Radar Detection and Matched Filter

Submitter: Ron Bartov

Contents

Introduction

This document provides a simulation analysis for a radar detection assignment.

The primary objectives include analyzing received radar signals in both time and frequency domains, comparing different simulation schemes, such as rectangular and linear frequency modulation (LFM) pulses, improving signal-to-noise ratio (SNR) using matched filtering, and evaluating the detection performance under different noise conditions using various noise averaging methods for defining the threshold.

Preliminary Knowledge

This section includes key concepts and equations, serving as the basis for later calculations and analysis.

Peak Sidelobe Level

Compares the size of the highest sidelobe to the size of the main lobe

$$PSL = 20\log_{10}\left(\frac{Highest\ Side\ Lobe\ Amplitude}{Main\ Lobe\ Amplitude}\right)$$

This measure provides information about the spread of the energy. High sidelobe level (lower difference in peaks) means more unwanted energy spread, which can cause range ambiguity or false detections in radar.

Fourier Transform of Rectangular Pulse

$$\mathcal{F}\left\{rect\left(\frac{t}{T}\right)\right\} = T \cdot sinc\left(\frac{\Omega}{2\pi/T}\right)$$

Time Frequency and Time Shift

$$(1)\ \ x(t-t_0) \leftrightarrow e^{j\Omega t_0}X(\Omega)$$
$$(2)\ \ e^{j\Omega_0 t}x(t) \leftrightarrow X(\Omega-\Omega_0)$$

Spectrum and Time Convolution and Multiplication Relation

$$(1)\ \ x(t) * y(t) \leftrightarrow X(\Omega)\cdot Y(\Omega)$$
$$(2)\ \ x(t)\cdot y(t) \leftrightarrow \frac{1}{2\pi}X(\Omega)\cdot Y(\Omega)$$

Matched Filter

The goal in matched filtering is to correlate a known delayed signal (a template), with an unknown signal to detect the presence of the template in the unknown signal. This is equivalent to convolving the unknown signal with a conjugated time-reversed version of the template.

MATLAB Simulations Notes

Calculating the Delay for Discrete Signals

In MATLAB, we simulate each scenario in the discrete time domain, thus it is important for us to understand the relation between a continuous time interval and discrete time interval (i.e. number of samples). For a given sampling rate f_s, the sampling time interval is $t_s = \frac{1}{f_s}$, thus in the discrete time domain, the number of samples correspond to a continuous time interval T is $\#S = \frac{T}{t_s}$.

According to the above, calculating the discrete time delay of the pulse to get back from the target, and the discrete time pulse duration will be implemented by $\#S_{delay} = \frac{t_{round}}{t_s}$, and $\#S_{pulse} = \frac{\tau}{t_s}$ correspondingly.

Question 1: Rectangular Pulse Signal Analysis

Objective

Analyze a received radar signal transmitted using a rectangular pulse, examine its time and frequency domain characteristics and apply matched filtering to improve SNR.

Solution

Section a: Time Representation of the Received Signal

Figure [1] shows the time domain representation of the received signal due to a transmission of a rectangular pulse with pulse width of $\tau = 50\ usec$, from a target located $R = 50km$ from the radar.

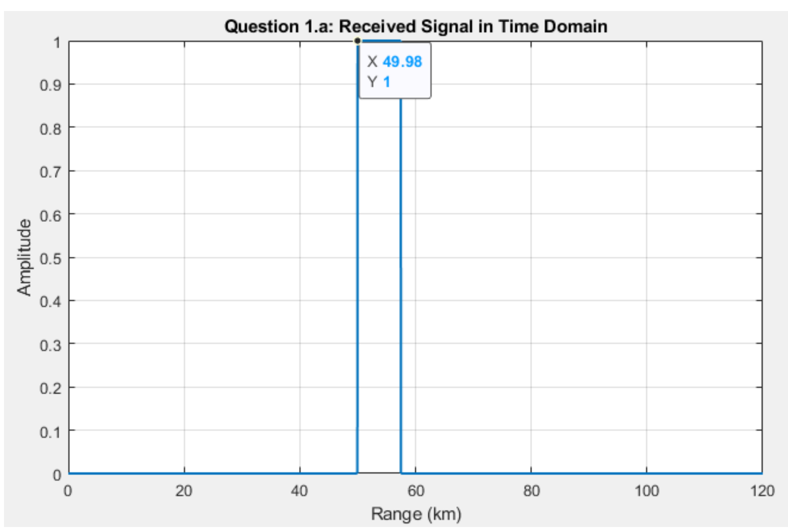

Figure 1: Question 1.a: Received Signal in Time Domain

After the transmission, the pulse wave will travel to the target and then back again to the receiver. The time representing this duration is called the **"round-trip time"**, and because the propagation speed in the radar case is the speed of light, we are getting the following relation:

$$2 \cdot R_{target} = t_{round} \cdot c \rightarrow t_{round} = \frac{2R_{target}}{c}$$

We can see from the plot that the front side of the pulse (the left side of the rectangular when the wave is returning to the receiver) is indeed mapped into the range of approximately 50km.

