Efficient Grid-Less Target Detection Algorithms

Master's thesis in Systems, Control and Mechatronics Programme

USAMA ABDULLAH
ZUBIA MUMTAZ

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Efficient Grid-Less Target Detection Algorithms

USAMA ABDULLAH
ZUBIA MUMTAZ

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Signal Processing and Biomedical Engineering
Chalmers University of Technology
Gothenburg, Sweden 2024

Efficient Grid-Less Target Detection Algorithms
USAMA ABDULLAH
ZUBIA MUMTAZ

© USAMA ABDULLAH, ZUBIA MUMTAZ, 2024.

Supervisor: Tomas McKelvey, Electrical Engineering
Examiner: Tomas McKelvey, Electrical Engineering

Master's Thesis 2024
Department of Electrical Engineering
Division of Signal Processing and Biomedical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Efficient Grid-Less Target Detection Algorithms
Usama Abdullah
Zubia Mumtaz
Department of Electrical Engineering
Chalmers University of Technology

Abstract

In this thesis, performance metrics are established to evaluate the efficiency and accuracy of various grid-less multiple target detection algorithms in radar. A high-fidelity radar simulation environment is established with a liberty to change parameters affecting the radar base-band frequency estimation. The radar data is generated using a base-band sinusoidal signal model, in which clutter and noise can be added or removed. Different settings of data generation are used in order to replicate multiple realistic scenarios. The detection algorithms are rigorously tested under these scenarios including high noise and dense clutter environment. Special cases when two target frequencies are very close to each other, are also monitored. The errors are recorded for various noise levels, signal lengths and clutter spread and strengths. A mapping metric i.e. GOSPA, is used to correlate estimated and actual parameters in order to perform an effective evaluation. The aim of this work is to provide a detailed evaluation platform for detection algorithms. Moreover, different clutter removal techniques are also introduced for the detection algorithms.

Keywords: radar, target detection, grid-less detection

Acknowledgements

We would like to express our deepest gratitude and appreciation to our supervisor, Tomas McKelvey, for the exceptional guidance, unwavering support and outstanding feedback throughout this journey.
We would like to mention our kids, Amina and Ali, for being constant source of motivation for us. A special thanks to our parents for their perpetual prayers and moral support.

Usama Abdullah, Zubia Mumtaz, Gothenburg, February 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AUC	Area Under Curve
CFAR	Constant False Alarm Rate
CRLB	Cramer Rao Lower Bound
DFT	Discrete Fourier Transform
ESPRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
FFT	Fast Fourier Transform
fs-ESPRIT	frequency selective ESPRIT
GOSPA	Generalized optimal sub-pattern assignment metric
LMMSE	Linear Minimum Mean Square Error
LRT	Likelihood Ratio Test
MC	Monte Carlo
OMP	Orthogonal Matching Pursuit
PDF	Probability Density Function
RMS	Root Mean Square
ROC	Receiver Operating Characteristic
SNR	Signal to Noise Ratio
SVD	Singular Value Decomposition

Contents

List of Figures

List of Tables

1
Introduction

Target detection in a radar system is the primary task to determine if an object is present at a certain location in space. A classical approach to solve this problem is to test if a target is present in any of a set of discrete hypothetical target locations. If the volume to be searched for targets is large and the grid on which the search is performed over is dense, the computational complexity becomes a practical problem to be feasible for real-time operation. In some radar configurations, it is possible to perform so called grid-less detection where the location of a possible target is determined by solving an associated estimation problem. In contrast to grid-based detection, grid-less detection requires more advanced signal processing algorithms. In a radar scenario where the sensing array senses signals from multiple targets returns and ground clutter, the computational complexity of the grid-less detection may also increases. In order to reduce this complexity, beamspace transformation can be done which divides the input into multiple beams and data from each beam is processed in parallel.

1.1 Aim

The aim of the project is to investigate various grid-less estimation algorithms that can be applied to beamspace transformed data, to determine how they can be adopted to such data, and characterize the performance of the overall methodology. To do the performance evaluation, a high-fidelity radar simulation environment is established. In the end, it should be possible to demonstrate accuracy of multiple estimation algorithms.

1.2 Thesis Outline

Chapter 1 gives the introduction to the topic and explains the aim of the project. Chapter 2 describes the theoretical background of radar target detection and estimation, grid-less detection techniques and various performance metrics that can be used to evaluate these algorithms. It also illuminates the theory behind some other algorithms used in this project like GOSPA metric etc. The theoretical background of signal model and clutter is also presented in this chapter. Chapter 3 describes the methodology adopted to carry out various tasks in the project. It also gives a brief description on the estimation algorithms used in the simulation. Chapter 4 consists of all simulation results for various test cases. Moreover, these results are presented along with detailed discussions in which investigations have been made

behind results and how they can be improved. The final chapter concludes the thesis and briefly describes the outcome of the thesis.

2

Theory

This chapter gives a brief introduction to radar theory, the multiple target detection problem, multiple target estimation algorithms and metrics to evaluate their performance.

2.1 Basic Theory of Radar detection

Radar stands for *Radio Detection and Ranging*. It is a device which can be used to detect location, distance and speed of the object in its vicinity. The working principle of Radar is based on transmission of electromagnetic waves into space and to detect objects from the reflected signals. Typically a radar system consists of a transmitter and an antenna which radiate electromagnetic waves, and a receiver with antenna to collect the reflected waves. A radar could use same or different antennas for transmission and reception depending on system design. But nonetheless, a radar typically does not receive signals while it is transmitting. Depending upon the complexity of system, a radar may use array of antennas. A basic block diagram of radar is shown in figure 2.1.

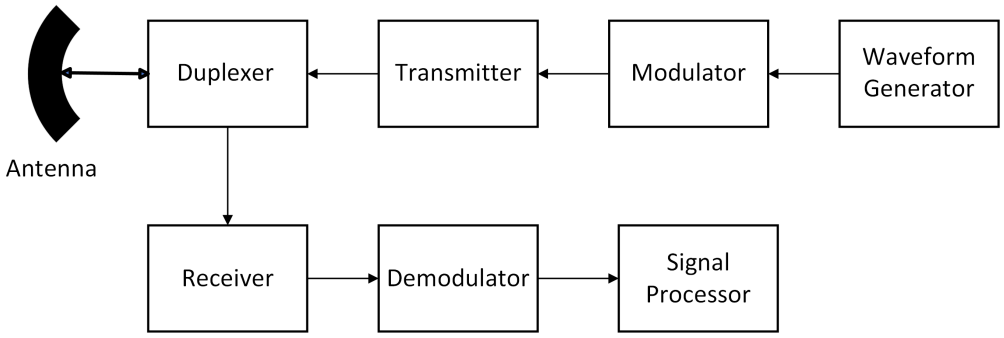


Figure 2.1: Basic Block Diagram of Radar

- The waveform generator generates the complex transmission signal. The pulse duration of transmitted pulse is controlled here.
- The Modulator modulates the waveform pulse to carrier frequency, which is ranges from 5MHz to 130GHz in modern radars.
- The transmitter combined with antenna radiates the waves. Usually a high power amplifier is needed in order to transmit such high energy waves.
- The receiver collects the reflected waves from targets.

- The demodulator demodulates the received signal back to base-band frequency i.e, frequencies closer to zero.
- In Signal Processor, matched filtering is applied to received signal in order to maximize it's signal-to-noise ratio (SNR). Subsequently, detection and estimation algorithms are applied to detect if there is any potential target in the echoe and different techniques can be applied to estimate its range and Doppler.

One of the major challenges in radar signal processing is to determine the presence of valid target with certainty. As the signal received by the signal processor, has been through the whole chain of radar system thus deteriorating it at each step. Moreover, it is also corrupted by interference from the environment, as well as the Tx/Rx chain itself. The radar detector needs to determine if the signal contains any useful target information or is it just noise. This problem can be formulated as a *binary hypothesis test* [1].

$$\begin{cases} H_0: & \text{Interference only} \\ H_1: & \text{target and interference} \end{cases} \tag{2.1}$$

where H_0 is called *Null Hypothesis* and H_1 is called *Alternative Hypothesis*

2.1.1 Signal Model

Depending upon the type of radar, different types of information can be extracted from the received signal. In Pulse-Doppler radar, the most common information that is usually extracted from received echoes is of the target's:

- Range: How far is the target from Radar centre?
- Doppler: At what speed, the target is approaching towards or moving away from radar?
- Azimuth: Direction to the target. It is also called angle of arrival or direction of arrival (DOA).

After some initial pre-processing of data, for a single point target present at range n with Doppler k and on azimuth i, the signal is a product of sinusoids of frequencies corresponding to target's range, Doppler and azimuth.

$$\boldsymbol{x}(n,k,i) = \sum_{t=1}^{T} \alpha_t e^{j2\pi f_r n} e^{j2\pi f_d k} e^{j2\pi f_a i} \tag{2.2}$$

α_t = amplitude of target
f_r = frequency corresponding to range
f_d = frequency corresponding to Doppler
f_i = frequency corresponding to azimuth
T = Total number of targets

FFT can be taken in each dimension to generate a three dimensional data cube as shown in figure 2.2 [13].

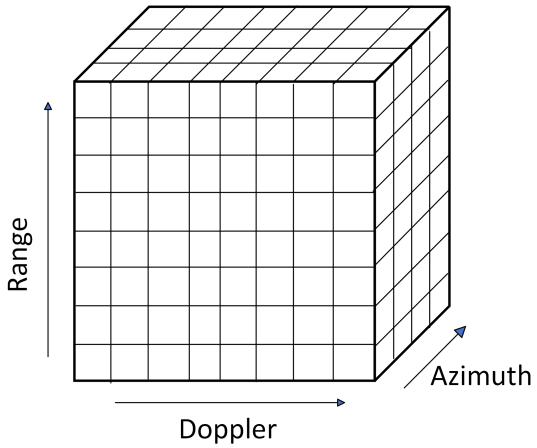


Figure 2.2: 3-D data cube

For the sake of simplicity and due to repetitive nature of these cube dimensions, this thesis assumes data in only one dimension.

2.1.2 Noise Model

As mentioned earlier, it is inevitable for received signal to have some interference and noise. There are several noise generating sources in a radar system. Some of these are listed below:

- Thermal noise is generated by the electronic components in radar processing chain, caused by agitation of electrons. It is usually the receiver which contributes to thermal noise.
- Atmospheric noise is caused by natural phenomenon such as lightning discharges and cosmic radiations etc.
- External interference such as electromagnetic interference either caused by nearby devices or intentionally caused by hostile jammers.
- Clutter is unwanted echoes from buildings, vegetation, sea and terrain etc.

All of these contributing factors can be modelled and compensated in radar simulation, if required. However in this thesis, only additive zero mean complex white Gaussian noise is considered. Moreover, few cases with ground clutter are also discussed.
The aforementioned complex white Gaussian noise has a PDF of the form [14]:

$$p(\tilde{x}) = \frac{1}{\pi\sigma^2} \exp[-\frac{1}{\sigma^2}|\tilde{x} - \tilde{\mu}|^2] \tag{2.3}$$

and is denoted by $CN(\tilde{\mu}, \sigma^2)$ where $\tilde{\mu}$ is mean and σ^2 is the variance of noise. For zero mean noise, $\tilde{\mu} = 0$.

2.1.2.1 Signal-to-Noise Ratio

Signal-to-Noise Ratio (SNR) is a fundamental metric used to quantify the quality of a signal relative to the level of noise present in a system. It characterizes the ratio of the power or amplitude of the desired signal to the power or amplitude

of the unwanted noise. SNR is often expressed in logarithmic form using decibels (dB), which provides a more convenient and meaningful representation of the ratio. Mathematically, SNR is defined as:

$$\mathrm{SNR(dB)} = 10 \log_{10} \frac{P_{sig}}{P_{noise}} \tag{2.4}$$

where P_{sig} is power of signal and P_{noise} is power of noise.
A higher SNR means that a signal is stronger and more distinguishable from the noise. A lower SNR indicates that signal is buried in noise and more difficult to detect, which may lead to degradation of accuracy and reliability of system.

