**FMCW Waveform**

Consider an automotive long range radar (LRR) used for adaptive cruise control (ACC). This kind of radar usually occupies the band around 77 GHz, as indicated in [1]. The radar system constantly estimates the distance between the vehicle it is mounted on and the vehicle in front of it, and alerts the driver when the two become too close. The figure below shows a sketch of ACC.



A popular waveform used in ACC system is FMCW. The principle of range measurement using the FMCW technique can be illustrated using the following figure.



The received signal is a time-delayed copy of the transmitted signal where the delay, , is related to the range. Because the signal is always sweeping through a frequency band, at any moment during the sweep, the frequency difference, , is a constant between the transmitted signal and the received signal.  is usually called the beat frequency. Because the sweep is linear, one can derive the time delay from the beat frequency and then translate the delay to the range.

In an ACC setup, the maximum range the radar needs to monitor is around 200 m and the system needs to be able to distinguish two targets that are 1 meter apart. From these requirements, one can compute the waveform parameters.

fc = 77e9;

c = 3e8;

lambda = c/fc;

The sweep time can be computed based on the time needed for the signal to travel the unambiguous maximum range. In general, for an FMCW radar system, the sweep time should be at least 5 to 6 times the round trip time. This example uses a factor of 5.5.

range\_max = 200;

tm = 5.5\*range2time(range\_max,c);

The sweep bandwidth can be determined according to the range resolution and the sweep slope is calculated using both sweep bandwidth and sweep time.

range\_res = 1;

bw = range2bw(range\_res,c);

sweep\_slope = bw/tm;

Because an FMCW signal often occupies a huge bandwidth, setting the sample rate blindly to twice the bandwidth often stresses the capability of A/D converter hardware. To address this issue, one can often choose a lower sample rate. Two things can be considered here:

1. For a complex sampled signal, the sample rate can be set to the same as the bandwidth.
2. FMCW radars estimate the target range using the beat frequency embedded in the dechirped signal. The maximum beat frequency the radar needs to detect is the sum of the beat frequency corresponding to the maximum range and the maximum Doppler frequency. Hence, the sample rate only needs to be twice the maximum beat frequency.

In this example, the beat frequency corresponding to the maximum range is given by

fr\_max = range2beat(range\_max,sweep\_slope,c);

In addition, the maximum speed of a traveling car is about 230 km/h. Hence the maximum Doppler shift and the maximum beat frequency can be computed as

v\_max = 230\*1000/3600;

fd\_max = speed2dop(2\*v\_max,lambda);

fb\_max = fr\_max+fd\_max;

This example adopts a sample rate of the larger of twice the maximum beat frequency and the bandwidth.

fs = max(2\*fb\_max,bw);

The following table summarizes the radar parameters.

System parameters Value

----------------------------------

Operating frequency (GHz) 77

Maximum target range (m) 200

Range resolution (m) 1

Maximum target speed (km/h) 230

Sweep time (microseconds) 7.33

Sweep bandwidth (MHz) 150

Maximum beat frequency (MHz) 27.30

Sample rate (MHz) 150

With all the information above, one can set up the FMCW waveform used in the radar system.

waveform = phased.FMCWWaveform('SweepTime',tm,'SweepBandwidth',bw,...

 'SampleRate',fs);

This is a up-sweep linear FMCW signal, often referred to as sawtooth shape. One can examine the time-frequency plot of the generated signal.

sig = waveform();

subplot(211); plot(0:1/fs:tm-1/fs,real(sig));

xlabel('Time (s)'); ylabel('Amplitude (v)');

title('FMCW signal'); axis tight;

subplot(212); spectrogram(sig,32,16,32,fs,'yaxis');

title('FMCW signal spectrogram');



**Target Model**

The target of an ACC radar is usually a car in front of it. This example assumes the target car is moving 50 m ahead of the car with the radar, at a speed of 96 km/h along the x-axis.

The radar cross section of a car, according to [1], can be computed based on the distance between the radar and the target car.

car\_dist = 43;

car\_speed = 96\*1000/3600;

car\_rcs = db2pow(min(10\*log10(car\_dist)+5,20));

cartarget = phased.RadarTarget('MeanRCS',car\_rcs,'PropagationSpeed',c,...

 'OperatingFrequency',fc);

carmotion = phased.Platform('InitialPosition',[car\_dist;0;0.5],...

 'Velocity',[car\_speed;0;0]);

The propagation model is assumed to be free space.

channel = phased.FreeSpace('PropagationSpeed',c,...

