Imaging Radar
EE355/GP265
HW 4 Solutions

Question 1:

a) How many valid range bins are found in the range compressed data?

The number of complex range samples is:

$$n_{echo} = \frac{10218-412}{2} = 4903 \text{ samples}$$

The number of samples in the chirp is:

$$n_{chirp} = \lfloor \tau f_s \rfloor = \lfloor (37.12\mu s)(18.96\, MHz) \rfloor = 703 \text{ samples}$$

So, the number of valid range bins is:

$$n_{valid} = n_{echo} - n_{chirp} = 4903 - 703 = 4200 \text{ valid range bins}$$

b) What is the minimum fft size for range processing?

The minimum fft size for range processing is the same as the length of the original echo, which is 4903 samples. If we want this to be a power of 2, I can zero pad up to 8192 samples, keeping in mind that I should only keep the valid data after compressions (4200). With our small data set, zero-padding should not be necessary.

c) What is the effective spacecraft velocity?

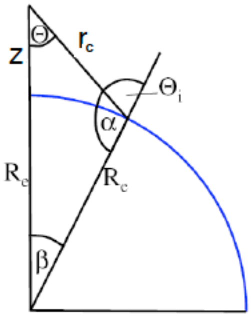


Figure 1: Spherical earth geometry.

The range to the swath center r$_c$ is:

$$r_c = r_0 + \Delta r \frac{n_{valid}}{2} = r_0 + \frac{c}{2f_s}\frac{n_{valid}}{2} = 83000\, m + \frac{(2.99792458e8\frac{m}{s})}{2(18.96e6\, Hz)}\frac{4200}{2} = 846602.43 \text{ m}$$

We can solve for the platform altitude z, given the spherical earth geometry in Figure 1. From the law of cosines, I have (θ is the look angle to swatch center, R$_e$ is the radius of the earth, r$_c$ is the range to the center of the swatch):

$$\cos\theta = \frac{r_c^2 + (z + R_e)^2 - R_e^2}{2r_c(z + R_e)}$$

$$\ggg z = r_c\cos\theta - R_e + \sqrt{R_e^2 - r_c^2\,(\sin\theta)^2}$$

Plugging in $\theta = 23^{\circ}$, R$_e$ = 6378000 m, and r$_c$ = 846602.43 m, I get z = 770717.58 m.

Then, the effective spacecraft velocity is (v = 7550 m/s as the platform velocity):

$$v_{eff} = v\sqrt{\frac{R_e}{z + R_e}} = \left(7550\frac{m}{s}\right)\sqrt{\frac{6378000}{770717.58+6378000}} = 7131.41 \text{ m/s}$$

d) What is the range bin spacing in meters?

The slant range bin spacing Δr is:

$$\Delta r = \frac{c}{2f_s} = \frac{2.99792458e8\ m/s}{2(18.96e6\ Hz)} = 7.91 \text{ m/pixel}$$

The slant range resolution δr is:

$$\delta r = \frac{c}{2(BW)} = \frac{c}{2s\tau} = \frac{2.99792458e8\ m/s}{2(4.189166e11\frac{Hz}{s})(37.12e{-6}\ s)} = 9.64 \text{ m}$$

To get the ground range resolution and bin spacing, we first need the incidence angle θ_i at the center of the swath, which is larger than the original look angel, since we have a spherical earth geometry. By using the law of sines (here, β is the angle at the center of the earth):

$$\frac{\sin\beta}{r_c} = \frac{\sin\theta}{R_e} ==> \beta = \sin^{-1}\left(\frac{r_c}{R_e}\sin\theta\right)$$

$$\theta_i = \theta + \beta = 23^{\circ} + 2.973^{\circ} = 25.973^{\circ}$$

The ground range bin spacing is then:

$$\Delta r_g = \frac{\Delta r}{\sin\theta_i} = \frac{7.91\ m/pixel}{\sin 25.973^{\circ}} = 18.05 \text{ m/pixel}$$

And the ground range resolution δr_g is then:

$$\delta r_g = \frac{\delta r}{\sin\theta_i} = \frac{9.64\ m}{\sin 25.973^{\circ}} = 22.01 \text{ m}$$

Question 2:

a) Determine the approximate Doppler centroid of the data

To determine the Doppler Centroid frequency, we use the average phase change method from HW 3.