2.2 Multiple target detection in digital Radar

The multiple target detection problem in a digital radar can be illustrated by the figure 2.3. Targets can be in any direction and a grid-less method will estimate the locations of all targets present. The computational complexity for the grid-less detection problem may be too large for real-time operation due to many simultaneous targets and clutter returns.

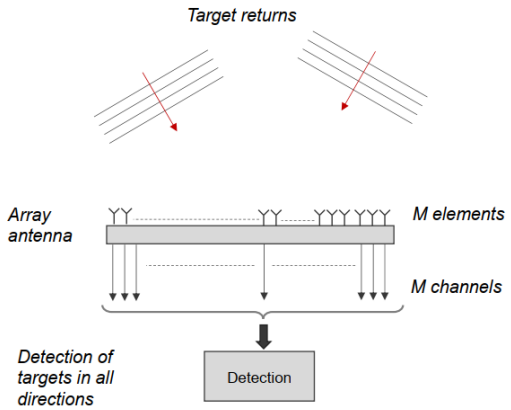


Figure 2.3: Detection without beam forming

In the figure 2.4, a potential solution to reduce the complexity is illustrated. In a first step, the signal from the array is transformed by a beamspace transformation into N virtual beams and the data from each beam can be processed in parallel. Naturally, the number of potential targets in each beam is now drastically reduced and it is possible to employ a grid-less detection method in each beam at a reduced computational cost.

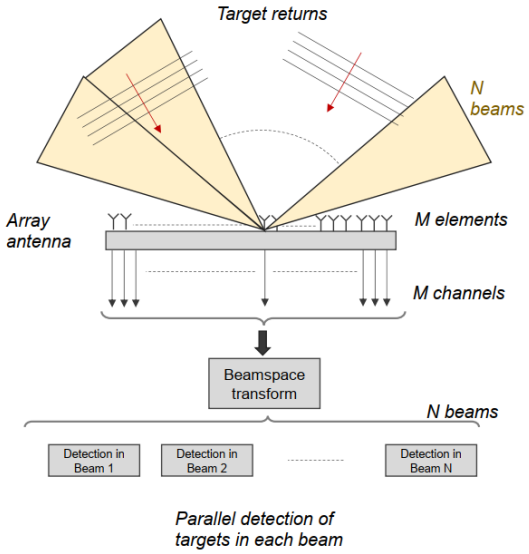


Figure 2.4: Detection with beam forming

2.3 Target Detection and Estimation

The significance of radar target detection and estimation algorithms can be understood through their ability to extract relevant information from radar signals. These algorithms employ sophisticated mathematical and statistical techniques to distinguish targets from clutter, noise, and environmental interference. By effectively isolating the desired signals, they enable the precise detection and tracking of objects of interest.

As mentioned earlier, the radar detection problem can be formulated as a *binary hypothesis test* (2.1). A very basic method to determine which hypothesis is correct for a given received signal is *Likelihood Ratio Test* (LRT) [2]. It is a statistical hypothesis testing method that compares two competing hypotheses using the likelihood ratio as the test statistic. A likelihood function for both the *Null hypothesis*, $f_L(L|H_0)$, and *Alternative Hypothesis*, $f_L(L|H_1)$ is established using sufficient statistic under H_0 and H_1. And the LRT can then be expressed as:

$$\frac{f_L(L|H_1)}{f_L(L|H_0)} \gtrless_{H_0}^{H_1} \gamma \tag{2.5}$$

A generalized form of this test is called *Generalized Likelihood Ratio Test* (GLRT) [1] which maximizes both likelihood functions in (2.5), and then determines their ratio.

Over the years, various techniques have been developed to precisely detect the presence of valid target in radar echo and to estimate it's characteristics such as range, doppler, amplitude etc. A mere identification of targets and removal of clutter could

be done using simpler algorithms like Moving Target Indicator (MTI) [3]. One of the classical radar detection algorithm is the Matched Filter [4]. It correlates the received radar signal with a known reference signal (typically the transmitted waveform) to maximize the detection of targets with the same characteristics as the reference signal.

With the advent of advanced technology, it was possible to enable more sophisticated techniques such as the Constant False Alarm Rate (CFAR) [5], which enabled adaptive thresholding to maintain a consistent false alarm rate in the presence of varying clutter levels. This technique revolutionized clutter rejection and improved detection performance in complex environments. Some other techniques like Adaptive Matched Filtering [6] [7] also bear CFAR property. And many variants of CFAR have also been developed. Some examples are cell averaging CFAR (CA-CFAR), order statistic CFAR (OS-CFAR), greatest of CFAR (GO-CFAR) etc.

Even better techniques were developed with the rise of digital signal processing, for example CLEAN [8] [9] and Orthogonal Matching Pursuit (OMP) [10]. They are the grid version of radar detection which first place the received signal on grid (usually FFT grid) and then remove all strong echoes from it in a repetitive manner. With the constant technological development, it is possible to increase the efficacy of these type of techniques by simply increasing the length of the grid.

2.4 Performance Metric for Estimation Algorithms

The development of several performance evaluation metrics to assess the efficiency of radar target estimation algorithms hold a significant value in the realm of radar technology. By thoroughly measuring the algorithmic performance through these metrics, engineers and researchers can effectively enhance radar system design, optimize target detection capabilities, and enhance the overall efficacy and reliability of radar technology in diverse applications such as surveillance, navigation, and defense.
Several metrics have been developed in particularity to determine radar target detection and estimation algorithms. Some of these are briefly discussed below:

2.4.1 Binary Hypothesis test

A binary hypothesis test as mentioned in (2.1), is a statistical procedure used to make a decision between two competing hypotheses, *Null Hypothesis*, H_0 and *Alternative Hypothesis*, H_1 regarding the absence or presence of a target signal in the received radar data respectively. The figure 2.5 (reference: [23]) illustrates Gaussian probability density functions for binary hypothesis test.
The threshold, γ determines which hypothesis will be selected. If the signal power is above this threshold, H_1 is selected and if it's below this value, then H_0 is selected. The PDFs of these two hypothesis is shown in the figure. Two important metrics, probability of detection and probability of false alarm can determine the efficacy of this test. These metrics are defined below:

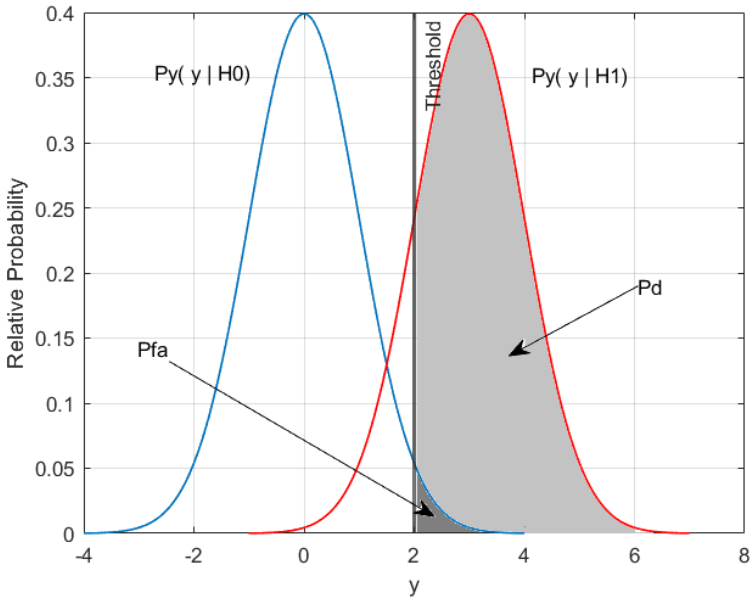


Figure 2.5: PDFs for BHT

- **Probability of Detection**, P_D is a conditional probability that the alternative hypothesis, H_1, is selected if the received signal has both interference and target. This metric quantifies the algorithm's ability to correctly detect and identify targets in the presence of noise and clutter. A higher P_D indicates a better detector.
- **Probability of False Alarm**, P_{FA} is a conditional probability that the alternative hypothesis, H_1 (2.1), is selected if the received signal has just interference and no target. This metric measures the algorithm's tendency to generate false alarms or spurious detection in the absence of a target. A lower P_{FA} indicates a better detector.

It can be seen from figure 2.5, by increasing γ, $P_D \to 0$ and $P_{FA} \to 0$. And by decreasing the threshold, $\gamma \to 0$, $P_D \to 1$ and $P_{FA} \to 1$. Therefore, there is a trade-off between the two and one has to select the threshold carefully.
The detectors with CFAR property are designed to maintain a constant false alarm rate regardless of the changes in noise levels and clutter. With having the guarantee of a consistent false alarm rate, it is easier for the operator in decision-making.

2.4.2 Receiver Operating Characteristic, ROC curves

Another metric that can be used is Receiver Operating Characteristic curves. ROC curves provide a graphical representation of the trade-off between P_D and P_{FA} for varying Signal-to-Noise ratios. They illustrate the algorithm's performance across a range of operating conditions, allowing for visual comparison and selection of algorithms based on desired performance characteristics. An example of ROC curve graph is shown in figure 2.6.

As mentioned earlier, it is desirable to have a higher P_D and lower P_{FA}, but it can

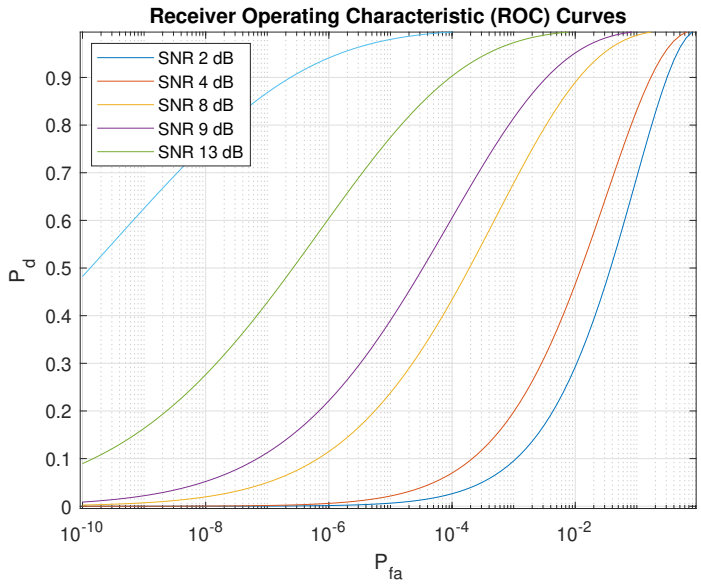


Figure 2.6: MATLAB generated ROC curves for different values of SNR

be seen in the figure that for lower values of SNR, this is not possible. A radar equivalent SNR can be increased by transmitting more power, increasing antenna gain, optimizing receiver sensitivity, using coherent pulse integration and increasing the number of pulses in one coherent pulse interval.

