CLUTTER DETECTION IN PULSE-DOPPLER RADAR SYSTEMS

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND

ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Ahmet Güngör

August 2010

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Sinan Gezici (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Orhan Arıkan

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. İbrahim Körpeoğlu

Approved for the Institute of Engineering and Sciences:

Prof. Dr. Levent Onural
Director of Institute of Engineering and Sciences

ABSTRACT

CLUTTER DETECTION IN PULSE-DOPPLER RADAR SYSTEMS

Ahmet Güngör
M.S. in Electrical and Electronics Engineering
Supervisor: Assist. Prof. Dr. Sinan Gezici
August 2010

Among various types of radar systems, the pulse-Doppler radar is the most widely used one, especially in military applications. Pulse Doppler radars have a primary objective to detect and estimate the range and the radial velocity of the targets. In order to have a basis for the detection, first reflected echo signals are matched filtered and then the time-alligned pulse returns are transformed to the Fourier domain to obtain the range-Doppler matrix. The resulting range-Doppler matrix is input to target detection algorithms. For this purpose, constant false alarm rate (CFAR) algorithms are run on the range-Doppler matrix. It is useful to run different CFAR algorithms inside the clutter region and outside the clutter region because the statistics are different inside and outside of the clutter. In order to achieve this discrimination, the position of the clutter has to be detected in the range-Doppler matrix. Moreover, the clutter may not always appear around zero Doppler frequency when realistic terrain models and moving platforms are considered. Two algorithms for clutter detection using range-Doppler matrix elements are investigated and their performance analysis is presented in this thesis. The first algorithm has higher error rates but lower computational complexity,

whereas, the second one has lower error rates but higher computational complexity. Both algorithms detect clutter position by filtering the range-Doppler matrix elements via non-linear filters. In addition to the probabilistic error rate analysis, simulation results on some realistic cases are presented. It is concluded that the first algorithm is a good choice for low clutter-to-noise ratio values when a low-complexity algorithm is required. On the other hand, the second algorithm has better performance in all clutter-to-noise ratio values but it requires more computational power.

Keywords: Pulse-Doppler Radar, Realistic Terrain Generation, Matched Filtering, Pulse-Doppler Processing, Clutter Detection, Kernel Density Estimation, Kullback-Leibler Divergence.

ÖZET

DARBE-DOPPLER RADAR SİSTEMLERİNDE PARAZİT YANKI TESPİTİ

Ahmet Güngör
Elektrik ve Elektronik Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Yrd. Doç. Dr. Sinan Gezici
Ağustos 2010

Çeşitli radar sistemleri arasında darbe Doppler radar sistemleri, özellikle askeri uygulamalarda en çok kullanılanıdır. Darbe Doppler radar sistemlerinin birincil görevi hedefleri tespit edip, hedeflerin menzil ve radyal hızlarını kestirmektir. Tespit yapabilmek için, yansıyan sinyaller uyumlu süzgeçten geçirilip, darbe Doppler işlemine tabi tutulur. Sonuçta elde edilen mesafe-Doppler matrisi bazı tespit algoritmalarına girdi olarak kullanılır. Hedefleri tespit edebilmek için, Sabit Yanlış Alarm Oranlı (SYAO) algoritmalar mesafe-Doppler matrisi üzerinde koşturulur. Parazit yankı bölgesinde ve bu bölgenin dışında farklı SYAO algoritmaları koşturmak yararlıdır. Çünkü hücre istatistiği parazit yankının içinde ve dışında farklıdır. Bu ayrımı elde edebilmek için öncelikle mesafe-Doppler matrisinde parazit yankının yeri tespit edilmelidir. Gerçekçi arazi modelleri ve hareketli platformlar göz önüne alındığında, parazit yankı sıfır Doppler frekansı etrafında bulunmayabilir. Bu tez çalışmasında, mesafe-Doppler matrisi elemanlarını kullanarak, parazit yankı tespit edilmesine yönelik iki algoritma ve performans analizleri sunulmaktadır. Algoritmalardan birincisi daha yüksek hata oranına ancak daha düşük işlem karmaşıklığına, ikincisi ise daha düşük hata

oranına ancak daha yüksek işlem karmaşıklığına sahiptir. Algoritmalar mesafe-Doppler matrisi elemanlarını doğrusal olmayan süzgeçlerden geçirerek parazit yankının konumunu tespit etmektedir. Olasılıksal hata oranı analizlerine ek olarak, bazı gerçekçi durumların benzetim sonuçları da sunulmaktadır. Birinci algoritmanın, düşük işlem karmaşıklığı gerektiren durumlarda, düşük parazit-yankı-gürültü oranı değerleri için kullanılmasının iyi bir seçim olduğu; öte yandan, daha yüksek işlem karmaşıklığına sahip ikinci algoritmanın bütün parazit-yankı-gürültü oranları için daha iyi performansa sahip olduğu gözlenmektedir.

Anahtar Kelimeler: Darbe Doppler Radarı, Gerçekçi Arazi Modellemesi, Uyumlu Filtreleme, Darbe Doppler İşleme, Parazit Yankı Tespiti, Çekirdek Yoğunluk Kestirimi, Kullback-Leibler Iraksaması.

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisor Assist. Prof. Dr. Sinan Gezici for his invaluable supervision, suggestions and encouragement throughout the development of this thesis.

I am also indebted to Prof. Dr. Orhan Arıkan for being very helpful to me with his experience and suggestions on my thesis topic. In addition, I would like to extend my special thanks to Assist. Prof. Dr. İbrahim Körpeoğlu for his valuable comments and suggestions on the thesis.

I wish to thank to all my friends and colleagues, the staff and the professors in our department for their collaboration and support.

Finally, my deepest gratitude goes to my family. Not just during my graduate studies but during all my life, they believed in me and helped me achieve my goals. Their support has always been invaluable.

Contents

List of Figures

List of Tables

To my family...

Chapter 1

INTRODUCTION

A wide spectrum of components ranging from defensive or offensive devices to the strategic integration of more complex technologies designed to defeat an enemy constitute weapons systems [4]. Weapon systems must be capable of accomplishing some functions like target detection, classification, localization and tracking in order to destruct or neutralize the target. Countless target and environmental characteristics including clutter location, target location, target speed and target direction emerges a need for complex detection systems. Noise generated by the detection system itself complicates the target detection. Therefore, appropriate design is needed to distinguish target signals from noise [4].

Radar is an acronym for Radio Detection and Ranging [5]-[8]. Some objects can be detected and located at far greater distances than a naked eye can see, using the applications of electromagnetic waves. This sensing is not affected by most obstacles to ordinary vision like night, cloud, fog and smoke. In addition, radar permits the accurate measurement of the range and velocity of what it senses with a precision cannot be obtainable by a human operator. Some other aspects of radar performance are poorer than that of the eye [4].

Pulse-Doppler radar systems are used in order to detect both locations and radial velocities of the targets. Airborne pulse-Doppler radars are operating in an environment being exposed to strong echoes reflected from the ground [9]. These reflections constitute clutter signals. Target detection in the presence of clutter becomes difficult. An approach to detect targets includes finding the position of the clutter and then running different constant false alarm rate (CFAR) algorithms in the clutter zone, in the vicinity of the clutter zone and in the far field of the clutter zone.

In this thesis, we study the detection of clutter in the range-Doppler plane. This plane is observed after the pulse-Doppler processing of the signals reflected from the environment. In airborne pulse-Doppler radars, the clutter is always present and some targets may also be present in the range-Doppler plane.

In most radar applications, clutter is eliminated instead of locating its position in order to detect targets. However, this approach allows to detect moving targets and it is called as moving target indication (MTI). Stationary targets are also cancelled out with the clutter because they have a spectrum falling into the clutter spectrum. Ground looking stationary radars are exposed to fixed ground clutter which is around zero Doppler. In this way they distinguish the moving targets. An airborne ground looking moving radar is also exposed to ground clutter but in this case it is not fixed along the motion of the radar and it is not always around the zero Doppler. Moving targets can also be detected in this case because they have different spectra than the clutter [10]-[14]. In order to detect the stationary targets, first the clutter should be detected because the desired targets lie inside the clutter. When the clutter is detected, it is easy to find the moving targets in the rest of the process. There are some approaches to find the clutter position using the range-Doppler plane. Because this is a military

application, it is hard to find this type of techniques that are uncovered. Our algorithms are useful to solve this problem.

In [15], an improved signal processing technique is presented. It is suitable for airborne Doppler radars. The primary objective is to distinguish targets from the mainbeam clutter. The technique does not require accurate measurements of the velocity or the altitude of the moving radar platform [15]. After obtaining the range-Doppler matrix, each cell of the matrix is compared to a scaled mean of the background noise. If the value in the cell exceeds the threshold, a logic one is set to that cell. If the value in the cell is lower than the threshold, a logic zero is set to that cell. In the logic matrix, the clutter signals occur in many contiguous cells and target signals are present in only a few isolated cells or in a single cell [15]. The problem with this technique is the calculation of the threshold value. It is the scaled mean of the background noise but the noise level is not accurately known. For the case in which the noise is known accurately, it is again not defined how the mean of the background noise is scaled.

In [16], a method for clutter detection is presented. In this technique, if a large number of cells that have large amplitudes are present in a range, those cells are identified as clutter. This procedure is repeated on each range cell. The problem with the discussed method is that the word "large" is not defined accurately.

In [17], a new model to detect objects in a given image is proposed. The model can detect objects, the boundaries of which are not necessarily defined by gradient. It is used in radar imaging to reveal targets. In addition, this method can be applied to the clutter detection problem. The range-Doppler matrix is also an image. The edges of the clutter and targets can be detected. If there is only clutter in the matrix, the result of this method gives the clutter edges. However, if there are targets, in addition to the clutter, in the matrix, the result

of this method gives both the clutter and target edges and a further processing is needed to distinguish the clutter from targets. Moreover, this technique is a slow one because it is an iterative algorithm.

In our work, we propose two algorithms for clutter detection. The first one is a simple and fast algorithm. The second one is a more complex and slower algorithm. Simplicity of the first algorithm would cause a drop in the performance for the sake of higher speed. The second one would have better performance in a longer period of running time. Both algorithms are based on simple nonlinear filters. They can be used to detect clutter regardless of its Doppler value. Namely, clutter can be around zero Doppler or it can be around any other value of Doppler frequency.

The first algorithm includes a single nonlinear filter. This filter selects the maximum valued cell as the clutter cell in a range bin. It would have good performance when the range-Doppler plane is exposed to low noise and no target is present in the plane. When high noise or a target is present in the plane, the algorithm would begin to give erronous results.

The second algorithm includes one more nonlinear filter. In addition to the maximum selecting filter, a median filter is present. It filters the bin numbers of the cells that are chosen as clutter by the first algorithm. The median filter used here would apply a correction on the erronous results of the first algorithm. The run time increases with the length of the median filter. For the longer lengths of the median filter, the algorithm would give improved results.

In Chapter 2, the system model is introduced. The simulation environment consists of a terrain model, a target model and a radar model. Each model is discussed in detail in that chapter. The steps of the pulse-Doppler processing is

also mentioned. In Chapter 3, the clutter detection algorithms are explained in detail. The probability of error formulas of the algorithms are obtained. Moreover, a series of performance analysis is conducted for some different cases. The results of different cases are also compared in that chapter. In Chapter 4, simulation results on realistic clutter and target data are discussed. The results of the algorithms on realistic scenarios are presented. The difference in the performance of the algorithms can also be observed in that chapter. Finally, Chapter 5 concludes the thesis.

Chapter 2

SYSTEM MODEL

In this chapter, the radar system model is introduced in order to explain the environment in which the clutter detection algorithms are run. First of all, a terrain must be generated in order to simulate an air platform flying on the terrain. The air platform carries a pulse-Doppler radar system on it. It transmits the radar signal from a specified height and elevation for a 360-degree coverage of azimuth and listens for the reflected signals from the terrain. The reflections from the terrain constitute the raw clutter signal. After pulse-Doppler processing of the received signals, the range-Doppler matrix is formed, which indicates the Doppler and range of the clutter. In addition, realistic target models are used in some simulations. The target is processed individually and then added to the clutter signal.

2.1 Terrain Generation

In [1], some fractal terrain generating algorithms are discussed. The algorithms consist of midpoint displacement in one dimension, height maps, triangular-edge

subdivision, diamond-square subdivision and square-square subdivision. A realistic terrain should not include creases and its roughness should be in accordance with the real geography. Among the algorithms, the diamond-square subdivision technique is used to generate the realistic terrain. It is an algorithm that produces realistic terrains and can easily be implemented [1].

The diamond-square subdivision algorithm is divided into two stages as the name implies: square stage and diamond stage. The process begins with a square matrix. Some values should be assigned to the corners of the matrix for the initiation of the process and then the subdivision begins [18], [19]. In Figure 2.1(a), the blue point in the center is generated from the average of the values initially generated,the remaining blue points, plus a random value. Now, half diamonds are created. The red points in the same figure represents the recently created points. They stay in the center of a full diamond. Their values are the average of the points forming the diamond plus a random value. In Figure 2.1(b), the blue points are coming from the previous stage and they form squares. The recently created points are the red ones. They are in the center of each square and their values are the average of the four values forming each square plus a random value. In Figure 2.1(c), diamonds are formed again. The iterations continue following this pattern until all the points in the matrix have a value. The random numbers added should be selected from a reduced range from iteration to iteration in order to get a smooth terrain. The other terrain generating algorithms are explained in detail in [1]. A sample terrain generated using the diamond-square subdivision method is illustrated in Figure 2.2 and Figure 2.3. The terrain consists of small patches and their sizes are determined by the resolution. Each echo of the transmitted radar signal is reflected from a patch.

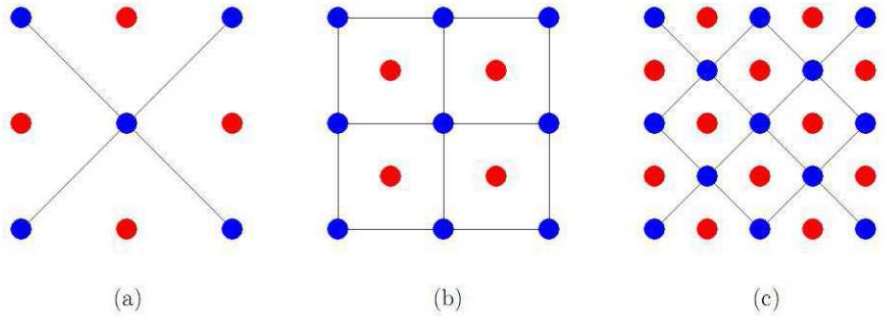

Figure 2.1: (a) Diamond stage. (b) Square stage. (c) Diamond stage again [1].

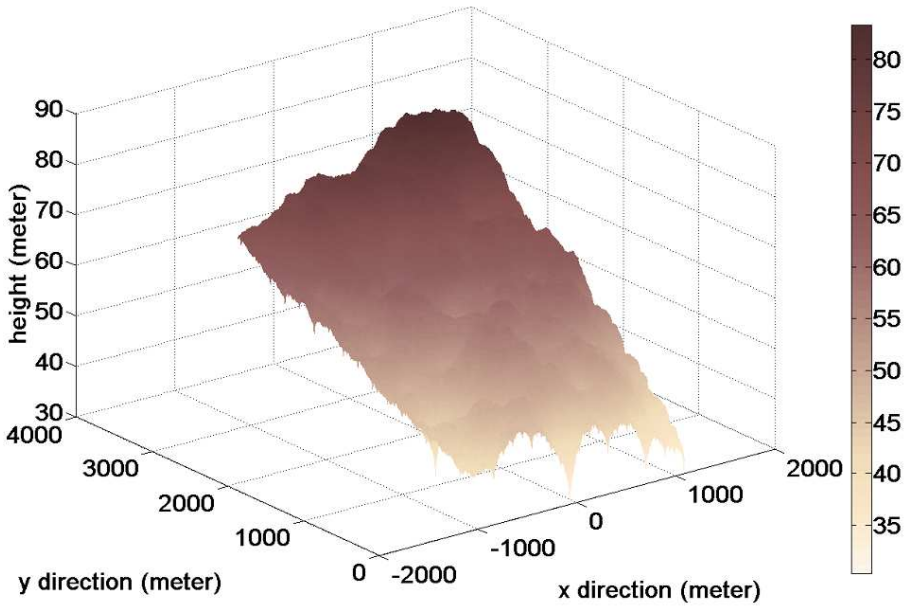

Figure 2.2: Side view of the terrain [1].

2.2 Target Model

In some simulations, different from the point targets, a realistic target model is used with a realistic terrain model. In [20], some realistic target models are discussed. The models include a Russian T-62 tank, a French VAB (Vehicule de l'Avant Blinde) wheeled armoured vehicle and another Russian tank T-72. Among the realistic target models, the model for T-62 tank is used.

The T-62 tank has a 6.63 m length, a 3.52 m width and a 2.4 m height. An illustration of the tank can be seen in Figure 2.4. In [3], a monopulse radar

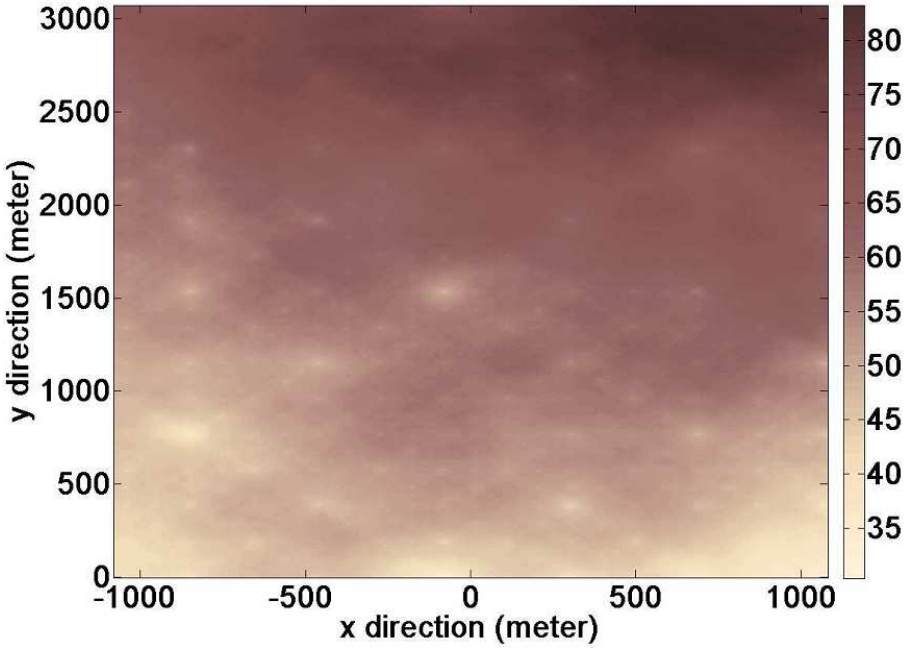

Figure 2.3: Top view of the terrain [1].

at 95 GHz is used to measure the radar cross section (RCS) values of the T-62 tank, which is placed on a turntable. In order to model the tank, it is first assumed that the target is rectangular in shape. The greatest RCS values are formed on the edges and corners. For this reason, the main scatterers are located on the corners, the middle point of the edges and the center of the rectangular target. These scatterers are responsible for the average value of the RCS. Then, about 60 minor scatterers are located on the target in a random manner. These points have small RCS values. In order to add the effect of the view angle, the RCS values of some angles are used from the measurements to fit a polynomial function. As a result, a function is generated including all the angles. Finally, all the RCS values of the scatterers are summed and the result is multiplied with the function value of the view angle. The realistic result can be seen in Figure 2.5 [20].