Compute the phase shift $\Delta\phi(ir)$ at each range bin ir from line i_{az} to line i_{az} -1 in azimuth (where R is the range-compressed image):

$$P(ir) = \sum_{iaz=2}^{Naz} R(ir, iaz)R^*(ir, iaz-1)$$

$$\Delta\phi(ir) = \tan^{-1}\frac{Im(P(ir))}{Re(P(ir))}$$

Then average $\Delta\phi(ir)$ over all range bins and relate the phase to frequency, which will then be the Doppler centroid f_d.

$$\Delta\phi = \frac{\sum_{ir=1}^{Nr}\Delta\phi(ir)}{Nr}$$

$$\frac{\Delta\phi}{2\pi} = \frac{f_d}{prf}$$

$$f_d = prf\frac{\Delta\phi}{2\pi}$$

My Doppler Centroid is at -300 Hz.

b) Write a multipatch SAR processor for this data set. Set the azimuth beamwidth to 80% of the full illuminated beamwidth to keep the reference function below 1024 points. Process as many patches as you can from the data set given.

Each patch has 2048 azimuth lines, as specified in the problem. To process as many patches as possible, we need the offset between patches to be as small as possible. This happens at the farthest possible range $r_{dc,max}$, which will have the maximum illumination time $\tau_{az,max}$, and therefore the largest number of azimuth chirp samples.

$$r_{max} = r_0 + (n_{valid} - 1)\Delta r = 863196.95 \text{ m}$$

$$r_{dc,max} = \sqrt{r_{max}^2 + \left(\frac{f_d\Delta r\lambda}{2v_{eff}}\right)^2} = 863197.57 \text{ m}$$

$$\tau_{az,max} = \frac{r_{dc,max}\lambda}{v_{effl}}0.8 = 0.5481 \text{ s}$$

$$N_{ref,max} = \lfloor\tau_{az,max}prf\rfloor = 920 \text{ samples}$$

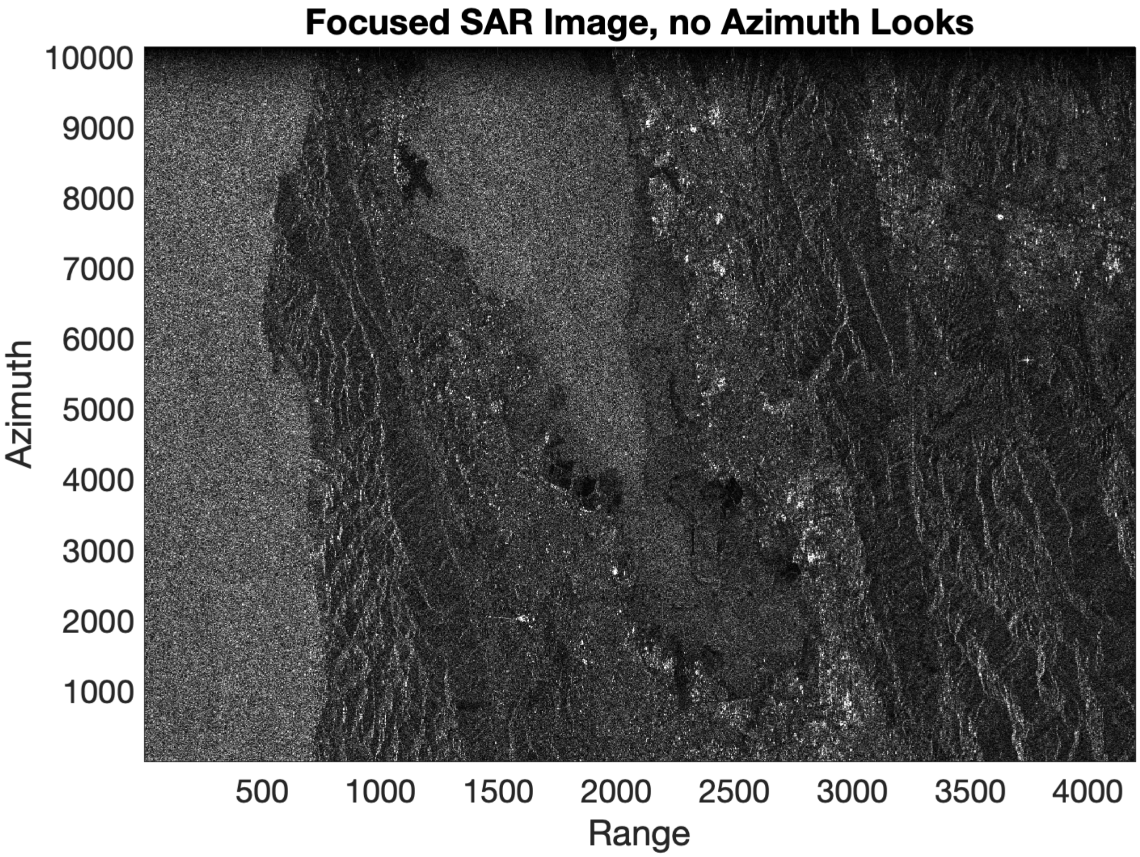
The offset between patches, which will be the number of valid azimuth samples in each patch, is 2048-920 = 1128. So, for each patch, we will use 2048 azimuth samples to correlate with the azimuth reference chirp, and keep only 1128 of the output pixels.