2.4.3 AUC: Area Under Curve

The AUC in radar represents the area under the ROC curve. The AUC value ranges between 0 and 1, where a higher AUC indicates better discrimination capability and overall performance of the radar system. A radar system with an AUC value close to 1 suggests a high P_D and a low P_{FA} across a range of threshold values. This indicates that the radar system is effective in detecting targets while maintaining a low false alarm rate. A detector with AUC equal to 1 is considered as a perfect detector.
The performance metrics discussed so far are the classical methods which give a very deep insight of a detector efficiency, but they can only be used for a single target. In case of multiple targets, some enhanced metrics are used to sum up the overall performance of a detector in the presence of interference, clutter and several targets. Some of these are mentioned below:

2.4.4 Confusion Matrix

Yet another way to measure the accuracy of radar detector, Confusion Matrix gives a tabular representation of the relationships between predicted and actual target states. The confusion matrix consists of four distinct elements: true positive rate (TPR), false positive rate (FPR), true negative rate (TNR), and false negative rate (FNR). The true positives signify the cases in which the radar system correctly identifies the presence of a target when one actually exists. The false positives occur

when the radar system erroneously indicates the presence of a target when there is none, leading to false alarms. The true negatives denote the instances in which the radar system correctly determines the absence of a target when there is no target present. The false negatives refer to the situations where the radar system fails to detect an actual target, leading to a missed detection. By organizing detection outcomes into this matrix, the confusion matrix provides a detailed assessment of the performance of radar detection algorithms. The sum of each column in this matrix equals one. Also, the TPR is equal to the Probability of Detection, P_D and FPR equals Probability of False Alarm, P_{FA}, mentioned in section 2.4.1. A typical confusion matrix is represented as below:

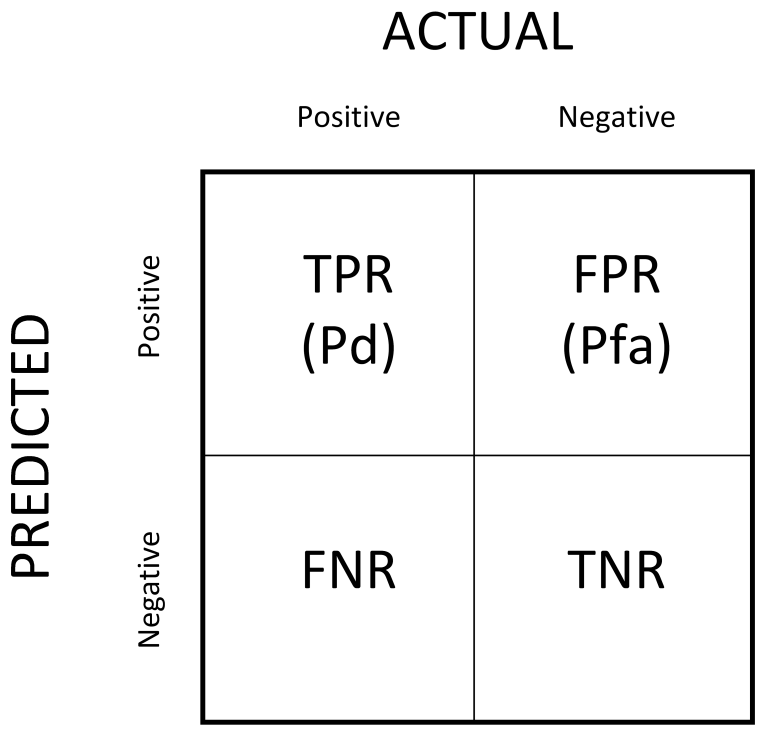


Figure 2.7: Confusion Matrix Representation

2.4.5 False Alarm Rate

False Alarm Rate, FAR measures the rate at which false alarms occur per unit time or per scan (360 degree span of radar antenna). It indicates the frequency of incorrect target detection when there are no actual targets present. Target detection algorithms that have CFAR property ensure to maintain a constant false alarm rate.

2.4.6 RMS Errors

The root mean squared error (RMSE) is a statistical metric that provides a comprehensive assessment of the overall accuracy and goodness of fit between observed and predicted values of various target parameters. The RMSE is defined as:

$$\text{RMSE}_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\|x_i - \hat{x}_i\|^2} \tag{2.6}$$

where:

x_i = actual value
$\hat{x}_i$ = estimated value
n = length of estimated parameter vector

Depending upon which RMSE is desired to be computed, n could be number of targets or number of Monte Carlo iterations. The state x can be detected Doppler, amplitude, range or azimuth of target. Thus, RMSE provides a wide range of possibilities in which detectors can be evaluated. This is the reason why this thesis also utilizes this performance metric to evaluate several estimation algorithms for multiple targets.

2.5 Target parameter Mapping

The outcome of the multiple target estimation algorithms isn't necessarily in one to one correspondence with the ground truth. In most of the cases the number of target estimates could also be different from actual number of targets present in data. In order to evaluate these algorithms correctly, it is essential to have one to one mapping between ground truth and target estimates. One of the techniques that can be used for this purpose is the GOSPA metric.
The Generalized optimal sub-pattern assignment (GOSPA) metric is the sum of localization errors for properly detected frequencies and a penalty for false and missed frequencies. The implementation of this metric is done according the Definition 1 mention in [29].

$$d_p^{(c,\alpha)}(X,Y) \triangleq \left(\min_{\pi\in\Pi_{|Y|}} \sum_{i=1}^{|X|} d^{(c)}(x_i, y_{\pi_i})^p + \frac{c^p}{\alpha}(|Y| - |X|)\right)^{\frac{1}{p}}, \text{ for } |X| \leq |Y| \tag{2.7}$$

where:

c = cut-off metric and $c > 0$
α = parameter of GOSPA metric and it is specified in the range [0,2]
p = switching penalty and $1 \leq p \leq \infty$
$d(x,y)$ = metric for any $x, y \in \mathbb{R}^N$
$d^{(c)}(x,y)$ = $\min(d(x,y), c)$ is its cut-off metric
Π_n = set of all permutations of $\{1, ...n\}$ for any $n \in N$
π = Any element $\pi \in \Pi_n$ is a sequence $(\pi(1)...\pi(n))$
X = $\{x_1, ...x_{|X|}\}$, a finite subset of $\mathbb{R}^N$
Y = $\{y_1, ...y_{|Y|}\}$, a finite subset of $\mathbb{R}^N$

The parameter c determines the maximum allowable localization error and, along with parameter α, it also determines the error due to cardinality mismatch ($|Y| - |X|$). The parameter p is used for penalty for false targets. The larger the value of p, the more the outliers are penalized.

2.6 The Cramer Rao Lower Bound

CRLB is a lower bound on the variance of any unbiased estimator. CRLB can be used to create a benchmark against which all other estimators can be measured. This can give us insight on which estimator to choose from many.

Let $\hat{\boldsymbol{\theta}}$ be an unbiased estimate of $\boldsymbol{\theta}$ and let $\boldsymbol{P}$ be covariance matrix of $\hat{\boldsymbol{\theta}}$ [30]:

$$\boldsymbol{P} = E\{(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T\} \tag{2.8}$$

Then, there is a matrix called Cramer Rao Bound Matrix, $\boldsymbol{P_{cr}}$, such that [31]:

$$\boldsymbol{P} \geq \boldsymbol{P_{cr}} \tag{2.9}$$

and that the difference $(\boldsymbol{P} - \boldsymbol{P_{cr}})$ is a positive semi-definite matrix. A good starting point to find CRLB is to calculate the Fisher Information Matrix, which always exist. And if the inverse of Fisher Information Matrix, F, exists then:

$$\boldsymbol{P_{cr}} = \boldsymbol{F}^{-1} \tag{2.10}$$

The focus is to estimate frequency and complex amplitude of target, $\boldsymbol{\theta}$ is complex. Therefore the discussion is restricted to complex data vectors that have complex Gaussian PDF of the form [33]:

$$p(\boldsymbol{x}; \boldsymbol{\theta}) = \frac{1}{\pi^N \det(\boldsymbol{C}_x(\boldsymbol{\theta}))} \exp[-(\boldsymbol{x} - \boldsymbol{\mu}(\boldsymbol{\theta}))^H \boldsymbol{C}_x^{-1}(\theta)(\boldsymbol{x} - \boldsymbol{\mu}(\boldsymbol{\theta}))] \tag{2.11}$$

where parameter vector $\boldsymbol{\theta}$ is to be estimated based on complex data $\boldsymbol{x}$. Here $\boldsymbol{\mu}$ is the complex mean and $\boldsymbol{C}$ is the covariance matrix of $\boldsymbol{x}$. Both are dependent on parameter vector $\boldsymbol{\theta}$.
Since $\boldsymbol{\theta}$ may have both real and complex components, therefore the vector of real parameters is denoted as $\boldsymbol{\xi}$. So now if we want to estimate a complex amplitude, a, the vector $\boldsymbol{\xi} = [a_r \quad a_i]^T$.
The Fisher Information Matrix is given by [31]:

$$[\boldsymbol{F}(\boldsymbol{\xi})]_{ij} = \operatorname{tr}\left[\boldsymbol{C}_x^{-1}(\boldsymbol{\xi})\frac{\partial \boldsymbol{C}_x(\boldsymbol{\xi})}{\partial \xi_i}\boldsymbol{C}_x^{-1}(\boldsymbol{\xi})\frac{\partial \boldsymbol{C}_x(\boldsymbol{\xi})}{\partial \xi_j}\right] + 2\operatorname{Re}\left[\frac{\partial \boldsymbol{\mu}^H(\boldsymbol{\xi})}{\partial \xi_i}\boldsymbol{C}_x^{-1}(\boldsymbol{\xi})\frac{\partial \boldsymbol{\mu}(\boldsymbol{\xi})}{\partial \xi_j}\right] \tag{2.12}$$

The CRLB Matrix can then be calculated using equation 2.10.

2.7 Clutter

Clutter in radar systems can be described as the compound effect of undesirable returns originating from terrain, buildings, atmospheric conditions, electromagnetic interference, and other non-target objects present within the radar's field of view. Clutter usually covers a wide region in returned signals, much greater than a radar range resolution cell. In addition to that a *point clutter* may also be present, usually

a return from a tower, pole or a bird. The echoes from land or sea are called *surface clutter*, and the echoes from rain and other atmospheric phenomenon such as lightning, wind, cosmic radiations etc, and chaffs are know as *volume clutter*. These type of clutter can have a significant Doppler content. For airborne radars, dealing with clutter could be even more complicateD as the clutter could produce a lot of Doppler. This makes it difficult to reject clutter on basis of near-to-zero Doppler criteria.

Clutter poses a significant challenge in radar signal processing and target detection tasks. The complex nature of clutter originates from its variability in intensity, spatial distribution and frequency distribution, making it difficult to distinguish from genuine target echoes. Sophisticated techniques are employed to mitigate clutter effects in radar systems. These techniques involve the use of advanced signal processing algorithms, adaptive filtering, statistical models, and clutter maps derived from historical data. By modeling and analyzing the statistical properties of clutter, sophisticated algorithms can discriminate between clutter returns and actual target echoes, allowing for more accurate and reliable target detection and tracking.

3
Methods

This chapter includes the process in which the simulation was prepared and different tests were carried out on various estimation algorithms. It also describes the evaluation methodology adopted to categorize these algorithms in different scenarios. Some additional simulator features e.g. noise threshold, addition and removal of clutter, and usage of GOSPA and Cramer Rao Lower Bound are also elucidated here.

