Linear Frequency Modulation Pulse Compression Technique on Generic Signal Model

Uttam K. Majumder

ABSTRACT

This paper examines the Linear Frequency Modulation (LFM) pulse compression technique on a generic signal model. Pulse compression allows achieving the performance of a shorter pulse using a longer pulse and hence gain of a large spectral bandwidth. The pulse compression technique plays a very important role for designing a radar system. Since a short pulse requires a high peak power which is unattainable for many constraints such as voltage breakdown, dimension of waveguide etc, the radar system uses a longer pulse and pulse compression technique. For high range resolution radar, the need for pulse compression is inevitable. The focus of this paper is time frequency autocorrelation and ambiguity functions' role in waveform design and then application of LFM pulse compression technique to a generic signal waveform.

Keywords: LFM, pulse compression, Time Frequency Autocorrelation Function (TFACF), Ambiguity Function (AF)

1. INTRODUCTION

One fundamental issue in designing a good radar system is it's capability to resolve two small targets that are located at long range with very small separation between them. This requires a radar system to transmit a long pulse which will have enough energy to detect a small target at long range. However, a long pulse degrades range resolution. Hence, frequency or phase modulation of the signal is used to achieve a high range resolution when a long pulse is required. The capabilities of short-pulse and high range resolution radar are significant [1] p.342. For example, high range resolution allows resolving more than one target with good accuracy in range without using angle information. Many other applications of short-pulse and high range resolution radar are clutter reduction, glint reduction, multipath resolution, target classification, and Doppler tolerance.

There are several methods of pulse compression that have been used in the past. The most popular of them is linear frequency modulation (LFM) which was invented by R.H. Dickie in 1945 [1]. The other pulse compression techniques are Binary phase codes, Polyphase codes, Barker codes, Costas codes, Nonlinear Frequency Modulation etc. In this research, we developed Matlab code to study a generic ambiguity function waveform model and the LFM pulse compression technique with chirp diversity. All results in this paper correspond to the simulation parameters found in Table 1 unless otherwise noted. We verified expected results for an LFM transmit waveform from the ambiguity surface plots with the EENG 668 course notes p.69, figure 20. This paper has been organized in the following manner: time frequency autocorrelation and ambiguity functions' role in waveform design, expected result for an uncompressed transmitted waveform, LFM pulse compression technique, result of LFM pulse compression technique, Doppler tolerance issue of LFM signal, and finally aliasing issues.

Table 1: Simulation parameters for model verification.

Parameter	Value
PRI T_r	3
Pulse Width τ	1
Number of Pulses M	1 and 2
Number of Chips P	1
Number of Chip Points	8
Continuous Amplitude Weighting $a(t)$	$a(t)$=1
Discrete Amplitude Weighting W_{mp}	W_{mp} = 1

2. TIME-FREQUENCY AUTOCORRELATION AND AMBIGUITY FUNCTIONS' ROLE IN WAVEFORM DESIGN

Suppose a signal $S(t)$ is transmitted from a radar system. If there is no range delay or frequency shift, the matched filter output of the received signal will be exactly the same as the transmitted signal. However, in a practical radar system there is always range delay and /or Doppler shift. Therefore analysis must be done for the case when there is received signal mismatch with the transmit signal. The time-frequency autocorrelation function describes matched filter output when the transmit signal does not match with the received signal in either Doppler or time delay. From the EENG 668 course notes (pp. 13-15), the following equations mathematically describe the matched filter output, TFACF, and AF:

$$y(\lambda) = \int_{-\infty}^{\infty} r(t) s^{*}(\lambda - t) dt \qquad (1)$$

$$\Phi(T_R, f_d) = y(T_R, f_d) = \int_{-\infty}^{\infty} s(t - T_R) s^{*}(t) e^{j2\pi f_d t} dt \qquad (2)$$

$$\chi(T_R, f_d) = \left|\Phi(T_R, f_d)\right|^2 \qquad (3)$$

In (1) – (3), $r(t)$ is received signal, $s^{*}(-t)$ is the matched filter impulse response, T_R is range delay, f_d is Doppler frequency shift.

If we evaluate AF at $(T_R, f_d) = (0,0)$, we will find that the matched filter output is perfectly matched with received signal. If we evaluate AF where (T_R, f_d) is nonzero, we will get the matched filter output of a received signal with range delay and/or Doppler shift.

3. EXPECTED RESULTS FOR AN UNCOMPRESSED TRANSMIT WAVEFORM

If we plot the ambiguity function of a single pulse of an uncompressed waveform, it will look like a ridge. The result of the simulation is shown below:

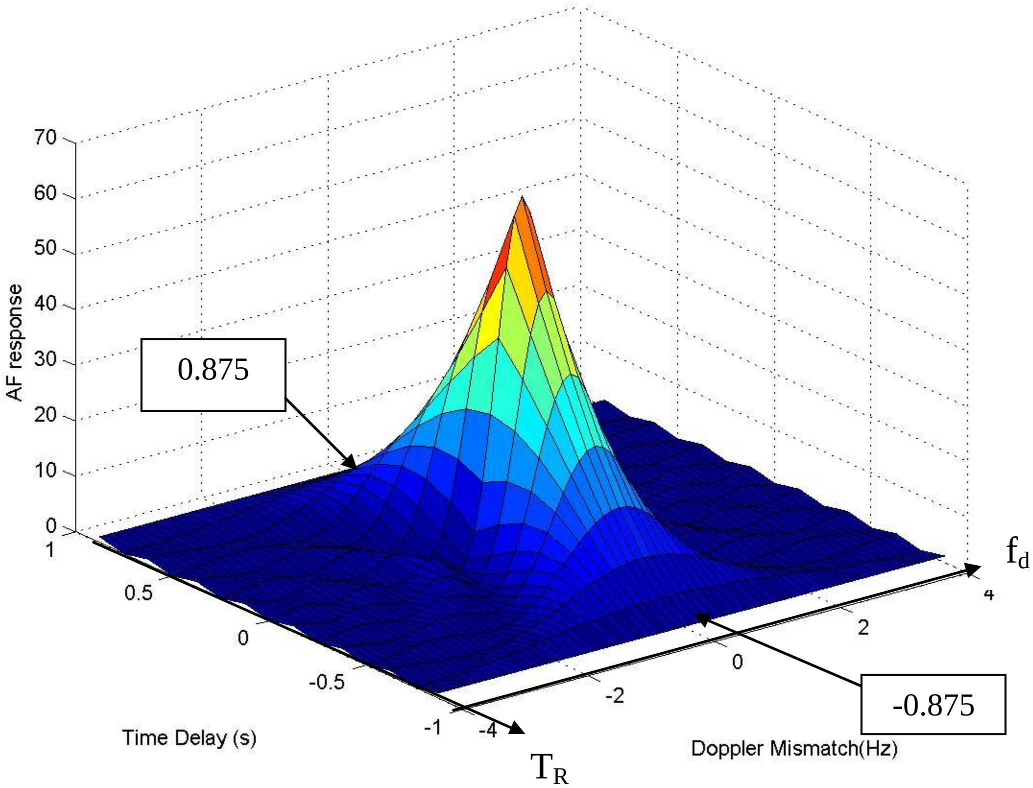

Figure 1: Ambiguity surface with Matlab's surf command
Pulse width [-0.875   0.875] and Frequency [-4 4]