Additionally, in the range axis we will get a pulse width proportional to the pulse time duration τ,

$$BW_R = \frac{t_{pulse} \cdot c}{2} = \frac{\tau \cdot c}{2} = 7.5km$$

Section b: Frequency-Domain Representation

Figure [2] shows the received signal in the frequency domain.

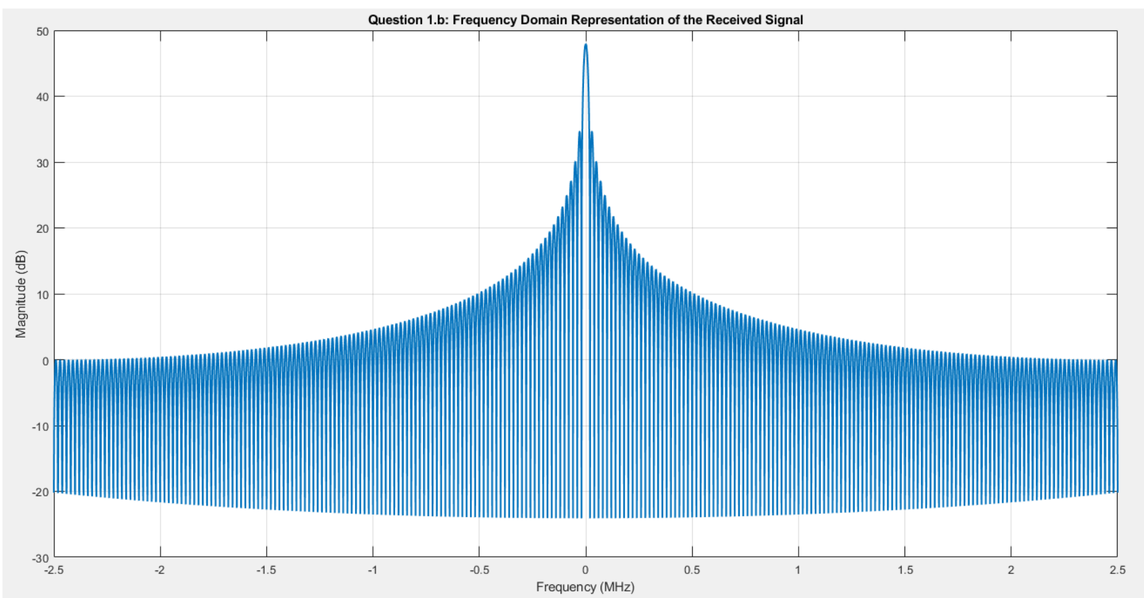

Figure 2: Question 1.b: Frequency Domain Representation of the Received Signal

The mathematical representation of the transmitted and the received pulse can be expressed as $rect\left(\frac{t}{\tau}\right)$, and according to the Fourier Transform of Rectangular Pulse we will expect to get a sinc function in the frequency domain, as we got in the plot.

From the simulation results, the difference absolute value between the two highest peaks (PSL) in the spectrum is $13.263\ dB$ and this is the expected value for the rectangular pulse case.

Section c: Matched Filtering Implementation

Figure [3] shows the received signal in the time domain, after filtering it with a Matched Filter.

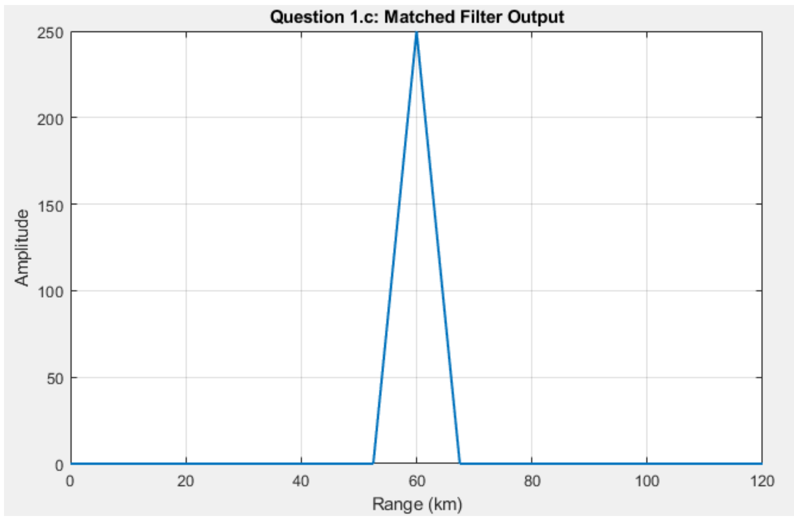

Figure 3: Question 1.c: Matched Filter Output

In our case, because the transmitted signal is a rectangular pulse, convolving it with its reflected version (also a rectangular pulse) will yield a triangular shape.