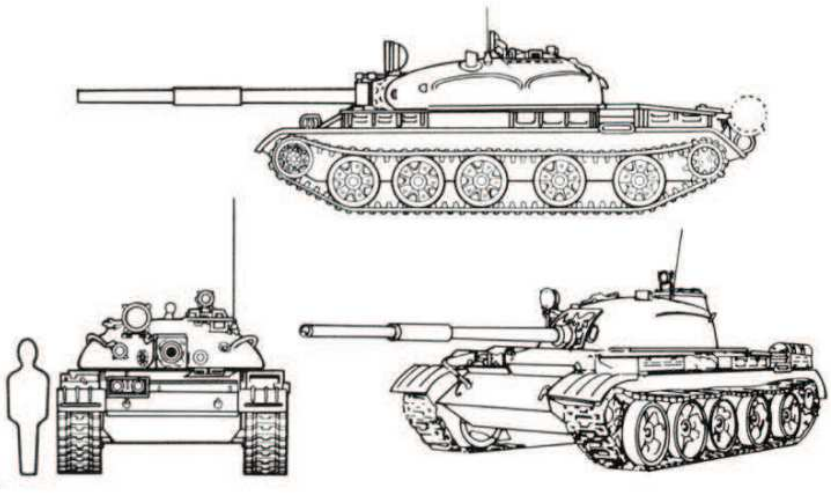

Figure 2.4: T-62 tank [2].

2.3 Radar Model and Pulse-Doppler Processing

After having a realistic terrain, a radar model which is mounted on an air platform flying over the terrain is needed. The radar is modeled as a pulse-Doppler radar because of the widespread use of it [9]. The generic radar waveform and its complex envelope can be expressed in (2.1) and (2.2) respectively.

$$\tilde{x}(t) = a(t)\, e^{j[\Omega t + \theta(t)]} \tag{2.1}$$

$$x(t) = a(t)\, e^{j\theta(t)} \tag{2.2}$$

where $a(t)$ is the amplitude modulation of the radio frequency (RF) carrier, Ω is the RF carrier frequency and $\theta(t)$ is the phase or frequency modulation of the carrier [21]. Here, $a(t)$ and $\theta(t)$ determines the type of the radar signal. In a pulsed radar, $a(t)$ has the form of a single pulse. However, for a better Doppler resolution, pulse burst waveforms must be used [22]-[25]. Accordingly, $a(t)$ has the form of a sum of shifted pulses. In a pulse burst radar system using a single antenna for transmission and reception, listening for the echoes continues when $a(t)$ is zero. Otherwise, transmission is performed. The phase or frequency modulation, $\theta(t)$, allows a better range resolution than an unmodulated signal

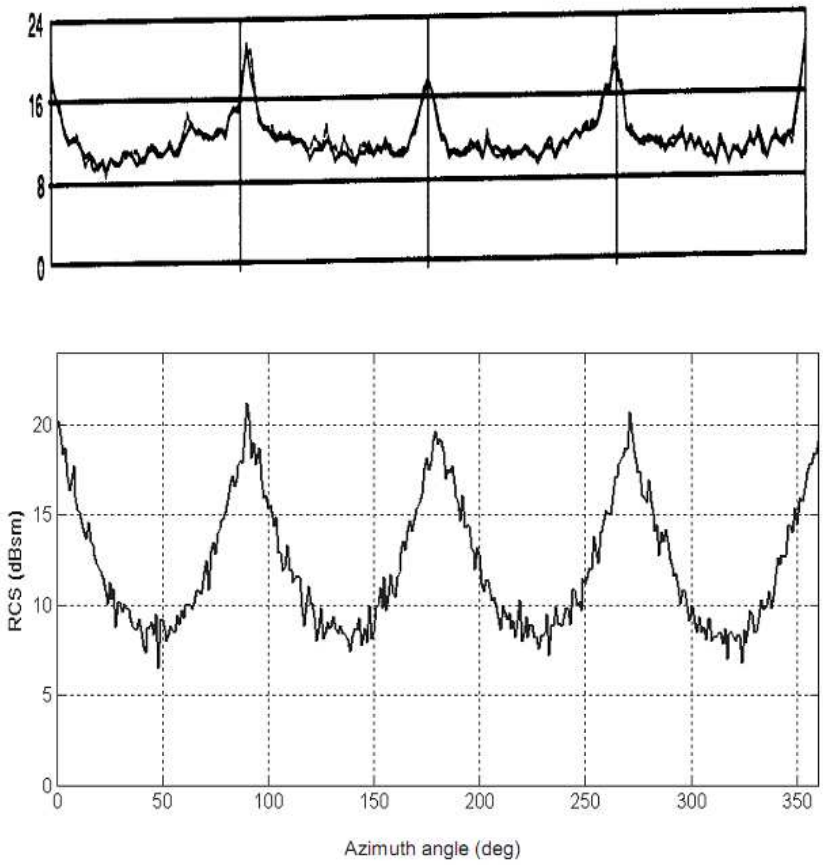

Figure 2.5: (a) Measurement of RCS values for the T-62 tank. (b) RCS values of the model for the T-62 tank [3].

[22]-[25]. A chirp modulated pulse burst waveform is used as the radar signal. A sample waveform can be seen in Figure 2.6.

After the echoes arrive at the antenna, a receiver that maximizes the signal-to-noise ratio (SNR) is needed in order to have improved detection performance. This can be achieved by a matched filter. A matched filter has an impulse response obtained by time-reversing and conjugating the reference waveform. Convolving the received signal with this impulse response maximizes the SNR in the presence of additive noise [26]-[28]. Actually, it is a correlator function using the transmitted waveform as the reference signal. Phase or frequency modulated signals have narrower autocorrelation functions [26]-[28]. In this way, they have better range resolution.

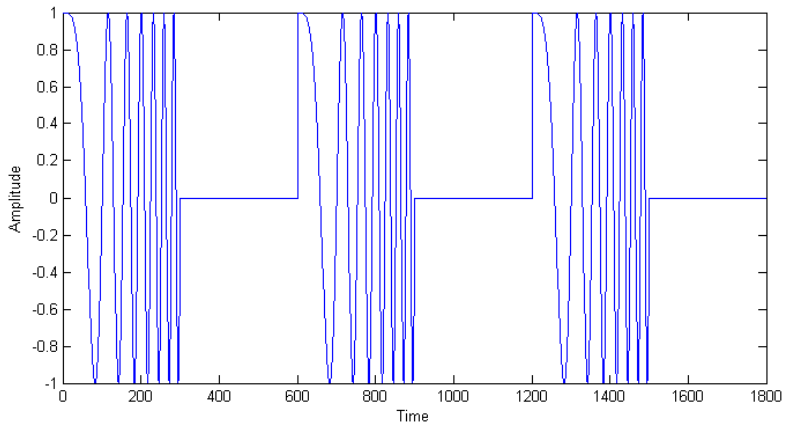

Figure 2.6: A sample radar signal waveform.

After filtering each pulse in the received pulse burst waveform using a single pulse in the transmitted burst as the reference signal, the individual filtered pulses are aligned one under the other in a matrix form. The matrix is an M x N one where M is the number of pulses and N is the number of samples collected in the receiving period. It is followed by the pulse-Doppler processing.

The pulse-Doppler processing is a technique for the spectral analysis for each column of the matrix obtained after matched filtering [26]-[28]. The columns in the matrix represent the delays. Here, delay corresponds to range. In each range bin, a spectral analysis using the discrete Fourier transform (DFT) is performed. Here, the DFT of each column is periodic with the pulse repetition interval. The result of the DFT is periodic with the same period independent of the number of pulses. So, if the number of pulses is higher, the period of the discrete Fourier transformed sequence is sampled more intensively. As a result, the Doppler resolution improves [20].

The matrix observed after the spectral analysis becomes the input for the clutter detection algorithms presented in the next chapter.

Chapter 3

CLUTTER DETECTION ALGORTIHMS AND PERFORMANCE ANALYSIS

In this chapter, two clutter detection algorithms are presented. Notation and assumptions used in performance analysis are explained. After getting the theoretical probability of error formulas for each case, they are compared with algorithms' performances on synthetically generated data using point targets in some cases. Performances of the algorithms are also compared with each other.

3.1 Notation

Here, the notation used in the analysis will be described. $A(r, \nu)$ is the range-Doppler matrix where r is the range index and ν is the Doppler frequency, $P_C(\cdot)$ is the probability density function (p.d.f.) of the clutter, $P_T(\cdot)$ is the p.d.f. of the target, $P_N(\cdot)$ is the p.d.f. of the noise, $F_C(\cdot)$ is the cumulative distribution function (c.d.f.) of the clutter, $F_T(\cdot)$ is the c.d.f. of the target, $F_N(\cdot)$ is the c.d.f.

of the noise, n is the number of noise bins in each range cell, and N_W is the median filter length which is used in Algorithm 2 in Section 3.3.

3.2 Assumptions

Here, the assumptions made in the analysis will be described. Noise cells are assumed to be independent and identically distributed (i.i.d.) which is consistent with the realistic case. Clutter cells are assumed to be i.i.d. This assumption may not be true in the realistic case. It has no effect on Algorithm 1 because each range cell is treated separately. However in Algorithm 2, this assumption leads to approximate results in the analysis. When more than one clutter cells are present in range cells, this assumption results in an approximation for Algorithm 1 as well.

3.3 Clutter Detection Algorithms

The algorithms will be described here are mainly developed for radars having low Doppler resolution or clutter having low Doppler spread. Namely, the algorithms are designed for detecting clutter which is present only in one Doppler bin in each range cell in the range-Doppler matrix. The algorithms are described as follows:

Algorithm 1:

$$C_r = \arg\max_{\nu} |A(r,\nu)|, \;\; r = 1, 2, ... \tag{3.1}$$

where C_r denotes the index of the clutter in the r^{th} range cell.

In this algorithm, along each range cell the maximum valued bin is selected as the clutter bin. Noise and targets may cause errors in the detection of the clutter. This algorithm runs a single nonlinear filter. So, it is expected to run faster than Algorithm 2 because it runs two nonlinear filters as follows:

Algorithm 2. Find $f(r,i)$'s and apply a nonlinear ordered statistics filter (median filter with filter length N_W). Namely,

$$f(r,i) = arg\max_{\nu} |A(r+i,\nu)| \tag{3.2}$$

$$C_r = median\{f(r,-j),...,f(r,0),...,f(r,j)\},\ r=1,2,... \tag{3.3}$$

where $j = \frac{N_W-1}{2}$, $f(r,i)$ denotes the index of the maximum Doppler frequency bin in $(r+i)^{th}$ range cell, and C_r denotes the index of the clutter in the r^{th} range cell.

In this algorithm, a further step is included. After deciding the maximum valued bins in each range cell, a median filter of length N_W is applied to correct any errors that are caused because of noise and targets. The level of the correction depends on the median filter length N_W. Running two nonlinear filters, this algorithm is expected to run slower than Algorithm 1. However, it is expected that there may also be performance difference in the algorithms. This will be investigated in the following sections.

As mentioned before, the algorithms are designed for detecting the clutter, which is present only in one Doppler bin in each range cell. The algorithms can be further modified to detect clutters which can be present in more than one Doppler bin in each range cell. For a clutter present in two Doppler cells, Algorithm 1 can again run a nonlinear filter but this time it decides the maximum and the second biggest valued bins as clutters. Similarly, Algorithm 2 can use the results obtained from Algorithm 1 and can again run a median filter. In

Algorithm 2, there may be two approaches for this case, which will be reviewed in a non-theoretical manner.

3.4 Performance

In the performance analysis, the clutter and the target cells are modeled to have the Rice distribution and the noise has the Rayleigh distribution. The error probabilities of the clutter detection for both algorithms will be calculated. Then the theoretical and simulation results will be compared to investigate whether they are consistent. P.d.f.s and c.d.f.s for clutter, target and noise are as follows: The p.d.f. for the clutter is given by

$$P_C(z) = \frac{z}{\sigma_C^2} \exp\left(\frac{-(z^2+\mu_C^2)}{2\sigma_C^2}\right) I_0\left(\frac{z\mu}{\sigma_C^2}\right) \tag{3.4}$$

where I_0 is the modified Bessel funciton of the first kind with order zero, and the corresponding c.d.f. is

$$F_C(z) = 1 - Q_1\left(\frac{\mu_C}{\sigma_C}, \frac{z}{\sigma_C}\right) \tag{3.5}$$

where Q_1 is the Marcum Q-function.

The p.d.f. for the target is

$$P_T(z) = \frac{z}{\sigma_T^2} \exp\left(\frac{-(z^2+\mu_T^2)}{2\sigma_T^2}\right) I_0\left(\frac{z\mu}{\sigma_T^2}\right) \tag{3.6}$$

where I_0 is the modified Bessel funciton of the first kind with order zero, and the corresponding c.d.f. is given by

$$F_T(z) = 1 - Q_1\left(\frac{\mu_T}{\sigma_T}, \frac{z}{\sigma_T}\right) \tag{3.7}$$

where Q_1 is the Marcum Q-function.

The p.d.f. for the noise is expressed as

$$P_N(z) = \frac{z}{\sigma_N^2} e^{-z^2/2\sigma_N^2} \tag{3.8}$$

and the corresponding c.d.f. is

$$F_N(z) = 1 - e^{-z^2/2\sigma_N^2}. \tag{3.9}$$

In the plots, the probability of error is changing with respect to the clutter power $E[|C|^2]$, where the noise power $E[|N|^2]$ is constant. $E[|C|^2] = 2\sigma_C^2 + \mu_C^2$ for the Rice distribution and $E[|N|^2] = 2\sigma_N^2$ for the Rayleigh distribution.

In the plots, $n = 10$, $\sigma_N = 2$ (so $E[|N|^2] = 8$) and $\sigma_C = \sqrt{2}$ (so $E[|C|^2] = 4 + \mu_C^2$). The values for μ_C and $E[|C|^2]$ are shown in Table 3.1.

μ_C	$E[\|C\|^2]$
0.5	4.25
1	5
1.5	6.25
2	8
2.5	10.25
3	13
3.5	16.25
4	20
4.5	24.25
5	29
5.5	34.25
6	40

Table 3.1: Values for μ_C and $E[|C|^2]$.

3.5 Performance of Algorithm 1 for No Target Case with One Clutter Cell in Each Range

A description of this case is provided in Figure 3.1. The probability of error formula for this case is shown in (3.10) and the proof of the formula can be seen through (3.11) to (3.31).

Probability of Error:

$$P_e = 1 - \int (1 - F_C(z))\, n\, [F_N(z)]^{n-1} P_N(z) dz \tag{3.10}$$

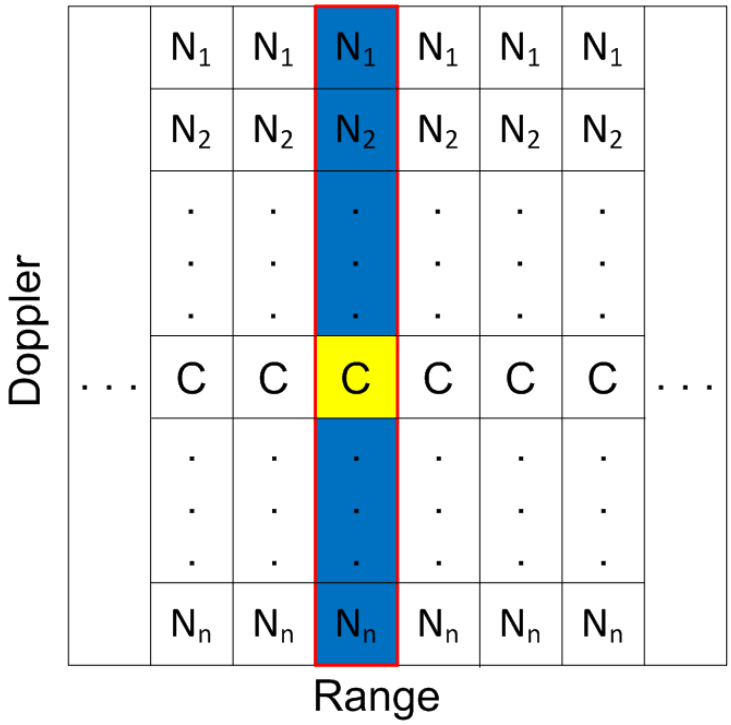

Figure 3.1: Description of Algorithm 1 for no target case with one clutter cell in each range.