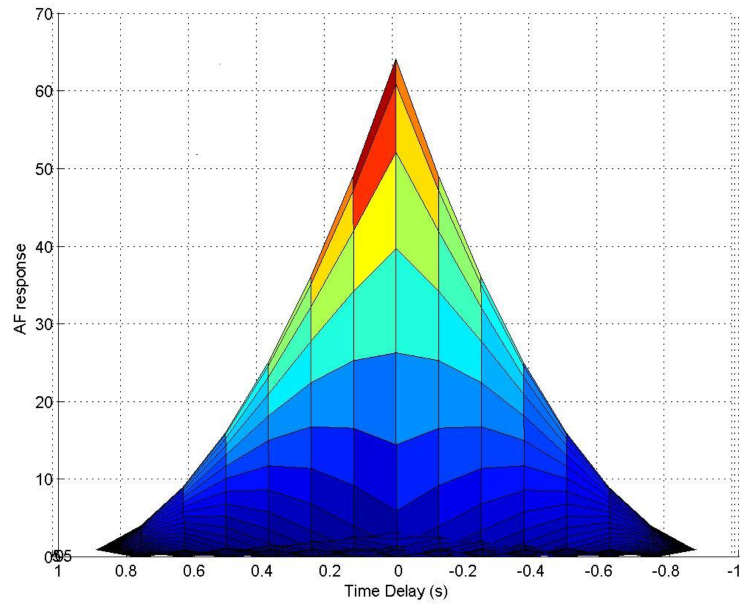

Figure 2: Ambiguity Surface cut along Time Delay (sec)

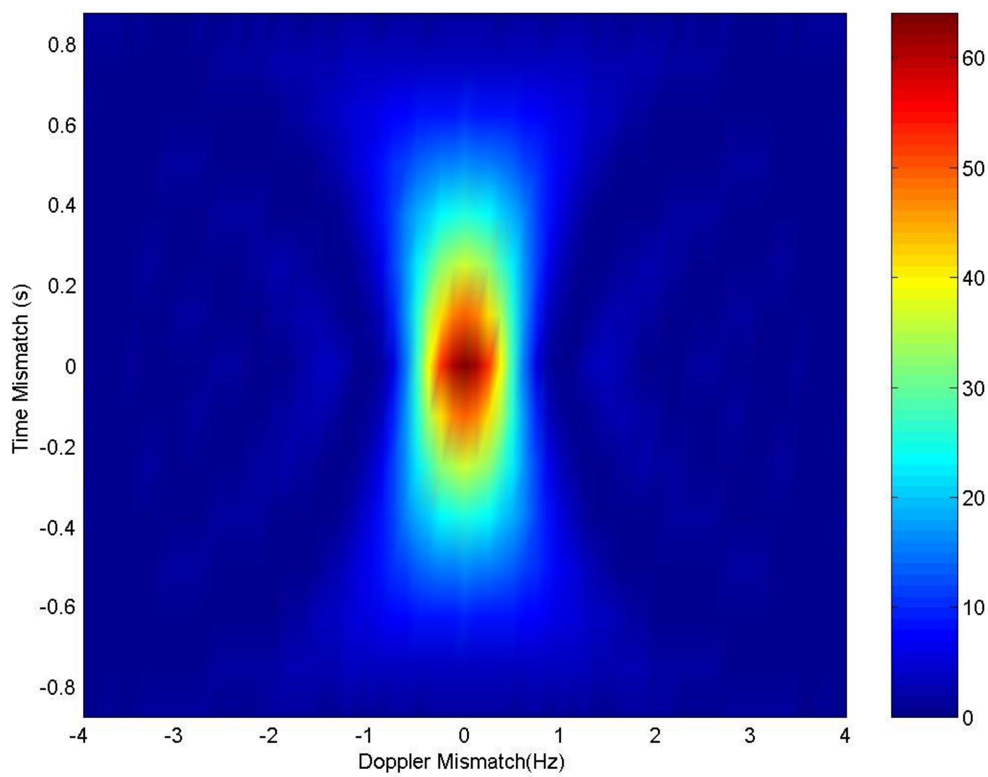

Figure 3: Ambiguity surface using Matlab's pcolor command

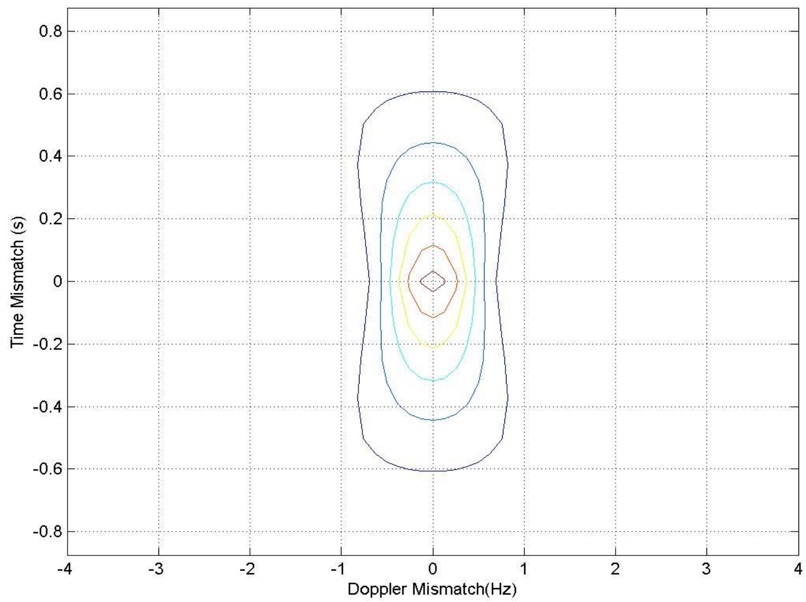

Figure 4: Ambiguity surface with Matlab'contour command

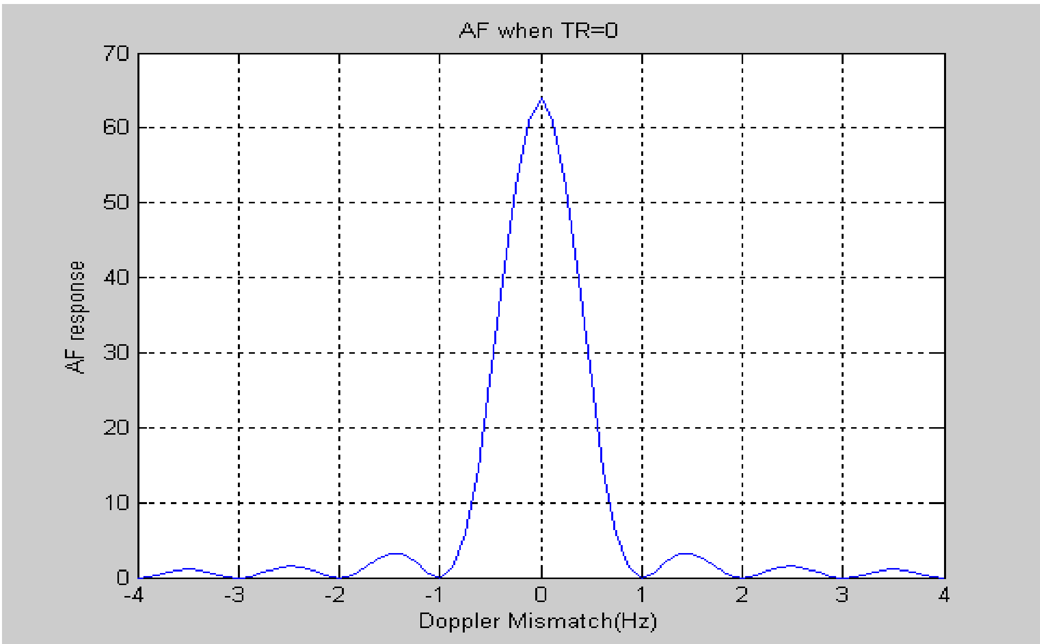

Figure 5: Ambiguity Function surface cut when $T_R = 0$.
This has been derived from magnitude square of TFACF