3.1 Data Generation

The signal length, N and number of targets, K are user configurable. The data is generated according to following equation:

$$\begin{aligned}\boldsymbol{x} &= \sum_{k=1}^{K} \alpha_k \begin{bmatrix} e^{j2\pi f_k 0} \\ e^{j2\pi f_k 1} \\ \vdots \\ e^{j2\pi f_k (N-1)} \end{bmatrix} \\ &= \alpha_1 \begin{bmatrix} e^{j2\pi f_1 0} \\ e^{j2\pi f_1 1} \\ \vdots \\ e^{j2\pi f_1 (N-1)} \end{bmatrix} + \alpha_2 \begin{bmatrix} e^{j2\pi f_2 0} \\ e^{j2\pi f_2 1} \\ \vdots \\ e^{j2\pi f_2 (N-1)} \end{bmatrix} \cdots \alpha_K \begin{bmatrix} e^{j2\pi f_K 0} \\ e^{j2\pi f_K 1} \\ \vdots \\ e^{j2\pi f_K (N-1)} \end{bmatrix}\end{aligned} \tag{3.1}$$

where:

K = number of total targets
f_k = normalized frequency of each target, range [0,1)
α_k = complex amplitude of each target

3.1.1 Monte Carlo Simulation

The Complex White Gaussian Noise $\boldsymbol{v}$ is added to the generated targets data $\boldsymbol{x}$. Monte Carlo simulations are performed to produce sample based statistics on the performance of the estimation algorithms. The number of Monte Carlo iterations is user-configurable and in every Monte Carlo iteration noise is added to data:

$$\begin{aligned}\boldsymbol{x} &= \boldsymbol{x} + \boldsymbol{v} \\ \boldsymbol{v} &\sim \mathrm{CN}(0, \sigma_v^2 \boldsymbol{I})\end{aligned}$$

where:

σ_v^2 = noise variance
$\boldsymbol{v}$ = vector of length N

3.1.2 Formulation of Data Scenarios

The estimation algorithms are evaluated in both ideal and non-ideal situations. Therefore, different scenarios are generated such as with fewer and more targets, well-apart and neighboring target frequencies, cluttered and clutter-free environment and different noise levels and signal lengths. These scenarios are explained in detail below:

3.1.2.1 Data Scenario 0 - Single target case

This is the simplest scenario where there is just one target present. This scenario is useful to evaluate the performance of estimation algorithms with respect to the Cramer Rao Lower Bound. (see section 2.6)

3.1.2.2 Data Scenario 1 - Target proximity case

There are two targets in the data where the frequency of one target is fixed whilst the other target is given a frequency sweep in the range Δf. The number of resolution points and total range of Δf can be set in the simulation.
The RMS errors in frequency and amplitudes can then be plotted against Δf. During the frequency sweep, when the two frequencies are equal to each other, singularity occurs. And in order to keep the plots readable, these points are omitted.
The algorithms can be evaluated in ideal situation with zero noise and no clutter and also with change in noise levels and clutter.

3.1.2.3 Data Scenario 2 - Distinct targets

The target frequencies are well apart and fixed. In case of lower noise levels and no clutter, this scenario should yield smaller errors. The table 3.1 shows the example followed in simulator.

Target	Frequency	Amplitude
1	0.1	0.7071 + 0.7071i
2	0.2	0.0000 + 0.7500i
3	0.5	0.3400 + 0.5889i
4	0.6	0.0000 + 0.5200i
5	0.8	0.3182 + 0.3182i

Table 3.1: Target Parameters for Data Scenario 2

This data is tested with and without clutter and for different noise levels and signal lengths.

3.1.2.4 Data Scenario 3 - Closely spaced targets

This is more challenging data scenario where five targets are closely spaced in frequency. This data scenario is used to evaluate the performance of different algorithms in extreme situations. The environment can be further complicated by adding clutter and increasing noise levels. The table 3.2 shows the example target parameters used in simulator.

Target	Frequency	Amplitude
1	0.1	0.7071 + 0.7071i
2	0.11	0.0000 + 0.7500i
3	0.131	0.3400 + 0.5889i
4	0.17	0.0000 + 0.5200i
5	0.2	0.3182 + 0.3182i

Table 3.2: Target Parameters for Data Scenario 3

3.2 Estimation Algorithms

As mentioned in section 2.3 of chapter 2, there are various detection and estimation algorithms used for target detection. In this thesis, two FFT based algorithms CLEAN and Orthogonal Matching Pursuit (OMP) are used to benchmark the performance of advanced ESPRIT and fs-ESPRIT.

3.2.1 CLEAN Algorithm

A well-known classical CLEAN approach sequentially estimates the strongest component in the received signal and then removes it until only white noise is left in the signal [8]. It is also used to remove interference from side lobes [9]. The concept is to clean the signal from dominant targets.

In the implementation of CLEAN, the frequency of each dominant target is estimated by detecting the global maxima on the DFT grid. The index value of the maxima is then converted to normalized estimated frequency, $\hat{f}_k$. Knowing the formation of transmitted signal from (3.1), the estimated frequency, $\hat{f}_k$, can be used to mimic the component of original signal, corresponding to one target. This signal is called steering vector $\boldsymbol{s}$. This leads to a least square minimization problem where we need to estimate the target amplitude, α_k.

$$\min_{\alpha_k} \|\boldsymbol{x} - \boldsymbol{s}\alpha_k\|^2 \tag{3.2}$$

This least square problem can be solved for α_k by taking derivative of (3.2) w.r.t. α_k and equating it to zero. The algorithm to calculate f_k and α_k is described below:

$$\boldsymbol{x_{res}} = \boldsymbol{x} \tag{3.3}$$

while $k < K$:

$$\boldsymbol{x_{fft}} = \text{fft}(\boldsymbol{x_{res}}, N_{FFT}) \tag{3.4}$$

$$\hat{f}_k = \arg\max_{f_k} \left| \frac{\boldsymbol{x_{fft}}}{N} \right| \tag{3.5}$$

$$\boldsymbol{s} = \begin{bmatrix} e^{j2\pi\hat{f}_k 0} \\ e^{j2\pi\hat{f}_k 1} \\ \vdots \\ e^{j2\pi\hat{f}_k (N-1)} \end{bmatrix} \tag{3.6}$$

$$\hat{\alpha_k} = \frac{\boldsymbol{s}^H \boldsymbol{x_{res}}}{\boldsymbol{s}^H \boldsymbol{s}} \tag{3.7}$$

$$\boldsymbol{x_{res}} = \boldsymbol{x_{res}} - \boldsymbol{s}\hat{\alpha_k} \tag{3.8}$$

where:

$\boldsymbol{x}$	= input vector
k	= An integer representing target number with total targets K
$\hat{f}_k$	= estimated frequency of one target
$\boldsymbol{x_{fft}}$	= FFT of the residual vector $\boldsymbol{x_{res}}$
N	= Signal length i.e. length of vector $\boldsymbol{x}$
N_{FFT}	= length of zero padding in FFT
$\hat{\alpha_k}$	= estimated amplitude (complex) of one target.

The equation (3.7) is derived by solving least squares problem of (3.2). In CLEAN algorithm, there can be several ways to set the stopping criteria, for example, in [8], the signal is cleaned from all the strong echoes until just white noise remains. But this thesis follows a rather simpler approach to the said problem. The above mentioned approach can lead to a difference between number of target estimates and number of actual targets. This can generate more complexity when mapping the estimates to ground truth. Therefore, the stopping criteria used in this work is predefined number of given targets, K.
List of all configurable parameters for this algorithm is given in table 3.3.

Parameter	Description	Range
K	Number of Targets	{0,1,2,3,...}
N_{FFT}	FFT zero padding	$\geq N$
f_s	Freq start point	0-1
f_e	Freq end point	0-1

Table 3.3: Hyper-parameters for CLEAN Algorithm

It is also possible to select the frequency range in which the algorithm should find the target peaks. This can be done by setting two other parameters f_s and f_e with range 0-1. The vector $\boldsymbol{x_{fft}}$ will then be truncated as below:

$$\boldsymbol{x_{fft}} = \boldsymbol{x_{fft}}(f_s : f_e) \tag{3.9}$$

3.2.1.1 Effect of Zero Padding on Algorithm Performance

The minimum range for N_{FFT} is the data length, N, and the increase in zero padding highly affects the results in some scenarios. To understand this, we first need to see that in CLEAN technique, we try to maximize the following expression:

$$\max_{\boldsymbol{f}} \left| \boldsymbol{s}^H(\boldsymbol{f})\boldsymbol{x} \right| \tag{3.10}$$

$$\boldsymbol{f} \in \left[0, \frac{1}{N_{FFT}}, \frac{2}{N_{FFT}}, \cdots, \frac{N_{FFT}-1}{N_{FFT}}\right] \tag{3.11}$$

As increasing the zero padding increases the resolution of DFT grid. Hence the distance between the true frequency and grid point decreases. This results in decrease in error.

3.2.2 Orthogonal Matching Pursuit (OMP) Algorithm

The Orthogonal Matching Pursuit (OMP) is an iterative greedy algorithm which recovers the non-zero components from a sparse signal [10]. For target detection, the algorithm is same as CLEAN except that the $\hat{\alpha_k}$ estimates for previous targets are updated when new estimates are calculated. Instead of a vector $\boldsymbol{s}$, the matrix $\boldsymbol{S}$ is now formed which contains the $\boldsymbol{s}$ vectors for the current and all previous frequencies. The detection is repeated in a loop for all targets.

$$\boldsymbol{x_{res}} = \boldsymbol{x} \tag{3.12}$$

while $k < K$:

$$\boldsymbol{x_{fft}} = \text{fft}(\boldsymbol{x_{res}}, N_{FFT}) \tag{3.13}$$

$$\hat{f}_k = \arg\max_{f_k} \left| \frac{\boldsymbol{x_{fft}}}{N} \right| \tag{3.14}$$

$$\boldsymbol{s_k} = \begin{bmatrix} e^{j2\pi\hat{f}_k 0} \\ e^{j2\pi\hat{f}_k 1} \\ \vdots \\ e^{j2\pi\hat{f}_k (N-1)} \end{bmatrix} \tag{3.15}$$

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{s_1} & \boldsymbol{s_2} & \cdots & \boldsymbol{s_k} \end{bmatrix} \tag{3.16}$$

$$\boldsymbol{S} = \begin{bmatrix} e^{j2\pi\hat{f}_1 0} & e^{j2\pi\hat{f}_2 0} & \cdots & e^{j2\pi\hat{f}_k 0} \\ e^{j2\pi\hat{f}_1 1} & e^{j2\pi\hat{f}_2 1} & \cdots & e^{j2\pi\hat{f}_k 1} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j2\pi\hat{f}_1 (N-1)} & e^{j2\pi\hat{f}_2 (N-1)} & \cdots & e^{j2\pi\hat{f}_k (N-1)} \end{bmatrix} \tag{3.17}$$

$$\hat{\boldsymbol{\alpha}}_{\boldsymbol{k}} = (\boldsymbol{S}^H\boldsymbol{S})^{-1}\boldsymbol{S}^H\boldsymbol{x} \tag{3.18}$$

$$\boldsymbol{x_{res}} = \boldsymbol{x} - \boldsymbol{S}\hat{\boldsymbol{\alpha}}_{\boldsymbol{k}} \tag{3.19}$$

where:

$\boldsymbol{x}$ = input vector
k = An integer representing target number with total targets K
$\hat{f}_k$ = estimated frequency of one target
$\boldsymbol{x_{fft}}$ = FFT of the residual vector $\boldsymbol{x_{res}}$
N = Signal length i.e. length of vector $\boldsymbol{x}$
$\hat{\boldsymbol{\alpha}}_{\boldsymbol{k}}$ = vector of estimated amplitude (complex) of all target k and all previous targets $1-(k-1)$.

Apart from calculation of $\boldsymbol{S}$ and $\hat{\boldsymbol{\alpha}}_{\boldsymbol{k}}$, there is another difference from CLEAN algorithm evident from (3.19). In case of OMP, the amplitudes for all estimated targets are re-estimated in each iteration.
The stopping criteria for OMP is same as in CLEAN. The frequency selection is also possible. The hyper-parameters for OMP is also same as CLEAN and are mentioned in table 3.3. The effect of zero padding is also the same as discussed in section 3.2.1.1.