Proof: The probability of error can be expressed as one minus the probability of correct clutter cell selection:

$$P_{err} = 1 - P_{cor} \tag{3.11}$$

The algorithm will make a correct decision if the value of the clutter C is greater than the maximum of noise values.

$$\begin{aligned} P_{cor} &= P(C > N_i, i = 1, ..., n) && (3.12)\\ &= P(C > \max(N_1, ..., N_n)) && (3.13)\\ &= P(C > \tilde{N}) && (3.14)\\ &= \int P(C > z) P_{\tilde{N}}(z) dz && (3.15) \end{aligned}$$

where $\tilde{N} \triangleq max(N_1, ..., N_n)$ and $N_i's$ are i.i.d. Then, the cdf of $\tilde{N}$ is obtained as follows:

$$\begin{aligned} F_{\tilde{N}}(z) &= P(\tilde{N} < z) && (3.16)\\ &= P(max(N_1, ..., N_n) < z) && (3.17)\\ &= P(N_1 < z, ..., N_n < z) && (3.18) \end{aligned}$$

$N_i's$ being independent,

$$= P(N_1 < z)...P(N_n < z) \quad (3.19)$$

$N_i's$ being identical,

$$\begin{aligned} &= [P(N_1 < z)]^n && (3.20)\\ &= [F_{N_1}(z)]^n && (3.21) \end{aligned}$$

The p.d.f. of $\tilde{N}$ is given by

$$\begin{aligned} P_{\tilde{N}}(z) &= \frac{d}{dz} F_{\tilde{N}}(z) && (3.22)\\ &= \frac{d}{dz} [F_{N_1}(z)]^n && (3.23)\\ &= n[F_{N_1}(z)]^{n-1} \frac{d}{dz} F_{N_1}(z) && (3.24)\\ &= n[F_{N_1}(z)]^{n-1} P_{N_1}(z) && (3.25) \end{aligned}$$

The last unknown in the probability of correct clutter cell selection equation is expressed as

$$\begin{aligned} P(C > z) &= 1 - P(C \leq z) && (3.26)\\ &= 1 - F_C(z) && (3.27) \end{aligned}$$

The probability of correct clutter cell selection is

$$\begin{aligned} P_{cor} &= P(C > \tilde{N}) && (3.28)\\ &= \int (1 - F_C(z))\, n\, [F_{N_1}(z)]^{n-1} P_{N_1}(z) dz && (3.29) \end{aligned}$$

Finally, the probability of error is given by

$$\begin{aligned} P_{err} &= 1 - P_{cor} && (3.30)\\ &= 1 - \int (1 - F_C(z))\, n\, [F_{N_1}(z)]^{n-1} P_{N_1}(z) dz && (3.31) \end{aligned}$$

In Figure 3.2, it can be observed that the theoretical and the simulation results are consistent.

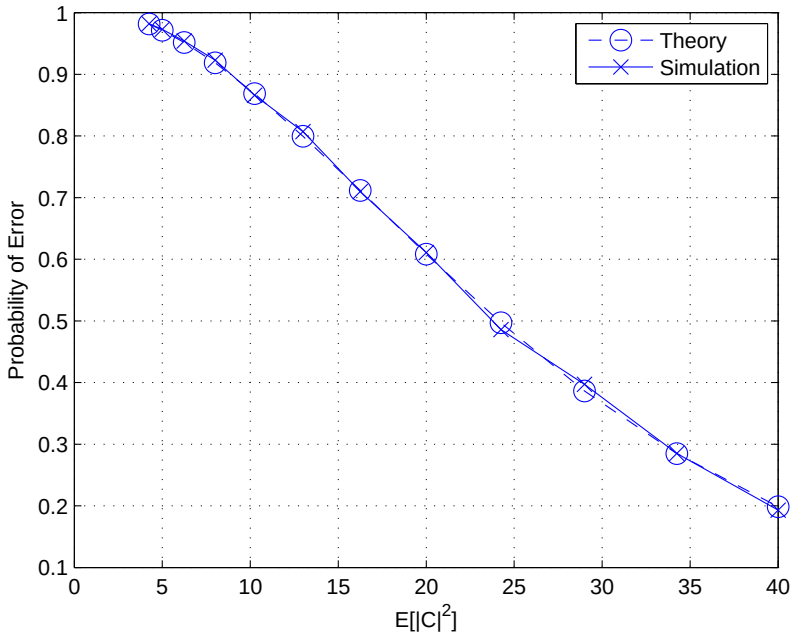

Figure 3.2: Probability of error for Algorithm 1 for no target case with one clutter cell in each range.

3.6 Performance of Algorithm 2 for No Target Case with One Clutter Cell in Each Range

A description of this case is provided in Figure 3.3. The probability of error formula for this case is shown through (3.32) to (3.35) and the proof of the formula can be seen through (3.36) to (3.41).

Probability of Error: For $N_W = 2K+1$:

$$\begin{aligned} P_e &= (2K+1)!\left[\sum_{a=K+1}^{2K+1}\frac{x^a}{a!}\left(\sum_{b=0}^{2K+1-a}\frac{c^b y^{(2K+1-a-b)}}{b!(2K+1-a-b)!}\right)\right] \\ &+ (2K+1)!\left[\sum_{a=K+1}^{2K+1}\frac{y^a}{a!}\left(\sum_{b=0}^{2K+1-a}\frac{c^b x^{(2K+1-a-b)}}{b!(2K+1-a-b)!}\right)\right] \end{aligned} \tag{3.32}$$

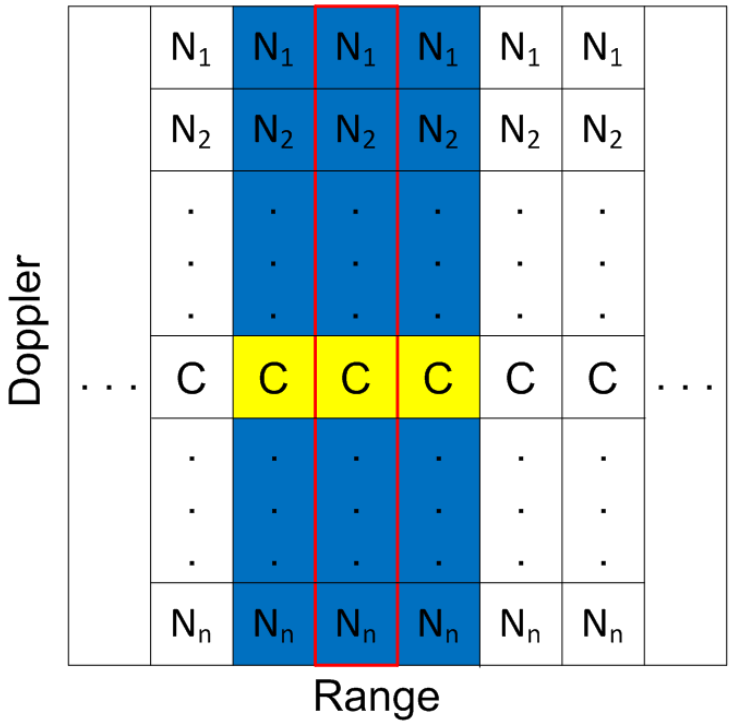

Figure 3.3: Description of Algorithm 2 for no target case with one clutter cell in each range.

where c is the correct decision probability for clutter position, x is the probability of deciding a bin that is above the clutter bin and y is the probability of deciding a bin that is below the clutter bin in each range. Namely,

$$c = \int (1 - F_C(z))\, n\, [F_N(z)]^{n-1} P_N(z) dz \tag{3.33}$$

$$x = (1-c)\epsilon \tag{3.34}$$

$$y = (1-c)(1-\epsilon) \tag{3.35}$$

where $\epsilon = \frac{1}{2}$ in the case of the clutter that is in the middle of each range cell. Actually, it is the ratio of the number of noise bins lying above the clutter bin to the total number of noise bins.

Proof: The median filter length is $N_W = 2K + 1$. If the number of bins lying above the clutter bin, namely x's, in the median window are more than K, the median filter results in an error. Similarly, if the number of bins lying below the clutter bin, namely y's, in the median window are more than K, the median filter

result is again an error. To illustrate, consider a median window with length 3. There are x, c and y values in this window. To have the result to be an error, the following ordered combinations should occur in the window: xxc, xxx, cyy, yyy. Note that the given combinations are ordered sets. The median filter orders its inputs and gives the one in the middle as the output. Therefore, the sets do not have to be in an ordered form. Their unordered combinations have to be taken into account. All of these requirements lead to the following proof for the generic case:

$$P_e = P_{ex} + P_{ey} \tag{3.36}$$

where

$$\begin{aligned} P_{ex} &= \frac{(2K+1)!}{(K+1)!}x^{K+1}\sum_{a=0}^{K}\frac{c^a y^{(K-a)}}{a!(K-a)!} \\ &+ \frac{(2K+1)!}{(K+2)!}x^{K+2}\sum_{a=0}^{K-1}\frac{c^a y^{(K-1-a)}}{a!(K-1-a)!} \\ &+ ... \\ &+ \frac{(2K+1)!}{(2K+1)!}x^{2K+1}\sum_{a=0}^{0}\frac{c^a y^{(0-a)}}{a!(0-a)!}. \end{aligned} \tag{3.37}$$

After arranging the above equation, it reduces to the following equation:

$$P_{ex} = (2K+1)!\left[\sum_{a=K+1}^{2K+1}\frac{x^a}{a!}\left(\sum_{b=0}^{2K+1-a}\frac{c^b y^{(2K+1-a-b)}}{b!(2K+1-a-b)!}\right)\right]. \tag{3.38}$$

Similarly,

$$\begin{aligned} P_{ey} &= \frac{(2K+1)!}{(K+1)!}y^{K+1}\sum_{a=0}^{K}\frac{c^a x^{(K-a)}}{a!(K-a)!} \\ &+ \frac{(2K+1)!}{(K+2)!}y^{K+2}\sum_{a=0}^{K-1}\frac{c^a x^{(K-1-a)}}{a!(K-1-a)!} \\ &+ ... \\ &+ \frac{(2K+1)!}{(2K+1)!}y^{2K+1}\sum_{a=0}^{0}\frac{c^a x^{(0-a)}}{a!(0-a)!}. \end{aligned} \tag{3.39}$$

After arranging the above equation, it reduces to the following equation:

$$P_{ey} = (2K+1)!\left[\sum_{a=K+1}^{2K+1}\frac{y^a}{a!}\left(\sum_{b=0}^{2K+1-a}\frac{c^b x^{(2K+1-a-b)}}{b!(2K+1-a-b)!}\right)\right]. \tag{3.40}$$

Combining the above equations, the resulting equation is as follows:

$$\begin{aligned} P_e &= (2K+1)! \left[\sum_{a=K+1}^{2K+1} \frac{x^a}{a!} \left(\sum_{b=0}^{2K+1-a} \frac{c^b y^{(2K+1-a-b)}}{b!(2K+1-a-b)!} \right) \right] \\ &+ (2K+1)! \left[\sum_{a=K+1}^{2K+1} \frac{y^a}{a!} \left(\sum_{b=0}^{2K+1-a} \frac{c^b x^{(2K+1-a-b)}}{b!(2K+1-a-b)!} \right) \right]. \end{aligned} \tag{3.41}$$

In Figure 3.4 it can be observed that the theoretical and the simulation results are consistent. In Figure 3.5 the effect of the median filter length can be seen. When the length of the filter is increased, the performance of the algorithm is also increased. In other words, the probability of error decreases with the increase in the median filter length. In Figure 3.6 the performance difference between the two algorithms can be observed. Expectedly, the performance of Algorithm 2 is better than that of Algorithm 1. Even with a small window length, the probability of error of Algorithm 2 is less than the probability of error of Algorithm 1 because Algorithm 2 uses more data to decide the position of the clutter. In other words, Algorithm 2 achieves a lower probability of error for a given clutter power, or equivalently requires a lower power of clutter for a specific probability of error compared to Algorithm 1.

3.7 Performance of Algorithm 1 for No Target Case with Two Clutter Cells in Each Range

A description of this case is provided in Figure 3.7. The probability of error formula for this case is shown in (3.42) and the proof of the formula can be seen through (3.43) to (3.51).

Probability of Error:

$$P_e = 1 - \int \left(1 - F_C(z)\right)^2 n \left[F_N(z)\right]^{n-1} P_N(z) dz \tag{3.42}$$

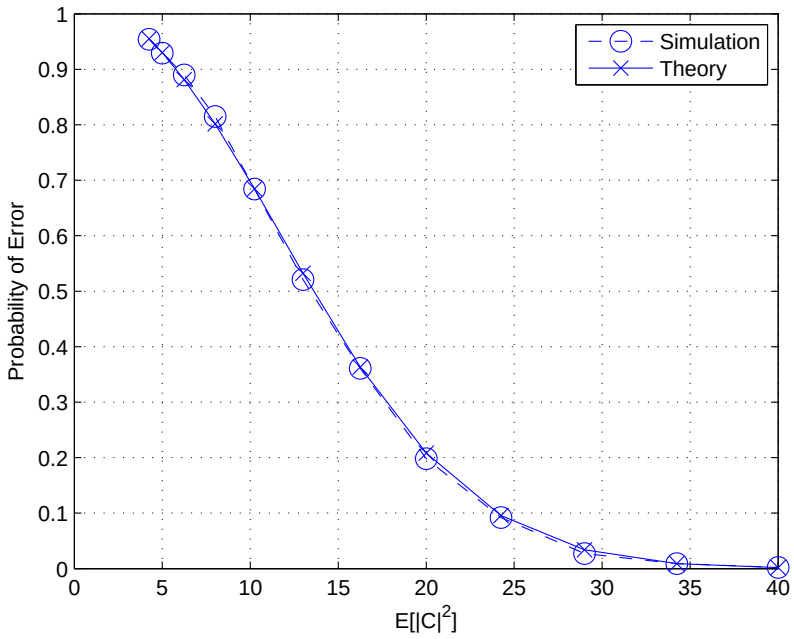

Figure 3.4: Probability of error for Algorithm 2 for no target case with one clutter cell in each range.

Proof:

$$P_{err} = 1 - P_{cor} \tag{3.43}$$

The algorithm will make a correct decision if the value of the minimum of C_1 and C_2 is greater than the maximum of the noise values.

$$\begin{aligned}
P_{cor} &= P(\min(C_1, C_2) > N_i, i = 1, ..., n)) && (3.44)\\
&= P(\min(C_1, C_2) > \max(N_1, ..., N_n)) && (3.45)\\
&= P(\min(C_1, C_2) > \tilde{N}) && (3.46)\\
&= P((C_1 > \tilde{N})\&(C_2 > \tilde{N})) && (3.47)\\
&= \int P(C_1 > z)P(C_2 > z)P_{\tilde{N}}(z)dz && (3.48)\\
&= \int P(C > z)^2 P_{\tilde{N}}(z)dz && (3.49)\\
&= \int (1 - F_C(z))^2 \, n \, [F_N(z)]^{n-1} P_N(z)dz && (3.50)
\end{aligned}$$

Then, from (3.43) and (3.50), P_{err} is obtained as

$$P_{err} = 1 - \int (1 - F_C(z))^2 \, n \, [F_N(z)]^{n-1} P_N(z)dz. \tag{3.51}$$

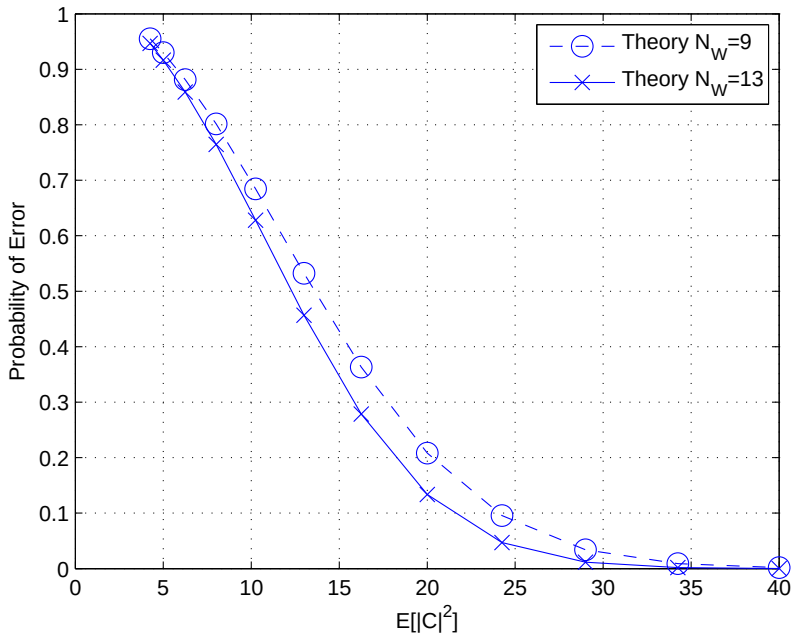

Figure 3.5: Comparing the effect of N_W in probability of error for Algorithm 2 for no target case with one clutter cell in each range.

In Figure 3.8 it can be seen that the theoretical and the simulation results are consistent. In Figure 3.9 there is a comparison between the cases including 1 clutter cell and 2 clutter cells. As expected, the performance of the algorithm is decreased when the number of clutter cells increases. In other words, the probability of error is increased when the number of clutter cells increases because there are more clutters to detect in each range cell.

3.8 Performance of Algorithm 1 for No Target Case with M Clutter Cells in Each Range

Following from Section 3.5 and Section 3.7, a general formula can be induced as in (3.52).

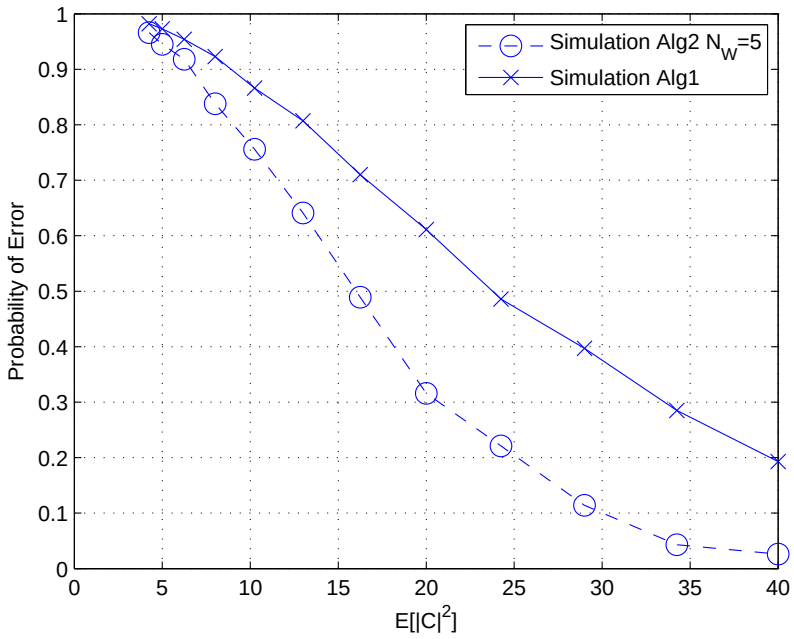

Figure 3.6: Comparing the probability of error for Algorithm 1 and Algorithm 2 for no target case with one clutter cell in each range.