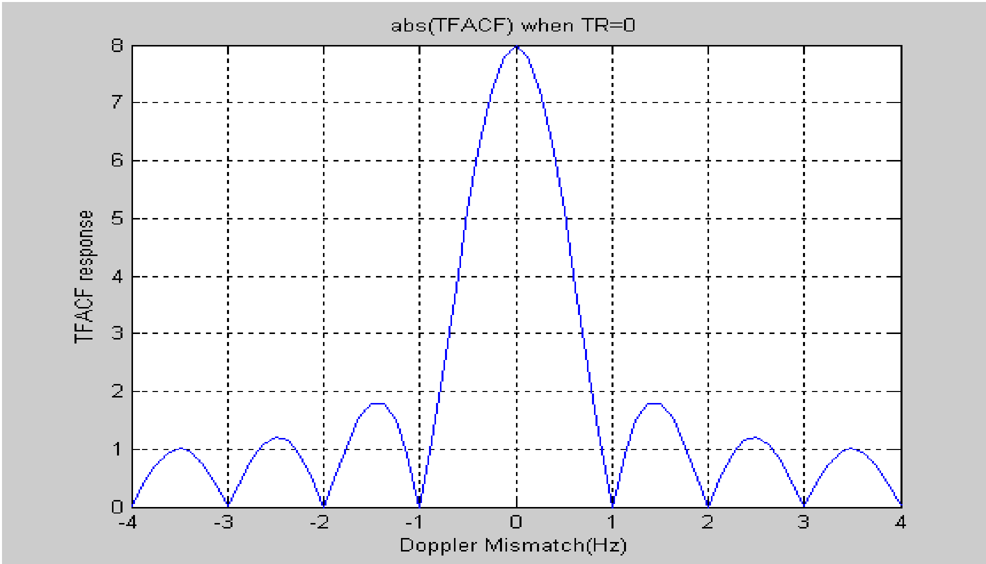

Figure 6: Ambiguity Function surface cut when $T_R = 0$.
This has been derived from absolute value of TFACF

When $f_d = 0$, equation (2) becomes $\Phi(T_R, 0) = y(T_R, 0) = \int_{-\infty}^{\infty} s(t - T_R)s^*(t)dt$.
If we plot the above equation, we observe the triangular function on the time axis.

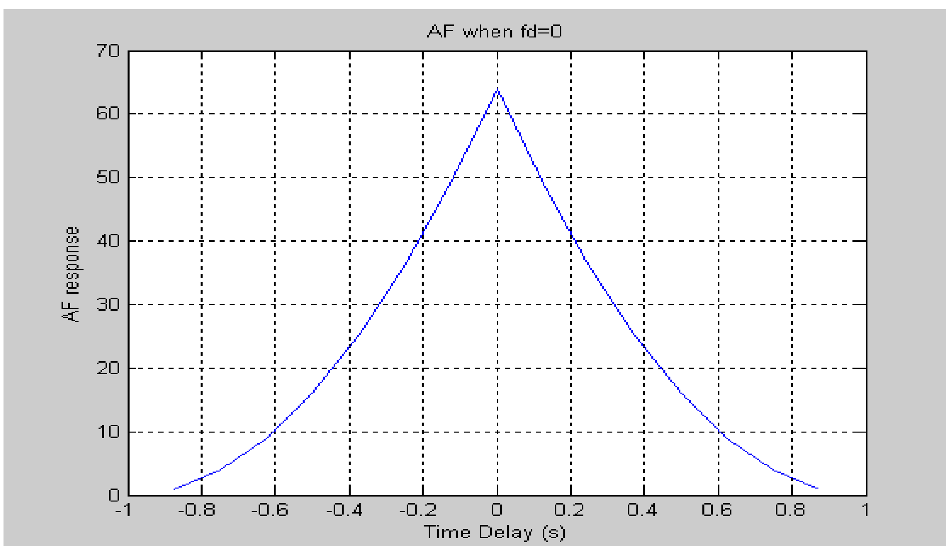

Figure 7: Ambiguity Function surface cut when $f_d = 0$.
This has been derived from magnitude square of TFACF

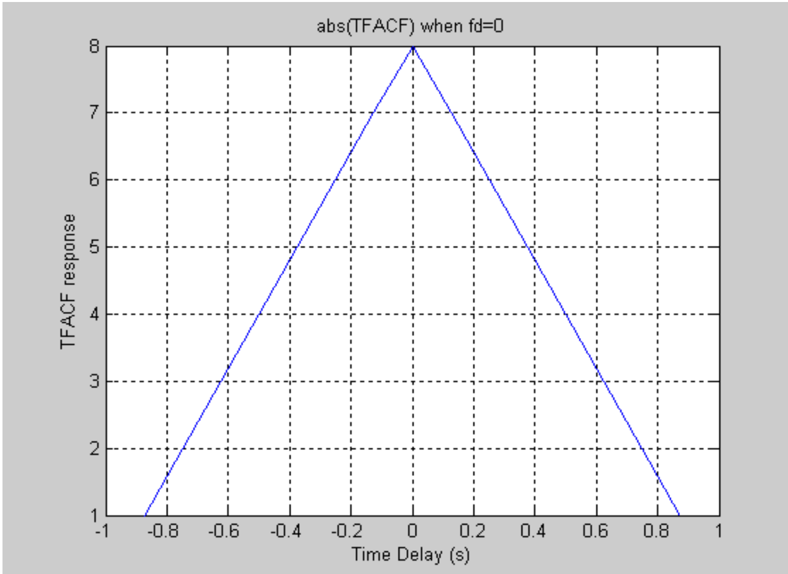

Figure 8: Ambiguity Function surface cut when $f_d = 0$.
This has been derived from absolute value of TFACF