This phenomenon is sometimes called as **pulse compression**, since we are compressing the signal. In terms of power per range, this action increases the peak of the received signal in a region close to the target range.

Note that due to the convolution operation, there is some offset, i.e. the triangular peak is not aligned with the target range. Convolution is an integral operation between one static signal shape and one shifted signal shape. In our case these are two positive rectangular pulses, thus the maximum will be achieved when they overlap perfectly.

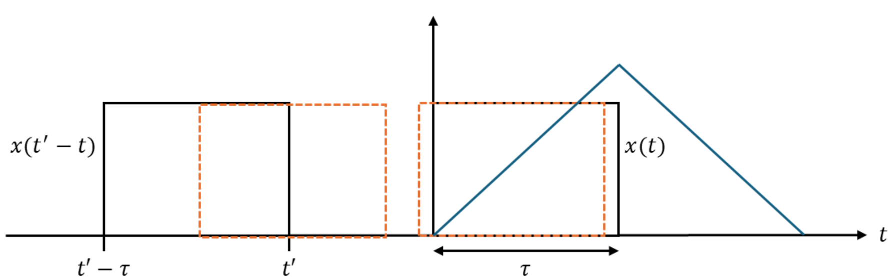

Figure 4: Convolution operation

Section d: Impact of Noise

Figure [5] shows the noisy received signal.

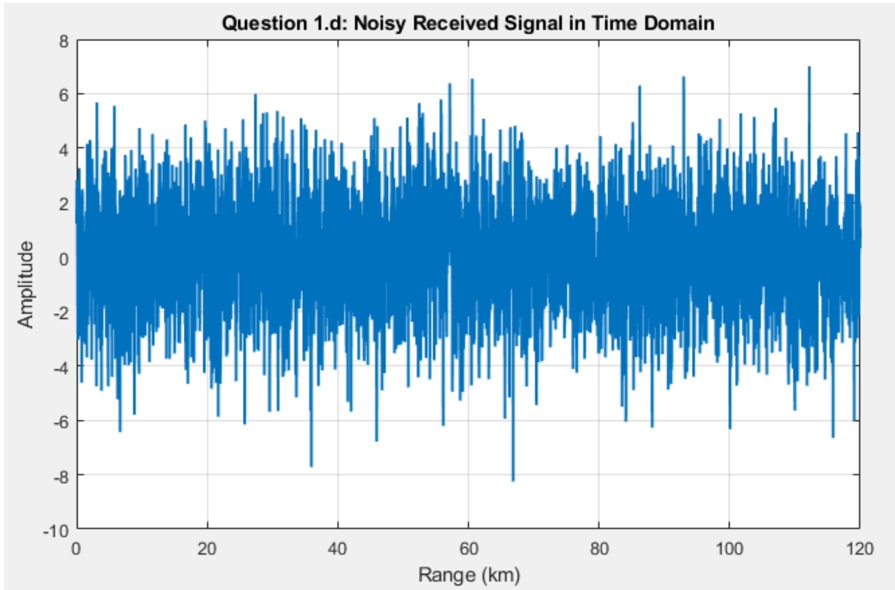

Figure 5: Question 1.d: Noisy Received Signal in Time Domain

In figure [6] we can see the result of the noisy received signal filtered by the matched filter.

Because the matched filter is highly correlated with the shape of the received pulse, but not with the additive noise, then the pulse compression operation helps to increase the peak in the target region while less increasing the noise.