 'OperatingFrequency',fc,'SampleRate',fs,'TwoWayPropagation',true);

**Radar System Setup**

The rest of the radar system includes the transmitter, the receiver, and the antenna. This example uses the parameters presented in [1]. Note that this example models only main components and omits the effect from other components, such as coupler and mixer. In addition, for the sake of simplicity, the antenna is assumed to be isotropic and the gain of the antenna is included in the transmitter and the receiver.

ant\_aperture = 6.06e-4; % in square meter

ant\_gain = aperture2gain(ant\_aperture,lambda); % in dB

tx\_ppower = db2pow(5)\*1e-3; % in watts

tx\_gain = 9+ant\_gain; % in dB

rx\_gain = 15+ant\_gain; % in dB

rx\_nf = 4.5; % in dB

transmitter = phased.Transmitter('PeakPower',tx\_ppower,'Gain',tx\_gain);

receiver = phased.ReceiverPreamp('Gain',rx\_gain,'NoiseFigure',rx\_nf,...

 'SampleRate',fs);

Automotive radars are generally mounted on vehicles, so they are often in motion. This example assumes the radar is traveling at a speed of 100 km/h along x-axis. So the target car is approaching the radar at a relative speed of 4 km/h.

radar\_speed = 100\*1000/3600;

radarmotion = phased.Platform('InitialPosition',[0;0;0.5],...

 'Velocity',[radar\_speed;0;0]);

**Radar Signal Simulation**

As briefly mentioned in earlier sections, an FMCW radar measures the range by examining the beat frequency in the dechirped signal. To extract this frequency, a dechirp operation is performed by mixing the received signal with the transmitted signal. After the mixing, the dechirped signal contains only individual frequency components that correspond to the target range.

In addition, even though it is possible to extract the Doppler information from a single sweep, the Doppler shift is often extracted among several sweeps because within one pulse, the Doppler frequency is indistinguishable from the beat frequency. To measure the range and Doppler, an FMCW radar typically performs the following operations:

1. The waveform generator generates the FMCW signal.
2. The transmitter and the antenna amplify the signal and radiate the signal into space.
3. The signal propagates to the target, gets reflected by the target, and travels back to the radar.
4. The receiving antenna collects the signal.
5. The received signal is dechirped and saved in a buffer.
6. Once a certain number of sweeps fill the buffer, the Fourier transform is performed in both range and Doppler to extract the beat frequency as well as the Doppler shift. One can then estimate the range and speed of the target using these results. Range and Doppler can also be shown as an image and give an intuitive indication of where the target is in the range and speed domain.

The next section simulates the process outlined above. A total of 64 sweeps are simulated and a range Doppler response is generated at the end.

During the simulation, a spectrum analyzer is used to show the spectrum of each received sweep as well as its dechirped counterpart.

specanalyzer = dsp.SpectrumAnalyzer('SampleRate',fs,...

 'PlotAsTwoSidedSpectrum',true,...

 'Title','Spectrum for received and dechirped signal',...

 'ShowLegend',true);

Next, run the simulation loop.

rng(2012);

Nsweep = 64;

xr = complex(zeros(waveform.SampleRate\*waveform.SweepTime,Nsweep));

for m = 1:Nsweep

 % Update radar and target positions

 [radar\_pos,radar\_vel] = radarmotion(waveform.SweepTime);

 [tgt\_pos,tgt\_vel] = carmotion(waveform.SweepTime);

 % Transmit FMCW waveform

 sig = waveform();

 txsig = transmitter(sig);

 % Propagate the signal and reflect off the target

 txsig = channel(txsig,radar\_pos,tgt\_pos,radar\_vel,tgt\_vel);

 txsig = cartarget(txsig);

 % Dechirp the received radar return

 txsig = receiver(txsig);

 dechirpsig = dechirp(txsig,sig);

 % Visualize the spectrum

 specanalyzer([txsig dechirpsig]);

 xr(:,m) = dechirpsig;

end



From the spectrum scope, one can see that although the received signal is wideband (channel 1), sweeping through the entire bandwidth, the dechirped signal becomes narrowband (channel 2).

**Range and Doppler Estimation**

Before estimating the value of the range and Doppler, it may be a good idea to take a look at the zoomed range Doppler response of all 64 sweeps.

rngdopresp = phased.RangeDopplerResponse('PropagationSpeed',c,...