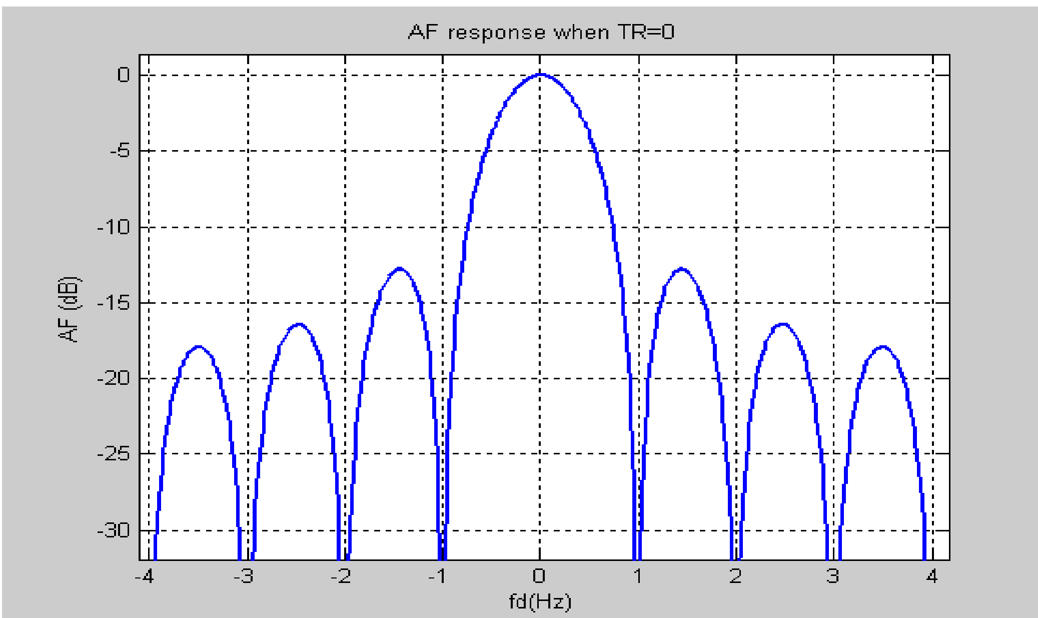

Figure 9: Normalized Ambiguity Function surface cut when $T_R = 0$.
From the definition, AF is calculated by taking magnitude square of TFACF.

4. LFM PULSE COMPRESSION TECHNIQUE

An LFM signal is a kind of signal in which the frequency of the transmitted signal is varied over a pulse duration of T_P. This variation of the frequency from low to high or vice a versa is known as “chirping”. Changing the frequency from low to high is called “up-chirp” or upsweep [3]. Similarly, changing the frequency from high to low is called called “down-chirp”. The technique of applying a different chirp rate for each pulse is known as “chirp diversity”., Following is a brief mathematical description of an LFM signal derived from EENG 668 course notes and Sumekh’s text book [3].

Consider for the transmitted frequency of f_0 and chirp slope b, the phase function is $\phi(t) = f_0 t + bt^2$. By taking the derivative of the phase function, instantaneous frequency can be calculated as $\omega_i(t) = f_0 + 2bt$. For a chirp pulse width $t \in [0, T_P]$, $\omega_i(0) = f_0$ is the minimum frequency and $\omega_i(T_P) = f_0 + 2bT_P$, is the maximum frequency. Therefore, bandwidth of the chirp pulse is, $\omega_i(T_P) - \omega_i(0) = 2bT_P$.

Following is a mathematical description of single pulse LFM, using a unit pulse function *u(t)* of width T_P derived from the EENG 668 course notes.

$$S(t) = u(t)e^{(j2\pi(f_0 t + bt^2))}$$

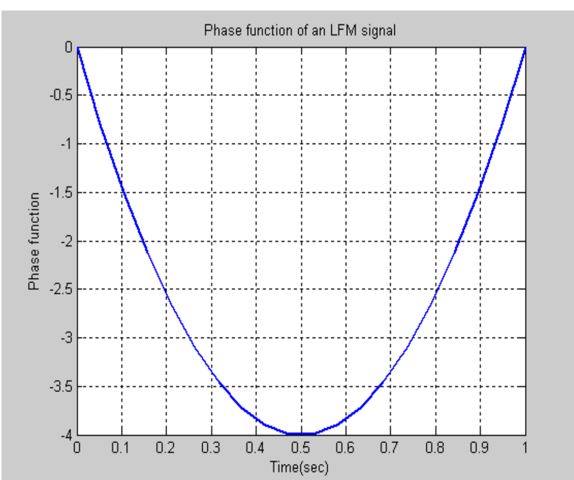

Figure 10: LFM Signal Phase function with $f_0 = -16, b = 16$

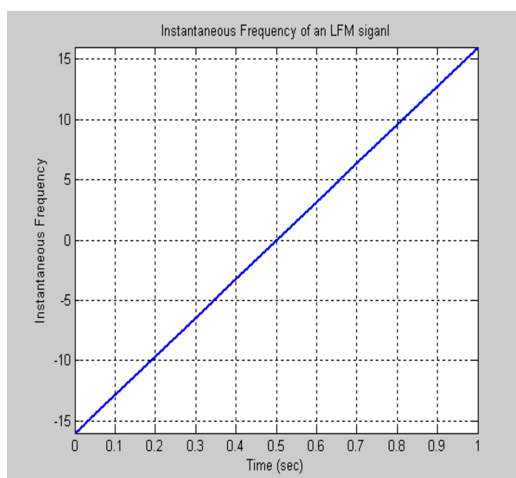

Figure 11: LFM signal instantaneous function with $f_0 = 16, b = 16$

5. RESULT OF LFM PULSE COMPRESSION TECHNIQUE

Figure 12 and 13 shows ambiguity surface of an LFM pulse compressed signal.

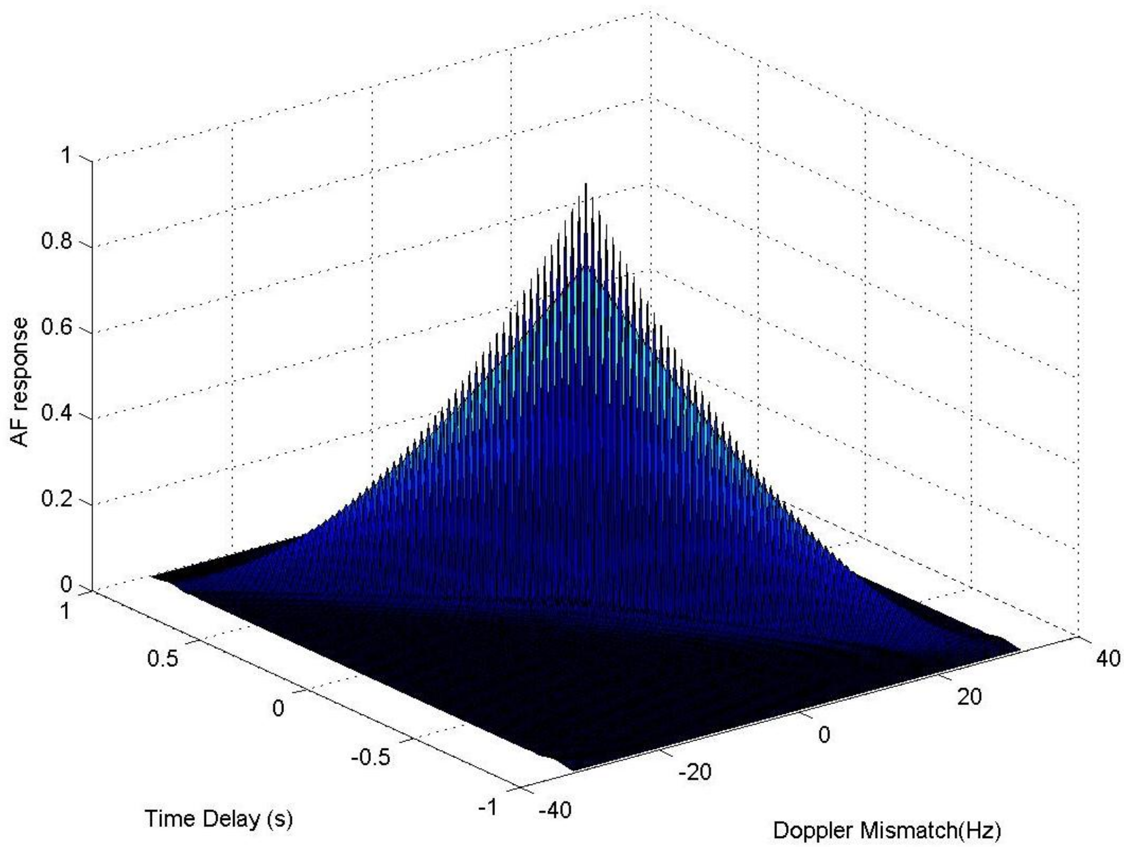

Figure 12: Ambiguity surface with Matlab's surf command. Transmitted frequency f0=-16 Hz, chirp slope b=16, Number of Pulse M=1, Number of Chip Points NCps=64