Probability of Error:

$$P_e = 1 - \int (1 - F_C(z))^M \, n \, [F_N(z)]^{n-1} P_N(z) dz \tag{3.52}$$

3.9 Performance of Algorithm 2 for No Target Case with Two Clutter Cells in Each Range

There are two approaches in this case. The two approaches below are presented in a non-theoretical manner because of the difficulty in constructing a relationship between the value in a cell and the bin number of the cell. However, simulation based results and comparisons are available.

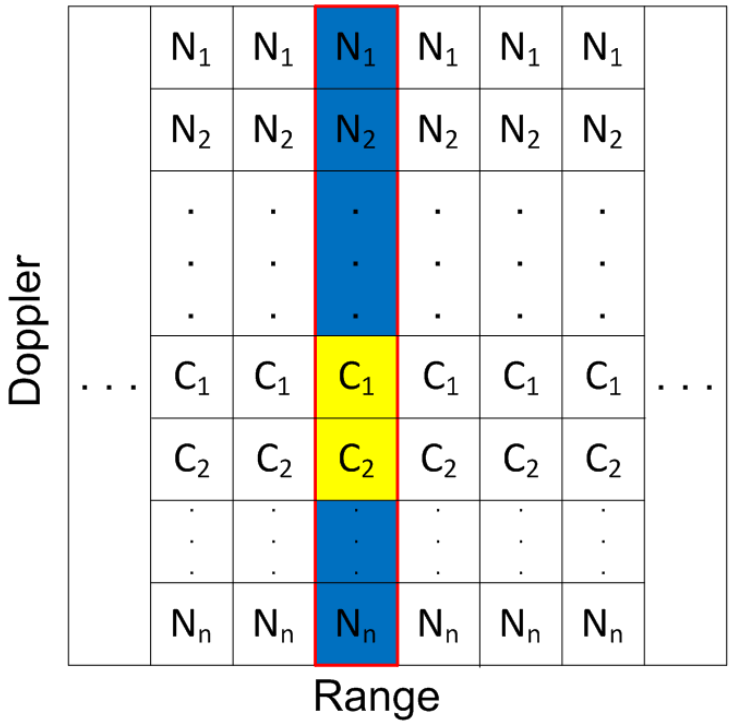

Figure 3.7: Description of Algorithm 1 for no target case with two clutter cells in each range.

3.9.1 Algorithm 2a

In this approach, the indices of the maximum valued and the second biggest valued cells are collected in different vectors. Then a median filter with length N_W is applied to both of the vectors. Individual outputs of the median filters are the indicators of the clutter.

In Figure 3.10 there is a comparison between Algorithm 1 and Algorithm 2a. Expectedly, Algorithm 2a has better performance than Algorithm 1. In other words, the probability of error of Algorithm 2a is less than the probability of error of Algorithm 1 because Algorithm 2a uses more data to decide the position of the clutter.

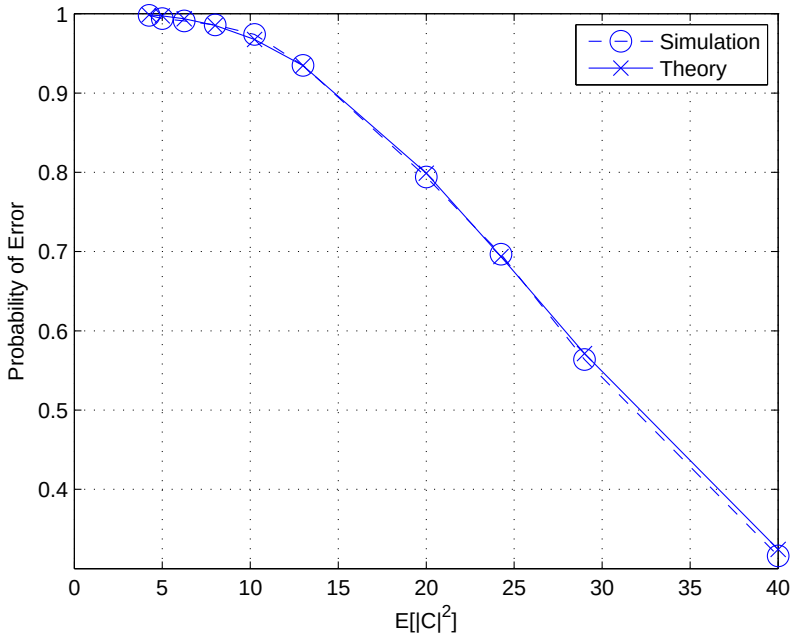

Figure 3.8: Probability of error for Algorithm 1 for no target case with two clutter cells in each range.

3.9.2 Algorithm 2b

In this approach, the indices of the maximum valued and the second biggest valued cells are collected in a matrix, the first row containing the maximum valued cell indices and the other row containing the second biggest valued cell indices. Then a $2\times N_W$ submatrix is selected. After combining them in a single vector, they are ordered. Then, two of the indices which are in the mid-position are selected as the clutter.

In Figure 3.11 there is a comparison between Algorithm 2a and Algorithm 2b. It can be seen that Algorithm 2b has better performance than Algorithm 2a. The probability of error of Algorithm 2b is less than that of Algorithm 2a. As a result, it is obvious that the 2b approach is better than the other one. Hence, Algorithm 2b should be used for this case of interest.

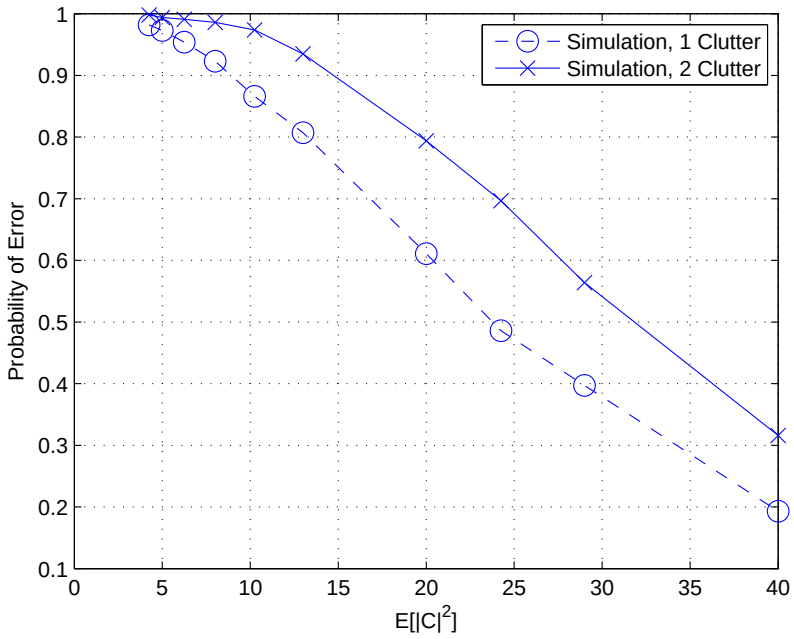

Figure 3.9: Comparing the probability of error for Algorithm 1 for no target case with one clutter cell and two clutter cells in each range.

3.10 Performance of Algorithm 1 for One Target Case with One Clutter Cell in Each Range

A description of this case is presented in Figure 3.12. The probability of error formula for this case is shown in (3.53) and the proof of the formula can be seen through (3.54) to (3.63).

Probability of Error:

$$P_e = 1 - \int [F_N(y)]^n F_T(y) P_C(y) dy \tag{3.53}$$

Proof:

$$P_{err} = 1 - P_{cor} \tag{3.54}$$

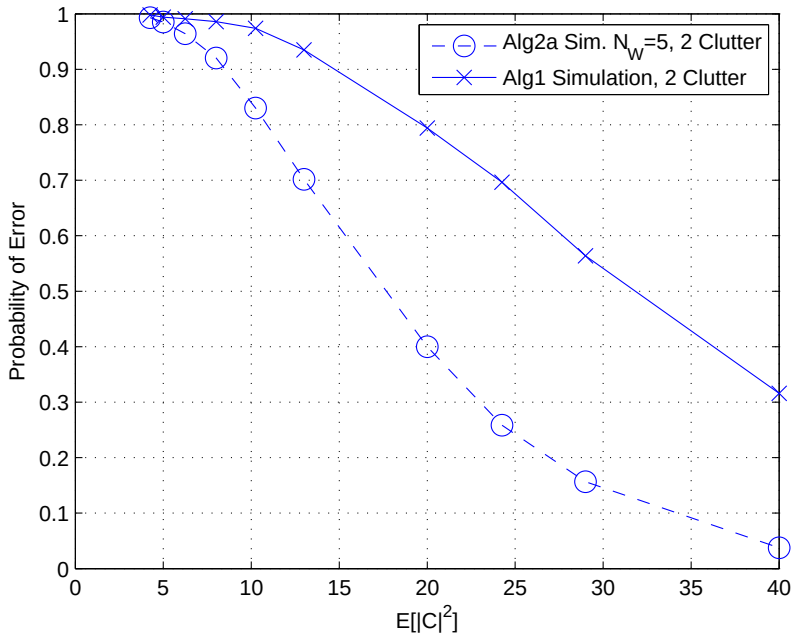

Figure 3.10: Comparing the probability of error for Algorithm 1 and Algorithm 2a for no target case with two clutter cells in each range.

The algorithm will make a correct decision if the value of C is greater than the maximum of the noise values and the value of the point target.

$$\begin{aligned} P_{cor} &= P((C > N_i, i = 1, ..., n)\&(C > T)) && (3.55)\\ &= P((C > \max(N_1, ..., N_n))\&(C > T)) && (3.56)\\ &= P((C > \tilde{N})\&(C > T)) && (3.57)\\ &= \int P(\tilde{N} < y)P(T < y)P_C(y)dy && (3.58)\\ &= \int F_{\tilde{N}}(y)F_T(y)P_C(y)dy && (3.59)\\ &= \int [F_N(y)]^n F_T(y)P_C(y)dy && (3.60) \end{aligned}$$

$$P_{cor} = \int [F_N(y)]^n F_T(y)P_C(y)dy \quad (3.61)$$

$$P_{err} = 1 - P_{cor} \quad (3.62)$$

$$P_{err} = 1 - \int [F_N(y)]^n F_T(y)P_C(y)dy \quad (3.63)$$

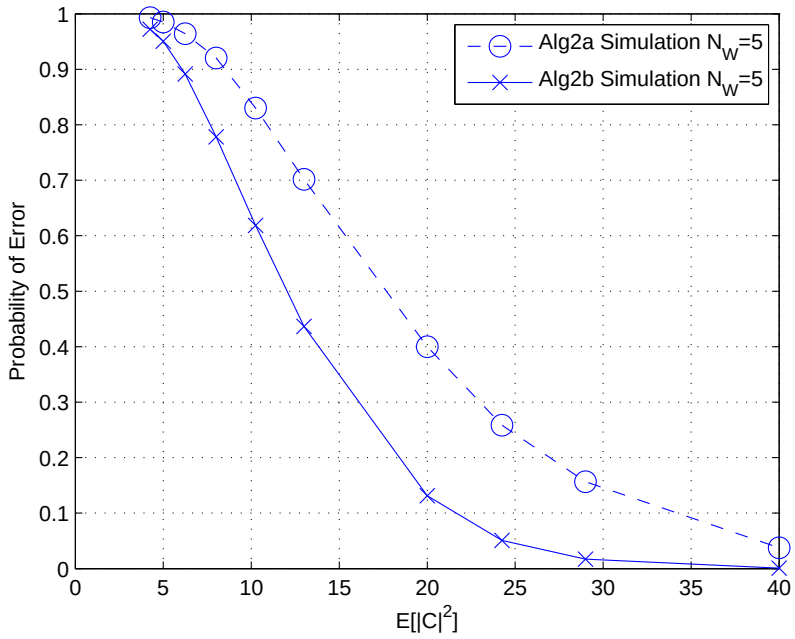

Figure 3.11: Comparing the probability of error for Algorithm 2a and Algorithm 2b for no target case with two clutter cells in each range.

In Figure 3.13 it can be seen that the theoretical and the simulation results are consistent.

3.11 Performance of Algorithm 1 for Two Target Case with One Clutter Cell in Each Range

A description of this case is presented in Figure 3.14. The probability of error formula for this case is shown in (3.64) and the proof of the formula can be seen through (3.65) to (3.75).

Probability of Error:

$$P_e = 1 - \int [F_N(y)]^n [F_T(y)]^2 P_C(y) dy \tag{3.64}$$

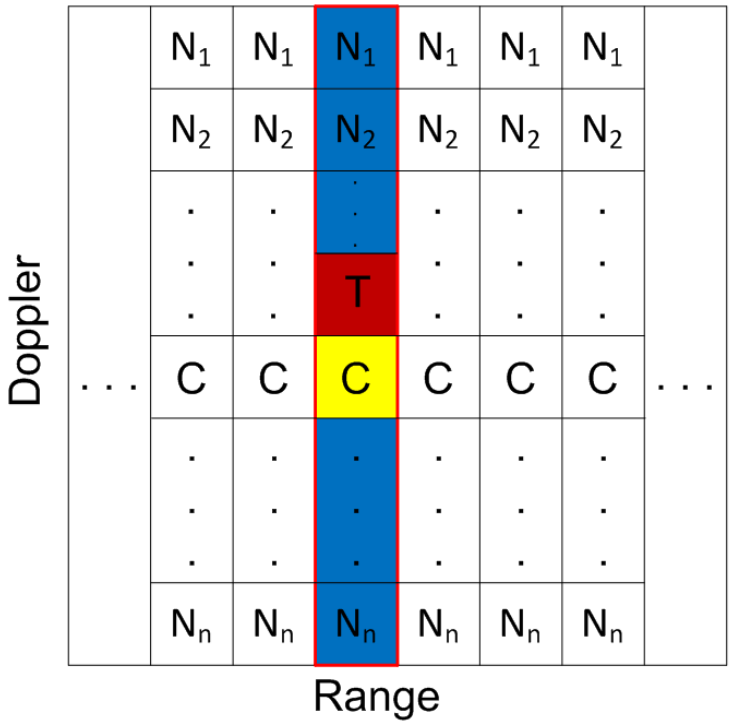

Figure 3.12: Description of Algorithm 1 for one target case with one clutter cell in each range.

Proof:

$$P_{err} = 1 - P_{cor} \tag{3.65}$$

The algorithm will make a correct decision if the value of C is greater than the maximum of the noise values and the values of the two point targets.

$$\begin{aligned}
P_{cor} &= P((C > N_i, i = 1, ..., n)\&(C > T_1)\&(C > T_2)) && (3.66)\\
&= P((C > \max(N_1, ..., N_n))\&(C > T_1)\&(C > T_2)) && (3.67)\\
&= P((C > \tilde{N})\&(C > T_1)\&(C > T_2)) && (3.68)\\
&= \int P(\tilde{N} < y)P(T_1 < y)P(T_2 < y)P_C(y)dy && (3.69)\\
&= \int P(\tilde{N} < y)P(T < y)^2 P_C(y)dy && (3.70)\\
&= \int F_{\tilde{N}}(y)\left[F_T(y)\right]^2 P_C(y)dy && (3.71)\\
&= \int \left[F_N(y)\right]^n \left[F_T(y)\right]^2 P_C(y)dy && (3.72)
\end{aligned}$$

$$P_{cor} = \int \left[F_N(y)\right]^n \left[F_T(y)\right]^2 P_C(y)dy \tag{3.73}$$

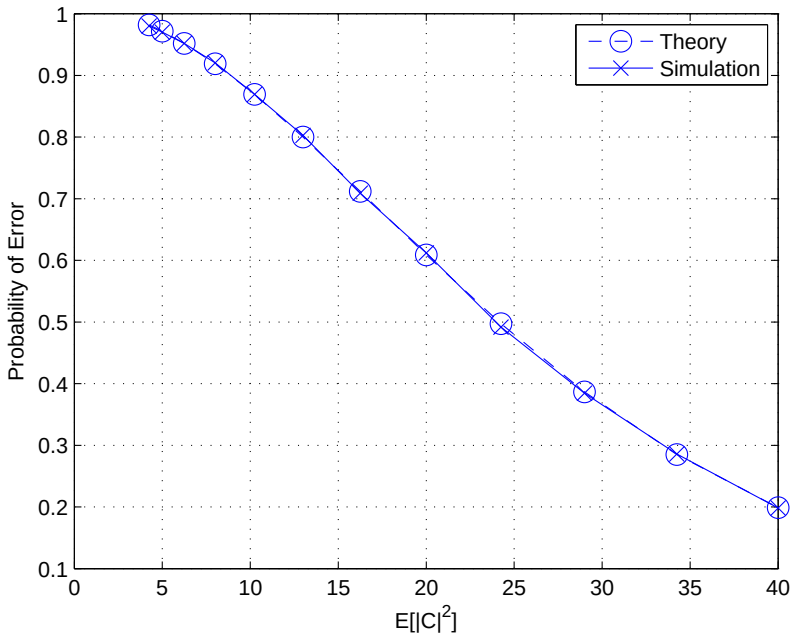

Figure 3.13: Probability of error for Algorithm 1 for one target case with one clutter cell in each range.

$$P_{err} = 1 - P_{cor} \tag{3.74}$$

$$P_{err} = 1 - \int [F_N(y)]^n \left[F_T(y)\right]^2 P_C(y) dy \tag{3.75}$$

In Figure 3.15 it can be seen that the theoretical and the simulation results are consistent.

3.12 Performance of Algorithm 1 for L Target Case with One Clutter Cell in Each Range

Following from Section 3.10 and Section 3.11, a general formula can be induced as in (3.76).