Since there are 10100 azimuth lines, the total number of patches we will have is $\lceil 10100/1128 \rceil =$ 9 patches. We will then have 1128*9 = 10152 azimuth lines in the final, non-multilooked image.

For each range bin, we need to compute the range to the center of the bin, the chirp rate, and the azimuth reference chirp pulse length. The ranges and pulse lengths can be computed using the same equations above (First 3 in section 2b), except instead of using the maximum range, r_{max}, we use the range bin index to calculate r_i, where we replace n_{valid} with i, the range bin index. Then in the following equations, use the new range results for each bin. The chirp rate at each range bin, i, is then:

$$f_{R,i} = -\frac{2v_{eff}^2}{\lambda r_{dc,i}}$$

Then, the reference function for azimuth compressions is a chirp with slope given by the chirp rate $f_{R,i}$, duration $\tau_{az,i}$, sample rate prf, and center frequency given by the Doppler centroid f_{dc}. We apply azimuth compression in the same way by performing the correlation in the frequency domain. We then save the magnitude of the compressed signal in the processed image, keeping only the valid bins, and combine the patches correctly.



c) What is the azimuth resolution for your 80% bandwidth processor? What is the azimuth point spacing on the ground?

The azimuth bandwidth product is given by:

$$BW_{az} = f_R \tau_{az} = \frac{2v_{eff}^2}{\lambda r_{dc}} 0.8 \frac{r_{dc}}{l v_{eff}} = 0.8 \frac{2v_{eff}}{l}$$

The azimuth resolution is then:

$$\delta_{az} = v_{eff}\delta_t = v_{eff}\frac{1}{BW_{az}} = v_{eff}\frac{l}{0.8 \cdot 2v_{eff}} = \frac{l}{0.8 \cdot 2} = \frac{10\ m}{0.8 \cdot 2} = 6.25 \text{ m}$$

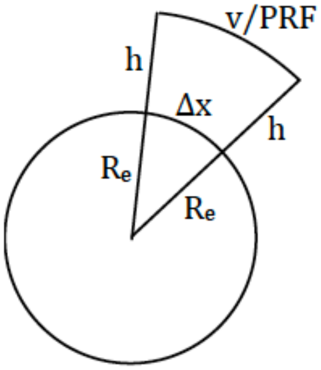


Figure 2: Spherical earth geometry to derive expression for azimuth ground pixel spacing.

