know that I am using T_I = 5 msec and from that I used to get step size of 100 hz for the range of my selected f_D (I got step size of 100 hz from (1/2)*(1/TI)).

Here are some of my confusions:

1) What does the Prof and u mean by f_D bin?

2) I came up with f_D range between 13 Khz and 19 Khz. So should we not simply use for loop and go through 13 Khz to 19 Khz with step increments of 100 hz? Are these roughly the ranges u came up with as well or are the min and max for f_D totally different?

3) So how do I incorporate my selected f_D range of 13 Khz and 19 Khz with my selected step of 100 hz into what u wrote (i.e. -f_D_range:f_D_bin:f_D_range)?

Ur input or anyone's else's input on this would help me a lot!

-Riz

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Nissa Smith **Date:** October 13, 2011 11:27 AM

If your range is 13kHz to 19kHz with bin size of 100 hz. then

(-f_D_range:f_D_bin:f_D_range) = (13kHz:100Hz:19kHz)

Though my range was larger.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Adam Harden **Date:** October 12, 2011 12:07 AM

I'm hoping someone can clarify something for me, as I seem to be stuck. What I've done is generate the in-phase signal and quadrature signal per the notes. I've then defined a reasonable lower and upper bound for what I think the Doppler shift could be, and I've been looping for that array of dopper shifts, stepping every (1/2)*(1/TI) where TI is the integration time.

Within each loop I'm cross-correlating the hw2_data array with my in-phase signal and then cross-correlating the hw2_data array with the quadrature array. Is this correct? I've found the notes a bit challenging to follow here, and admittedly, all I get is a noisy result with mean 0, which doesn't tell me anything.

Any tips would be appreciated. Thank you!

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Nissa Smith **Date:** October 13, 2011 4:15 PM

If you are not finding your peaks, I would suggest extending your range. I found mine at a very low value.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 13, 2011 7:44 PM

Thanks for the input Nissa. It helps a lot!

For PRN 22 satellite, I found my peak (maximum value in Z function) at f_d = 1 Kh and the max value in Z was 0.0750.

Now I am about to run PRN_27 data as well.

So the way I am going to conclude as to which satellite is present in Prof's data is by comparing MAX values of Z function for both PRN 22 and PRN 27 and whichever satellite's MAX value of Z is greater, then THAT satellite is present in the data that Prof gave us. Prof said atleast one satellite is present in the data.

Anyone is this how u guys are concluding as to which satellite is present in the data?

Thanks!
Riz

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 13, 2011 8:05 PM

I got the same thing as you. Did you get a delay of like 16000 something?

As for deciding if 27 is present I don't think it's as easy as saying that the 27 signal is smaller (it is). I got zmax of 27 as .02. I haven't done anything else yet, but I think it's going to come down to whether the values of the peaks are high enough above the noise that you can really say if the satellite is visible or not.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 13, 2011 9:41 PM

Hey Griffin,

For PRN 22:

I am not understanding how to get a delay value once I have determined my f_D (i.e. 1 Khz) at which Z is maximized.

In order to find the delay, do I do autocorrelation of the downconverted signal or I_L or something? How did u arrive at value of 16000?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 13, 2011 10:19 PM

I should clarify that ~16000 is just the position in the matrix the value was at. The way my Z matrix is I have doppler shifts as columns and steps as rows. You just have to convert your times to chips after that.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 13, 2011 10:40 PM

Oh I see.

In my Z matrix, my columns are delays going from 1 to 28570 and rows are f_D.

So for PRN 22, my delay is 12091 chips corresponding to Z_max of 0.0725.

Thanks for the clarification!

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 13, 2011 10:57 PM

Remember that those aren't really chips though. We only have 5115 chips and that we just elongated the code. I think it's a little odd that we have such different delays. Can anyone explain how that could happen and if it is an absolute? It seems strange that we would have the same Z and doppler.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 13, 2011 10:59 PM

Oh nevermind. Sorry about that. I have a very similar delay.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 13, 2011 11:47 PM

Great!

Thanks for the clarification. I keep forgetting this that we only have 5414 chips. So the way I have converted my delay of 12091 is as follows:

1/fs = 1.749999913e-7

delay in time = 1.749999913e-7 * 12091 = 0.0021159 seconds

So f_d = 1 Khz and delay of 0.0021159 seconds are what gives us MAXIMIZED Z function value of Max_Z = 0.0725.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Nissa Smith **Date:** October 14, 2011 9:04 AM

I did things a little differently because your cross correlation goes from a -tau to tau range. So the number for row or column that represents your tau steps needs to be adjusted as well.

I would suggest thatyou find the row/column number of where tau=0.

