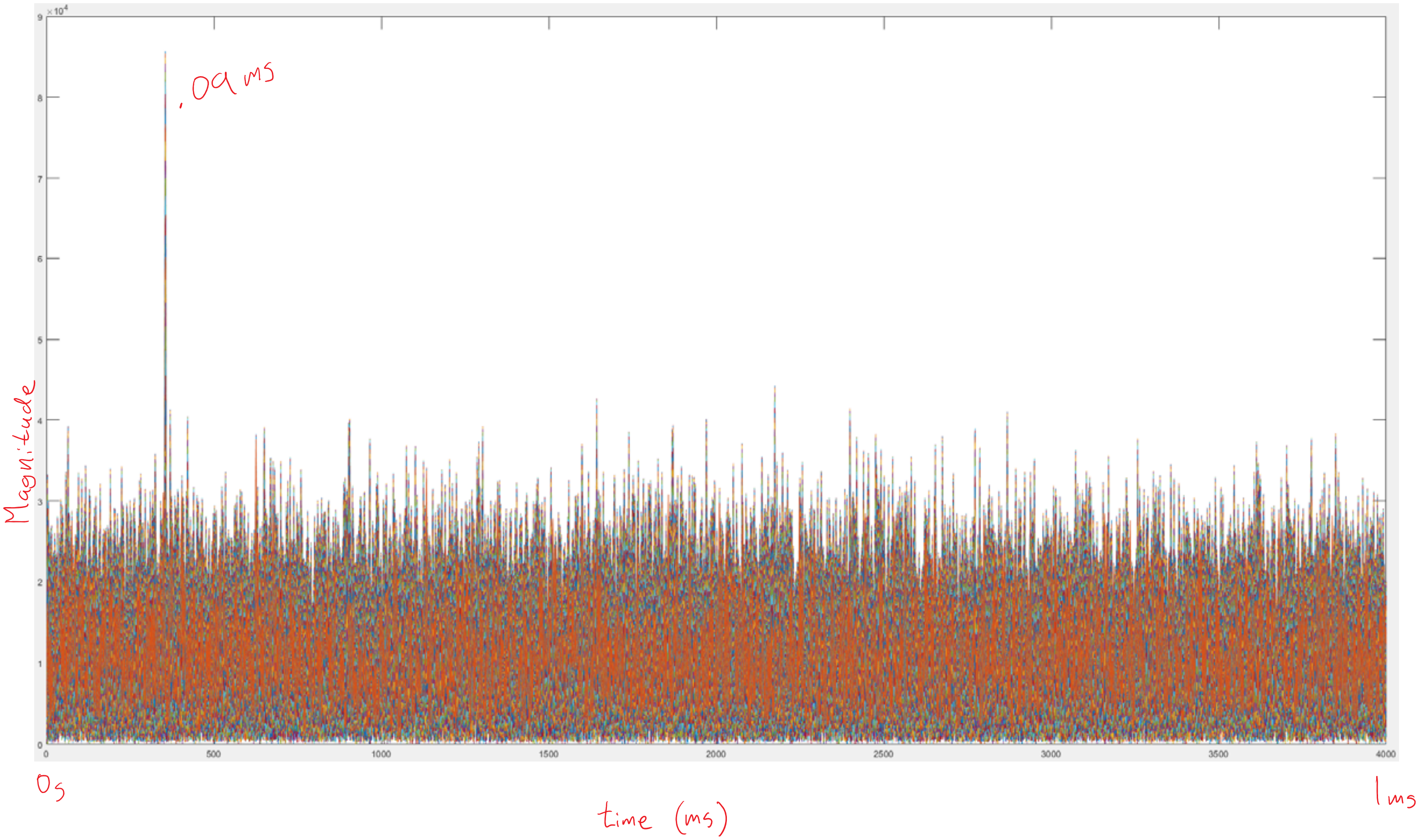
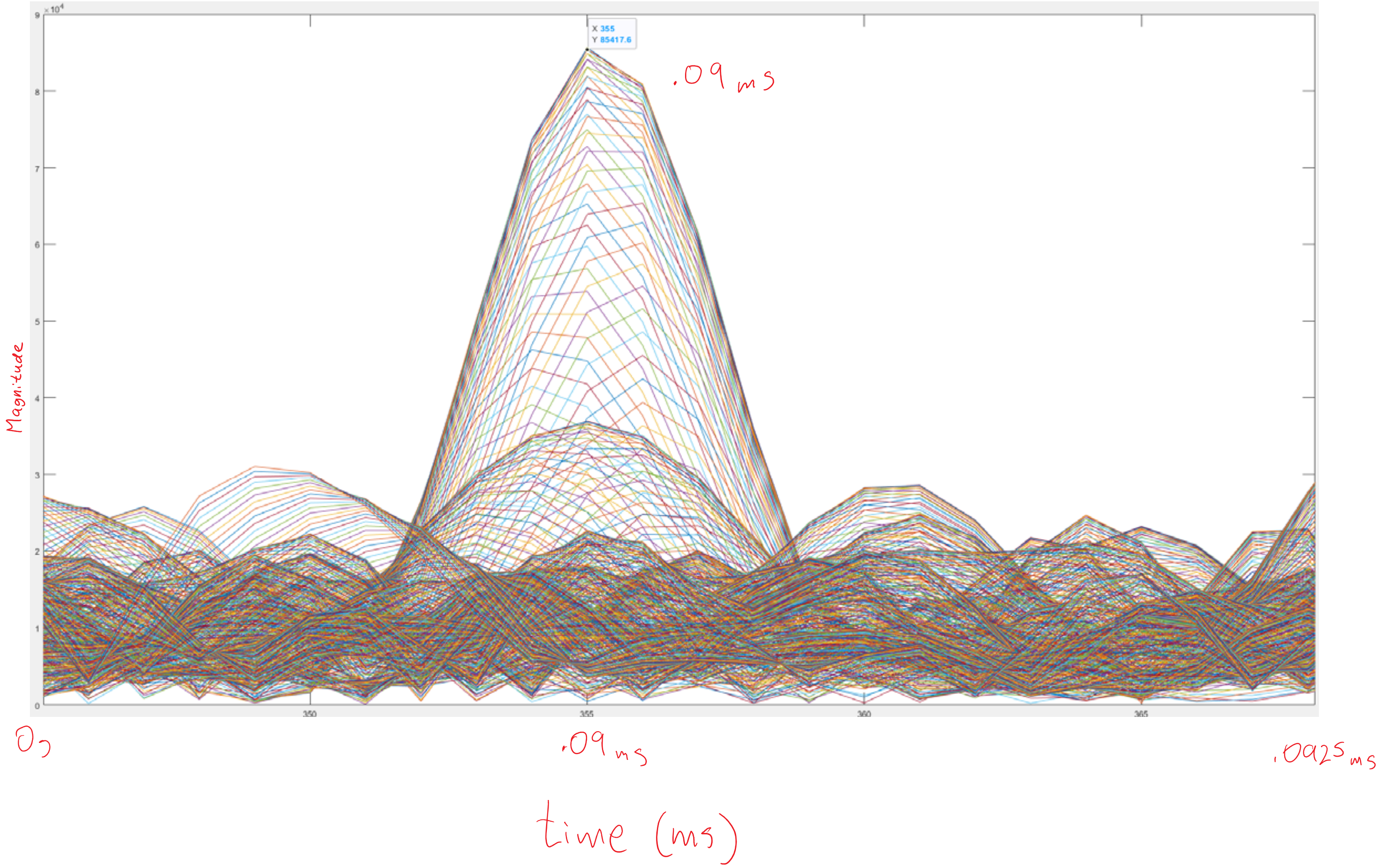
CA5 - pt. 2

Thursday, February 24, 2022 2:18 AM

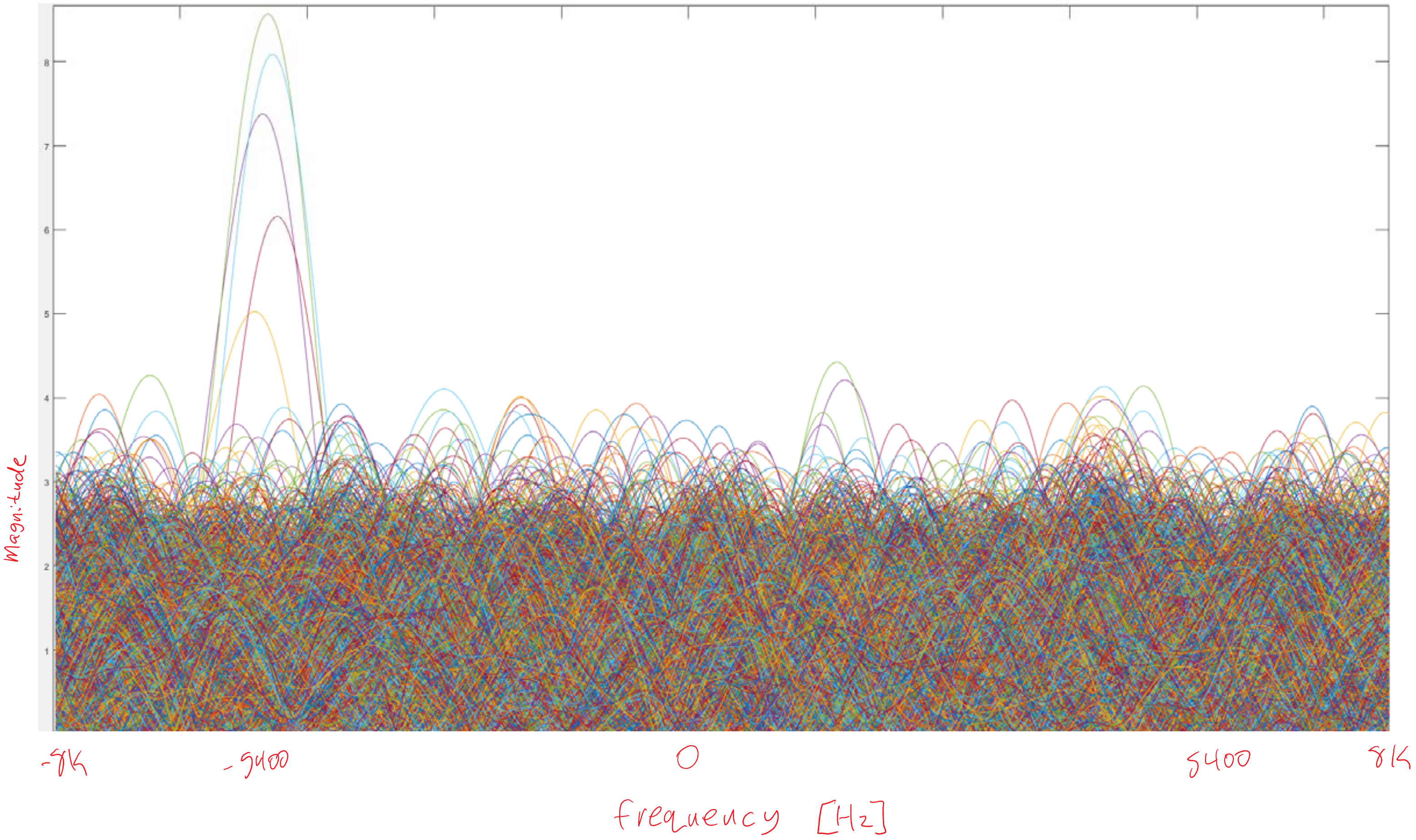
Satellite #11 time



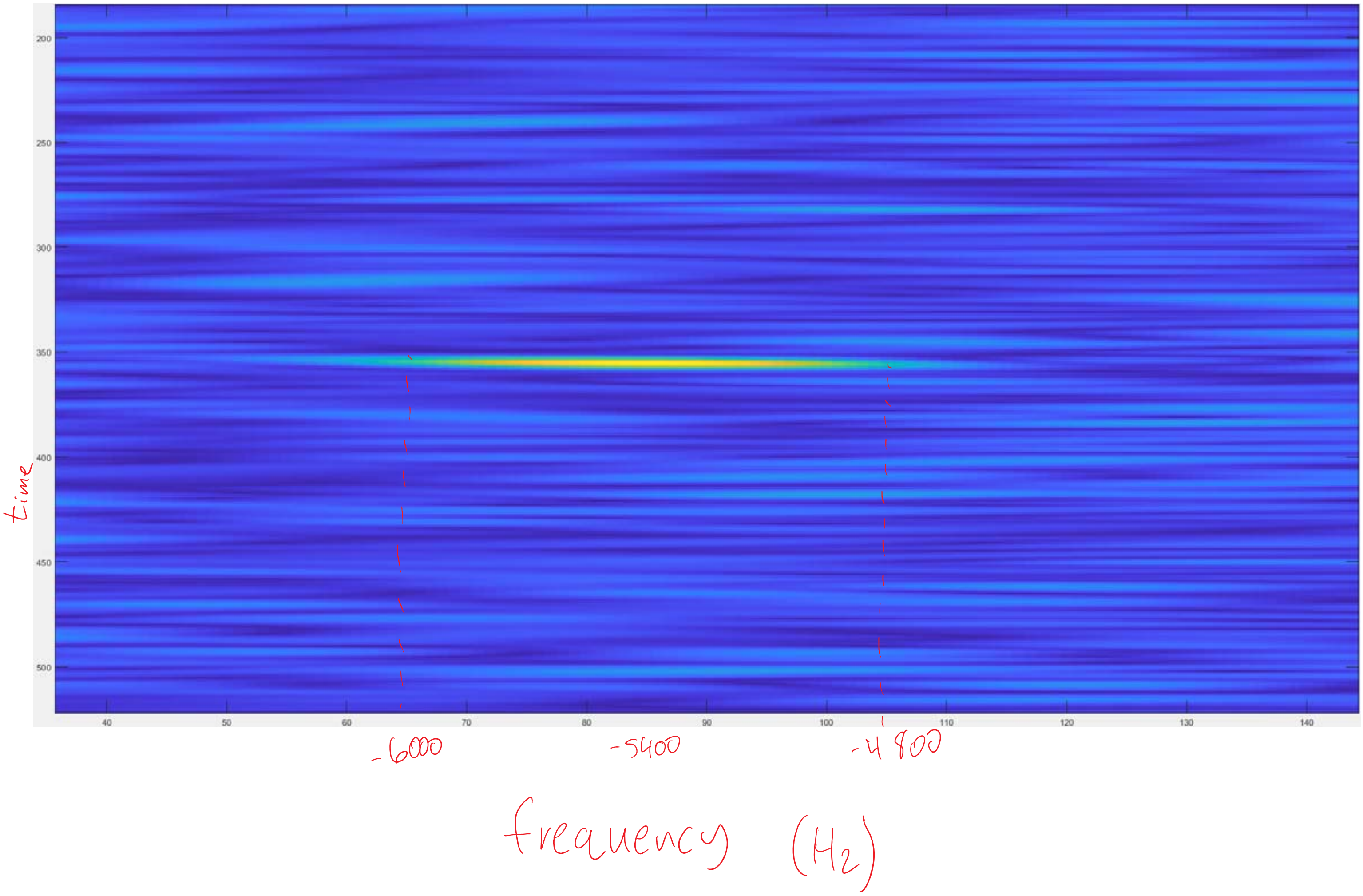
Satellite #11 time



Satellite #11 frequency



Satellite #11 Spectrogram



Visible Satellites:	Time delay (milliseconds)	Doppler Frequency (Hertz)
1	.997	-7000
11	.089	-5400
20	.313	-8000
32	.923	-6500

```
% Set up some important variables

Fs = 4e6; % Sample rate of data [samples/second]

% GPS matched filter impulse responses