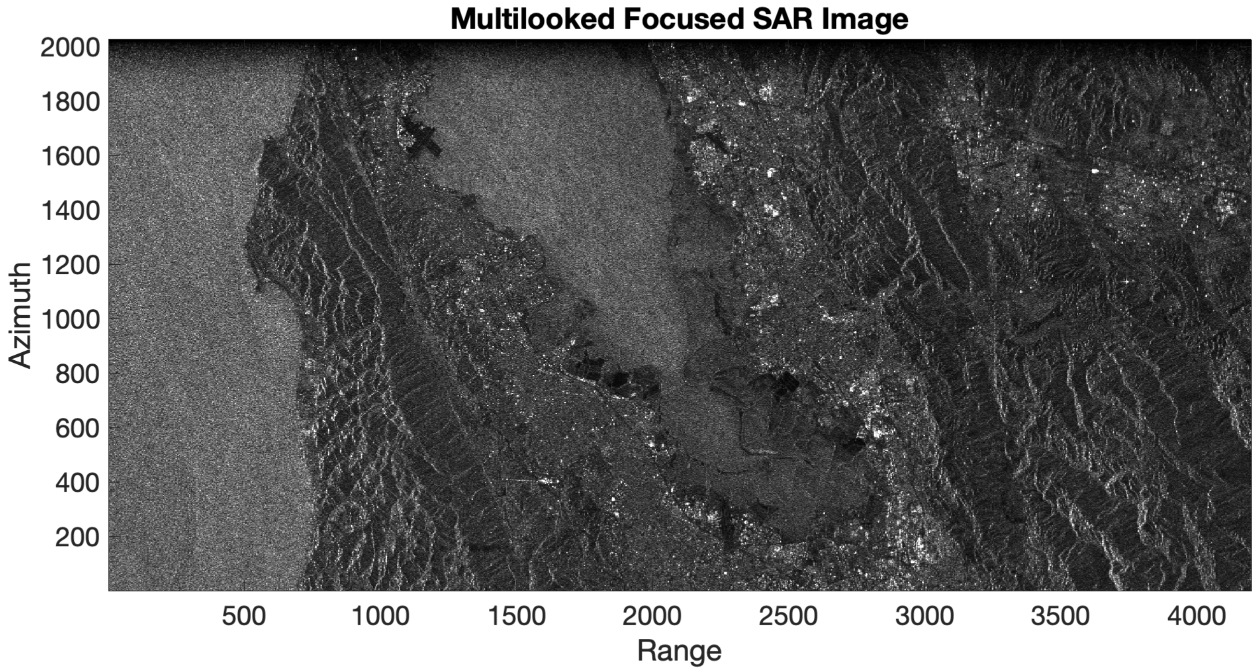
The azimuth ground pixel spacing is (deriving the expression from spherical earth geometry in Figure 2):

$$\Delta az_g = \frac{v}{prf}\left(\frac{R_e}{R_e+z}\right) = \frac{7550\ m/s}{1679.9\ Hz}\left(\frac{6378\ km}{6378\ km+770.71758\ km}\right) = 4.01 \text{ pixels}$$

d) Calculate looks to obtain approximately square pixels on the ground, and display the image. Submit electronically.

We need to take looks (average the amplitudes of the pixels) in the azimuth direction because the ground pixel spacing is much finer in azimuth than in range. Number of looks (rounding to the nearest integer):

$$n_{looks} = \frac{\Delta r_g}{\Delta az_g} = \frac{18.05\ m/pixel}{4.01\ m/pixel} = 5 \text{ looks}$$



Compare the image quality of the SAR processor with your unfocused processor from the previous homework assignment.