3.2.3 ESPRIT Algorithm

ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) is a subspace-based algorithm for estimating the parameters of multiple sinusoidal signals from their received measurements. It exploits the rotational invariance property of a uniform linear array (ULA) to decompose the received signal into two subspaces, enabling accurate and computationally efficient parameter estimation.
Consider a single element uniform linear array (ULA) that receives echoes from K distinct targets. The signal model can be expressed in form of:

$$\boldsymbol{x}(n) = \sum_{k=1}^{K} a_k e^{j2\pi f_k n} \tag{3.20}$$

$$\boldsymbol{x}(n) = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \phi^n \underline{a} = \mathbf{1}\phi^n \underline{a} \tag{3.21}$$

where:

$$\phi = \begin{bmatrix} e^{j2\pi f_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & e^{j2\pi f_K} \end{bmatrix} \implies \phi^n = \begin{bmatrix} e^{j2\pi f_1 n} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & e^{j2\pi f_K n} \end{bmatrix}$$

$$\underline{a} = \begin{bmatrix} a_1 & \cdots & a_K \end{bmatrix}^T$$

We note that the eigenvalues of ϕ are $\begin{bmatrix} e^{j2\pi f_1} & \cdots & e^{j2\pi f_K} \end{bmatrix}$. Now define the Hankel matrix with q rows when $q > K$

$$H = \begin{bmatrix} x(0) & x(1) & x(2) & \cdots & \vdots \\ x(1) & x(2) & x(3) & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x(q-1) & x(q) & x(q+1) & \cdots & x(N-1) \end{bmatrix} \tag{3.22}$$

Now insert values from signal model (3.21) in the Hankel matrix (3.22), we get

$$H = \begin{bmatrix} \mathbf{1}\underline{a} & \mathbf{1}\phi\underline{a} & \mathbf{1}\phi^2\underline{a} & \cdots & \mathbf{1}\phi^{N-q}\underline{a} \\ \mathbf{1}\phi\underline{a} & \mathbf{1}\phi^2\underline{a} & \mathbf{1}\phi^3\underline{a} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{1}\phi^{q-1}\underline{a} & \mathbf{1}\phi^q\underline{a} & \mathbf{1}\phi^{q+1}\underline{a} & \cdots & \mathbf{1}\phi^{N-1}\underline{a} \end{bmatrix} \tag{3.23}$$

The number of rows in H is q and the number of columns is $(N-1)-(q-1)+1 = N-q+1$. It is assumed that $N-q+1 \geq K$, which admits the factorization of rank K and resulting in k rows and k columns, hence

$$H = \begin{bmatrix} \mathbf{1} \\ \mathbf{1}\phi \\ \vdots \\ \mathbf{1}\phi^{q-1} \end{bmatrix} \begin{bmatrix} \underline{a} & \phi\underline{a} & \phi^2\underline{a} & \cdots & \phi^{N-q}\underline{a} \end{bmatrix} \tag{3.24}$$

Given the H (i.e. the signal $\boldsymbol{x}(n)$), this factorization is not unique.
If $(C, A, b) = (\mathbf{1}T, T^{-1}\phi T, T^{-1}\underline{a})$, then for some square invertible matrix T

$$H = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{q-1} \end{bmatrix} \begin{bmatrix} b & Ab & \cdots & A^{N-q}b \end{bmatrix} \tag{3.25}$$

We also note that

$$\begin{bmatrix} \mathbf{1} \\ \mathbf{1}\phi \\ \mathbf{1}\phi^2 \\ \vdots \\ \mathbf{1}\phi^{q-1} \end{bmatrix} T = \begin{bmatrix} \mathbf{1}T \\ \mathbf{1}\phi T \\ \mathbf{1}\phi^2 T \\ \vdots \\ \mathbf{1}\phi^{q-1}T \end{bmatrix} = \begin{bmatrix} \mathbf{1}T \\ \mathbf{1}TT^{-1}\phi T \\ \mathbf{1}TT^{-1}\phi^2 T \\ \vdots \\ \mathbf{1}TT^{-1}\phi^{q-1}T \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{q-1} \end{bmatrix} \tag{3.26}$$

This means for any rank-K factorization of H, i.e.

$H = UZ$ where U has K-columns

there exists (C, A, b) such that

$$U = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{q-1} \end{bmatrix} \text{ and } Z = \begin{bmatrix} b & Ab & \cdots & A^{N-q}b \end{bmatrix}$$

and some symmetric T such that $A = T^{-1}\phi T$, which shows that the eigenvalues of A are same as the eigenvalues of ϕ.
Since U has the structure that row $p+1$ is row p multiplied with A, then if $U_{\text{-}}$ are the $q-1$ first rows of U and U_{+} are the $q-1$ last rows of U, then

$$U_{\text{-}}A = U_{+} \tag{3.27}$$

Since it is assumed that $q > K$ we can solve for A if $U_{\text{-}}$ has full rank. Now since

$$U_{\text{-}} = \underbrace{\begin{bmatrix} \mathbf{1} \\ \mathbf{1}\phi \\ \vdots \\ \mathbf{1}\phi^{q-1} \end{bmatrix}}_{\text{Vandermonde matrix}} \quad T \tag{3.28}$$

and both matrices to the right in (3.28) have full rank, therefore from (3.27)

$$A = (U_{\text{-}}^H U_{\text{-}})^{-1} U_{\text{-}}^H U_+ \tag{3.29}$$

An eigenvalue decomposition of A now gives the eigenvalues $e^{j2\pi f_1} \cdots e^{j2\pi f_k}$ and the frequencies are the angles. The amplitude $a_1 \cdots a_k$ are obtained in the same way as the last step in OMP by solving the LS problem.
This construction only works as intended if $q > K$ and $N - q + 1 \geq K$. If we use the minimal dimension $q = K + 1$, we obtain $N - (K + 1) + 1 \geq K$,

$$\implies N \geq 2K$$

The Reversed Conjugate signal model

This can be expressed as

$$\boldsymbol{x}_r(n) = \overline{\boldsymbol{x}}(N - 1 - n) = \sum_{k=1}^{K} \overline{a_k} e^{-j2\pi f_k (N-1-n)} \tag{3.30}$$

$$\boldsymbol{x}_r(n) = \sum_{k=1}^{K} \overline{a_k} e^{-j2\pi f_k (N-1)} e^{-j2\pi f_k n} = \mathbf{1}\phi^n \underline{b} \tag{3.31}$$

where:

$\underline{b} = \phi^{1-N} \underline{a}$

Let H_r be the Hankel matrix for the x_r signal, then

$$\begin{bmatrix} H & H_r \end{bmatrix} P = \begin{bmatrix} \mathbf{1} \\ \mathbf{1}\phi \\ \mathbf{1}\phi^2 \\ \vdots \\ \mathbf{1}\phi^{q-1} \end{bmatrix} \begin{bmatrix} \underline{a} & \phi\underline{a} & \cdots & \phi^{N-q}\underline{a} & \underline{b} & \phi\underline{b} & \cdots & \phi^{N-q}\underline{b} \end{bmatrix} \tag{3.32}$$

$$= \begin{bmatrix} \mathbf{1} \\ \mathbf{1}\phi \\ \mathbf{1}\phi^2 \\ \vdots \\ \mathbf{1}\phi^{q-1} \end{bmatrix} \begin{bmatrix} \underline{a} & \phi\underline{a} & \cdots & \phi^{N-q}\underline{a} & \phi^{1-N}\underline{a} & \phi^{2-N}\underline{a} & \cdots & \phi^{1-q}\underline{a} \end{bmatrix} \tag{3.33}$$

The matrix to the right has the rank K. If it has at least K columns, then:

$$2(N - q + 1) \geq K \tag{3.34}$$
$$2(N - (K + 1) + 1) \geq K \tag{3.35}$$
$$2N - 2K \geq K \tag{3.36}$$
$$N \geq \frac{3K}{2} \tag{3.37}$$

This implies that $N \geq \frac{3K}{2}$ samples are needed for the reverse conjugate signal model also.
The factor $UZ = \begin{bmatrix} H & H_r \end{bmatrix}$ is computed using the truncated singular value decomposition,

$$\begin{bmatrix} H & H_r \end{bmatrix} \begin{bmatrix} \sum \\ \sum_{\perp} \end{bmatrix} \begin{bmatrix} V^H \\ V_{\perp}^H \end{bmatrix} \tag{3.38}$$

where:

$\sum = \text{diag} \begin{pmatrix} \sigma_1 & \cdots & \sigma_k \end{pmatrix}$ have the k-largest singular values.

In the noise free case, $\sum_{\perp} = 0$, so

$$\begin{bmatrix} H & H_r \end{bmatrix} = U \sum V^H = UZ \tag{3.39}$$

In the noisy case, $\begin{bmatrix} H & H_r \end{bmatrix}$ has a rank larger than K but we use $U \sum V^H$ as the rank K factorization.

The computational cost of the ESPRIT algorithm is dominated by the SVD operation. However, the ESPRIT algorithm based on partial SVD and fast Hankel matrix–vector multiplications in [12] has much lower cost. Using the cyclic convolution property of discrete Fourier Transform and applying fast Fourier Transform (FFT), one obtains a fast evaluation of a circulant matrix-vector product.

The summary of ESPRIT via complete/partial SVD is as follows:
Input: Order d and data x of length N

- Form the rectangular hankel matrix using the input data.
- Compute the SVD of Hankel matrix.
- Compute the eigenvalues, $\boldsymbol{\lambda}$, and from these frequency estimates are calculated as.

$$\hat{\boldsymbol{f}} = \text{sort}(\frac{\angle \boldsymbol{\lambda}}{2\pi}) \tag{3.40}$$

- Compute the steering matrix, $\boldsymbol{S}$ from $\hat{\boldsymbol{f}}$

$$\boldsymbol{S} = \begin{bmatrix} e^{j2\pi\hat{f}_1 0} & e^{j2\pi\hat{f}_2 0} & \dots & e^{j2\pi\hat{f}_d 0} \\ e^{j2\pi\hat{f}_1 1} & e^{j2\pi\hat{f}_2 1} & \dots & e^{j2\pi\hat{f}_d 1} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j2\pi\hat{f}_1 (N-1)} & e^{j2\pi\hat{f}_2 (N-1)} & \dots & e^{j2\pi\hat{f}_d (N-1)} \end{bmatrix} \tag{3.41}$$

- Compute the complex amplitudes, $\hat{\boldsymbol{\alpha}}$.

$$\hat{\boldsymbol{\alpha}} = (\boldsymbol{S}^H \boldsymbol{S})^{-1} \boldsymbol{S}^H \boldsymbol{x} \tag{3.42}$$

The amplitudes can also be estimated by solving the problem as LMMSE problem. The benefit of using this approach is that it puts a bound on the amplitude errors and does not let them overshoot if the matrix $\boldsymbol{S}$ is not well conditioned. The LMMSE amplitude estimates can be calculated as:

$$\hat{\boldsymbol{\alpha}} = \left(\boldsymbol{S}^H \boldsymbol{S} + \frac{\sigma_v^2}{\sigma_a^2} \boldsymbol{I} \right)^{-1} \boldsymbol{S}^H \boldsymbol{x} \tag{3.43}$$

where:
σ_a^2 = assumed amplitude variance
σ_v^2 = assumed noise variance

The proof of (3.43) is attached in Appendix A.
In this thesis, σ_a^2 is assumed to be always 1. Therefore only σ_v^2 will be considered. The hyper-parameters for ESPRIT are listed in table 3.4

Parameter	**Description**	**Range**
order	Order of the algorithm	$\{0, 1, 2, 3, \ldots\}$
q	Number of rows in Hankel matrix	$\leq N$
σ_v	Noise std	$\in \mathbb{R}$

Table 3.4: Hyper-parameters for ESPRIT Algorithm

Here, *order* of the algorithm should equal to total number of targets. In case of data with clutter, this number needs to be greater than actual number of targets present. Moreover, the default value for the parameter q is set as $\frac{N}{2}$.