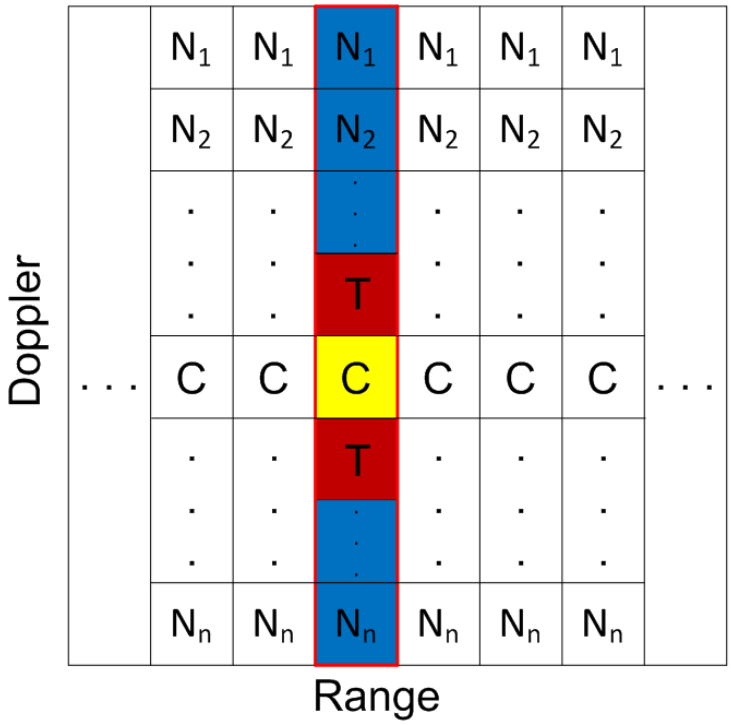

Figure 3.14: Description of Algorithm 1 for two target case with one clutter cell in each range.

Probability of Error:

$$P_e = 1 - \int [F_N(y)]^n [F_T(y)]^L P_C(y) dy \tag{3.76}$$

3.13 Performance of Algorithm 1 for One Target Case with Two Clutter Cells in Each Range

A description of this case is provided in Figure 3.16. The probability of error formula for this case is shown in (3.77) and the proof of the formula can be seen through (3.78) to (3.102).

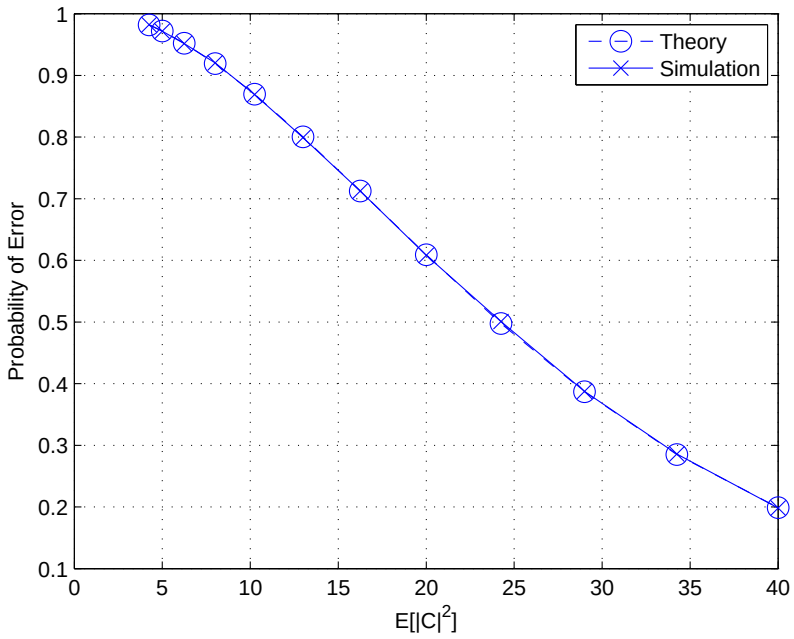

Figure 3.15: Probability of error for Algorithm 1 for two target case with 1 clutter cell in each range.

Probability of Error:

$$P_e = 1 - \int \left[F_N(y)\right]^n F_T(y) \left[2P_C(y) - 2F_C(y)P_C(y)\right] dy \tag{3.77}$$

Proof:

$$P_{err} = 1 - P_{cor} \tag{3.78}$$

The algorithm will make a correct decision if the value of the minimum of C_1 and C_2 is greater than the maximum of the noise values and the value of the point target.

$$\begin{aligned}
P_{cor} &= P((\min(C_1, C_2) > N_i, i = 1, ..., n)\&(\min(C_1, C_2) > T)) && (3.79)\\
&= P((\min(C_1, C_2) > \max(N_1, ..., N_n))\&(\min(C_1, C_2) > T)) && (3.80)\\
&= P((\min(C_1, C_2) > \tilde{N})\&(\min(C_1, C_2) > T)) && (3.81)\\
&= P((\tilde{C_2} > \tilde{N})\&(\tilde{C_2} > T)) && (3.82)\\
&= \int P(\tilde{N} < y)P(T < y)P_{\tilde{C_2}}(y)dy && (3.83)
\end{aligned}$$

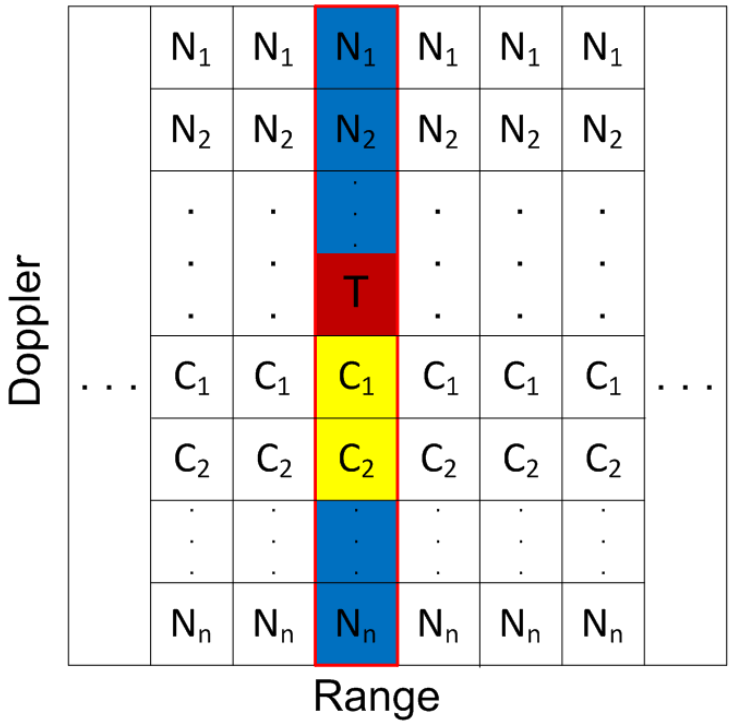

Figure 3.16: Description of Algorithm 1 for one target case with two clutter cells in each range.

where $\tilde{C_2} = \min(C_1, C_2)$ and C_1 and C_2 are i.i.d. Then,

$$
\begin{aligned}
F_{\tilde{C_2}}(y) &= P(\tilde{C_2} < y) && (3.84)\\
&= P(\min(C_1, C_2) < y) && (3.85)\\
&= 1 - P(\min(C_1, C_2) > y) && (3.86)\\
&= 1 - P((C_1 > y)\&(C_2 > y)) && (3.87)
\end{aligned}
$$

C_i's being independent, (3.88)

$$
\begin{aligned}
&= 1 - P(C_1 > y)P(C_2 > y) && (3.89)\\
&= 1 - [(1 - P(C_1 < y))(1 - P(C_2 < y))] && (3.90)\\
&= 1 - [(1 - F_{C_1}(y))(1 - F_{C_2}(y))] && (3.91)\\
&= 1 - [1 - F_{C_2}(y) - F_{C_1}(y) + F_{C_1}(y)F_{C_2}(y)] && (3.92)\\
&= F_{C_1}(y) + F_{C_2}(y) - F_{C_1}(y)F_{C_2}(y) && (3.93)
\end{aligned}
$$

C_i's being identical, (3.94)

$$
\begin{aligned}
&= 2F_C(y) - [F_C(y)]^2 && (3.95)
\end{aligned}
$$

$$P_{\tilde{C}_2}(y) = \frac{d}{dy}F_{\tilde{C}_2}(y) \tag{3.96}$$

$$= \frac{d}{dy}\left(2F_C(y) - [F_C(y)]^2\right) \tag{3.97}$$

$$= 2P_C(y) - 2F_C(y)P_C(y) \tag{3.98}$$

$$P_{cor} = \int P(\tilde{N} < y)P(T < y)P_{\tilde{C}_2}(y)dy \tag{3.99}$$

$$= \int [F_N(y)]^n F_T(y)\left[2P_C(y) - 2F_C(y)P_C(y)\right]dy \tag{3.100}$$

$$P_{err} = 1 - P_{cor} \tag{3.101}$$

$$= 1 - \int [F_N(y)]^n F_T(y)\left[2P_C(y) - 2F_C(y)P_C(y)\right]dy \tag{3.102}$$

In Figure 3.17 it can be seen that the theoretical and the simulation results are consistent.

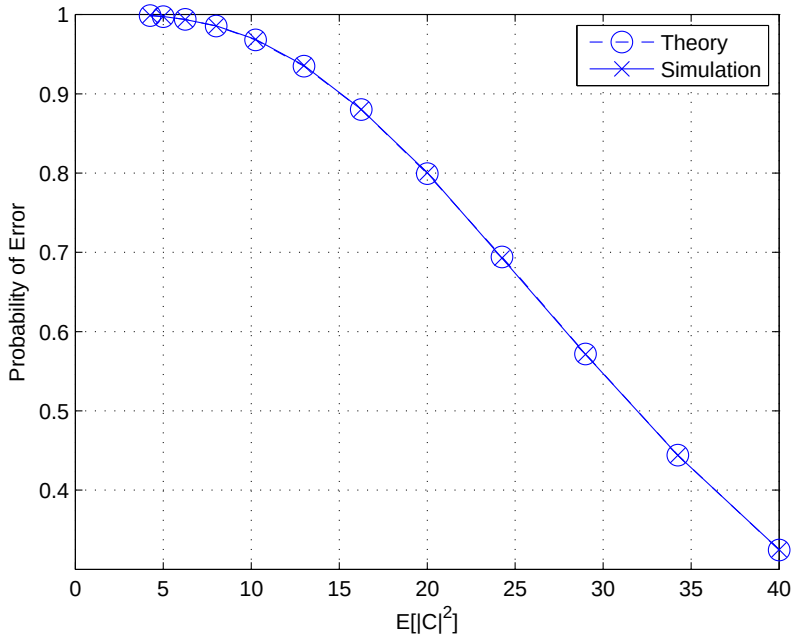

Figure 3.17: Probability of error for Algorithm 1 for one target case with two clutter cells in each range.

3.14 Performance of Algorithm 2 for One Target Case with One Clutter Cell in Each Range

A description of this case is presented in Figure 3.18. There can be no or one target in each range cell. In this case, the number of range cells that include no target, a target that is above the clutter and a target that is below the clutter in a median window must be given. The probability of error formula for this case is shown through (3.103) to (3.120).

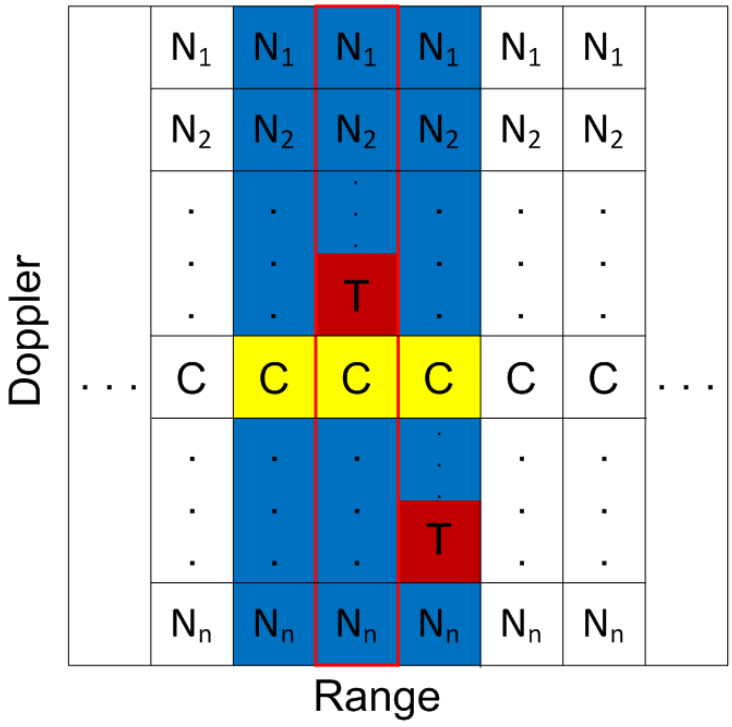

Figure 3.18: Description of Algorithm 2 for one target case with one clutter cell in each range.

Probability of Error:

$$P_e = (2K+1)!\left[\sum_{a=K+1}^{2K+1}\frac{x^a}{a!}\left(\sum_{b=0}^{2K+1-a}\frac{c^b y^{(2K+1-a-b)}}{b!(2K+1-a-b)!}\right)\right] + (2K+1)!\left[\sum_{a=K+1}^{2K+1}\frac{y^a}{a!}\left(\sum_{b=0}^{2K+1-a}\frac{c^b x^{(2K+1-a-b)}}{b!(2K+1-a-b)!}\right)\right] \quad (3.103)$$

where c is the correct decision probability of position of clutter, x is the probability of deciding a cell that is above the clutter cell and y is the probability of deciding a cell that is below the clutter cell in each range. Namely,

$$c = \frac{N_{T_n}}{N_{\mathrm{w}}}c_{T_n} + \frac{N_{T_a}}{N_{\mathrm{w}}}c_{T_a} + \frac{N_{T_b}}{N_{\mathrm{w}}}c_{T_b} \quad (3.104)$$

$$x = \frac{N_{T_n}}{N_{\mathrm{w}}}x_{T_n} + \frac{N_{T_a}}{N_{\mathrm{w}}}x_{T_a} + \frac{N_{T_b}}{N_{\mathrm{w}}}x_{T_b} \quad (3.105)$$

$$y = \frac{N_{T_n}}{N_{\mathrm{w}}}y_{T_n} + \frac{N_{T_a}}{N_{\mathrm{w}}}y_{T_a} + \frac{N_{T_b}}{N_{\mathrm{w}}}y_{T_b} \quad (3.106)$$

where N_{T_n} is the number of range cells that include no target, N_{T_a} is the number of range cells that include a target lying above clutter, N_{T_b} is the number of range cells that include a target lying below the clutter and $N_W = 2K+1$ is the median filter length. Here, c_{T_n} is the correct decision probability of clutter when there is no target in a range cell, and c_{T_a} and c_{T_b} are the correct decision probability of clutter when there is one target in a range cell. Actually, c_{T_a} is equal to c_{T_b}. Up to now all the defined variables above are derived previously as follows:

$$c_{T_n} = \int (1 - F_C(z))\, n\, [F_N(z)]^{n-1} P_N(z) dz \quad (3.107)$$

$$c_{T_a} = c_{T_b} = \int [F_N(y)]^n F_T(y) P_C(y) dy \quad (3.108)$$

In (3.105), x_{T_n} is the probability of deciding a bin that is above the clutter bin when there is no target in a range cell, x_{T_a} is the probability of deciding a bin that is above the clutter when a target lies above the clutter in a range cell, and x_{T_b} is the probability of deciding a bin that is above the clutter when a target lies below the clutter in a range cell. Similarly in (3.106), y_{T_n} is the probability of deciding a bin that is below the clutter when there is no target in a range cell,

y_{T_a} is the probability of deciding a bin that is below the clutter when a target lies above the clutter in a range cell, and y_{T_b} is the probability of deciding a bin that is below the clutter when a target lies below the clutter in a range cell. Namely,

$$x_{T_n} = (1-c)\varepsilon \tag{3.109}$$

$$y_{T_n} = (1-c)(1-\varepsilon) \tag{3.110}$$

$$x_{T_a} = (1-c-t)\varepsilon + t \tag{3.111}$$

$$y_{T_a} = (1-c-t)(1-\varepsilon) \tag{3.112}$$

$$x_{T_b} = (1-c-t)\varepsilon \tag{3.113}$$

$$y_{T_b} = (1-c-t)(1-\varepsilon) + t \tag{3.114}$$

where ε is the ratio of the number of noise bins that is above the clutter bin to the total number of noise bins in a range cell and t is the correct decision probability of position of target, which is derived as follows:

$$t = P((T > C)\&(T > N_i, i = 1, ..., n)) \tag{3.115}$$

$$= P((T > C)\&(T > \max(N_1, ..., N_n))) \tag{3.116}$$

$$= P((T > C)\&(T > \tilde{N})) \tag{3.117}$$

$$= \int P(C < t)P(\tilde{N} < t)P_T(t)dt \tag{3.118}$$

$$= \int F_C(t)F_{\tilde{N}}(t)P_T(t)dt \tag{3.119}$$

$$= \int F_C(t)[F_N(t)]^n P_T(t)dt \tag{3.120}$$

In Figure 3.19 it can be observed that the theoretical and the simulation results are consistent.

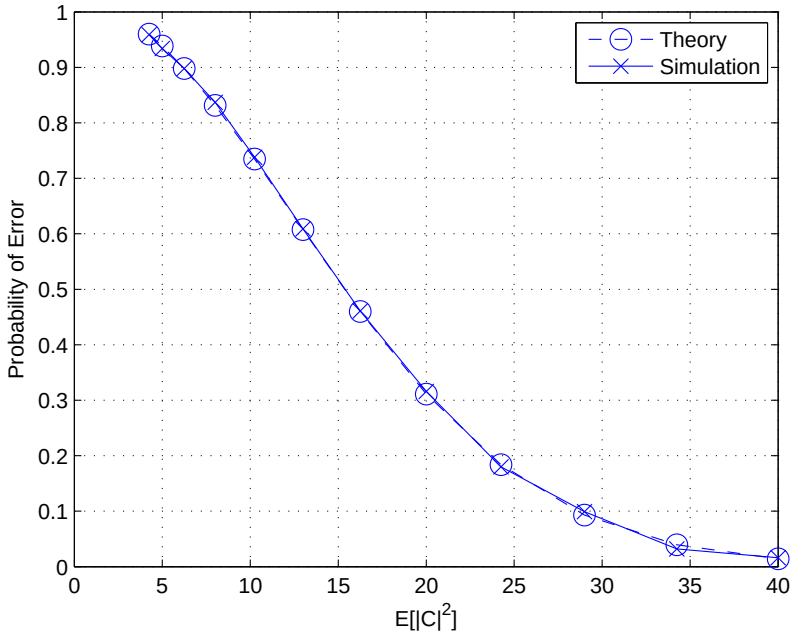

Figure 3.19: Probability of error for Algorithm 2 for one target case with one clutter cell in each range

Chapter 4

SIMULATION RESULTS ON REALISTIC DATA

In this chapter, the proposed clutter detection algorithms are run on some realistic clutter data. Here, the clutter data is not taken from a real radar that conducted an experiment on a real terrain. It is a synthetically generated data based on a reasonable model of the real clutter data. For this reason, the word "realistic" instead of "real" is used. The simulation results are compared with the theoretical results to see whether they are consistent.