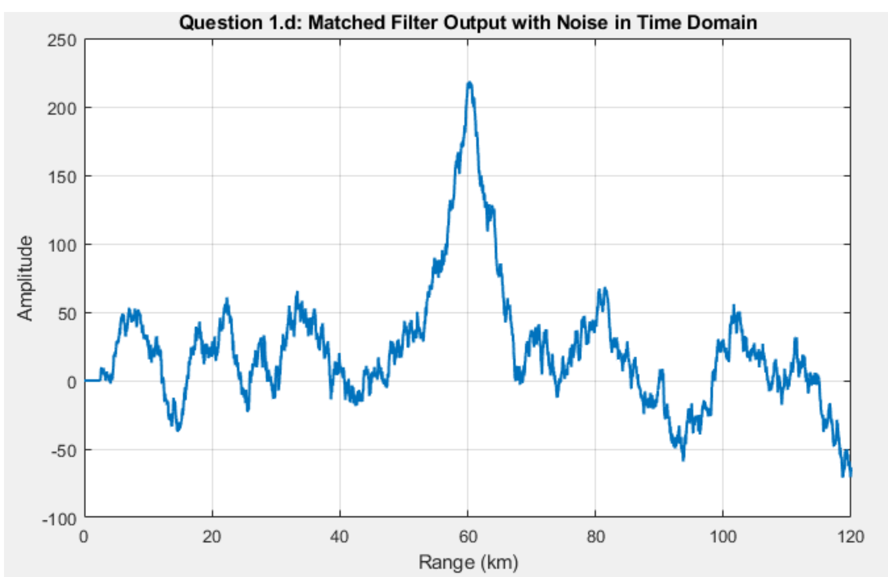

Figure 6: Question 1.d: Matched Filter Output with Noise in Time Domain

Question 2: LFM Pulse Signal Analysis

Objective

Analyze a received radar signal transmitted using a line frequency modulation (LFM) pulse. Examine its time and frequency domain characteristics and apply matched filtering to improve SNR.

Solution

Section a: Time Representation of the Received Signal

In the case of LFM transmission, the received signal will have the form of a complex exponential with a unit gain and some phase shift.

Figure [7] shows the absolute value of the received signal, thus we see constant amplitude and pulse width proportional to the pulse time duration ($7.5km$ as calculated in Question 1 section a)

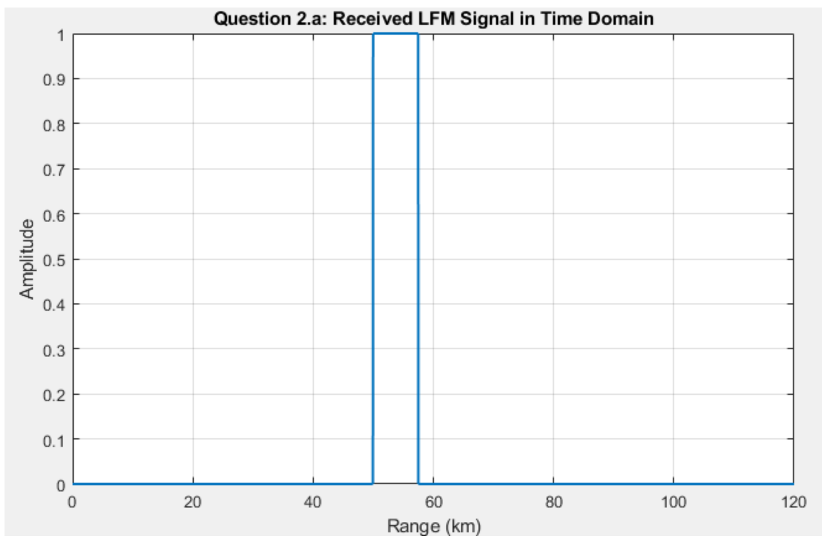

Figure 7: Question 2.a: Received LFM Signal in Time Domain

Section c: Matched Filtering Implementation

Figure [8] shows the result of the noisy received LFM signal filtered by the matched filter.

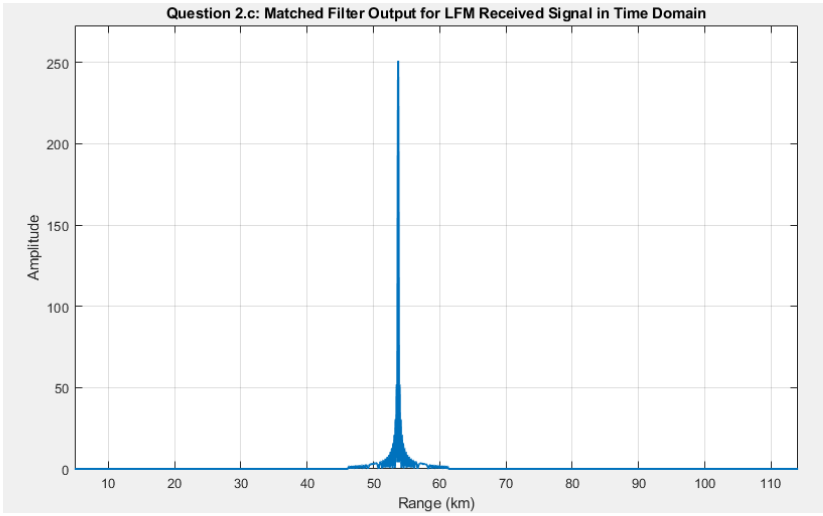

Figure 8: Question 2.c: Matched Filter Output for LFM Received Signal in Time Domain

In the LFM case, we still convolving with a reflected version of the pulse, but with different signals that have wider bandwidth (linear frequency modulation).

