FMCW Velocity Estimation using Triangular chirp

In this script, range and velocity estimation using FMCW radar is performed using triangular chirps.

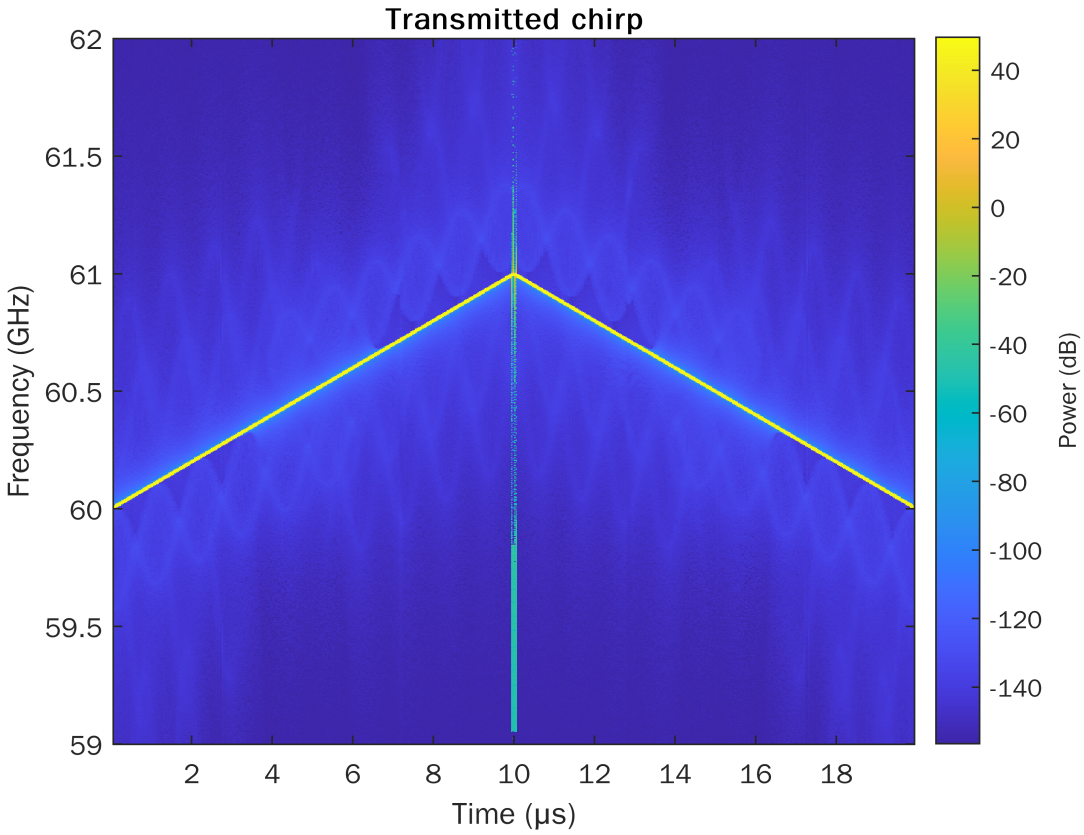
```
fs = 140e9; %Sampling frequency 10 GHz
ts = 1/fs; %Sampling interval
L = 14e5;
t0 = (0:L-1)*ts; %Time vector
t1 = (L:2*L-1)*ts;
t = [t0,t1];

l = length(t);
f = fs*(0:l/2)/l;
f_new = fs*(-l/2:l/2-1)/l;

f1 = 60*10^9; %Minimum frequency
f2 = 61*10^9; %Maximum frequency
K = (f2-f1)/((L-1)*ts); %Chirp slope

%transmitted chirps
x0 = 100*cos(2*pi*f1*t0 + pi*K*t0.^2);
x1 = 100*cos(2*pi*f2*t0 - pi*K*t0.^2);
x = [x0,x1];
c = 3*10^8;

lambda1 = c/f1;
d = 5.3; %Distance
td = 2*d/c; %Round trip delay

v_o = 1000 %original velocity
```

```
v_o = 1000
```