 'DopplerOutput','Speed','OperatingFrequency',fc,'SampleRate',fs,...

 'RangeMethod','FFT','SweepSlope',sweep\_slope,...

 'RangeFFTLengthSource','Property','RangeFFTLength',2048,...

 'DopplerFFTLengthSource','Property','DopplerFFTLength',256);

clf;

plotResponse(rngdopresp,xr); % Plot range Doppler map

axis([-v\_max v\_max 0 range\_max])

clim = caxis;



From the range Doppler response, one can see that the car in front is a bit more than 40 m away and appears almost static. This is expected because the radial speed of the car relative to the radar is only 4 km/h, which translates to a mere 1.11 m/s.

There are many ways to estimate the range and speed of the target car. For example, one can choose almost any spectral analysis method to extract both the beat frequency and the Doppler shift. This example uses the root MUSIC algorithm to extract both the beat frequency and the Doppler shift.

As a side note, although the received signal is sampled at 150 MHz so the system can achieve the required range resolution, after the dechirp, one only needs to sample it at a rate that corresponds to the maximum beat frequency. Since the maximum beat frequency is in general less than the required sweeping bandwidth, the signal can be decimated to alleviate the hardware cost. The following code snippet shows the decimation process.

Dn = fix(fs/(2\*fb\_max));

for m = size(xr,2):-1:1

 xr\_d(:,m) = decimate(xr(:,m),Dn,'FIR');

end

fs\_d = fs/Dn;

To estimate the range, firstly, the beat frequency is estimated using the coherently integrated sweeps and then converted to the range.

fb\_rng = rootmusic(pulsint(xr\_d,'coherent'),1,fs\_d);

rng\_est = beat2range(fb\_rng,sweep\_slope,c)

rng\_est =

 42.9976

Second, the Doppler shift is estimated across the sweeps at the range where the target is present.

peak\_loc = val2ind(rng\_est,c/(fs\_d\*2));

fd = -rootmusic(xr\_d(peak\_loc,:),1,1/tm);

v\_est = dop2speed(fd,lambda)/2

v\_est =

 1.0830

Note that both range and Doppler estimation are quite accurate.

**Range Doppler Coupling Effect**

One issue associated with linear FM signals, such as an FMCW signal, is the range Doppler coupling effect. As discussed earlier, the target range corresponds to the beat frequency. Hence, an accurate range estimation depends on an accurate estimate of beat frequency. However, the presence of Doppler shift changes the beat frequency, resulting in a biased range estimation.

For the situation outlined in this example, the range error caused by the relative speed between the target and the radar is

deltaR = rdcoupling(fd,sweep\_slope,c)

deltaR =

 -0.0041

This error is so small that we can safely ignore it.

Even though the current design is achieving the desired performance, one parameter warrants further attention. In the current configuration, the sweep time is about 7 microseconds. Therefore, the system needs to sweep a 150 MHz band within a very short period. Such an automotive radar may not be able to meet the cost requirement. Besides, given the velocity of a car, there is no need to make measurements every 7 microseconds. Hence, automotive radars often use a longer sweep time. For example, the waveform used in [2] has the same parameters as the waveform designed in this example except a sweep time of 2 ms.

A longer sweep time makes the range Doppler coupling more prominent. To see this effect, first reconfigure the waveform to use 2 ms as the sweep time.

waveform\_tr = clone(waveform);

release(waveform\_tr);

tm = 2e-3;

waveform\_tr.SweepTime = tm;

sweep\_slope = bw/tm;

Now calculate the range Doppler coupling.

deltaR = rdcoupling(fd,sweep\_slope,c)

deltaR =

 -1.1118

A range error of 1.14 m can no longer be ignored and needs to be compensated. Naturally, one may think of doing so following the same procedure outlined in earlier sections, estimating both range and Doppler, figuring out the range Doppler coupling from the Doppler shift, and then remove the error from the estimate.

Unfortunately this process doesn't work very well with the long sweep time. The longer sweep time results in a lower sampling rate across the sweeps, thus reducing the radar's capability of unambiguously detecting high speed vehicles. For instance, using a sweep time of 2 ms, the maximum unambiguous speed the radar system can detect using the traditional Doppler processing is

v\_unambiguous = dop2speed(1/(2\*tm),lambda)/2

v\_unambiguous =

 0.4870

The unambiguous speed is only 0.48 m/s, which means that the relative speed, 1.11 m/s, cannot be unambiguously detected. This means that not only the target car will appear slower in Doppler processing, but the range Doppler coupling also cannot be correctly compensated.