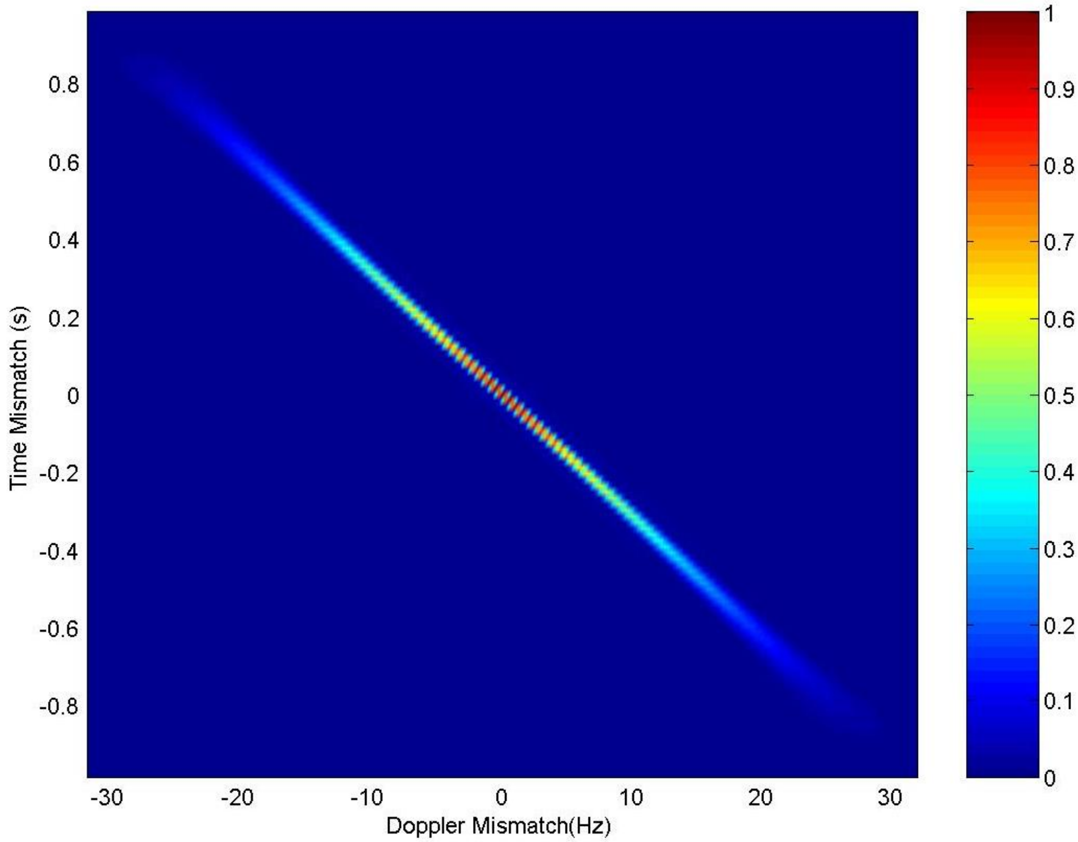

Figure 13: Ambiguity surface with Matlab's pcolor command. Transmitted frequency f0=-16 Hz, chirp slope b=16, Number of Pulse M=1, Number of Chip Points NCps=64

Figure 14 and 15 shows ambiguity surface of an LFM pulse compressed signal with chirp diversity.

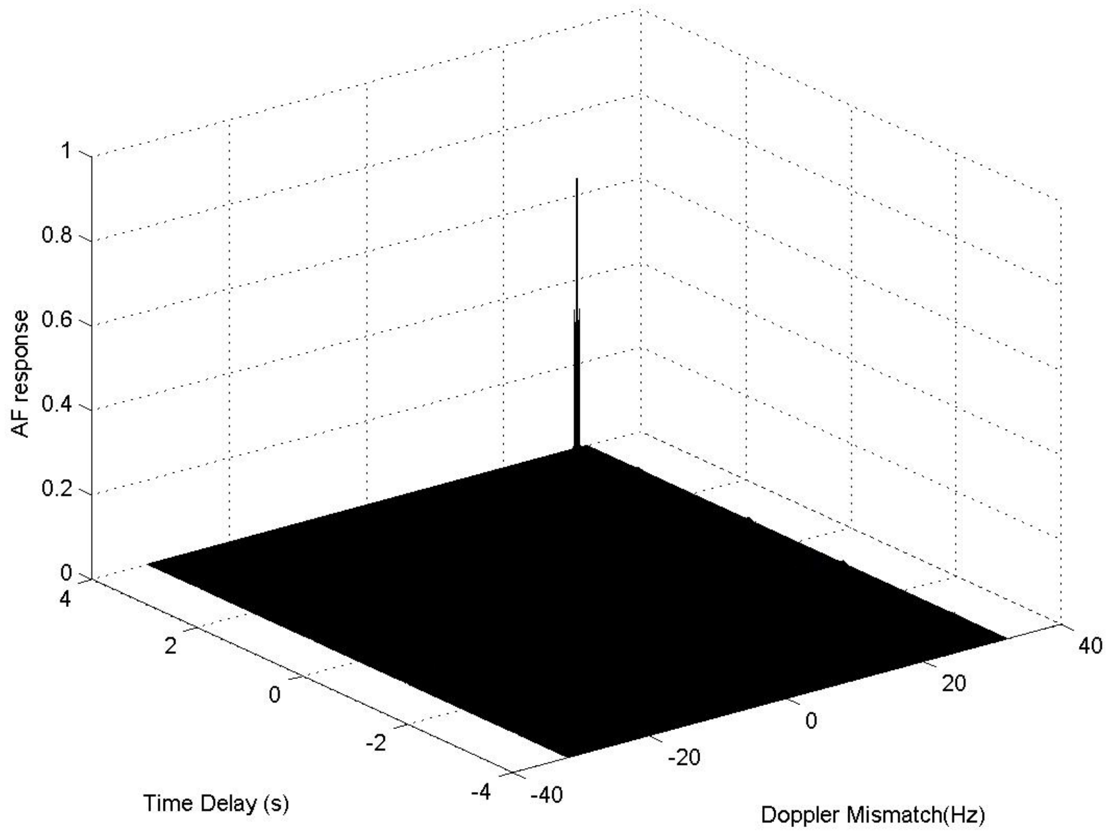

Figure 14: Ambiguity surface with Matlab's surf command. Transmitted frequencies f0 = [-16, 16] Hz, chirp slope b= [16, -16], Number of Pulses M=2, Number of Chip Points NCps=64