The focused image is much crisper, as the name implies. In the unfocused case, there was still some detail, but it looked a little blurred. Here, we can zoom in to even see the individual runways at SFO (looking at the original image when processed in matlab/python – it gets a bit blurrier when we convert to a png file). The bright spots in this image look sharper in azimuth and not so smeared out.

```
%% HW 4 - Focused SAR Processor

% Consider a Radar with the following parameters
clc; clearvars; close all;

%Range Modulation
s = 4.189166*10^11; %chirp slope (Hz/s)
tau = 37.12*10^-6; %pulse length (sec)
fs = 18.96*10^6; %sample rate (Hz)

%Other Parameters
prf = 1679.9; %Pulse-Repetition Frequency (Hz)
lambda = 0.0566; %wavelength (m)
r0_1 = 830000; %Range r0 for first bin (m)
vel = 7550; %platform velocity (m/s)
l = 10; %antenna length (m)
radius = 6378e3; %earth radius (m)
theta = 23; % Look angle in degrees

%% Question 1 - Part A
% How many valid range bins are found in the range compressed data,
% assuming as before the range record length is 10,218 bytes of
% which 412 are header bytes?
```

```
range = 10218; %number of columns in ers data
azimuth = 10100; %number of lines in ers data
nheader = 412; %number of header pixels

num = (range-nheader)/2; %length of returned echo (pixels)
nvalid = ceil(num-(tau*fs)); %valid range bins (pixels)

%% Question 1 - Part B
% What is the minimum fft size for range processing?

fft_length = num;
fft_length2 = pow2(ceil(log2(num))); % next power of 2

%% Question 1 - Part C
%What is the effective spacecraft velocity veff?

dt = 1/fs; % Sample spacing in time domain
df = fs/round(tau*fs); % Sample spacing in frequency domain
c = 2.99792458e8; %2.99792458e8; %speed of light (m/s)

theta = 23; % Look angle in degrees
deltar = c/2/fs; % Slant range bin spacing
r0_mid = r0_1 + deltar*nvalid/2; % middle range in m
alpha = asind(sind(theta)*r0_mid/radius); % angle from center of earth
third_angle = 180-(theta+alpha); % last angle in triangle
inc = 180-third_angle; % incidence angle

hgt = radius*sind(inc)/sind(theta) - radius; % Platform height (km)

v_eff = vel*sqrt(radius/(hgt+radius)); % Effective velocity

%% Question 1 - Part D
% What is the range bin spacing in meters?

% Slant range bin spacing and resolution
deltar = c/2/fs; % Slant range bin spacing
dr = 1/s/tau*c/2; % Slant range resolution

% Get the middle range and max range
r0_mid = r0_1 + deltar*nvalid/2; % middle range in m

% What are the ground spacings?
deltar_g = deltar/sind(inc); % ground range bin spacing
dr_g = dr/sind(inc); % Ground range resolution

%% Question 2: Download the raw data file ersdata.hw3 that we used last time.
% As before, the quantization level is 5 bits with average I and Q values
% of 15.5

% Open the Raw Data
fid = fopen('ersdata.hw3');
ERS = fread(fid,[range azimuth],'uint8');

% Create the Reference Chirp
```

```
chp = makechirp(s,tau,fs,0,1,fft_length);

chp_fft = fft(chp); % transform to frequency domain

sig = ERS(nheader+1:end,:); % Extract the non-header columns

sig_even = sig(2:2:end,:)-15.5; % Get complex component
sig_odd = sig(1:2:end,:)-15.5; % Get real component
signal = sig_odd + 1i*sig_even; % Combine real and complex components

% Range Compress the raw data
signal_fft = zeros(fft_length,azimuth);

sig_fft = fft(signal,[],1);

compressed_signal = zeros(fft_length,azimuth);
for k=1:azimuth
   compressed_signal(:,k) = ifft(sig_fft(:,k).*conj(chp_fft.')); % transform
back to time domain
end

cs_valid = compressed_signal(1:nvalid,:); %extract valid

figure(1)
imagesc(abs(cs_valid)');
title('Range Compressed Image');
xlabel('range'); ylabel('azimuth');
colormap('gray');
colorbar
axis on
axis image
cax = caxis;
caxis(0.5*cax);
%% Question 2 - Part A
% Determine the approximate Doppler Centroid of the Data

azimuth_fft = zeros(nvalid,azimuth);
for k=1:nvalid
    azimuth_fft(k,:) = fft(cs_valid(k,:));
end

% Calculate the dB magnitude