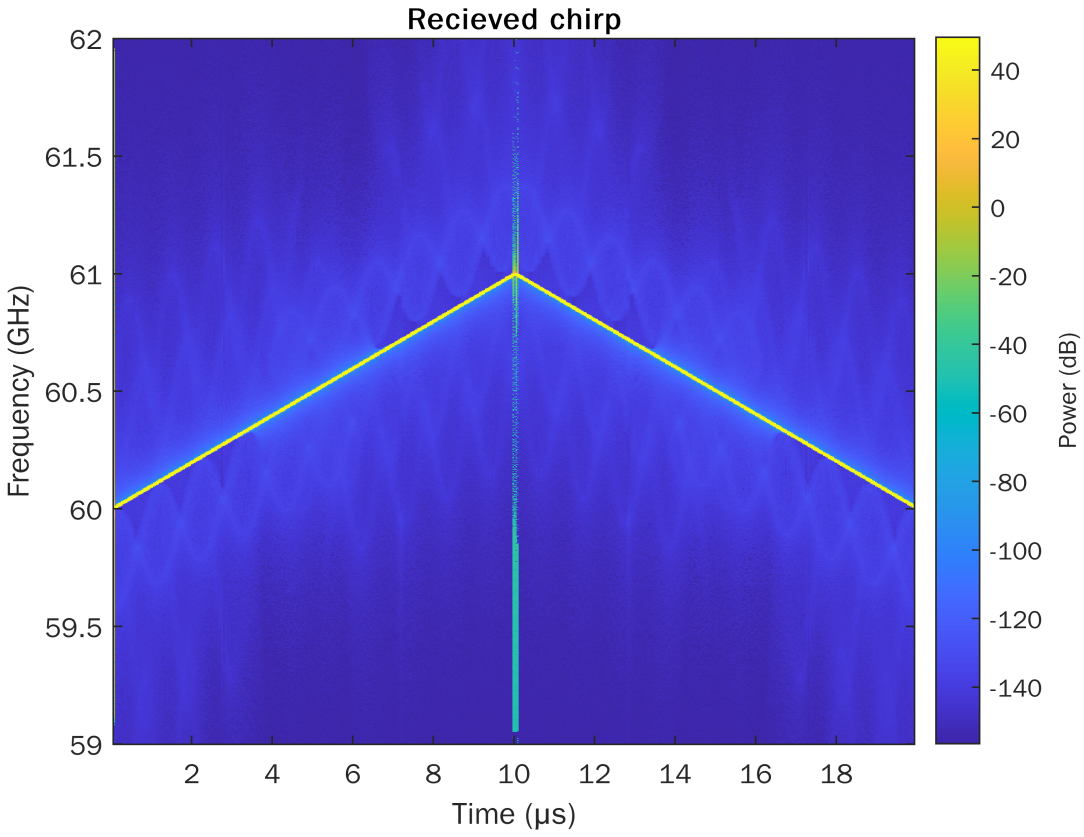
```
fd = 2*v_o/lambda1 %doppler shift due to velocity
```

```
fd = 400000
```

```

%Recieved chirps (without time delay but with doppler shifts)
s0 = 100*cos(2*pi*(f1+fd)*t0 + pi*K*t0.^2);
s1 = 100*cos(2*pi*(f2+fd)*t0 - pi*K*t0.^2);
s = [s0,s1];

y = (delayseq(s',td,fs))'; %recieved chirp


figure();
pspectrum(x,fs,'spectrogram','Reassign',true,'FrequencyLimits',[59*10^9,62*10^9],'Reass
title('Transmitted chirp');
```