According to the uncertainty principle we know that a wider bandwidth in frequency domain will yield a narrower bandwidth in the time domain, thus comparing to question 1, the convolution operation between the two LFM signals will yield a narrower shape. Additionally, from the plot we can see that also the peak of the LFM matched filter output is higher, which is better for a target detection.

Section d: Impact of Noise

Figures [9] and [10] shows the noisy LFM received signal before and after the matched filtering.

Comparing the results to question 1, we can see that in the LFM case, there is a higher signal peak at the target location and a more compressed noise.

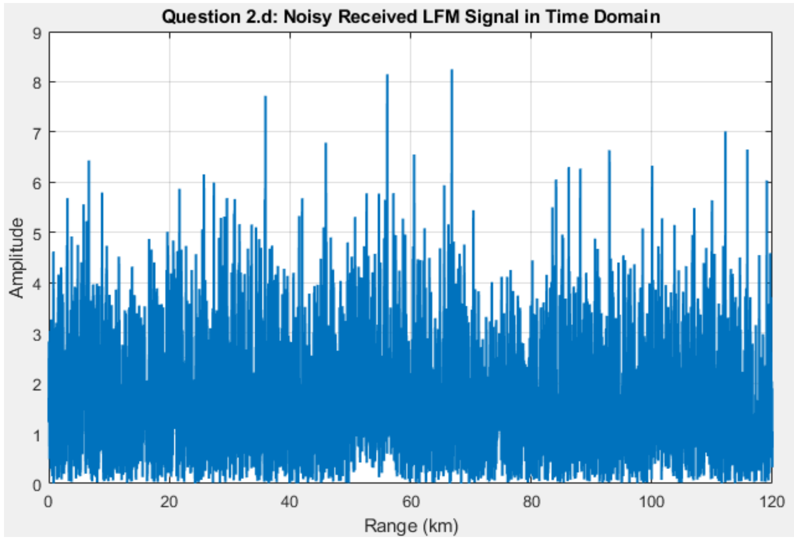

Figure 9: Question 2.d: Noisy Received LFM Signal in Time Domain

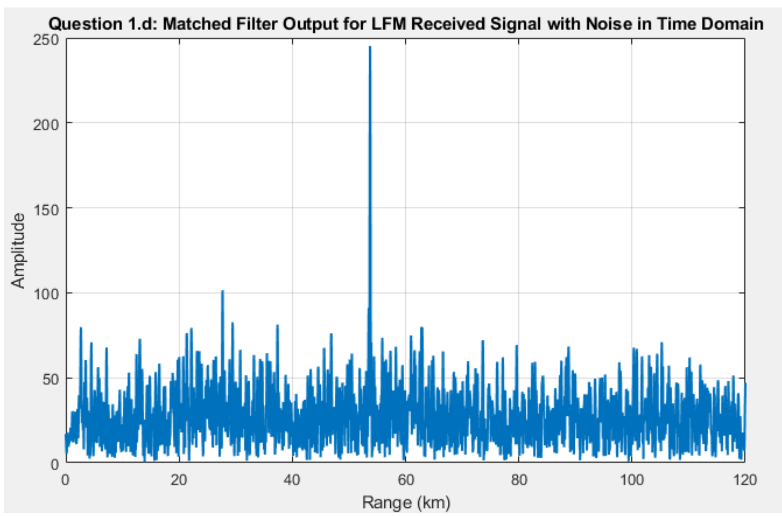

Figure 10: Question 1.d: Matched Filter Output for LFM Received Signal with Noise in Time Domain

To conclude, the main advantage in using LFM is to increase the signal gain with respect to the noise (provide better resolution), due to its wider bandwidth.

Question 3: Target Detection Using LFM Radar

Objective

Analyze a real-world scenario by adding a significant noise for specific ranges and compare different target detection methods using different threshold definition algorithms.

Figure [11] shows the received noisy LFM signal, before matched filtering and additional significant noise.