3.2.4 fs-ESPRIT Algorithm

The fs-ESPRIT algorithm is derived for the scalar valued case $y(n) \in \mathbb{C}$. This algorithm was first presented in [15] and further developments and applications are reported in [16, 17, 18, 19, 20].
We assume the noise free signal is given by

$$y(n) = \boldsymbol{c} \boldsymbol{A}^n \boldsymbol{b} \tag{3.44}$$

$$\boldsymbol{x}(n) = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \phi^n \underline{a} = \mathbf{1} \phi^n \underline{a} \tag{3.45}$$

where:

$$\boldsymbol{c} = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}$$

$$\boldsymbol{A} = \begin{bmatrix} e^{j2\pi f_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & e^{j2\pi f_K} \end{bmatrix}$$

$$\boldsymbol{b}^T = \begin{bmatrix} a_1 & \cdots & a_K \end{bmatrix}^T$$

K = Total number of targets

The signal is hence a weighted sum of K complex exponentials. It is also assumed that the frequencies $f_i \in (-0.5, 0.5]$ are distinct and $a_i \neq 0$ for all i. This implies that the signal $y(n)$ cannot be described by another model with fewer than K complex exponentials. In the following steps, it is assumed that we have access to N samples $y(0), y(1), \cdots, y(N-1)$. If all samples are considered as a vector, then we can describe the signal as

$$\boldsymbol{y} = \begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N-1) \end{bmatrix} = \boldsymbol{S}\boldsymbol{b} \tag{3.46}$$

where:

$$\begin{aligned} \boldsymbol{S} &= \begin{bmatrix} \boldsymbol{s}(f_1) & \boldsymbol{s}(f_2) & \cdots & \boldsymbol{s}(f_K) \end{bmatrix} \quad \text{is a Vandermonde matrix and} \\ \boldsymbol{s}^T(f) &= \begin{bmatrix} 1 & e^{j2\pi f} & e^{j4\pi f} & \cdots & e^{j2\pi(N-1)f} \end{bmatrix} \text{ is the Vandermonde vector} \end{aligned}$$

Since the frequencies are assumed distinct, the Vandermonde matrix $\boldsymbol{S}$ has full rank, see e.g. [21].
We directly see that the rows in the Vandermonde matrix is the vectors which generate the Krylov (row-)subspace associated with row vector $\boldsymbol{c}$ and matrix $\boldsymbol{A}$.

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{c} \\ \boldsymbol{c}\boldsymbol{A} \\ \boldsymbol{c}\boldsymbol{A}^2 \\ \vdots \\ \boldsymbol{c}\boldsymbol{A}^{q-1} \end{bmatrix} \tag{3.47}$$

3.2.4.1 DFT relations

The DFT of the N samples is

$$Y(k) = \sum_{n=0}^{N-1} y(n) e^{-j2\pi \frac{nk}{N}} = \boldsymbol{c}\Big(\sum_{n=0}^{N-1} \boldsymbol{A}^n e^{-j2\pi \frac{nk}{N}}\Big)\boldsymbol{b} = \boldsymbol{c}\boldsymbol{A}(k)\boldsymbol{b} \tag{3.48}$$

where we have introduced

$$\boldsymbol{A}(k) = \sum_{n=0}^{N-1} \boldsymbol{A}^n e^{-j2\pi \frac{nk}{N}} \tag{3.49}$$

as the DFT of the length N matrix valued signal $\boldsymbol{A}^n$. To simplify notation below, we introduce $z_k = e^{j2\pi \frac{k}{N}}$ and we have

$$\boldsymbol{A}(k) = \sum_{n=0}^{N-1} \boldsymbol{A}^n z_k^{-n} \tag{3.50}$$

A key algebraic relation is

$$\begin{aligned} z_k \boldsymbol{A}(k) &= \sum_{n=0}^{N-1} \boldsymbol{A}^n z_k^{-(n-1)} = \boldsymbol{A} \sum_{n=1}^{N-1} \boldsymbol{A}^{n-1} z_k^{-(n-1)} + z_k \boldsymbol{I} \\ &= \boldsymbol{A} \sum_{n=1}^{N-1} \boldsymbol{A}^{n-1} z_k^{-(n-1)} + z_k \boldsymbol{I} - z_k^{1-N} \boldsymbol{A}^N \\ &= \boldsymbol{A} \sum_{n=0}^{N-1} \boldsymbol{A}^n z_k^{-n} + z_k \boldsymbol{I} - z_k \boldsymbol{A}^N \\ &= \boldsymbol{A}\boldsymbol{A}(k) + z_k(\boldsymbol{I} - \boldsymbol{A}^N) \end{aligned}$$

where we used the equality $z_k^{-N} = 1$. A repeated use of this relation yields

$$z_k^r \boldsymbol{A}(k) = \boldsymbol{A}^r \boldsymbol{A}(k) + \sum_{i=1}^{r} z_k^i \boldsymbol{A}^{r-i}(\boldsymbol{I} - \boldsymbol{A}^N) \tag{3.51}$$

Applying this to the DFT of $y(n)$ results in

$$z_k^r \boldsymbol{Y}(k) = z_k^r \boldsymbol{c}\boldsymbol{A}(k)\boldsymbol{b} = \boldsymbol{c}\boldsymbol{A}^r \boldsymbol{A}(k)\boldsymbol{b} + \sum_{i=1}^{r} z_k^i \boldsymbol{c}\boldsymbol{A}^{r-i}(\boldsymbol{I} - \boldsymbol{A}^N)\boldsymbol{b} \tag{3.52}$$

3.2.4.2 Subspace equations

With the definition $\tilde{\boldsymbol{b}} = (\boldsymbol{I} - \boldsymbol{A}^N)\boldsymbol{b}$ and (3.52), we can form the vector equality

$$\underbrace{\begin{bmatrix} 1 \\ z_k \\ z_k^2 \\ \vdots \\ z_k^{q-1} \end{bmatrix}}_{\boldsymbol{Y}_k} Y(k) = \begin{bmatrix} \boldsymbol{c} \\ \boldsymbol{c}\boldsymbol{A} \\ \boldsymbol{c}\boldsymbol{A}^2 \\ \vdots \\ \boldsymbol{c}\boldsymbol{A}^{q-1} \end{bmatrix} \boldsymbol{A}(k)\boldsymbol{b} + \underbrace{\begin{bmatrix} 0 & \cdots & \cdots & 0 \\ \boldsymbol{c}\tilde{\boldsymbol{b}} & 0 & \cdots & 0 \\ \boldsymbol{c}\boldsymbol{A}\tilde{\boldsymbol{b}} & \boldsymbol{c}\tilde{\boldsymbol{b}} & \vdots & \vdots \\ \vdots & \ddots & \cdots & 0 \\ \boldsymbol{c}\boldsymbol{A}^{q-2}\tilde{\boldsymbol{b}} & \boldsymbol{c}\boldsymbol{A}^{q-2}\tilde{\boldsymbol{b}} & \cdots & \boldsymbol{c}\tilde{\boldsymbol{b}} \end{bmatrix}}_{\Gamma} \underbrace{\begin{bmatrix} z_k \\ z_k^2 \\ \vdots \\ z_k^{q-1} \end{bmatrix}}_{\boldsymbol{Z}_k} \tag{3.53}$$

With the introduced notation in (3.53), we have

$$\boldsymbol{Y}_k = \boldsymbol{S}\boldsymbol{A}(k)\boldsymbol{b} + \Gamma \boldsymbol{Z}_k \tag{3.54}$$

Let $k_1, k_2, ..., k_M$ be M distinct integers such that $0 \leq k_i \leq N$. Then based on (3.54), we have the matrix equality.

$$\boldsymbol{Y} = \boldsymbol{S}\boldsymbol{X} + \Gamma \boldsymbol{Z} \tag{3.55}$$

where:

$$\begin{aligned} \boldsymbol{Y} &= \begin{bmatrix} \boldsymbol{Y}_{k_1} & \boldsymbol{Y}_{k_1} & \cdots & \boldsymbol{Y}_{k_M} \end{bmatrix} \\ \boldsymbol{X} &= \begin{bmatrix} \boldsymbol{A}(k_1)\boldsymbol{b} & \boldsymbol{A}(k_2)\boldsymbol{b} & \cdots & \boldsymbol{A}(k_M)\boldsymbol{b} \end{bmatrix} \\ \boldsymbol{Z} &= \begin{bmatrix} \boldsymbol{Z}_{k_1} & \boldsymbol{Z}_{k_1} & \cdots & \boldsymbol{Z}_{k_M} \end{bmatrix} \end{aligned}$$

If $M \geq q-1$ then the matrix $\boldsymbol{Z}$ has a nullspace of dimension d_n where $d_n = M-q+1$ since $\boldsymbol{Z}$ is a Vandermonde matrix and has full rank. Let $\boldsymbol{Z}_\perp \in \mathbb{C}^{M \times d_n}$ be a matrix where the range space of $\boldsymbol{Z}_\perp$ spane the nullspace of $\boldsymbol{Z}$ then, as $\boldsymbol{Z}\boldsymbol{Z}_\perp = 0$

$$\boldsymbol{Y}\boldsymbol{Z}_\perp = \boldsymbol{S}\boldsymbol{X}\boldsymbol{Z}_\perp \tag{3.56}$$

which directly tells that the range space of $\boldsymbol{Y}\boldsymbol{Z}_\perp$ is a subspace of the range space of $\boldsymbol{S}$. If $M \geq K+q-1$ it can be shown along the lines presented in [22] that the range space of $\boldsymbol{Y}\boldsymbol{Z}_\perp$ is equal to the range space of $\boldsymbol{S}$. This, in turn, implies that if $q \geq K$ then $rank(\boldsymbol{Y}\boldsymbol{Z}_\perp) = K$. Let $\boldsymbol{Z}_Y \in \mathbb{C}^{q \times K}$ be a matrix which span the range space of $\boldsymbol{Y}\boldsymbol{Z}_\perp$. It then follows that there exists a non-singular matrix $\boldsymbol{P} \in \mathbb{C}^{K \times K}$ such that

$$\boldsymbol{S}_Y\boldsymbol{P} = \boldsymbol{S} \tag{3.57}$$

From the structure of $\boldsymbol{S}$ we can see that

$$\boldsymbol{S}_Y = \boldsymbol{S}\boldsymbol{P}^{-1} = \begin{bmatrix} \boldsymbol{c}\boldsymbol{P}^{-1} \\ \boldsymbol{c}\boldsymbol{P}^{-1}\boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1} \\ \vdots \\ \boldsymbol{c}\boldsymbol{P}^{-1}(\boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1})^{q-1} \end{bmatrix} \triangleq \begin{bmatrix} \boldsymbol{c}_Y \\ \boldsymbol{c}_Y\boldsymbol{A}_Y \\ \vdots \\ \boldsymbol{c}_Y\boldsymbol{A}_Y^{q-1} \end{bmatrix} \tag{3.58}$$

and hence $\boldsymbol{A}_Y = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}$ which tells us that $\boldsymbol{A}_Y$ and $\boldsymbol{A}$ are similar and hence have same set of eigenvalues. From the shift structure in (3.58), we see that

$$\begin{bmatrix} \boldsymbol{I}_{q-1} & \boldsymbol{0}_{(q-1)\times 1} \end{bmatrix} \boldsymbol{S}_Y\boldsymbol{A}_Y = \begin{bmatrix} \boldsymbol{0}_{(q-1)\times 1} & \boldsymbol{I}_{q-1} \end{bmatrix} \boldsymbol{S}_Y \tag{3.59}$$

and if $q > K$ then $\begin{bmatrix} \boldsymbol{I}_{(q-1)p} & S_{(q-1)p\times p} \end{bmatrix} \boldsymbol{S}_Y$ has full rank K and we have a unique solution for $\boldsymbol{A}_Y$.

$$\boldsymbol{A}_Y = (\begin{bmatrix} \boldsymbol{I}_{q-1} & \boldsymbol{0}_{(q-1)\times 1} \end{bmatrix} \boldsymbol{S}_Y)^+ \begin{bmatrix} \boldsymbol{0}_{(q-1)\times 1} & \boldsymbol{I}_{q-1} \end{bmatrix} \boldsymbol{S}_Y \tag{3.60}$$

where $(.)^+$ denotes the Moore-Penrose pseudo inverse. The eigenvalues of $\boldsymbol{A}_Y$ is $\boldsymbol{\lambda}$, which is the set $\{e^{j2\pi f_i}\}_{j=1}^K$. The frequency estimates can then be calculated as.

$$\hat{\boldsymbol{f}} = \text{sort}(\frac{\angle\boldsymbol{\lambda}}{2\pi}) \tag{3.61}$$

The steering matrix $\boldsymbol{S}$ and amplitude estimates $\hat{\boldsymbol{\alpha}}$ can be calculated in a similar way shown in equations 3.41 and 3.43.
The hyper-parameters for fs-ESPRIT are listed in table 3.5.
In this thesis, σ_a^2 is assumed to be always 1. Therefore only σ_v^2 will be considered. Here, *order* of the algorithm should equal to total number of targets. Moreover, the default value for the parameter q is set as $\frac{N}{2}$.