magnitude_azimuth = 20*log10(abs(azimuth_fft) + 1*10^-30);
mag_az = mean(magnitude_azimuth,1);
freq = linspace(-prf/2,prf/2,azimuth);

figure
plot(freq,fftshift(mag_az));
set(gca,'fontsize',20)
title('Average dB Magnitude Plot of Azimuth Spectrum');
xlabel('frequency (Hz)'); ylabel('Magnitude (dB)');
grid on

% Average Phase Shift Calculation

for k=1:nvalid
```

```
    bin=cs_valid(k,2:end).*conj(cs_valid(k,1:end-1));
    result(k)=sum(bin);
    phase(k)=atan(imag(result(k))/real(result(k)));
end

avgphase=mean(phase);
doppler_freq = avgphase/2/pi*prf;

disp(['The Doppler Centroid is Approximately ' num2str(doppler_freq) ' Hz']);

%% Question 2 - Part B
% Write a multipatch SAR processor for this data set. Set the azimuth
% beamwidth to 80% of the full illuminated beamwidth to keep the reference
% function below 1024 points. Process as many patches as you can from the
% data set given.

% Further stuff for azimuth compression, to get patch size and valid pixels
rmax = r0_1 + (nvalid-1)*deltar; % Maximum range
rdcmax = sqrt(rmax^2 + (doppler_freq*rmax*lambda/2/v_eff)^2); % Antenna
boresight range (m)
tauazmax = rdcmax*lambda/v_eff/l*0.8; % Azimuth illumination time (s)
Nrefmax=floor(tauazmax*prf); % Maximum number of azimuth samples in reference
chirp
Npatchsize = 2048; % Patch size, specified in the problem
Nvalidaz=Npatchsize-Nrefmax; % Number of valid data samples in aximuth per
patch (1125)

deltaaz=vel/prf*(radius/(radius+hgt)); % azimuth point spacing on the ground
daz = l/2/0.8; % azimuth resolution
azlook = round(deltar_g/deltaaz); % number of looks we need to take in
azimuth to get square pixels
Naveaz = floor(Nvalidaz/azlook); % number of lines per patch with looks

npatch = round(azimuth/Nvalidaz);
naztotal = Nvalidaz*npatch; % number of lines in the output image without
looks
data_new = zeros(nvalid,naztotal); % initialize outpud image matrix without
looks
numazave = Naveaz*npatch; % number of lines in the output image with looks
finalimage_ml = zeros(nvalid,numazave); % initialize output image matrix,
with looks

% Pad range compressed image with zeros - prevent out of bound errors
data_all = [cs_valid zeros(nvalid,Npatchsize)];

% Run the Azimuth Processing code and take looks in azimuth
for k = 1:npatch
    display(['Processing patch ' num2str(k) ' of ' num2str(npatch)]);
    for j = 1:nvalid
        rg = r0_1 + (k-1)*deltar;
        rdc = sqrt(rg^2 + (doppler_freq*rg*lambda/2/v_eff)^2); % Antenna bore
sight range
        tauaz = rdc*lambda/v_eff/l*0.8;
        frate = -2*v_eff^2/lambda/rdc;

        refaz = makechirp(frate,tauaz,prf,doppler_freq,1,Npatchsize);
        refspect = fft(refaz);
        signal = data_all(j,1 + (k-1)*Nvalidaz:Nvalidaz*(k-1) + Npatchsize);
        sigspect = fft(signal);
        newsig = ifft(sigspect.*conj(refspect));

        data_new(j,1+(k-1)*Nvalidaz:k*Nvalidaz) = newsig(1:Nvalidaz);
    end

    % Take looks in azimuth
    for j = 1:Naveaz
        startline = (k-1)*Nvalidaz+1;
        endline = k*Nvalidaz;
        finalimage_ml(:,j+(k-1)*Naveaz) = sum(abs(data_new(:,startline+(j-
1)*azlook:...
            startline+j*azlook - 1)),2);
    end
end

% Display the Results
figure
imagesc(abs(data_new)')
set(gca,'fontsize',20,'YDir','normal')
colormap('gray')
cax=caxis;
caxis(0.07*caxis);
title('Focused SAR Image, no Azimuth Looks')
xlabel('Range')
ylabel('Azimuth')

FinalImage = finalimage_ml';
figure
imagesc(FinalImage)
set(gca,'fontsize',20)
colormap('gray')
cax=caxis;
caxis(0.07*cax);
axis image
title('Multilooked Focused SAR Image')
xlabel('Range')
ylabel('Azimuth')
set(gca,'YDir','normal')

%%
function chirp = makechirp(s,tau,fs,fc,start,n)
%Function to compute chirp - reused in all problems
%s: slope
%tau: pulse length
%fs: sample rate
%fc: center frequency
%start: location of chirp
%n: the length of the chirp including zero
dt=1/fs;
npts=tau*fs;
t=(-npts/2:npts/2-1)*dt;
phase=pi*s*t.^2+2*pi*fc*t;
chirp=[zeros(1,start-1) exp(1i*phase) zeros(1,n-length(phase)-start+1)];
end
```