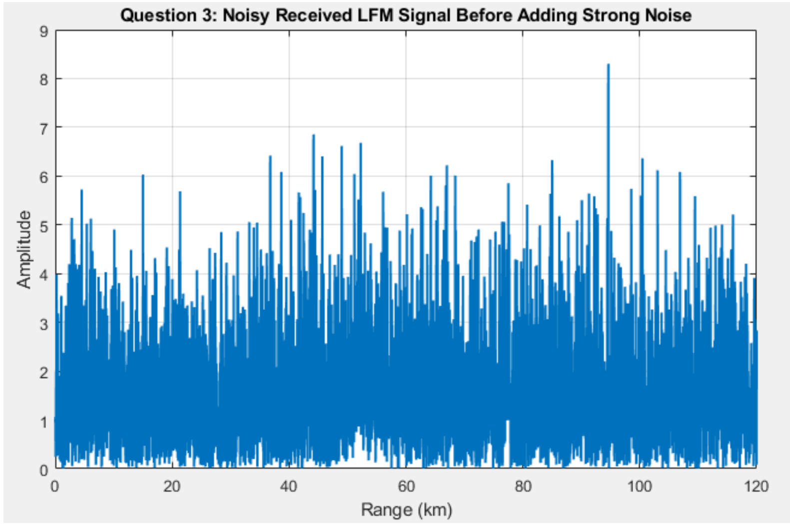

Figure 11: Question 3: Noisy Received LFM Signal Before Adding Strong Noise

Figure [12] shows the received noisy LFM signal, before matched filtering, but with an additional significant noise added in the ranges that are smaller than 30km.

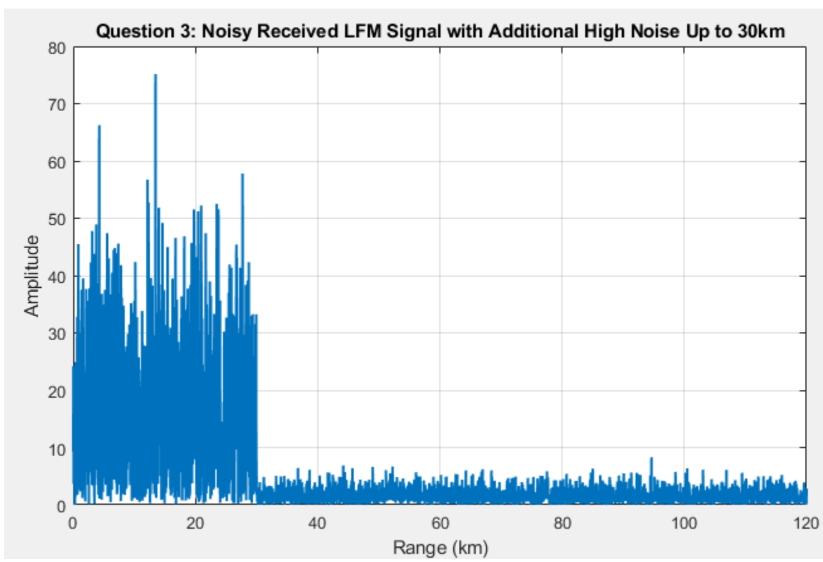

Figure 12: Question 3: Noisy Received LFM Signal with Additional High Noise Up to 30km

Figure [13] shows the previous mentioned signal, after matched filtering. This will be the signal which will be used for calculating the threshold for the target detection task.

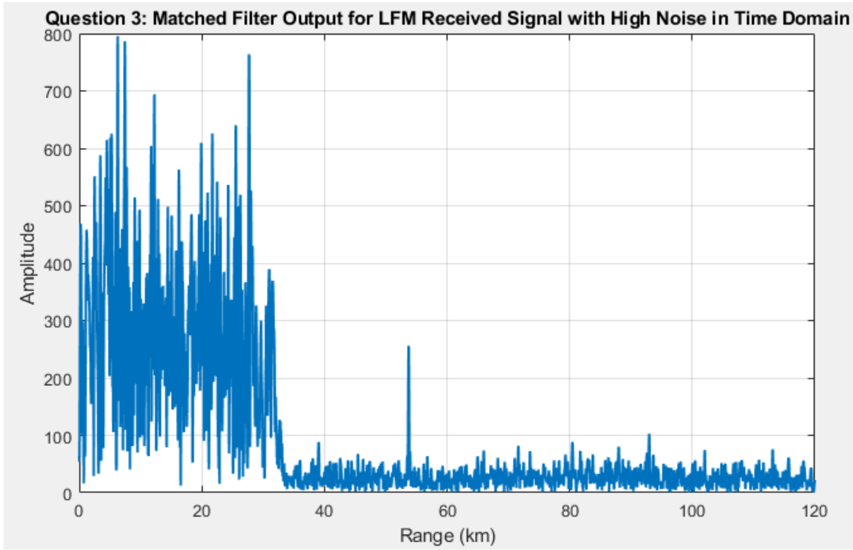

Figure 13: Question 3: Matched Filter Output for LFM Received Signal with High Noise in Time Domain

Solution

Section a: Average Global Threshold

For this section and those that follow, we will mark as $x(n)$ the significant noisy, matched filtered LFM received signal absolute value.