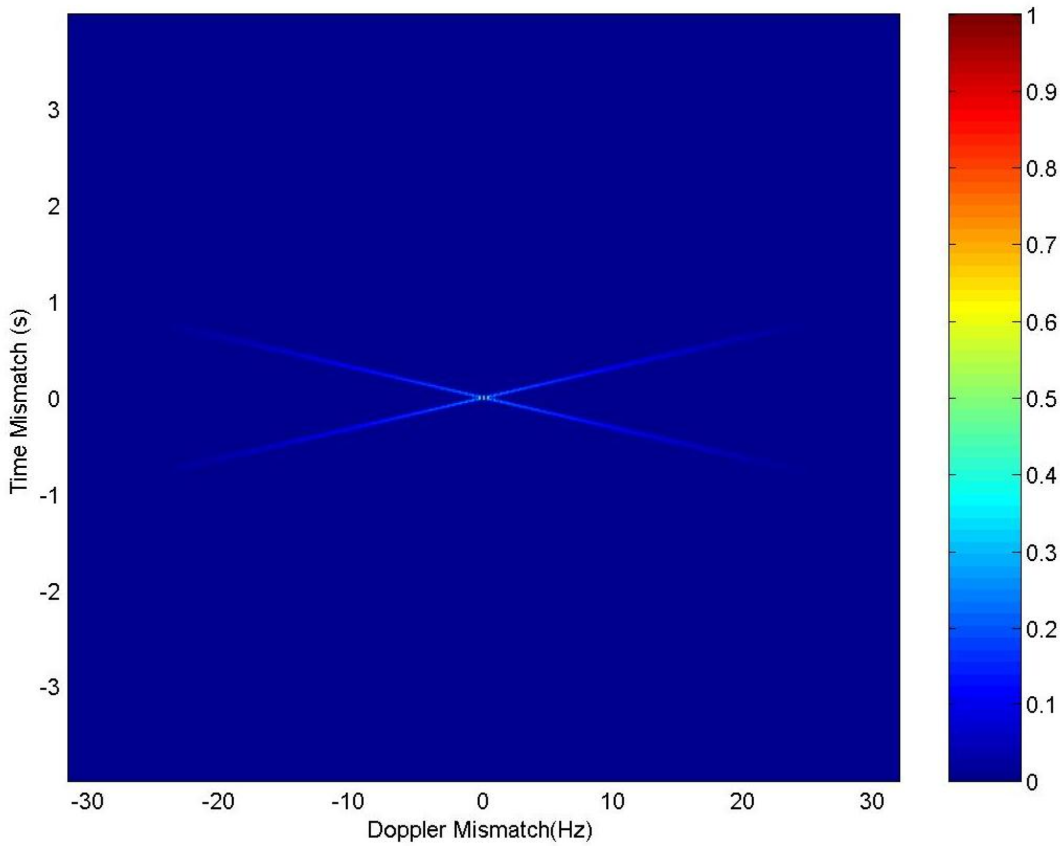

Figure 15: Ambiguity surface with Matlab's pcolor command. Transmitted frequencies f0= [-16, 16]Hz, chirp slope b=[16, -16], Number of Pulses M=2, Number of Chip Points NCps=64

Figure 16 and 17 shows ambiguity surface of an LFM pulse compressed signal with an increasing number of chip points (NCps=256)

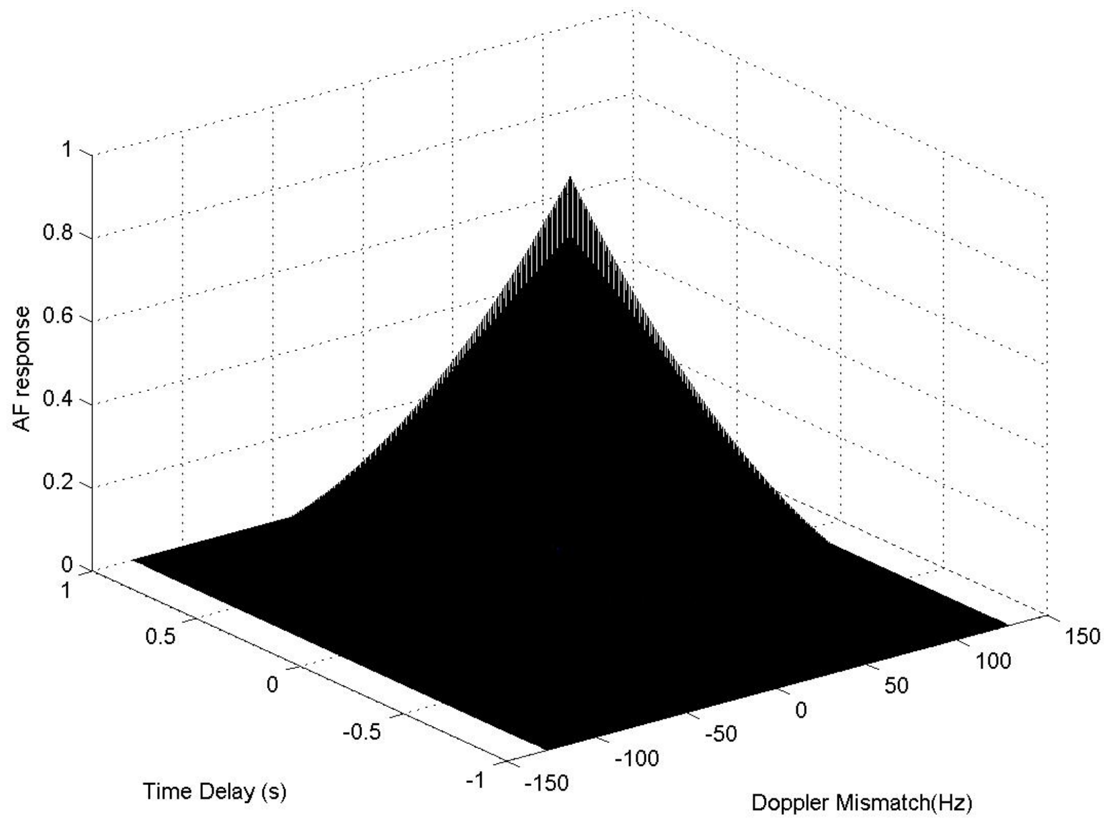

Figure 16: Ambiguity surface with Matlab's surf command. Transmitted frequency f0=-64Hz, chirp slope b=64, Number of Pulse M=1, Number of Chip Points NCps=256