In order to get the realistic clutter data, first, a synthetic terrain is generated. The terrain is generated in accordance with the reality [1]. It has vegetation on smooth hills. Then, a synthetic radar is generated. This radar is modeled as mounted on a helicopter platform. Therefore, the radar has a velocity. The transmitted radar signal is reflected from the environment and then the radar receives the reflected signal. After that, the signal is processed using pulse-Doppler processing techniques. Finally, the realistic clutter data is obtained on range-Doppler matrix for specific azimuth angles. The realistic data generating system model was explained in detail in Chapter 2.

In some cases while performing simulations, targets are present in the environment. For such cases, synthetically generated point targets are used. In addition, for some cases, a realistic target model is employed in the system. The realistic target model used in the simulations are explained in Chapter 2.

In the simulations, a noise signal is added to the clutter and target signals. Because the noise signal is synthetically generated, its parameter used in the theoretical calculations are known. However, the clutter parameter is unknown. In order to use the formulas derived in the previous chapter, the clutter parameter should be estimated.

4.1 Clutter Statistics

In the simulations, 70 and 90 degree azimuth angle signals are used and the results are presented. The 90 degree azimuth angle is the zero Doppler angle. Because the terrain is perpendicular to the velocity of the platform, the clutter appears in the zero Doppler region. The detection signal for the 90 degree azimuth angle can be seen in Figure 4.1 and the detection signal for the 70 degree azimuth angle is illustrated in Figure 4.2. More noisy versions of the detection signals can be seen in Figure 4.3 and Figure 4.4. In the 90 degree azimuth angle detection signal, the clutter is in the zero Doppler region as expected.

After having the detection signals, the histogram of the clutter is generated in order to have an idea about the distribution of the clutter. The histogram of the clutter lying on the 90 degree azimuth angle can be seen in Figure 4.5 and the histogram of the clutter lying on the 70 degree azimuth angle can be seen in Figure 4.6. The histograms of the more noisy versions of the clutter are shown in Figure 4.7 and Figure 4.8. Looking at the histogram figures, it can be observed

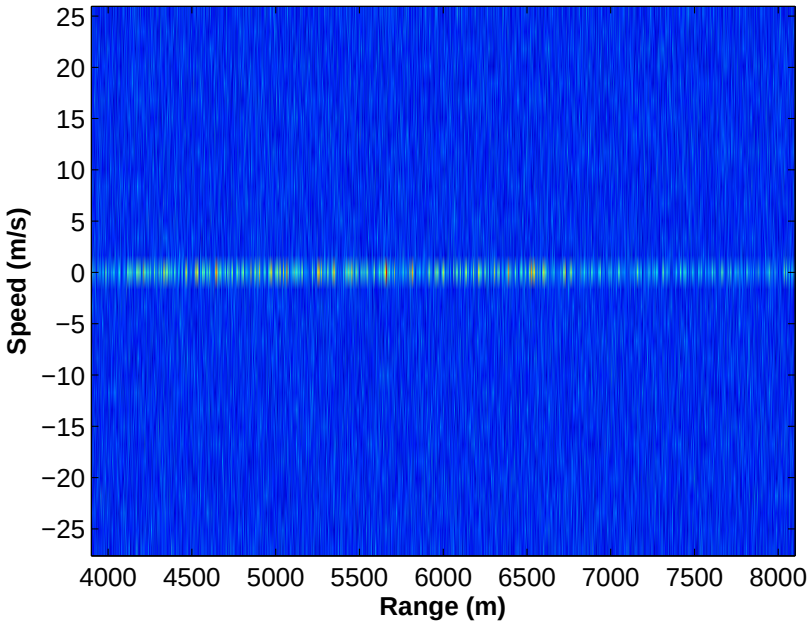

Figure 4.1: Detection signal for 90 degree azimuth angle.

that the histogram of the clutter resembles the Rayleigh distribution. Hence, the parameter of the Rayleigh distribution can be used as the clutter parameter in the theoretical formulas.

From now on, the clutter distribution is approximated as Rayleigh distribution. The Rayleigh parameter should be estimated from the clutter sample values. The estimation can be performed by moments matching (method of moments) [29]. One estimate can be obtained by using the sample mean and the other by using the sample variance. The mean of the Rayleigh distribution is given in (4.1) and the variance of the Rayleigh distribution is given in (4.2). From the sample mean, the parameter $\hat{\sigma}_m$ can be estimated as in (4.3) and from the sample variance, the parameter $\hat{\sigma}_v$ can be estimated as in (4.4). Another estimate can be obtained according to the maximum likelihood (ML) technique. The Rayleigh parameter $\hat{\sigma}_{ML}$ can be estimated using the ML technique as in (4.5).

$$\text{mean} = \sigma\sqrt{\frac{\pi}{2}} \tag{4.1}$$

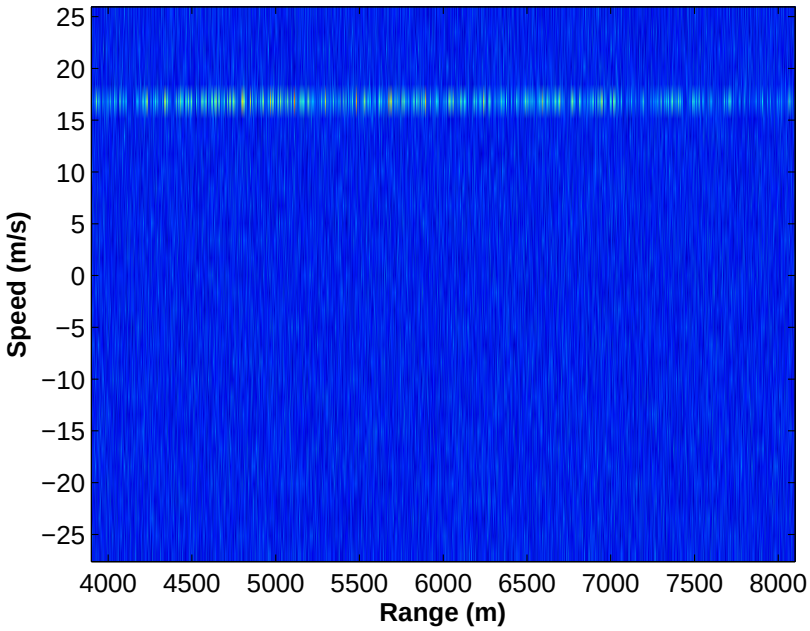

Figure 4.2: Detection signal for 70 degree azimuth angle.

$$\text{variance} = \frac{4-\pi}{2}\sigma^2 \tag{4.2}$$

$$\hat{\sigma}_m = \frac{\text{sample mean}}{\sqrt{\frac{\pi}{2}}} \tag{4.3}$$

$$\hat{\sigma}_v = \sqrt{\frac{\text{sample variance}}{\frac{4-\pi}{2}}} \tag{4.4}$$

$$\hat{\sigma}_{ML} = \sqrt{\frac{1}{2N_s}\sum_{i=1}^{N_s} k_i} \tag{4.5}$$

where k_i's are the samples and N_s is the number of samples.

In order to decide which estimate works the best on the realistic data, a comparison must be performed between the real distribution of the clutter and the distribution obtained from the estimates. To be able to make a comparison, the real distribution of the clutter must be obtained. This can be done by using kernel density estimation. The kernel density estimation of clutter's probability

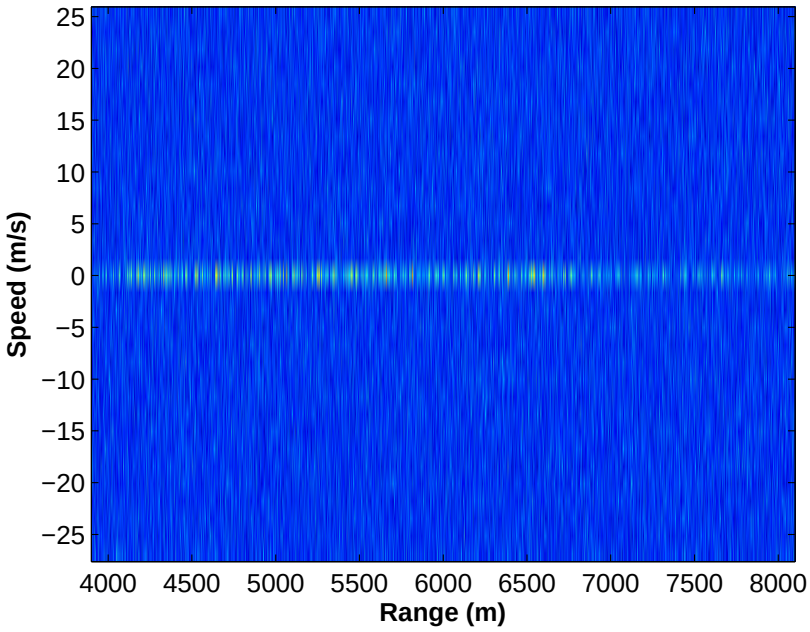

Figure 4.3: More noisy detection signal for 90 degree azimuth angle.

density function is given by

$$\hat{f}_h(x) = \frac{1}{N_s h} \sum_{i=1}^{N_s} K\left(\frac{x - k_i}{h}\right) \tag{4.6}$$

where k_i's are the samples, N_s is the number of samples, $K(\cdot)$ is some kernel and h is a smoothing parameter.

Generally K is taken to be the standard Gaussian function with zero mean and unit variance as [30]

$$K\left(\frac{x - k_i}{h}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-k_i)^2}{2h^2}}. \tag{4.7}$$

The kernel density estimated real distribution and the distributions obtained using the moments matching and ML estimators for 90 degree azimuth can be seen in Figure 4.9. The distributions for 70 degree azimuth can be seen in Figure 4.10. The distributions for more noisy clutter can be seen in Figure 4.11 and Figure 4.12.

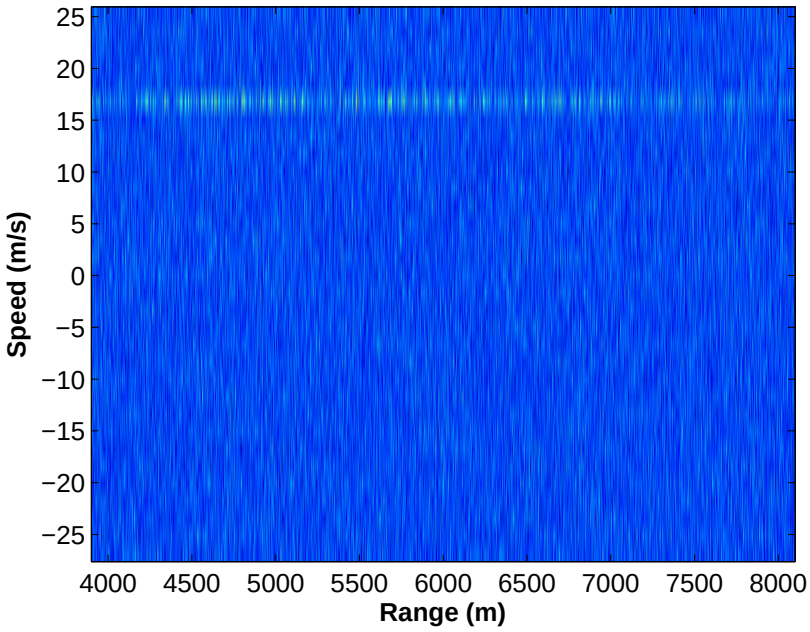

Figure 4.4: More noisy detection signal for 70 degree azimuth angle.

The comparison of the estimated density functions can be done using the Kullback-Leibler divergence, which is a measure of the difference between two probability distributions R and Q. Typically R represents a distribution which is precisely calculated, and Q typically represents an approximation of R. The KL distance is defined as [31]

$$D_{KL}\left(R||Q\right) = \int_{-\infty}^{\infty} r(x) \log \frac{r(x)}{q(x)} dx. \tag{4.8}$$

Here, R represents the clutter distribution calculated with kernel density estimation and Q represents the moments matching and ML estimators of the clutter distribution. The estimated parameters of the estimators and the results for the KL distance can be seen in Table 4.1 and Table 4.2.

Looking at the KL distance results, it is observed that the minimum distance to the clutter density is obtained with the density constructed using the ML estimator in each and every trial. Among the estimators used, the ML estimator has the best performance. From now on, in all the simulations using the realistic

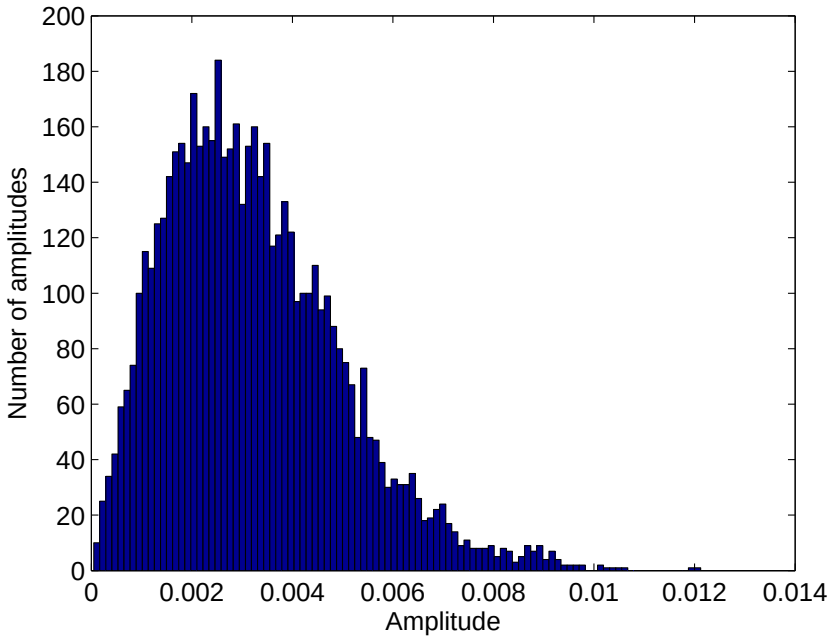

Figure 4.5: Histogram of clutter for 90 degree azimuth angle.

	90 degree azimuth	70 degree azimuth
$\hat{\sigma}_{ML}$	0.002615	0.002295
$\hat{\sigma}_m$	0.002591	0.002290
$\hat{\sigma}_v$	0.002702	0.002301
KL_{ML}	26.9430	1.33869
KL_m	28.0510	2.09471
KL_v	49.7358	29.1347

Table 4.1: Estimated parameters of clutter and KL distance results.

data, the distribution of the clutter used in the theoretical formulas will be obtained using the ML estimator.

4.2 Simulation Results

In this section, the probability of error values are obtained for different clutter-to-noise ratio (CNR) values. The simulation results are compared with the theoretical ones. The radar parameters and the platform on which the radar is deployed allows a single clutter bin in each range cell, and the analysis is performed accordingly.

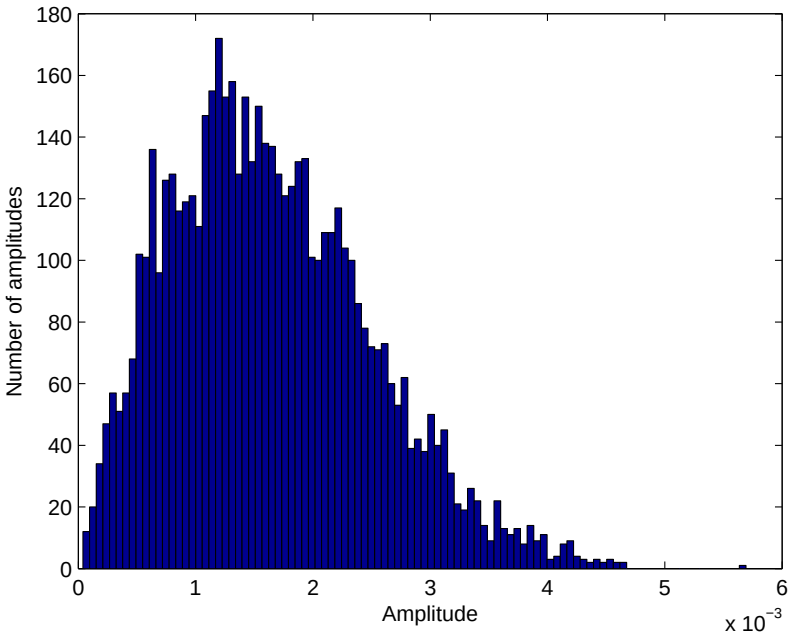

Figure 4.6: Histogram of clutter for 70 degree azimuth angle.

	90 degree azimuth	70 degree azimuth
$\hat{\sigma}_{ML}$	0.002767	0.002284
$\hat{\sigma}_m$	0.002734	0.002282
$\hat{\sigma}_v$	0.002887	0.002293
KL_{ML}	6.23070	25.5667
KL_m	8.496880	26.6492
KL_v	42.8982	30.2588

Table 4.2: Estimated parameters of more noisy clutter and KL distance results.

The comparison of the theoretical and simulation results with no target using Algorithm 1 for 90 degree azimuth is presented in Figure 4.13 and for 70 degree azimuth is shown in Figure 4.14.

The comparison of the theoretical and simulation results with no target using Algorithm 2 for 90 degree azimuth is presented in Figure 4.15 and for 70 degree azimuth is given in Figure 4.16.

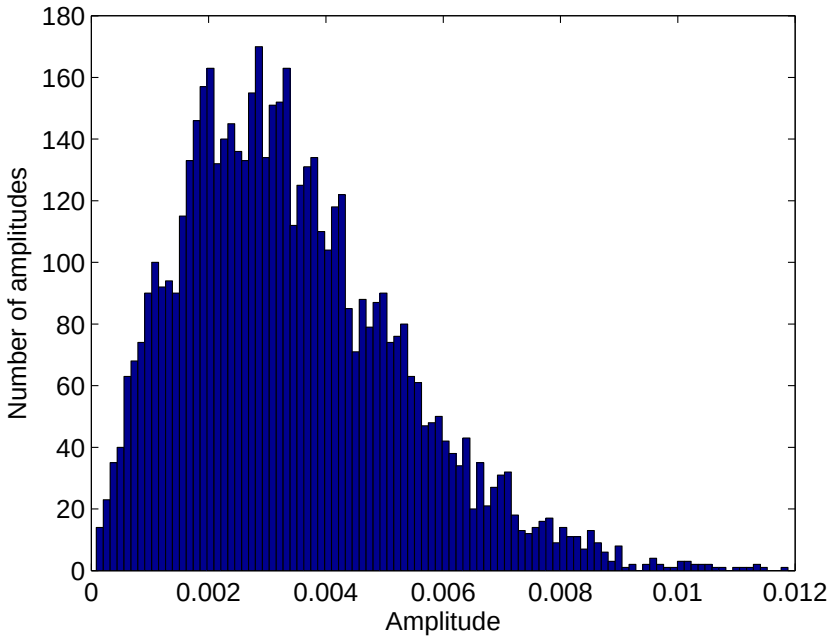

Figure 4.7: Histogram of more noisy clutter for 90 degree azimuth angle.