1/fs=1.75e-7
delay in time = (1/fs)*(position of peak - position of tau=0)

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Nissa Smith **Date:** October 14, 2011 9:17 AM

Sorry, I meant your lag range for the cross correlation goes from a - value to a + value, so roughly the first half of the rows or columns (depending how you set up your code) in your matrix should represent the negative side of the delays.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 14, 2011 9:31 AM

Good catch. Thanks Nissa.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 14, 2011 5:56 PM

Hey Griffin,

Just to complete your #s and #s comparison: For PRN 27 I have the following #'s:

I did search in f_d from 0 to 50 Khz and determined the following:

Z_max = 0.0259 and it occurs at f_d = 20.7 Khz and the delay at which Z_max occurs is 23390 (this is just the number from my Z_max and my columns are delays going from 0 to 28570).

I know ur Z_max was 0.2. BTW after Nissa's input, my new delay values are:

PRN 22 = 1.75e-07 * 2195 = 3.84125e-04 seconds

PRN 27 = 1.75e-07 * 9104 = 0.0015932 seconds.

It makes sense that we should measure delay from 0 reference point in the cross correlation plots of Z_max vs lag range.

I got the numbers of 2195 and 9104 from Cross correlation plots of Z_max vs. lag range going from -14285 to +14285.

My conclusion (as of now) will be mostly likely that PRN 22 is present in the data that Prof sent us. So delay calculation for PRN 27 shown above is just an exercise. Since PRN 27 is not present, hence no delay is there for it.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 14, 2011 6:23 PM

My PRN 22 results match yours, but my PRN 27 are different. Note I was able to get a match for PRN 27 because doppler shifts are not always positive, they can be negative.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 15, 2011 11:45 AM

I am getting a match for 27 at negative dopplers also.

As for the noise, is anyone getting a power to noise ratio of around 14.5?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 15, 2011 9:05 PM

Hey Griffin and Vikas,

For PRN 27: I also did get Z_max at negative dopplers as well. I took the range of f_ds from -3 Khz to + 3Kz with step sizes of usual 100 Hz.

So I found my Z_max = 0.0228 and corresponding f_d at which this Z_max occurs is: -2300 hz = -2.3 Khz. Also the delay range at which this occurs is at: 9270 (where in my Z matrix, my columns are delay ranges going from 0 to 28570).

Also I did calculation for Problem 4 and yes I get estimate of Signal to noise (SNR) to be:

SNR_for_PRN_22 = 0.07251/(0.01/2) = 14.502

I am glad we are getting same results after doing some quick calculations for Problem 4 as well!

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Adam Harden **Date:** October 15, 2011 9:17 PM

Hi Rizwan,

I've been following the discussion for a bit now. Could you clarify where the "(0.01/2)" comes from in your SNR calculation? Are you just looking at the plots of Z vs (delay,Doppler) and pulling some value?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 15, 2011 9:56 PM

Hey Adam,

See the attached plot for PRN 22. I have attached both .tiff file and .fig (u can open .fig in matlab window). This is the plot for Z magnitude vs. lag range between -14285 to +14285 when Z_max = 0.07251.

Now I get (0.01/2) term from this plot. Look at Prof's notes for Session 9 (09/20/2011) and look at his notes for slides: W4-25 and W4-26. U should watch the video too.

So on W4-26 slide Prof draws Cross-correlation plot and he calls the PEAK value to be CR (for us it is CR = 0.07251) and then he calls "PN" to be noise power and they are averages of the the noise that u see around the Peak value. (so for us PN = 0.01/2. I took the mean of noise that is present around my peak value of 0.07251 and looking at the plot, i estimated the mean to be 0.01/2).

On W4-26 slide, Prof then defines Signal to Noise Ratio (SNR) as: SNR = CR/PN. So SNR = 0.07251/(0.01/2) = 14.502. However this is NOT CR/No that Prof wants: Read ahead:

Now go to Session 11 (09/29/11)'s lecture and see that in Noise discussion (slides W5&6-33, W5&6-34, W5&6-35, Prof gives us more definitions of SNR:

SNR = CR/PN = (CR/N0)*T_I where T_I is our integration time. I choose T_I to be 0.005 seconds (length of our code).

So from here u can get ratio of CR/N0 that prof is really asking for.

So SNR = 14.502 = (0.07251/No)* 0.005. Solve for No: No = 2.5 x 10^-5 watts/hz.

Then (CR/No) ratio = 2900.4 hz = 2900.4/secs.

CR and PN has dimensions of [watts]

No has dimensions of [watts/hz] = [watts*secs]

T_I has dimensions of seconds.