Additionally in all sections, the threshold factor $TH_{fac} = 5.6 \; will \; be \; used.$

In the classic average method, the threshold is calculated by the following formula:

$$TH_{avg} = TH_{fac} \cdot \sum_{n=1}^{N} x(n)$$

Thus, the threshold in this case is a scalar, means a constant threshold line.

Figure [14] shows the results after classifying $x(n)$ samples based on the TH_{avg}.

As can be seen from the plot, there is a false alarm in this case.

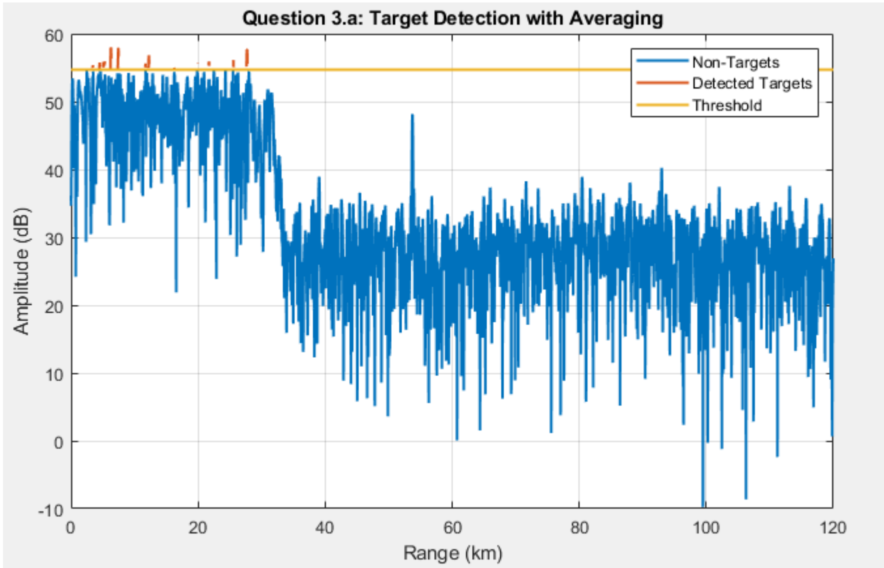

Figure 14: Question 3.a: Target Detection with Averaging

Section b: Moving Average Threshold

In the moving average method, a uniform filter is first defined:

$$h(n) = \frac{1}{32}; n = 1, \ldots, 32$$

The moving average threshold in this case is a vector and it is calculated by:

$$TH_{ma}(n) = TH_{fac} \cdot \{x * h\}(n); n = 1,2,3, \ldots, N$$

It means that each received signal sample will have its own threshold value based on a 32 samples neighbors (**including the sample itself**).

Figure [15] shows the result after classifying $x(n)$ samples based on the TH_{ma}.

In this case, no target is detected at all. It may make sense since for each sample, the filter h is averaging subset of the signal including the sample which the threshold should be calculated for, thus after additional gain by using TH_{fac}, the threshold will be far from the sample value.

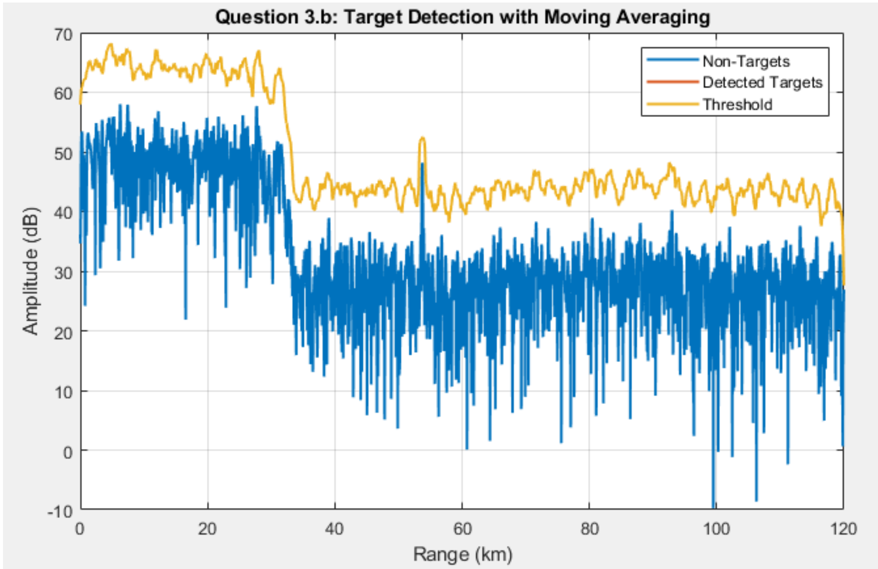

Figure 15: Question 3.b: Target Detection with Moving Averaging

Section c: Custom Filter-Based Threshold

In the custom filtering average method, the following filter is first defined:

$$h_{tmp} = \frac{1}{32} \cdot ones(1,16) \rightarrow h = [h_{tmp}, zeros(1,7), h_{temp}]$$

The threshold in this case is also a vector and it is calculated by:

$$TH_{custom}(n) = TH_{fac} \cdot \{x * h\}(n); n = 1,2,3, \dots, N$$

We can think on this filter as a notch filter, where we give more weight to the neighbors around each sample but not to the sample itself.