Parameter	Description	Range
order	Order of the algorithm	$\{0, 1, 2, 3, \ldots\}$
q	Number of rows in Hankel matrix	$\leq N$
$kset$	vector with set of integers for frequencies to include in the estimation	$1 \leq kset \leq N$
σ_v	Noise std	$\in \mathbb{R}$

Table 3.5: Hyper-parameters for fs-ESPRIT Algorithm

3.3 Frequency Mapping

The frequencies and amplitudes reported from these estimation algorithms are not necessarily in same order as the original frequencies. There is a need of establishing a mechanism through which the estimated frequencies can be mapped onto the true frequencies. The amplitudes will be automatically mapped in the same order. This step is essential to compute true errors.
Generalized optimal sub-pattern assignment metric (GOSPA) [29] metric is used for this purpose. It is a very robust method that can even help to map the frequencies which are very closely placed.

3.3.1 GOSPA Metric

The GOSPA parameters are set as following:

$$\alpha = 2$$
$$p = 2$$

The parameter c does not need to be assigned as there is no cardinality mismatch. The sets of true and detected frequencies are given as input to GOSPA. The matrix Π contains all the permutations of vector $vec = 1 : \text{length}(y)$, where y is set of estimated frequencies. So basically, Π contains all the possible orders of indices, and its each row is π which is a sequence $(\pi(1), , , \pi(n)$, where n is number of total estimated frequencies. The metric d is computed for true frequencies $x(i)$ and estimated frequencies $y(\Pi(j, i))$, where $i = \text{length}(x)$ and $j = \text{length}(\Pi)$. The sum in (2.7) is calculated for all combinations and are stored in 1-D array of length equal to number of rows in Π. The minima of this array then gives the loss and the set of indices that yield the least error. The estimated frequencies and amplitudes are then re-arranged according to these new indices.
The metric d could be any metric that determines the closeness of two numbers and the Euclidean distance can be used, however in our case the OMP and CLEAN algorithms return frequencies mapped between $[0, 1)$ whilst the ESPRIT and fs-ESPRIT algorithms return frequencies between $[-0.5, 0.5)$. Therefore, angle based distance is used, which can compute distance between two frequencies regardless of their mapping.

The $[-0.5, 0.5)$ frequencies can be mapped later onto the $[0, 1)$ range by:

$$f = \begin{cases} 1 + f & f < 0 \\ f & otherwise \end{cases} \tag{3.62}$$

3.3.1.1 Angle based distance

The angle is computed between two vectors where each vector is represented by a complex scalar in the form of the complex exponential.

$$d = \left| \frac{\angle e^{j2\pi f_{true}} \overline{e^{j2\pi f_{est}}}}{2\pi} \right| \tag{3.63}$$

where:

$f_{true}, f_{est} = \in [0, 1)$ or [-0.5,0.5)

3.4 Clutter

One of the most challenging problem in radar detection and estimation is posed by the presence of clutter in data. It can be ground clutter, clutter due to rain, sea clutter and so on. Each type of clutter has distinct characteristics with different frequency spreads and clutter power. Here, a simple example of clutter is incorporated with clutter spread close to zero frequency and clutter amplitude comparable to target's amplitude.

3.4.1 Clutter generation

The clutter is generated using similar equation as (3.1). But this time a number of frequencies are added which are closely spaced and very close to zero frequency. A clutter range is defined to determine clutter spread in normalized frequency. For example clutter range of 0.05 means the normalized frequency ranges from -0.05 to 0.05. Several exponentials with uniformly spaced frequencies in this range are added together to form clutter. The magnitude of clutter is then adjusted to scale it down in order to get it comparable to target strength.

$$\boldsymbol{c_{sig}} = \sum_{c=0}^{C} a_c \begin{bmatrix} e^{j2\pi f_c 0} \\ e^{j2\pi f_c 1} \\ \vdots \\ e^{j2\pi f_c (N-1)} \end{bmatrix} \tag{3.64}$$

$$\boldsymbol{c_{sig}} = a_0 \begin{bmatrix} e^{j2\pi f_0 0} \\ e^{j2\pi f_0 1} \\ \vdots \\ e^{j2\pi f_0 (N-1)} \end{bmatrix} + a_1 \begin{bmatrix} e^{j2\pi f_1 0} \\ e^{j2\pi f_1 1} \\ \vdots \\ e^{j2\pi f_1 (N-1)} \end{bmatrix} + \cdots a_C \begin{bmatrix} e^{j2\pi f_C 0} \\ e^{j2\pi f_C 1} \\ \vdots \\ e^{j2\pi f_C (N-1)} \end{bmatrix} \tag{3.65}$$

$$\boldsymbol{x_{clutter}} = \frac{\sqrt{\sigma_c^2}\boldsymbol{c_{sig}}}{|\boldsymbol{c_{sig}}|} \tag{3.66}$$

where:

C = total number of clutter frequencies.
f_c = clutter frequency.
a_c = complex amplitude of clutter for a frequency f_c.
σ_c^2 = clutter power which is user configurable.

In simulation total of 40 exponentials are added symmetrically around the zero frequency. An example of FFT of signal before and after clutter are shown in Figure 3.1

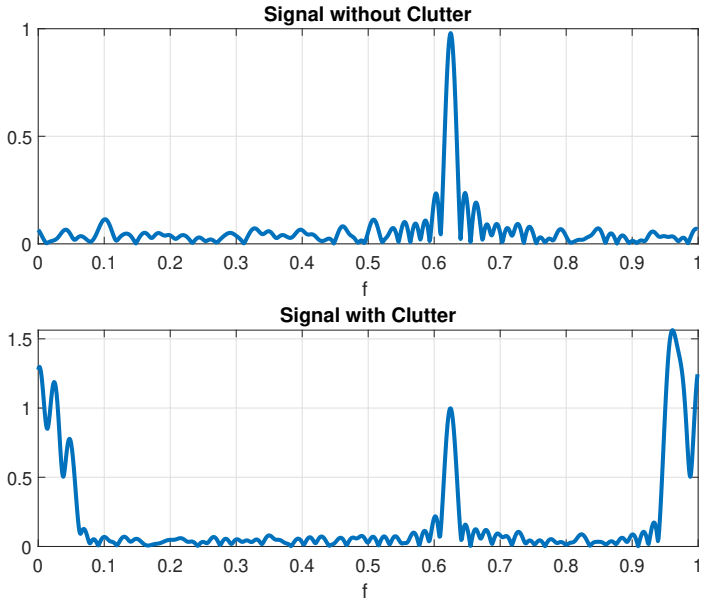


Figure 3.1: Signal with and without clutter

3.4.2 Clutter removal techniques

Some clutter removal techniques are implemented for all estimation algorithms, to remove the clutter from the data.

3.4.2.1 Clutter removal in OMP and CLEAN

For CLEAN and OMP, the frequencies in which clutter is present are truncated before finding the first maxima. The truncation depends upon the assumed clutter spread. For example, the FFT of signal with clutter spread of 5 percent is shown in Figure 3.2. From the figure it can be seen that it is good to start truncation before 0.1 frequency and end it before 0.9. The current implementation sets the truncation frequencies as 1.5 of the clutter spread from both sides. So, for 10 % spread the frequencies used for detection will be 0.15 to 0.85. This can be done by setting hyper-parameters f_s and f_e in table 3.3.

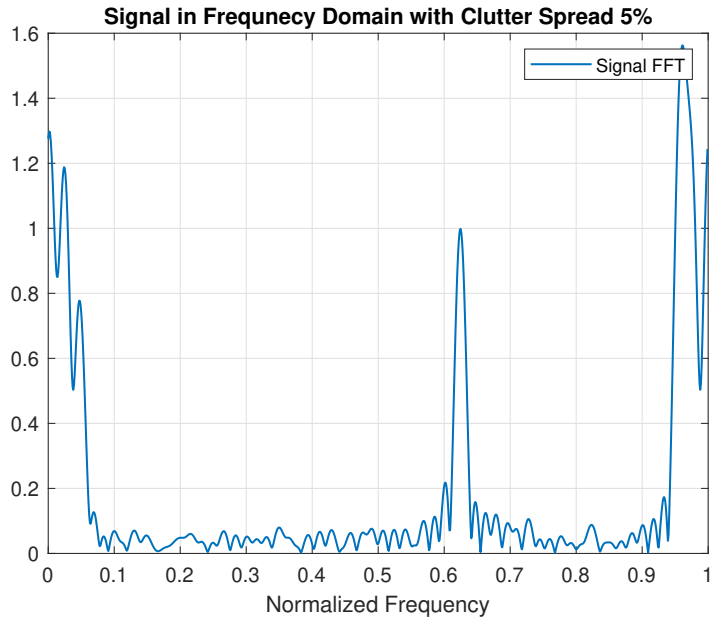


Figure 3.2: Signal with single frequency and clutter spread 5 %

3.4.2.2 Clutter removal in ESPRIT

In ESPRIT, it is not possible to select a range of frequencies to detect targets. Therefore, it is necessary to increase the order of the algorithm by some amount. If, for example, it is required to detect one target in a signal which is corrupted by clutter, then more than one target should be demanded from the algorithm, otherwise it will most probably return a frequency in clutter area. Hence in this case, the order is increased i.e., additional frequencies are demanded from the algorithm and later these frequencies can be removed in post processing. This is done by changing the hyper-parameter *order* mentioned in table 3.4. In post processing it is detected if any of the detected frequency is present in clutter area, and then it is removed.

3.4.2.3 Clutter removal in fs-ESPRIT

In fs-ESPRIT, it is easier to reject clutter based on clutter spread. The algorithm accepts the frequency range in which we need to detect targets. Thus, a set of frequencies based on clutter spread is given to the algorithm and clutter is naturally attenuated. Only those frequencies are returned that lie outside the clutter region. This is done by setting the value of parameter *kset* mentioned in table 3.5.

3.5 Evaluation Methodology

The following section explains the methodology of different types of test performed and their evaluation criteria. It explicitly states how the tests are carried out and how their respective performance metrics are calculated.

3.5.1 Error Computation

In simulation, two parameters for the evaluation of estimation algorithms are used:

- target frequency
- target amplitude

The frequency mapping returns the errors in frequency and amplitude for each target for a single Monte Carlo iteration. These errors are simple difference between ground truth and re-ordered target frequencies and amplitudes (after applying GOSPA). And the root mean square error is the performance metric for the estimation of these parameters.

$$\begin{aligned} RMSE_f &= \sqrt{\frac{1}{N_{MC}} \sum_{m=1}^{N_{MC}} \|f - \hat{f}\|^2} \\ RMSE_\alpha &= \sqrt{\frac{1}{N_{MC}} \sum_{m=1}^{N_{MC}} \|\alpha - \hat{\alpha}\|^2} \end{aligned} \tag{3.67}$$

where:

N_{MC} = Number of Monte Carlo iterations performed.