CR/No has dimensions of [hz] or 1/seconds.

Attachments: Problem_2_PRN_22_Z_max.fig Problem_2_PRN_22_Z_max.tif

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 15, 2011 9:59 PM

Hey Griffin,

The following is explanation that I wrote to Adam as to HOW to calculate CR/No. In my earlier response, I also get 14.502 but that is NOT CR/No that Prof is asking for. 14.502 is Signal to Noise ratio (SNR). Read below how to get CR/No that Prof really wants!

****

See the attached plot for PRN 22. I have attached both .tiff file and .fig (u can open .fig in matlab window). This is the plot for Z magnitude vs. lag range between -14285 to +14285 when Z_max = 0.07251 (ignore the word "chips" in the plot).

Now I get (0.01/2) term from this plot. Look at Prof's notes for Session 9 (09/20/2011) and look at his notes for slides: W4-25 and W4-26. U should watch the video too.

So on W4-26 slide Prof draws Cross-correlation plot and he calls the PEAK value to be CR (for us it is CR = 0.07251) and then he calls "PN" to be noise power and they are averages of the the noise that u see around the Peak value. (so for us PN = 0.01/2. I took the mean of noise that is present around my peak value of 0.07251 and looking at the plot, i estimated the mean to be 0.01/2).

On W4-26 slide, Prof then defines Signal to Noise Ratio (SNR) as: SNR = CR/PN. So SNR = 0.07251/(0.01/2) = 14.502. However this is NOT CR/No that Prof wants: Read ahead:

Now go to Session 11 (09/29/11)'s lecture and see that in Noise discussion (slides W5&6-33,

W5&6-34, W5&6-35, Prof gives us more definitions of SNR:

SNR = CR/PN = (CR/N0)*T_I where T_I is our integration time. I choose T_I to be 0.005 seconds (length of our code).

So from here u can get ratio of CR/N0 that prof is really asking for.

So SNR = 14.502 = (0.07251/No)* 0.005. Solve for No: No = 2.5 x 10^-5 watts/hz.

Then (CR/No) ratio = 2900.4 hz = 2900.4/secs.

CR and PN has dimensions of [watts]

No has dimensions of [watts/hz] = [watts*secs]

T_I has dimensions of seconds.

CR/No has dimensions of [hz] or 1/seconds.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: John Cook **Date:** October 15, 2011 7:34 AM

Hey Rizwan,

I thought I was doing this correctly, but your post makes me question my methods...my Z matrix has the same number of columns as yours, but how many rows were you using?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 15, 2011 4:00 PM

Hey John,

Yes please see what Griffin has said about doing problem 2 via a NEW approach.

My Z matrix is based on the NEW approach that people like Patrick explained really well here on this thread. Be sure to also read Patrick's really vital comments on the NEW approach. That is the NEW approach that Griffin is talking about. Now everyone on this thread (that I have talked to) is using the new approach that Patrick mentioned and that makes the code run fast.

Also previously I was only taking my f_d range from 0 to 50 khz for both PRN 22 and PRN 27. Luckily I matched the results for PRN 22 on the f_d = 1 Khz (on positive side of f_d) with Griffin and Vikas, but for PRN 27, I have to run my tests on negative side as well. Actually for both PRN 22 and PRN 27, I should run my sims on range of f_d which includes both negative side and positive side of f_d.

But I am glad results of PRN 22 match with others. I just need to run PRN 27 on negative side too.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 15, 2011 4:37 PM

I just ran 2 tests with one negative and one positive.

Have you done the noise part yet? I feel like my approach is reasonable but I'm not completely sure.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: John Cook **Date:** October 15, 2011 7:30 AM

Griffin,

Are you no longer handling the large arrays in your code? I just ask because currently if I loop from i=1:2585;, i get a an error that MATLAB has run out of memory : (Any advice would be appreciated!

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 15, 2011 10:22 AM

I basically started over on Thursday. The approach I was using was just too inefficient. If you are calculating each delay as a matrix you need to stop right now. There is no reason for it.

Once you have your elongated BPSK signal and your time vector, you can find the I matrix. The way I did this was using a for loop and and doing a loop inside each doppler test value. This wasn't really that slow. You just define your doppler range and how often you are going to fd_test= fd_min:fd_bin:fd_max. Then, when you define your for loop, you can write I as I(:,cycle)=...sin(...fd_dtest(cycle)...)

This applies for the rest of the values needed for the problem also.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 6, 2011 12:27 AM

I did something kind of strange but I think I may have bludgeoned my way through it. What I did is interp1(C/A time, C/A code, sample time). This gave me something resembling what I wanted, but the interpolation gave me some rough "edges" where values weren't equaling -1 or 1. What I did was turn the interpolated vector into 2 vectors; one with a ceil(interpolated), one with floor(interpolated). I then added the vectors together and redifined all of the values so that any values that were integers stay the same integer (-1 or 1) and the now rounded "edges" stay as -1 or 1.