The comparison of the theoretical and simulation results with one point target using Algorithm 1 for 90 degree azimuth is given in Figure 4.17 and for 70 degree azimuth is shown in Figure 4.18.

From Figure 4.13 to Figure 4.18, the figures for 90 degree azimuth and the ones for 70 degree azimuth are similar in each case. Also, the theoretical and the simulation results are not matched perfectly but it is observed that the results are very close. This situation has reasonable causes. One of the reasons is that the estimated clutter distribution is used instead of the real distribution of the clutter. From Figure 4.9 to Figure 4.12, it can be seen that the kernel density estimated distribution does not perfectly match the distribution obtained from the ML estimator. The other reason is that it is assumed that the clutter is present in only one cell for each range cell. Actually, the clutter mainly stays in one cell but a weak spread occurs along the range cell. Those reasons are behind the situation of the slight mismatch between the theory and the simulation.

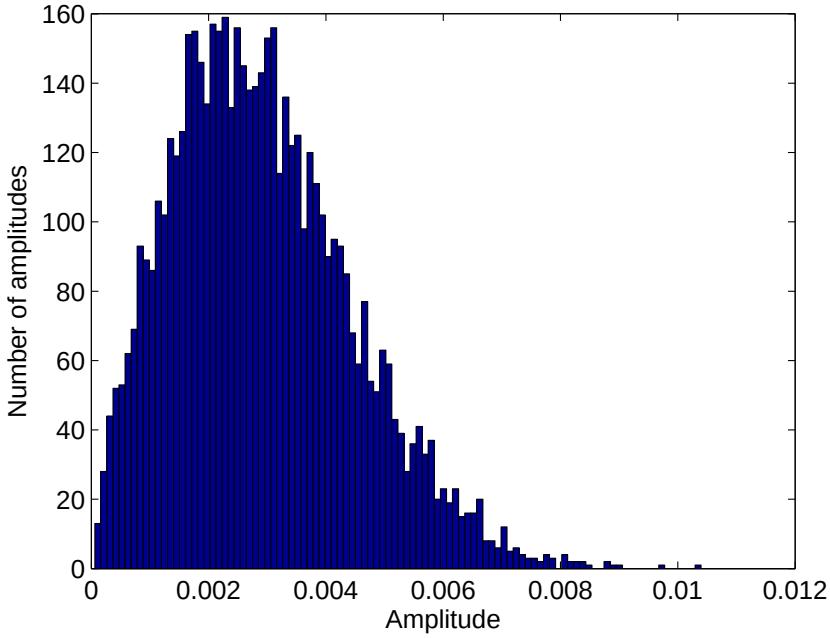

Figure 4.8: Histogram of more noisy clutter for 70 degree azimuth angle.

Comparing Figure 4.13 and Figure 4.15, it can be observed that Algorithm 2 has better performance than Algorithm 1, as expected. Algorithm 2 achieves a 0.05 probability of error with a 13 dB CNR while Algorithm 1 achieves the same probability of error with a 20 dB CNR.

Up to now, the comparison between the theoretical results and the simulation results are performed. The following discussion will be on the results of the algorithms on the range-Doppler matrix. In Figure 4.19, the noiseless detection signal with a realistic target for 90 degree azimuth can be seen. In Figure 4.20, the result of Algorithm 1 are shown and in Figure 4.21 the result of Algorithm 2 are illustrated. When noise is added to the noiseless detection signal, the results of Algorithm 1 and Algortihm 2 can be seen in Figure 4.22 and Figure 4.23, respectively.

As seen from the figures, both algorithms work well in the absence of noise. However, Algorithm 1 have errors when a target is present as expected. When

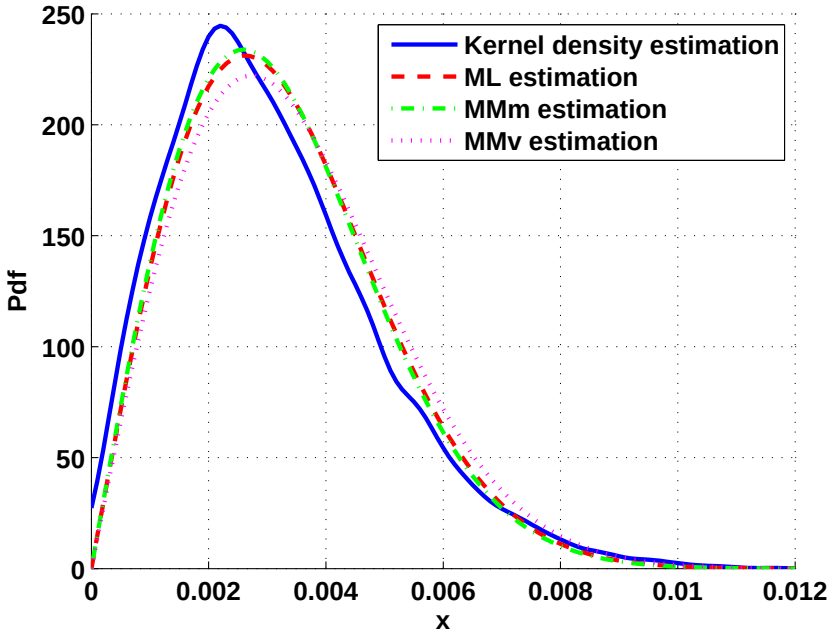

Figure 4.9: Clutter pdf's for different estimators for 90 degree azimuth.

noise is added to the pure detection signal, the performance difference between Algorithm 1 and Algorithm 2 becomes significant. In presence of the noise, Algortihm 2 works very well but Algorithm 1 have many errors.

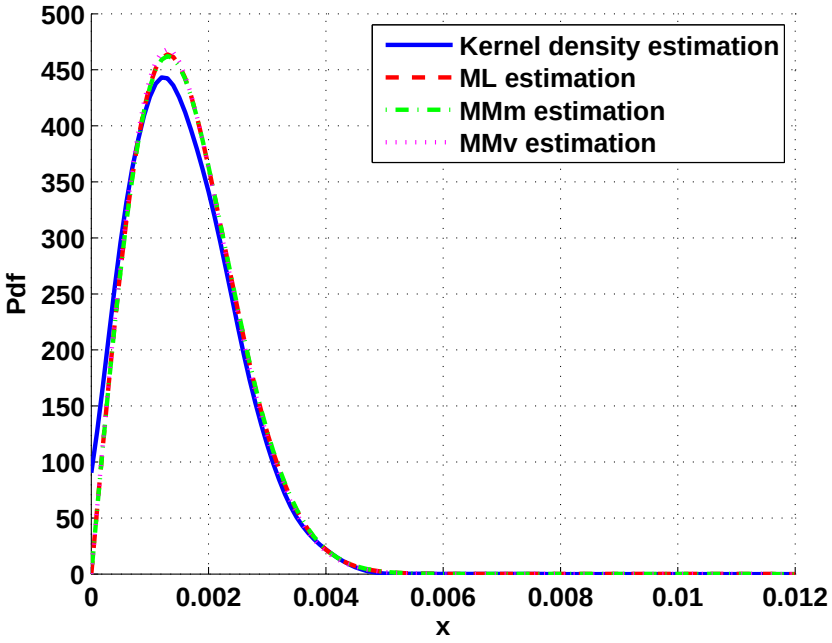

Figure 4.10: Clutter pdf's for different estimators for 70 degree azimuth.

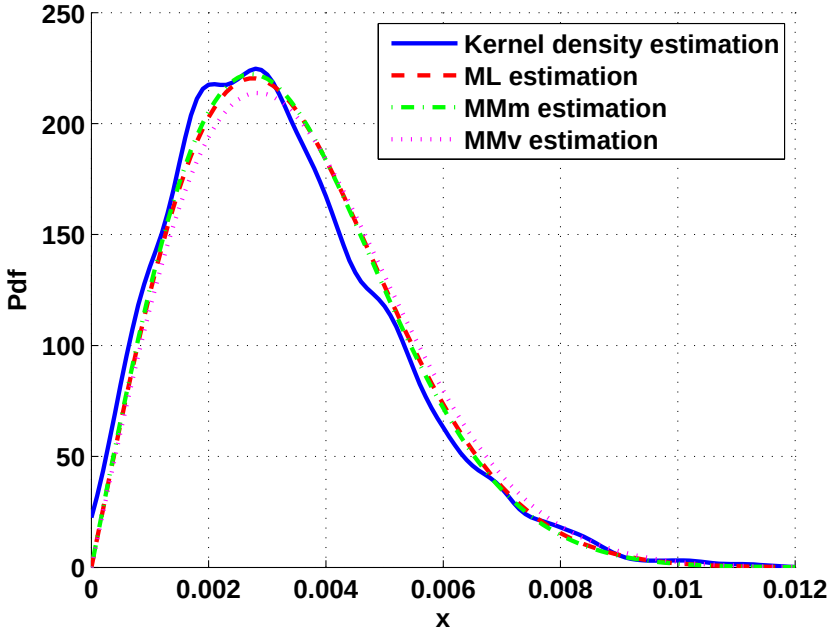

Figure 4.11: More noisy clutter pdf's for different estimators for 90 degree azimuth.

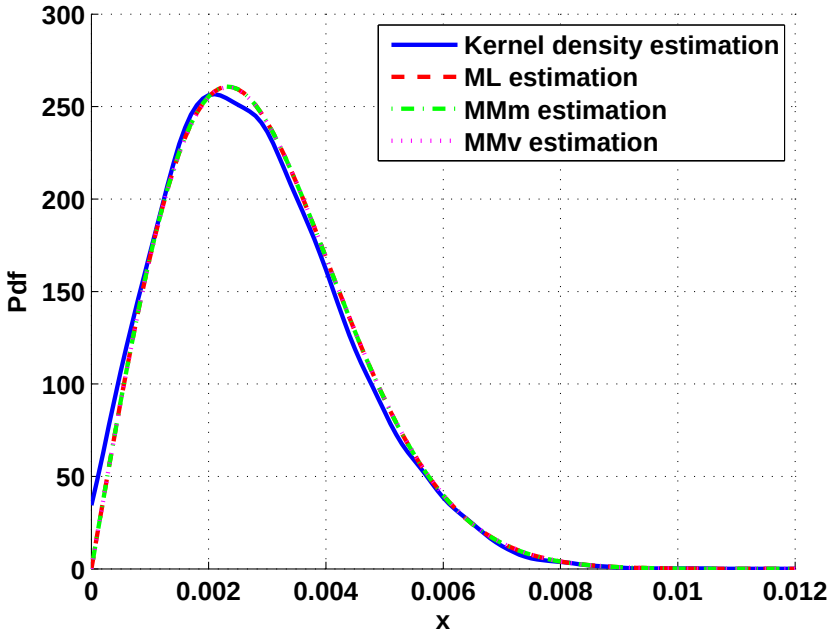

Figure 4.12: More noisy clutter pdf's for different estimators for 70 degree azimuth.

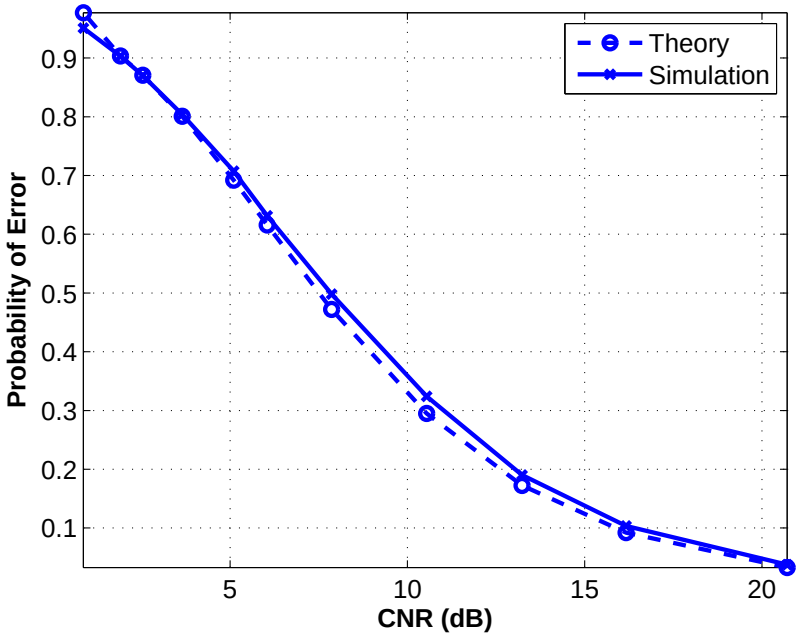

Figure 4.13: Comparing the simulation results with theoretical results for no target case using Algorithm 1 for 90 degree azimuth.

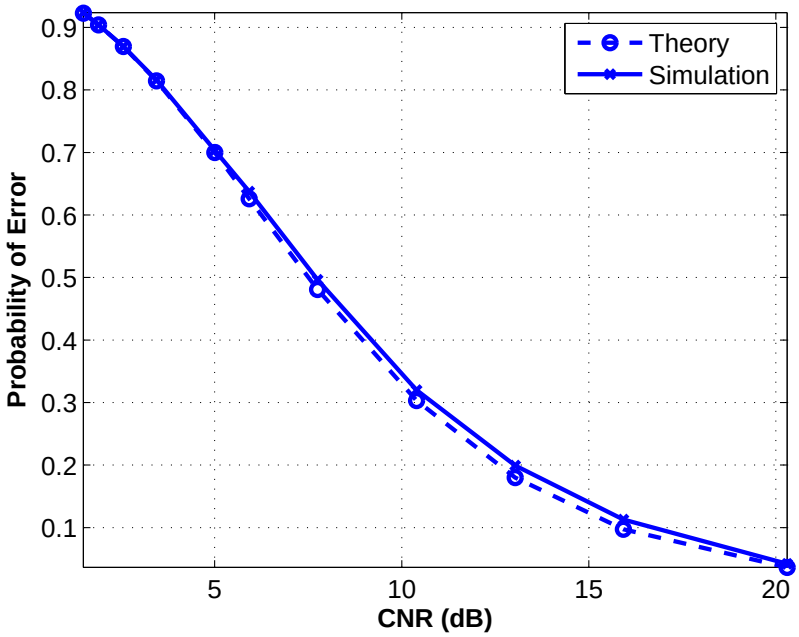

Figure 4.14: Comparing the simulation results with theoretical results for no target case using Algorithm 1 for 70 degree azimuth.

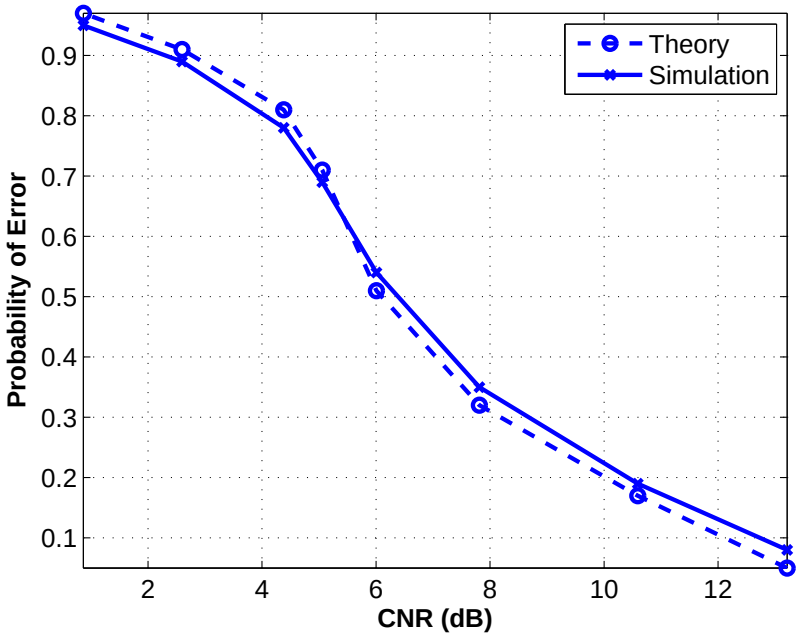

Figure 4.15: Comparing the simulation results with theoretical results for no target case using Algorithm 2 for 90 degree azimuth.

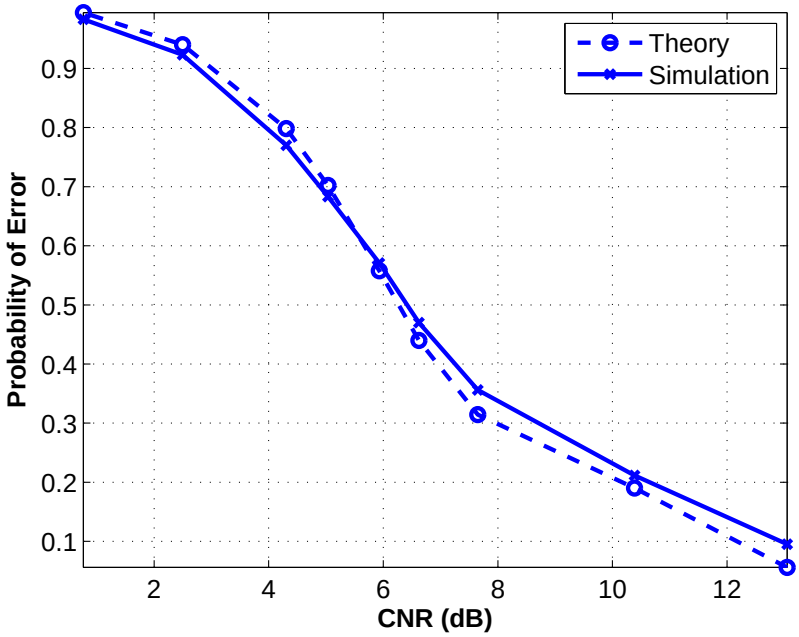

Figure 4.16: Comparing the simulation results with theoretical results for no target case using Algorithm 2 for 70 degree azimuth.