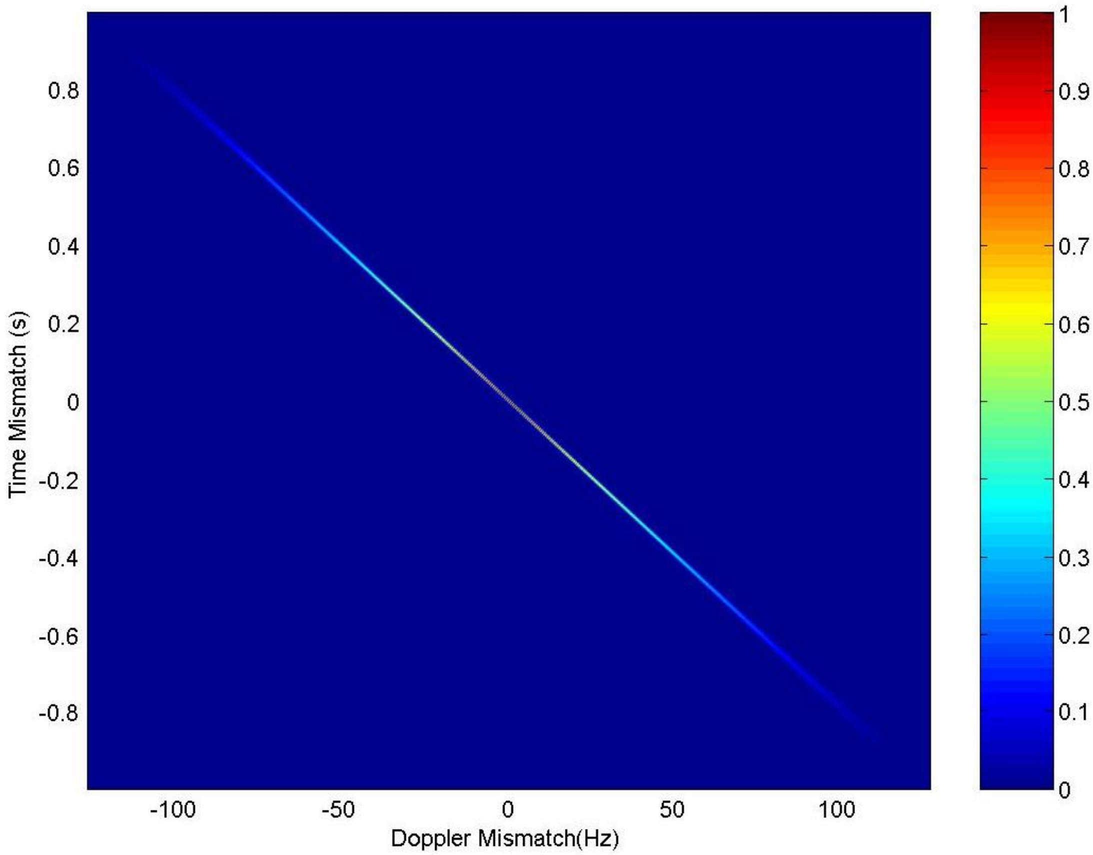

Figure 17: Ambiguity surface with Matlab's pcolor command. Transmitted frequency f0=-64Hz, chirp slope b=64, Number of Pulse M=1, Number of Chip Points NCps=256

6. RANGE RESOLUTION OF COMPRESSED (LFM) AND UNCOMPRESSED WAVEFORM

Consider an LFM waveform with bandwidth B = 0.5GHz and pulse width τ = 10ms. The range resolution of an uncompressed waveform is given by, $R_{uncomp} = \frac{c\tau}{2}$ where c is the speed of light and τ is the pulse width. To calculate the compressed range resolution, we need to calculate the compressed pulse width τ_{comp} which is calculated as $\frac{1}{B}$.

Therefore, compressed range resolution is given by $R_{comp} = \frac{c\tau_{comp}}{2}$.

Therefore, $R_{uncomp} = \frac{3x10^8 x10x10^{-3}}{2} = 1.5x10^6 \, meters$

And $R_{comp} = \dfrac{3x10^8\, x \dfrac{1}{0.5x10^9}}{2} = \dfrac{6x10^{-1}}{2} = \dfrac{3}{10} = 0.3 meter = 30cm$

For B=1GHz, $R_{comp} = 0.15 meter = 15cm$.

The Pulse Compression Ratio (PCR) is defined as $\dfrac{\tau}{(1/B)} = \tau B$. Therefore, as bandwidth B of a radar system is increased, and PCR is also increased, then a better range resolution could be achieved.

7. DOPPLER TOLERANCE OF THE LFM SIGNAL

Skolnik [1] p.335 describes Doppler tolerance as a measure of whether or not a single matched filter will be enough to produce a good output when there will be a case of large Doppler shift. Short pulses are Doppler tolerant; on the other hand, long pulses are not Doppler tolerant. Therefore a pulse compression technique should alleviate Doppler tolerant issue associated with the longer pulse and hence the LFM signal should be Doppler tolerant. However, there is a range-Doppler coupling issue inherent to LFM signal [1]. A range-Doppler coupling occurs due to a large Doppler shift. So the computed range may not be the true range of the target. Therefore, depending on the application and situation, an LFM signal may or may not be Doppler tolerant. In many applications, if the Doppler shift is small, range error is also small and hence this error could be ignored. To mitigate the range-Doppler coupling, chirp diverse LFM is used.

8. ALIASING ISSUE

In this simulation, we have to aware of the aliasing issues. Consider the uncompressed signal when $T_R = 0$. In this case equation (2) becomes

$$\Phi(0, f_d) = y(0, f_d) = \int_{-\infty}^{\infty} s(t) s^*(t) e^{j2\pi f_d t} dt .$$

If we plot the above equation, we observe the sinc function on the frequency axis.

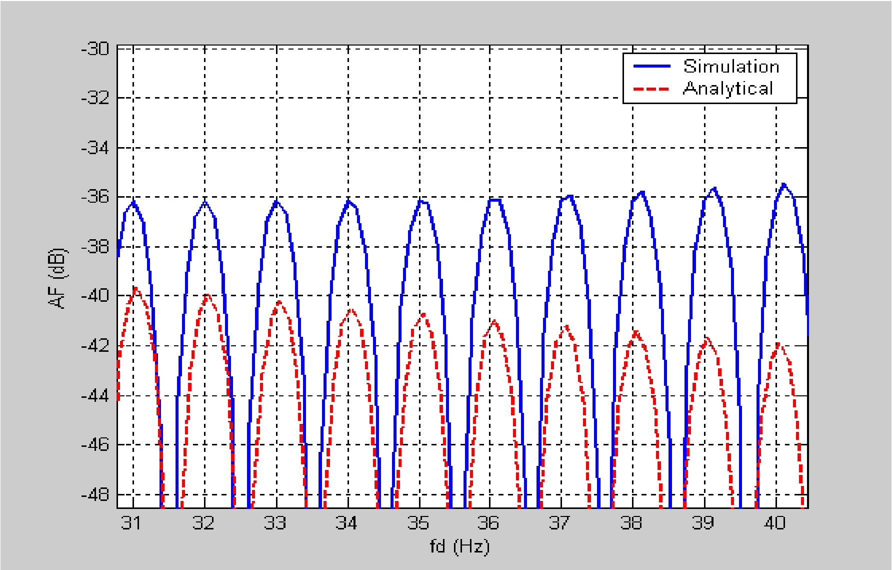

Figure 18: Aliasing effects. For TR=0 cut, we see the analytical and simulation values for the ambiguity surface. Aliasing is evident in the simulated result since sidelobes do not decrease below a certain level.

To alleviate this problem, we increase number of chip points. However, this creates computational burden. For a large number of chip points, this issue will be significant.