The resample sounds a lot more elegant and I am a little worried about some of my rounded edges being the opposite sign, but the graph looks pretty so we will see.

I haven't started on part 2 yet either. Will begin tomorrow.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 6, 2011 1:38 PM

Ok i'm trying to get my bearing for problem 2. I'm not going to say how exactly, but did anyone else figure about 13khz-18khz for Fd?

I am also a little confused about the D signal. Am I right in thinking that D is the data sent by the professor? I know it needs to be in BPSK, but the signal is not in binary. Has anyone figured out how to get D(BPSK) from the "1.5 bit" signal?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Vikas Vatsa **Date:** October 6, 2011 7:58 PM

I'm just starting problem 2 and I'm a little confused too about the 1.5 bit data that was given to us. I was actually thinking it was a digitized/sampled version of xi (down converted signal) on page 5&6-13. My hunch is that it wouldn't be a the D signal since we are assuming that D is not a function of time since all searching must be done before D changes value.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 6, 2011 8:38 PM

Yes, D(t) in this case is always 1 (i.e. before it changes state).

And re: slow computing ... I was able to structure my program so it uses parallel computing. Quite simply to do this:

matlabpool open local #number of parallel sessions ... depends on your compy

and when you're done

matlabpool close (THIS IS IMPORTANT!)

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 6, 2011 10:00 PM

I'm getting started on problem 2 and just want to run what i'm thinking by you. Are you thinking that we find I and Q using the equation on W5&6-17 and then crosscorrelating each one with the x(t) to find the best correlation? I'm a little confused because the ambiguity function has no dependance on x(t). Have either of you figured this out?

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic

Author: Patrick Mc Cormick **Date:** October 8, 2011 10:10 PM

The ambiguity function does depent on xi. Remember, xi is simply the downconverted signal, which is

xi = sqrt(C)D(t-tau)s(t-tau)*sin(2*pi*(fI+fD)t+phi) + noise (ignored)

there tau is the REAL WORLD delay, not the test value

And then we generate the "functions we multiply the downconverted signal" and then correlate on those two. Finally we calculate Z by simply doing sqrt(I^2+Q^2).

So, in the notes, he's already substituted for xi in the ambiguity function. It's just really verbose in the way the prof. put it in the notes.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Rizwan Qureshi **Date:** October 7, 2011 3:36 PM

Hey Patrick,

For HW 1, my old sampling rate was 1.023 Mhz and I had 1023 chips with entire code cycle being 0.001 seconds.

I resampled my old HW 1 PRN22 signal at new sampling rate of 5.714286 Mhz.

Now at new sampling rate of 5.714286 Mhz, the total # of chips I get is: 5714 chips with entire code cycle being same as for HW 1 (i.e. 0.001 seconds).

Do u get this as well? I would appreciate your response!

Rizwan

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Patrick Mc Cormick **Date:** October 8, 2011 12:11 AM

Rizwan,

Yes, that is correct. Your new BPSK code should have 1023*(f_S/f_C) samples where f_S is the sampling rate and f_C is the original chipping rate.

As for the actual plots ... I think people need to double-check their implementation of the resampling.

What you should get is your very same BPSK signal but with 5x the samples in it ... NOT the same sequence but at a faster rate. That is, if the original sequence was

[1 0 1 0 0] then the new sequence should be

[1 1 1 1 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0] or something like that

Check to make sure you get that and plot it out ... then use the resampled signal in the math.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 9, 2011 11:08 AM

You are basically going to need to resample at the sample rate in this new homework to get matching vector lengths. For example, each chip will now change every 5.7 samples about. I did this using the interp1 function and then messing with floors and ceiling. Patrick did it by using resample. Your BPSK should be 28570(not sure about the number exactly right now).

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Griffin Rowell **Date:** October 9, 2011 11:10 AM

Oh sorry, you meant for one period. Yeah, you are right. Patrick put a good example of what it should look like.

Reply
Forward

Subject: Re:HW2 **Topic:** Default Topic
Author: Adam Harden **Date:** October 15, 2011 8:55 PM

Hi all,

I'm having some difficulty deciphering the notes on determining the carrier to noise ratio C/N0. I've taken a look at Misra & Enge, but unfortunately it doesn't seem to clear my confusion. Can someone nudge me in the right direction to get started computing C/N0?

In Reply to: HW2

Reply
Forward

Close this window