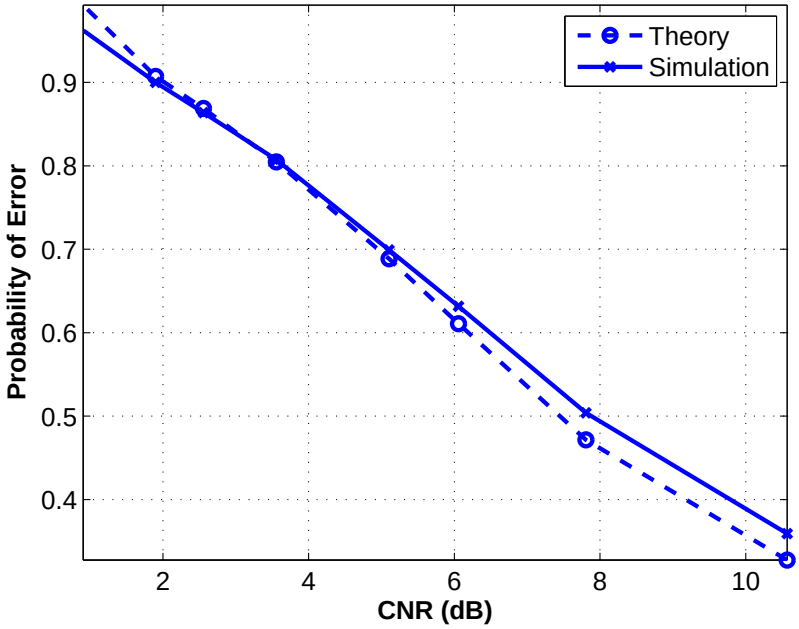

Figure 4.17: Comparing the simulation results with theoretical results for one point target case using Algorithm 1 for 90 degree azimuth.

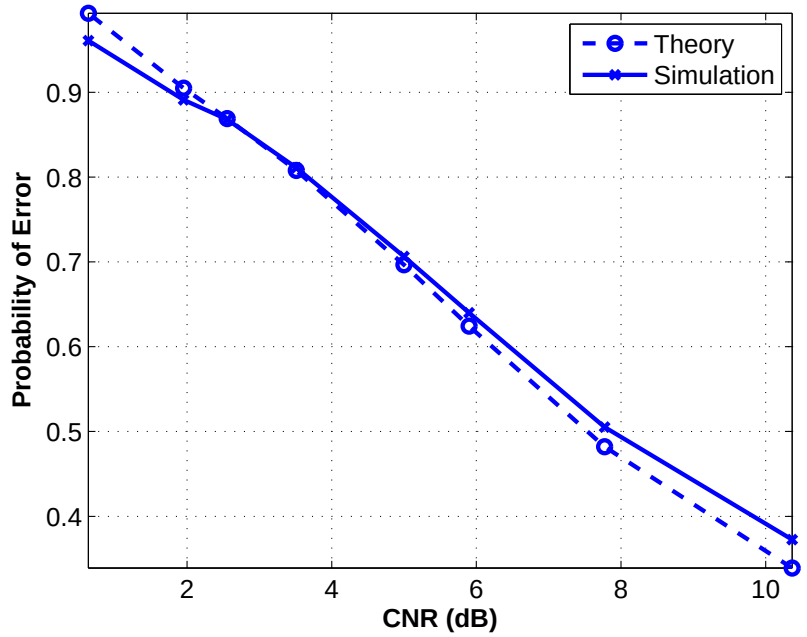

Figure 4.18: Comparing the simulation results with theoretical results for one point target case using Algorithm 1 for 70 degree azimuth.

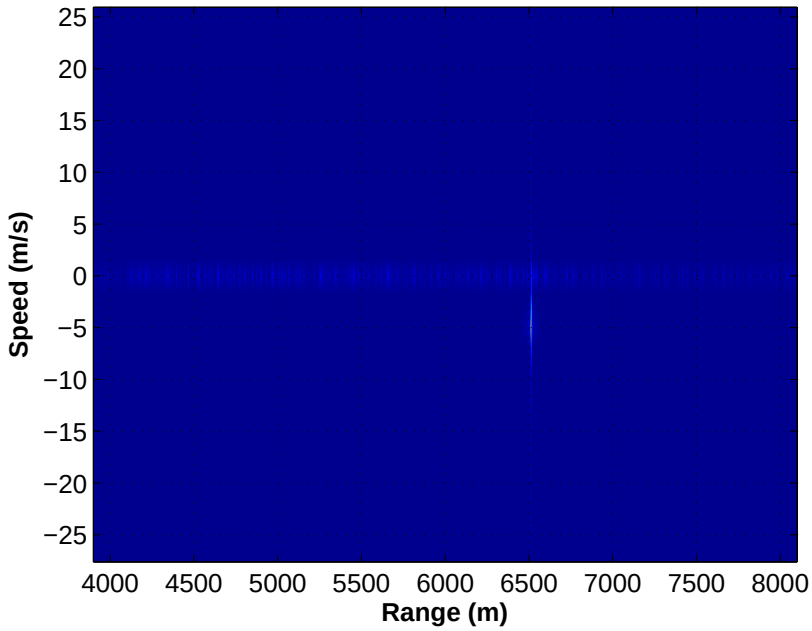

Figure 4.19: Pure detection signal with a realistic target for 90 degree azimuth.

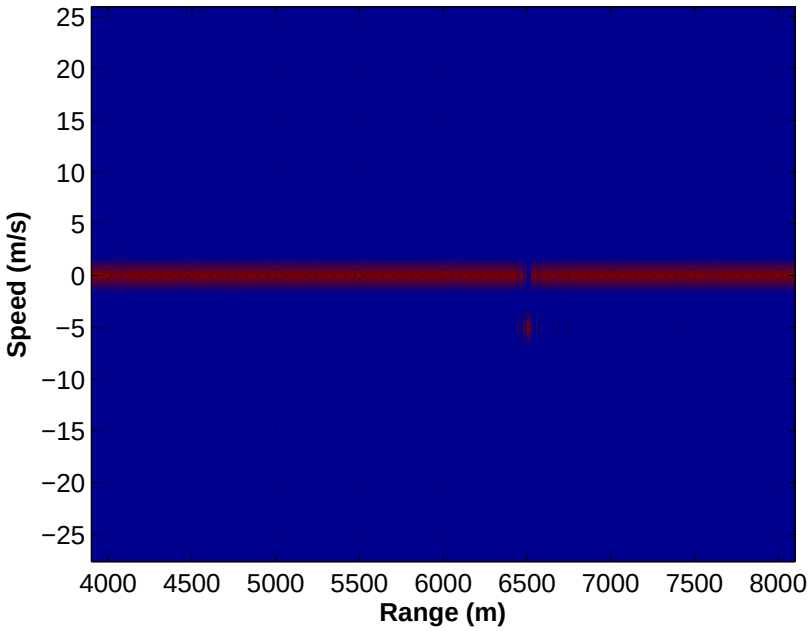

Figure 4.20: Detected clutter for 90 degree azimuth using Algorithm 1.

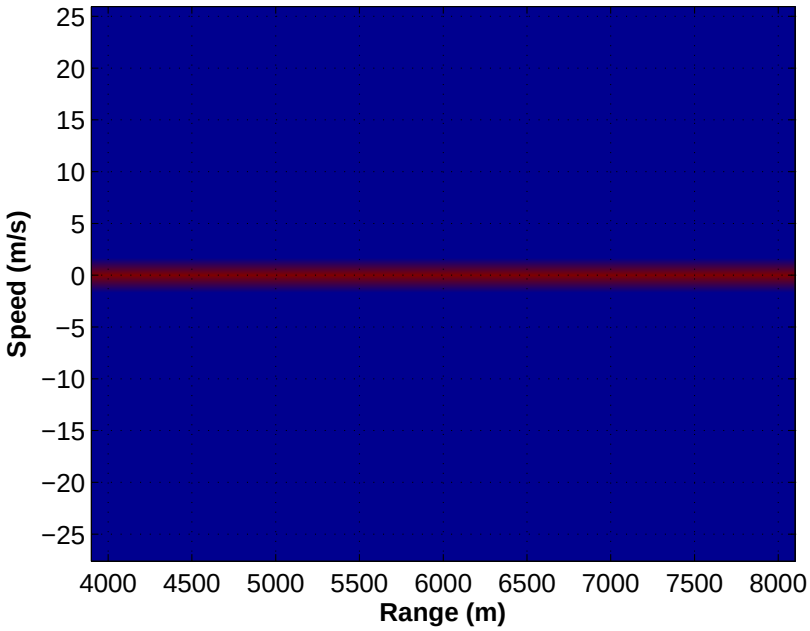

Figure 4.21: Detected clutter for 90 degree azimuth using Algorithm 2.

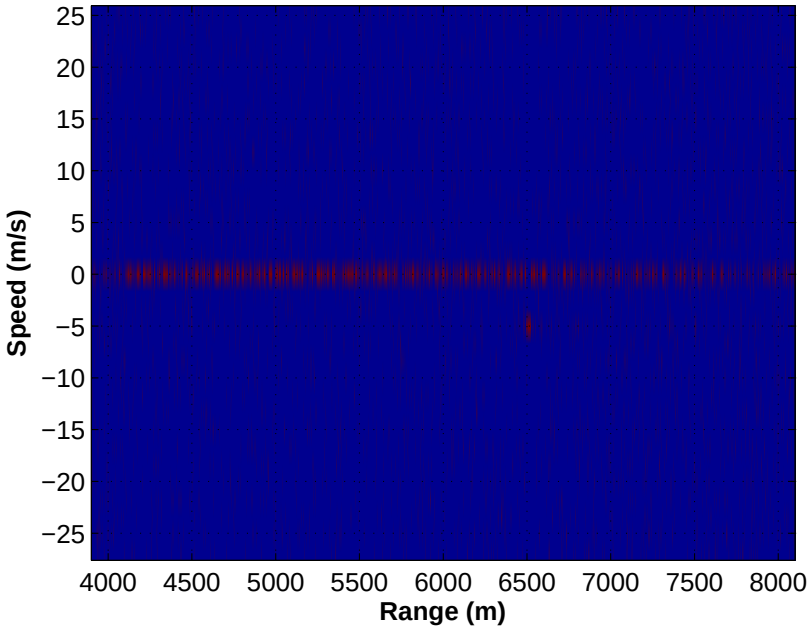

Figure 4.22: Detected clutter from noisy environment for 90 degree azimuth using Algorithm 1.

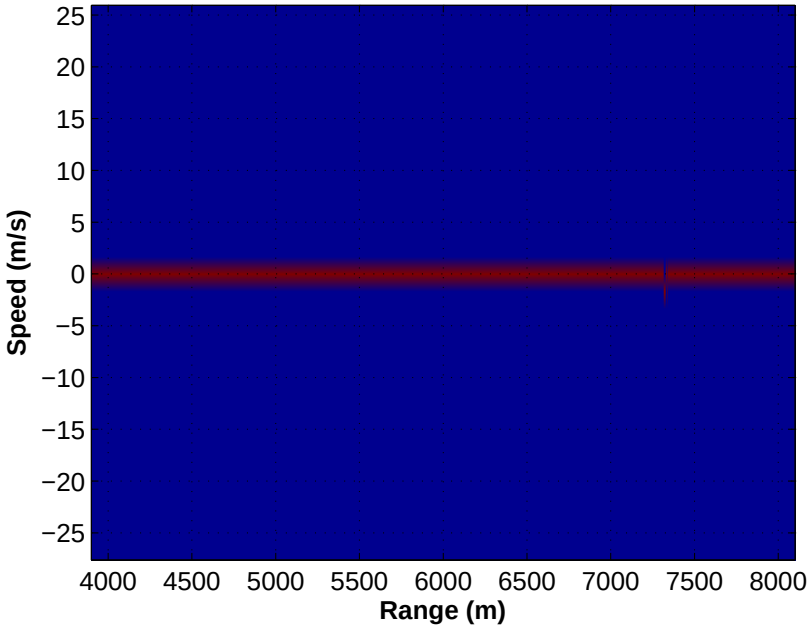

Figure 4.23: Detected clutter from noisy environment for 90 degree azimuth using Algorithm 2.

Chapter 5

CONCLUSIONS AND FUTURE WORK

In this thesis, various algorithms have been investigated for clutter detection in pulse-Doppler radar systems. The main idea behind the algorithms is that since the clutter generally has higher amplitude compared to noise values in a range-Doppler matrix, it can be used to perform precise detection of clutter. Since clutter detection is a real-time military process, it is vital to detect the clutter independent of other parameters as quick as possible with strictly defined algorithms unlike the algorithms proposed in literature. The first algorithm has low computational complexity. This aspect of the algorithm allows quick detection. Algorithm 1 selects the maximum valued cell as the clutter cell in each range in the range-Doppler matrix. When only clutter and noise are present, this algorithm works well. However, if a target is added to the range-Doppler matrix, the algorithm may cause erronous results because targets generally have higher amplitudes than clutter in the range-Doppler matrix. At this point, Algorithm 2 supplies a correction. The outputs of Algorithm 1 is median filtered and improved results can be obtained. However, this correction increases the computational

complexity of the algorithm, which can be lowered by filtering downsampled outputs of the Algorithm 1 and interpolating the obtained results. Both algorithms are designed for the clutter that is present in only one Doppler cell in each range. If the clutter is present in more than one Doppler cell, the algorithms should be modified accordingly. In this manner, prior information about the spread of the clutter must be learned beforehand based on some realistic simulations, consistent with radar system parameters.

The probabilistic analysis of the algorithms has been performed, which allows the user to investigate the probability of error of the algorithms for some desired cases, and closed form expressions for the probability of error have been obtained for some cases. In addition, several simulations have been conducted to show that Algorithm 2 has better performance than Algorithm 1. However, since the computational complexity of Algorithm 1 is lower, it can be an appropriate choice in applications with low CNR values because the performances of the algorithms are very close to each other in this region. On the other hand, Algorithm 2 provides a more complex detection but its performance is better than Algorithm 1 in all CNR values. Moreover, if the length of the median filter used in Algorithm 2 is increased, better results are obtained.

A model for prior information about the Doppler spread of the clutter is needed for the algorithms to be modified according to the prior information. Future work will focus on the generalization of the algorithms. Namely, the algorithms will be modified such that they work with unknown Doppler spreads of the clutter. The performance analysis for the generic algorithms will be performed. Furthermore, the speed of Algorithm 2 can be increased by downsampling the input samples and interpolating the obtained results. The performance of Algorithm 2 with lower complexity will be analyzed.

Bibliography

[1] O. Tan, "Terrain profile estimation over a synthetic terrain by using pulse-doppler radar," Master's thesis, Bilkent University, 2010.

[2] "T-62 Main Battle Tank." `http://www.inetres.com/gp/military/cv/tank/T-62.html`.

[3] G. H. Goldman, "Point-scatterer model for a Soviet T-62 tank at 95 GHz," *Army Research Laboratory*, April 1995.

[4] D. R. Frieden, *Principles of Naval Weapons Systems*. Annapolis, Maryland: Naval Institute Press, 1985.

[5] M. I. Skolnik, *Introduction to Radar Systems*. New York: McGraw-Hill, 1980.

[6] M. I. Skolnik, *Radar Handbook*. New York: McGraw-Hill, 1970.

[7] C. D. Corse and W. R. Barnett, *Introduction to Shipboard Engineering and Naval Weapons*. Annapolis: U.S. Naval Academy, 1971.

[8] G. E. Leydorf, *Naval Electronic Systems*. Annapolis: U.S. Naval Academy, 1971.

[9] G. W. Stimson, *Introduction to Airborne Radar*. SciTech Publishing, 1998.

[10] L. Goetz and J. Albright, "Airborne pulse-doppler radar," *IRE Transactions on Military Electronics*, pp. 116–126, April 1961.

[11] U. DiNardo and G. Nucci, "Synthesis criteria of a signal processor for pulse doppler radars," *Selenia Technical Review*, September 1972.

[12] R. O'Donnel et al., "Advanced signal processing for airport surveillance radars," *IEEE Eascon Record*, pp. 71–71F, 1974.

[13] R. Daly and J. Glass, "Digital signal processing for radar," *Electronic Progress*, pp. 24–30, Spring 1975.

[14] R. Mooney and G. Ralston, "Performance in clutter of airborne pulse mti, cw doppler and pulse doppler radar," *IRE Convention Record*, pp. 55–62, 1961.

[15] E. Pearson et al., *Method of Tracking Target in Presence of Clutter.* US Patent 4559537, 1985.

[16] S. Clancy et al., *Method and System for Tracking Targets in a Pulse Doppler Radar System.* US Patent 4450446, 1984.

[17] F. Tony and A. Luminita, "Active contours without edges," *IEEE Transactions On Image Processing*, pp. 266–277, February 2001.

[18] E. Ma, "Simulating nature," *Project A*, June 2002.

[19] G. Macklem, "Computer landscape generation and smoothing," *Masters Comprehensive Exam*, 2003.

[20] H. Soğancı, "Doppler frequency estimation in pulse doppler radar systems," Master's thesis, Bilkent University, 2009.

[21] M. A. Richards, *Fundamentals of Radar Signal Processing.* New York: McGraw-Hill, 2005.

[22] C. E. Cook and M. Bernfeld, *Radar Signals An Introduction to Theory and Application.* London: Artech House, 1993.

[23] N. Levanon and E. Mozeson, *Radar Signals.* New York: Wiley, 2004.

[24] T. W. Jeffrey, *Phased-Array Radar Design*. Raleigh, NC: SciTech Publishing, 2009.

[25] D. K. Barton, *Radar System Analysis and Modeling*. Norwood, MA: Artech House, 2005.

[26] G. Morris and L. Harkness, *Airborne Pulsed Doppler Radar*. Boston: Artech House.

[27] N. Levanon, *Radar Principles*. New York: Wiley, 1988.

[28] P. Z. Peebles, *Radar Principles*. New York: Wiley, 1998.

[29] M. Bulmer, *Principles of Statistics*. Dover, 1979.

[30] M. Jones and J. Marron, "A brief survey of bandwidth selection for density estimation," *Journal of the American Statistical Association*, pp. 401–407, Mar. 1996.

[31] S. Kullback and R. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, pp. 79–86, 1951.