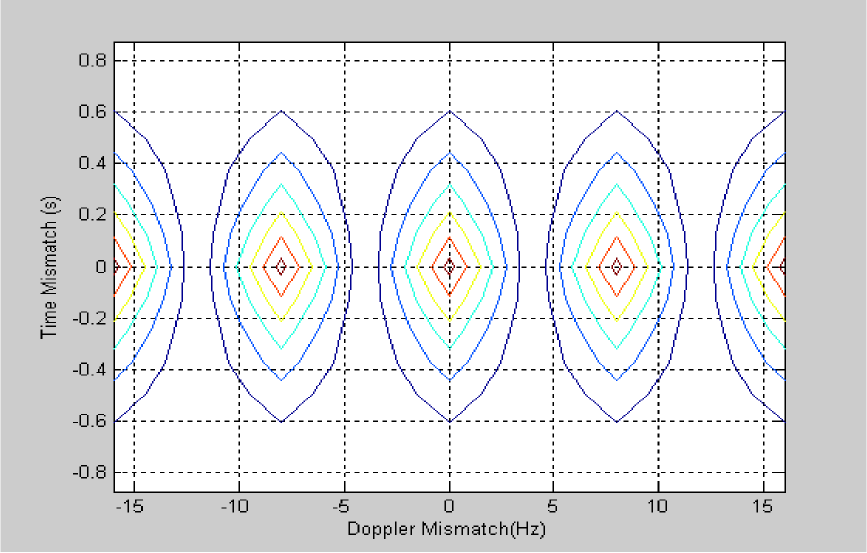

Figure 19: Aliasing artifacts due to sampling issue

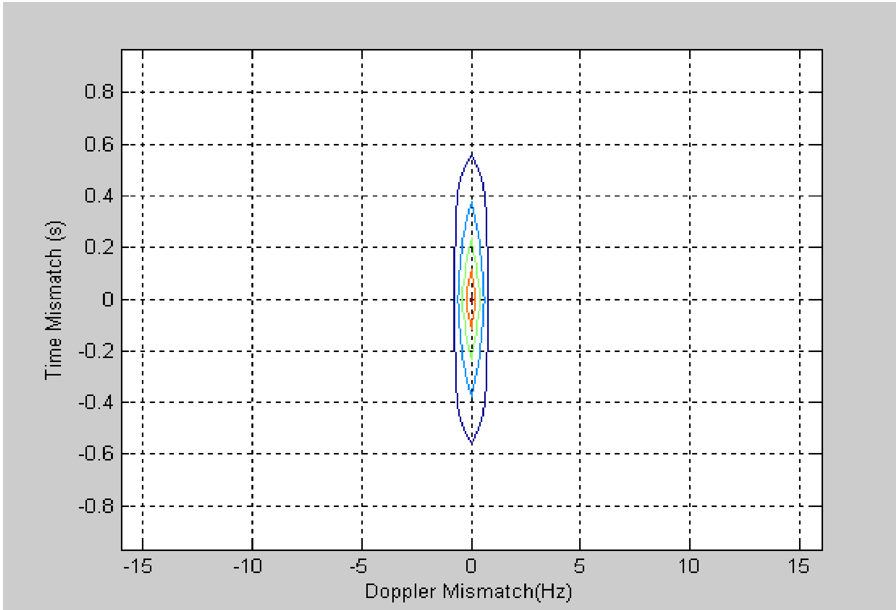

Figure 20: No aliasing after increasing chip points

10. CONCLUSION

In this paper, time frequency autocorrelation and ambiguity functions' role in waveform design has been discussed. TFACF and AF are very useful for radar designers for studying mathematical structure of different waveforms. They describe how different waveforms are useful for different applications. For example, to realize range Doppler resolution for a particular waveform, we need to know the ambiguity function. Also, to study the range-Doppler coupling of LFM signal, we study the ambiguity function. Finally, the LFM pulse compression technique, high range resolution from LFM pulse compression, range-Doppler coupling, and aliasing issue associated with sampling has been addressed.

REFERENCES

[1] Skolnik, M. I., *Introduction to Radar Systems.* McGraw-Hill, New York, 2001

[2] Hale, T.B., "EENG 668 Course Notes", Spring 2006

[3] Mehrdad, S., *Synthetic Aperture Radar Signal Processing*. John Wiley & Sons, Inc., New York, 1999

MATLAB CODE

```
clc; clear all
M= 1; % Number of Pulses: PRI/Waves
P =1; % Number of Chips within a Single Pulse
Tc = 1; % Chip Time: Chip/Second
NCps = 8; % Number of Chip Points: Samples/Chip

tau = P*Tc;
Tr =3; %Second/PRI
S=NCps*Tc*Tr*M; % Samples/Chip * Chip/Sec * Sec/PRI *
PRI/Wave = Samples/Wave

Psi_mp=zeros(M,P); % Phase weighting for each chip
Am=ones(1,M); %Continuous time weighting for each pulse
(inter-pulse weighting)
Wmp=ones(M,P); %Amplitude weight for each chip (intra- and
inter-pulse weighting)
Uc=ones(1,NCps);
Samp_sec=NCps/Tc; % Sample/second

for m=1:M
    pulse=[];
    for p=1:P
        Chip=Uc*Wmp(m,p)*exp(j*Psi_mp(m,p));
        pulse=[pulse Chip];
    end;
    pulse=pulse.*Am(m);
    indx=[(m-1)*Tr]*Samp_sec+1;
    S(indx:(indx+P*NCps-1))=pulse;
end

mf=xcorr(S,conj(S));
%plot(abs(mf));
T=[0:(length(S)-1)]/Samp_sec;
f=linspace(-Samp_sec/2, Samp_sec/2, length(S)+1);
% Following line Checks aliasing issues
%f=linspace(-2*Samp_sec, 2*Samp_sec, length(S)+1);
TFACF=[];
tic
for ff=f;
    XC=xcorr(S.*exp(j*2*pi*ff*T),conj(S));
    TFACF=[TFACF XC.'];
end
toc
```

```
AF = (abs(TFACF)).^2;
abs_AF=(abs(TFACF));

surf(f,[-T(end:-1:2) T],AF);
xlabel('Doppler Mismatch(Hz)');
ylabel('Time Delay (s)');
zlabel('AF response');

figure(2);
pcolor(f,[-T(end:-1:2) T],AF); shading interp
xlabel('Doppler Mismatch(Hz)');
ylabel('Time Mismatch (s)');
colorbar
figure(3)
contour(f,[-T(end:-1:2) T],AF);
xlabel('Doppler Mismatch(Hz)');
ylabel('Time Mismatch (s)');
grid on
% figure(4)
%  plot([-T(end:-1:2) T],abs_AF(:,33));
%  xlabel('Time Delay (s)');
%  ylabel('TFACF response');
%  title('abs(TFACF) when fd=0');
% grid
% figure(5)
%  plot(f,abs_AF(8,:));
%  xlabel('Doppler Mismatch(Hz)');
%  ylabel('TFACF response');
%  title('abs(TFACF) when TR=0');
%   grid
%   figure(6)
%  plot([-T(end:-1:2) T],AF(:,33));
%  xlabel('Time Delay (s)');
%  ylabel('AF response');
%  title('AF when fd=0');
% grid
% figure(7)
%  plot(f,AF(8,:));
%  xlabel('Doppler Mismatch(Hz)');
%  ylabel('AF response');
%  title('AF when TR=0');
%   grid
```