prn_code_id = [1:37]; %#ok<*NBRAK> % PRN code IDs for GPS satellites (32 + extras = 37)
for i=1:length(prn_code_id)
  mf(:,i) = build_matched_filter(prn_code_id(i),Fs);
end
[Nh,~] = size(mf); % Get the length of the matched filter impulse response

% Parameters for fast convolution
N = 2^17; % FFT size [samples]
MF = fft(mf,N); % Compute zero-padded DFT of matched filters for frequency-domain processing

% Parameters for doppler frequency search
Fmax = 8000; % Maximum doppler frequency [Hz]
Fmin = -Fmax; % Minimum doppler frequency [Hz]
Fbin = Fs/N; % FFT bin width [Hz]
KFmax = ceil(Fmax/Fbin); % Number of frequency bins (positive side)
KFmin = floor(Fmin/Fbin); % Number of frequency bins (negative side)
ksearch = [KFmin:KFmax]; % Set of bins for doppler frequency sesarch


% Read gps.dat
fid = fopen('gps.dat','rb'); % Open the file for binary read
Nx = N - Nh + 1; % Number of data samples so that Nx + Nh - 1 = N
x = fread(fid,2*Nx,'int16=>float32'); % Read samples and convert from short to float
fclose(fid); % Close the file
x = complex(x(1:2:end), x(2:2:end)); % Convert interleaved real-imaginary to complex


% Transform and search

X = fft(x,N); % Compute zero-padded DFT of samples in x
% Begin search
for i=1:size(MF,2) % For loop over satellites
   s = zeros(Nh, length(ksearch));
   ymax = 0; kmax = -1; imax = -1;

  for k=1:length(ksearch) % For loop over cyclic shifts (doppler frequency search)
       %shift, convolve, inverse FFT
     y = ifft(circshift(X,ksearch(k)).*MF(:,i));
     %trim
    y = y(Nh:2*Nh-1); % trim
     %add to matrix
    s(:,k) = abs(y(1:Nh));
     %calculate time/freq values
    [ypeak,ipeak] = max(abs(y(1:Nh)));
    if(ypeak>ymax)
      ymax = ypeak;
      imax = ipeak-1;
      kmax = k;
    end
  end
  i
  imax
  kmax
  figure(1); plot(s);
  figure(2); plot(s.');
  figure(3); imagesc(s);
  pause;
end
```