One way to resolve such ambiguity without Doppler processing is to adopt a triangle sweep pattern. Next section shows how the triangle sweep addresses the issue.

**Triangular Sweep**

In a triangular sweep, there are one up sweep and one down sweep to form one period, as shown in the following figure.



The two sweeps have the same slope except different signs. From the figure, one can see that the presence of Doppler frequency, , affects the beat frequencies ( and ) differently in up and down sweeps. Hence by combining the beat frequencies from both up and down sweep, the coupling effect from the Doppler can be averaged out and the range estimate can be obtained without ambiguity.

First, set the waveform to use triangular sweep.

waveform\_tr.SweepDirection = 'Triangle';

Now simulate the signal return. Because of the longer sweep time, fewer sweeps (16 vs. 64) are collected before processing.

Nsweep = 16;

xr = helperFMCWSimulate(Nsweep,waveform\_tr,radarmotion,carmotion,...

 transmitter,channel,cartarget,receiver);

The up sweep and down sweep are processed separately to obtain the beat frequencies corresponding to both up and down sweep.

fbu\_rng = rootmusic(pulsint(xr(:,1:2:end),'coherent'),1,fs);

fbd\_rng = rootmusic(pulsint(xr(:,2:2:end),'coherent'),1,fs);

Using both up sweep and down sweep beat frequencies simultaneously, the correct range estimate is obtained.

rng\_est = beat2range([fbu\_rng fbd\_rng],sweep\_slope,c)

rng\_est =

 42.9658

Moreover, the Doppler shift and the velocity can also be recovered in a similar fashion.

fd = -(fbu\_rng+fbd\_rng)/2;

v\_est = dop2speed(fd,lambda)/2

v\_est =

 1.1114

The estimated range and velocity match the true values, 43 m and 1.11 m/s, very well.

**Two-ray Propagation**

To complete the discussion, in reality, the actual signal propagation between the radar and the target vehicle is more complicated than what is modeled so far. For example, the radio wave can also arrive at the target vehicle via reflections. A simple yet widely used model to describe such a multipath scenario is a two-ray model, where the signal propagates from the radar to the target vehicle via two paths, one direct path and one reflected path off the road, as shown in the following figure.



The reflection off the road impacts the phase of the signal and the receiving signal at the target vehicle is a coherent combination of the signals via the two paths. Same thing happens on the return trip too where the reflected signal off the target vehicle travels back to the radar. Hence depending on the distance between the vehicles, the signals from different paths may add constructively or destructively, making signal strength fluctuating over time. Such fluctuation can pose some challenge in the successive detection stage.

To showcase the multipath effect, next section uses the two ray channel model to propagate the signal between the radar and the target vehicle.

txchannel = phased.TwoRayChannel('PropagationSpeed',c,...

 'OperatingFrequency',fc,'SampleRate',fs);

rxchannel = phased.TwoRayChannel('PropagationSpeed',c,...

 'OperatingFrequency',fc,'SampleRate',fs);

Nsweep = 64;

xr = helperFMCWTwoRaySimulate(Nsweep,waveform,radarmotion,carmotion,...

 transmitter,txchannel,rxchannel,cartarget,receiver);

plotResponse(rngdopresp,xr); % Plot range Doppler map

axis([-v\_max v\_max 0 range\_max]);

caxis(clim);



With all settings remaining same, the comparison of the resulting range-Doppler map with two-ray propagation and the range-Doppler map obtained before with a line of sight (LOS) propagation channel suggests that the signal strength dropped almost 40 dB, which is significant. Therefore, such effect must be considered during the design. One possible choice is to form a very sharp beam on the vertical direction to null out the reflections.

**Summary**

This example shows how to use FMCW signal to perform range and Doppler estimation in an automotive automatic cruise control application. The example also shows how to generate a range Doppler map from the received signal and discussed how to use triangle sweep to compensate for the range Doppler coupling effect for the FMCW signal. Finally, the effect on the signal level due to the multipath propagation is discussed.

**References**

[1] Karnfelt, C. et al.. *77 GHz ACC Radar Simulation Platform*, IEEE International Conferences on Intelligent Transport Systems Telecommunications (ITST), 2009.

[2] Rohling, H. and M. Meinecke. *Waveform Design Principle for Automotive Radar Systems*, Proceedings of CIE International Conference on Radar, 2001.