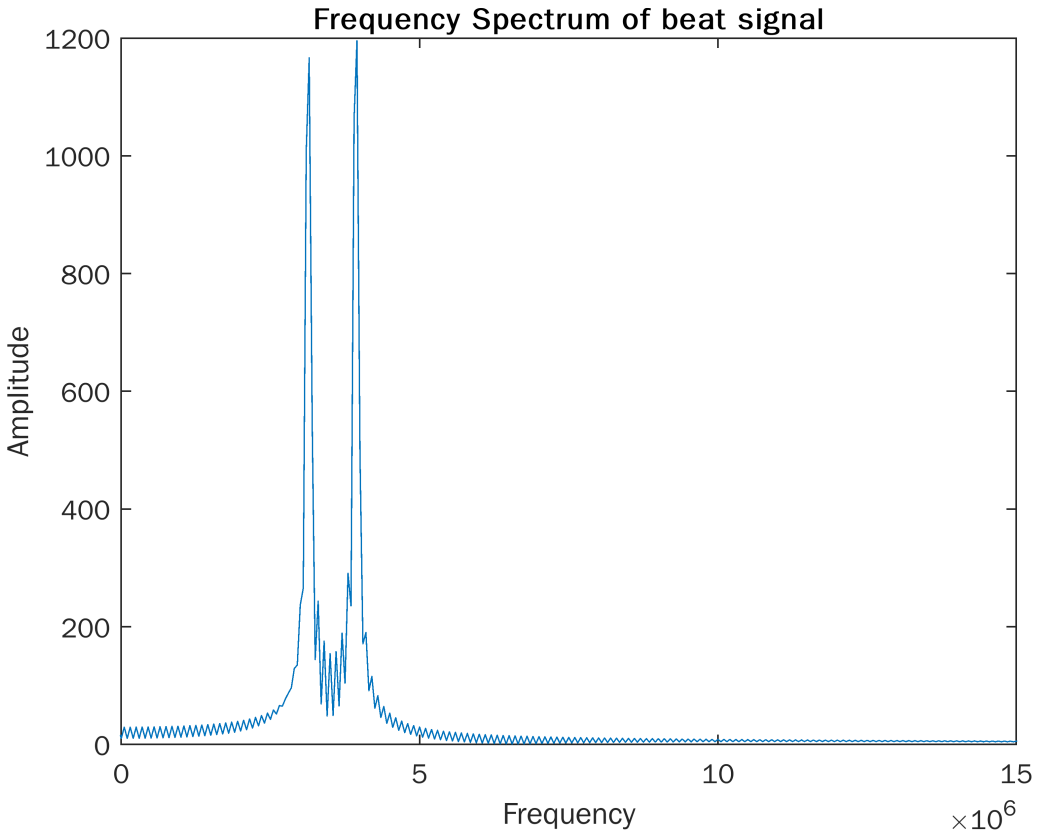



```
figure()
pspectrum(y,fs,'spectrogram','Reassign',true,'FrequencyLimits',[59*10^9,62*10^9],'Reass
title('Recieved chirp');
```



Dechirping

```
beat = dechirp(y',x'); %beat signal

beat_fft = (abs(fft(beat/l)));
P2 = (abs(fft(beat/l)));
P1 = P2(1:l/2+1);
P1(2:end-1) = 2*P1(2:end-1);
plot(f_new,fftshift(beat_fft));
xlabel("Frequency");ylabel("Amplitude");
title("Frequency Spectrum of beat signal");
xlim([0,1.5e7]);
```




```
[max_val,i] = max(P1)
```

```
max_val = 2.3911e+03
i = 80
```

```
f_beat_1 = f(i) %beat frequency
```

```
f_beat_1 = 3950000
```

```
P1(i) = 0;
[max_val,i] = max(P1)
```

```
max_val = 2.3335e+03
i = 64
```

```
f_beat_2 = f(i)
```

```
f_beat_2 = 3150000
```

```
f_beat = f_beat_1 - f_beat_2;
beat_f = f_beat_1 + f_beat_2;
velocity = lambda1*f_beat/4
```

```
velocity = 1000
```

```
velocity_error = abs(v_o-velocity)
```

```
velocity_error = 0
```

```
r = beat2range(beat_f,K,c)/2 %range
```

```
r = 5.3250
```

```
range_error = abs(r - d)
```

```
range_error = 0.0250
```