Figure [16] and [17] shows the result after classifying $x(n)$ samples based on the TH_{custom}.

In this case, the actual target is detected, means that this will be the optimal solution compared to the other.

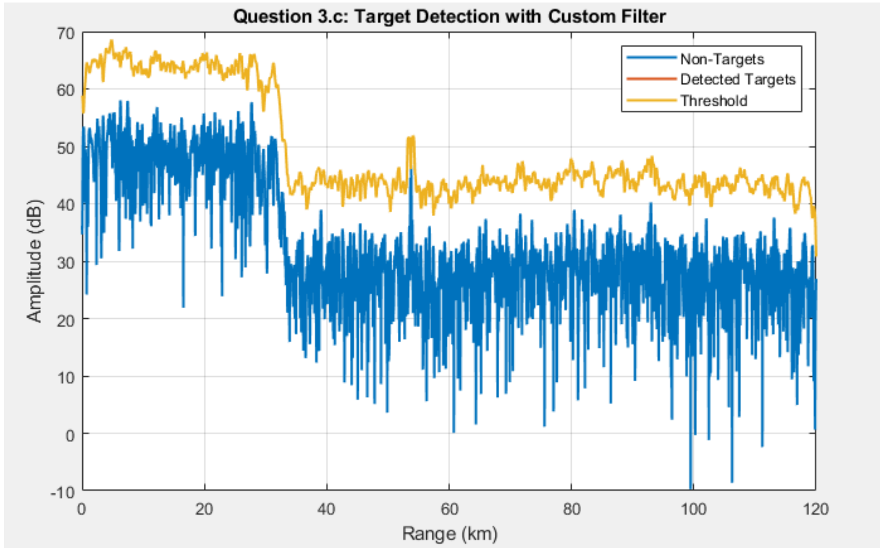

Figure 16: Question 3.c: Target Detection with Custom Filter

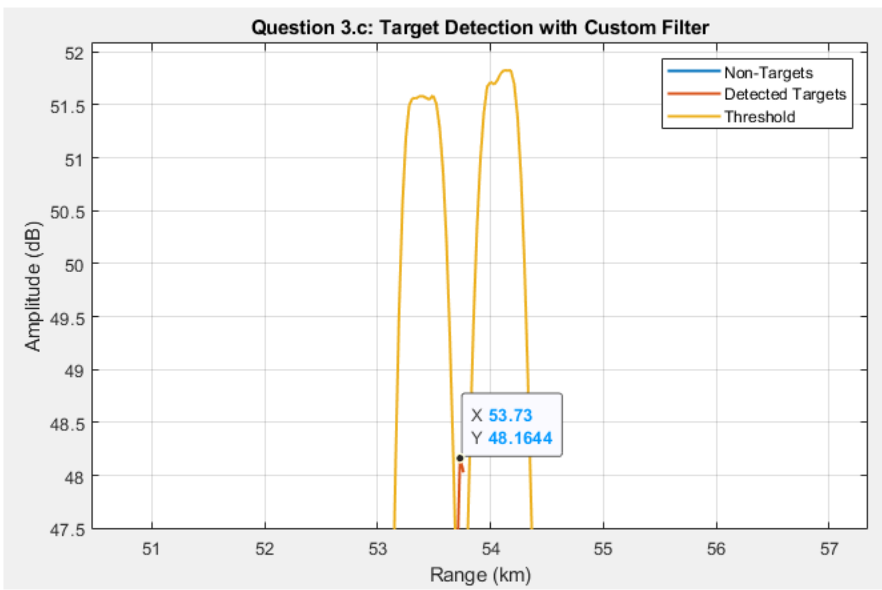

Figure 17: Question 3.c: Target Detection with Custom Filter (zoom in)

Section d: Comparison Between different Methods

In previous sections, we examined different methods to extract the threshold values for being able to classify and detect a target from a received noisy signal.

Based on the results from the graphs, we can conclude that the optimal solution is the third method.

Based on the full process that have been done to the received noisy signal (as described in this assignment), it makes sense that the third methods will be the optimal due to the fact that it calculate the threshold value by calculating the statistics around each sample and then try to check if the sample itself is an exceptional one in the series.