For the test cases that involve multiple frequencies, the RMS error is again computed for total number of targets to give a single error value.

$$\begin{aligned} RMSE_{tf} &= \sqrt{\frac{1}{K} \sum_{k=1}^{K} \|RMSE_f(k)\|^2} \\ RMSE_{t\alpha} &= \sqrt{\frac{1}{K} \sum_{k=1}^{K} \|RMSE_\alpha(k)\|^2} \end{aligned} \tag{3.68}$$

where:

K = Number of Targets
$RMSE_f$ = Target Frequency RMSE error for single target
$RMSE_\alpha$ = Target Amplitude RMSE error for single target

3.5.2 Noise threshold for Frequency Estimation

In frequency estimation, it is good to have high SNR to pick out the required signal from the noise floor. Figure 3.3 shows the signal peaks are identified easily because the SNR is high.

The threshold above which the signal peaks remain distinguishable from the noise floor depends on the number of data samples N and the SNR. [24] gives a relation to calculate threshold for single complex sinusoid in white gaussian noise.

$$\mathrm{SNR}\frac{N}{\ln N} \gg 1 \tag{3.69}$$

As a rule of thumb, the factor on left is 70 in practical applications. Furthermore, the presence of multiple sinusoids will increase the threshold.

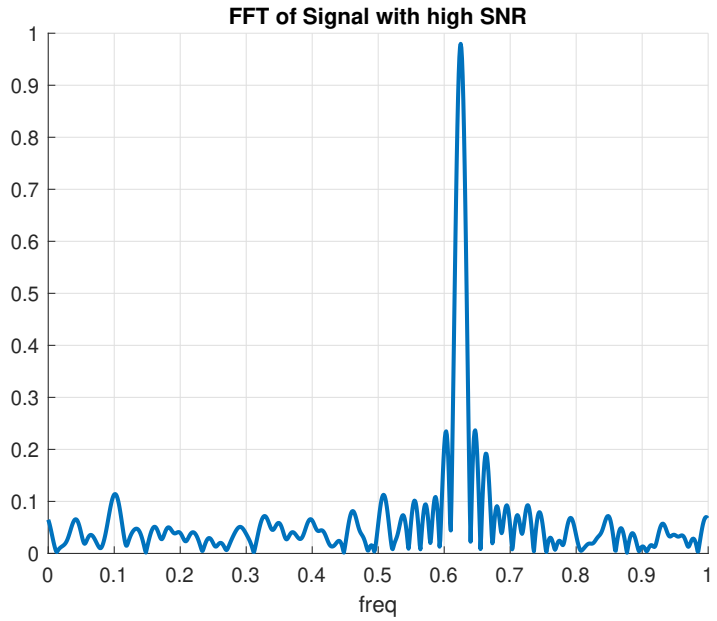


Figure 3.3: Signal with high SNR

3.5.3 Calculation and Usage of Cramer Rao Bound

The theory of Cramer Rao Lower bound is discussed in section 2.6. The usage is described here.
For the signal of the form:

$$\boldsymbol{x} = \boldsymbol{s}(\boldsymbol{f})\boldsymbol{a} + \boldsymbol{v} \tag{3.70}$$

$$\boldsymbol{x} \sim \mathrm{CN}(\boldsymbol{s}(\boldsymbol{f})\boldsymbol{a}, \sigma_v^2\boldsymbol{I}) \tag{3.71}$$

where:

$\boldsymbol{s}$ = steering matrix composed of complex exponentials with frequencies $\boldsymbol{f}$
$\boldsymbol{a}$ = corresponding complex amplitude vector
$\boldsymbol{v}$ = complex white Gaussian noise.

The PDF of $\boldsymbol{x}$ is given by:

$$f(\boldsymbol{x}, \boldsymbol{\theta}) = \mathrm{CN}(\boldsymbol{s}(\boldsymbol{f})\boldsymbol{a}, \sigma_v^2\boldsymbol{I}) \tag{3.72}$$

$$where \quad \boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{a} \end{bmatrix} \tag{3.73}$$

$$\boldsymbol{\mu} = \boldsymbol{s}(\boldsymbol{f})\boldsymbol{a} \tag{3.74}$$

$$\boldsymbol{C} = \sigma_v^2\boldsymbol{I} \tag{3.75}$$

The Fisher Information matrix, $\boldsymbol{F}$, can be calculated using (2.12) and the CRLB matrix can then be calculated using (2.10).
The lower bounds for the variance of θ are obtained from diagonal values of CRLB matrix.

For the case where signal is embedded in White Gaussian noise only and there is no other disturbance like clutter, Cramer Rao Lower Bound is calculated for single and dual frequency cases.

For our case, the first term in (2.12) is zero as the covariance matrix $\boldsymbol{C}$ doesn't depend upon frequency vector $\boldsymbol{f}$ and amplitude vector $\boldsymbol{a}$. And in order to calculate the second term, it is needed to compute all partial derivatives of $\boldsymbol{\mu}$. In this case, $\boldsymbol{\theta}$ is column vector obtained by concatenating the frequency vector $\boldsymbol{f}$ and amplitude vector $\boldsymbol{a}$. Note that the $\boldsymbol{a}$ is a complex vector. And taking partial derivative with respect to a complex value can be a complicated process. So, the derivative is taken for real and imaginary parts separately and the parameter vector is denoted by $\boldsymbol{\xi}$ instead, as discussed earlier in section 2.6
In the following sections, the calculations for partial derivatives are shown for single and dual frequency case:

3.5.3.1 CRLB for Single Frequency

For single frequency case, the $\boldsymbol{\xi}$ is defined as:

$$\boldsymbol{\xi} = \begin{bmatrix} f \\ a_r \\ a_i \end{bmatrix} \tag{3.76}$$

The equation (3.74) for this case will be:

$$\boldsymbol{\mu} = (a_r + ja_i) \begin{bmatrix} e^{j2\pi f0} \\ e^{j2\pi f1} \\ \vdots \\ e^{j2\pi f(N-1)} \end{bmatrix}$$

and the covariance matrix C is defined by (3.75). As we have discussed earlier that first term in (2.12) is zero. There will be three partial derivatives in this case:

$$\boldsymbol{\mu}_1' = \frac{\partial \boldsymbol{\mu}}{\partial f} = ja2\pi \begin{bmatrix} 0e^{j2\pi f0} \\ 1e^{j2\pi f1} \\ \vdots \\ (N-1)e^{j2\pi f(N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}_2' = \frac{\partial \boldsymbol{\mu}}{\partial a_r} = \begin{bmatrix} e^{j2\pi f0} \\ e^{j2\pi f1} \\ \vdots \\ e^{j2\pi f(N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}_3' = \frac{\partial \boldsymbol{\mu}}{\partial a_i} = j \begin{bmatrix} e^{j2\pi f0} \\ e^{j2\pi f1} \\ \vdots \\ e^{j2\pi f(N-1)} \end{bmatrix}$$

These derivatives are used to calculate Fisher Information Matrix using (2.12). And CRLB matrix and bounds are calculated for frequency and amplitude estimates.

3.5.3.2 CRLB for Dual Frequencies

For dual frequency case, the parameter vector $\boldsymbol{\xi}$ is defined as:

$$\boldsymbol{\xi} = \begin{bmatrix} f_1 \\ f_2 \\ a_{1r} \\ a_{1i} \\ a_{2r} \\ a_{2i} \end{bmatrix} \tag{3.77}$$

The equation (3.74) for this case will be:

$$\boldsymbol{\mu} = a_1 \begin{bmatrix} e^{j2\pi f_1 0} \\ e^{j2\pi f_1 1} \\ \vdots \\ e^{j2\pi f_1 (N-1)} \end{bmatrix} + a_2 \begin{bmatrix} e^{j2\pi f_2 0} \\ e^{j2\pi f_2 1} \\ \vdots \\ e^{j2\pi f_2 (N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu} = (a_{1r} + j a_{1i}) \begin{bmatrix} e^{j2\pi f_1 0} \\ e^{j2\pi f_1 1} \\ \vdots \\ e^{j2\pi f_1 (N-1)} \end{bmatrix} + (a_{2r} + j a_{2i}) \begin{bmatrix} e^{j2\pi f_2 0} \\ e^{j2\pi f_2 1} \\ \vdots \\ e^{j2\pi f_2 (N-1)} \end{bmatrix}$$

and the covariance matrix C is defined by (3.75). There will be six partial derivatives in this case:

$$\boldsymbol{\mu}'_1 = \frac{\partial \boldsymbol{\mu}}{\partial f_1} = j a_1 2\pi \begin{bmatrix} 0 e^{j2\pi f_1 0} \\ 1 e^{j2\pi f_1 1} \\ \vdots \\ (N-1) e^{j2\pi f_1 (N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}'_2 = \frac{\partial \boldsymbol{\mu}}{\partial f_2} = j a_2 2\pi \begin{bmatrix} 0 e^{j2\pi f_2 0} \\ 1 e^{j2\pi f_2 1} \\ \vdots \\ (N-1) e^{j2\pi f_2 (N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}'_3 = \frac{\partial \boldsymbol{\mu}}{\partial a_{1r}} = \begin{bmatrix} e^{j2\pi f_1 0} \\ e^{j2\pi f_1 1} \\ \vdots \\ e^{j2\pi f_1 (N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}'_4 = \frac{\partial \boldsymbol{\mu}}{\partial a_{1i}} = j \begin{bmatrix} e^{j2\pi f_1 0} \\ e^{j2\pi f_1 1} \\ \vdots \\ e^{j2\pi f_1 (N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}_5' = \frac{\partial \boldsymbol{\mu}}{\partial a_{2r}} = \begin{bmatrix} e^{j2\pi f_2 0} \\ e^{j2\pi f_2 1} \\ \vdots \\ e^{j2\pi f_2 (N-1)} \end{bmatrix}$$

$$\boldsymbol{\mu}_6' = \frac{\partial \boldsymbol{\mu}}{\partial a_{2i}} = j \begin{bmatrix} e^{j2\pi f_2 0} \\ e^{j2\pi f_2 1} \\ \vdots \\ e^{j2\pi f_2 (N-1)} \end{bmatrix}$$

These derivatives are used to calculate Fisher Information Matrix using (2.12). And CRLB matrix and bounds are calculated for frequency and amplitude estimates.

4

Testing and Results

In this chapter all the test cases and their results are discussed in detail. The chapter discusses the comparison between different estimation algorithms mentioned in section 3.2 under different conditions. The comparison is based on performance evaluation metrics discussed in section 2.4.6.

There are several factors which can affect an estimation algorithm performance. The test cases in order to evaluate the estimation algorithms are explained. The test cases are built in corresponding to following factors:

- target proximity
- noise variance
- signal length
- clutter strength
- clutter spread

4.1 Test Scenario 1: Target Proximity

This test can be performed using Data Scenario 1 (3.1.2.2), where two frequencies are generated with one frequency fixed and the other frequency is given a sweep. The errors can be observed for a given noise variance, σ_v^2. For each frequency pair, 200 Monte Carlo iterations are performed.
The RMS errors for N_{MC} Monte Carlo iterations for one frequency pair can be computed by using (3.67). These Monte Carlo simulations are repeated for all frequency pairs and the errors are plotted against Δf for each target and each algorithm. This type of analysis can be performed for different noise variance levels and different signal lengths. The purpose of this test is to evaluate the performance of different estimation algorithms especially when two targets are placed very close to each other in frequency i.e. $\Delta f \to 0$. The test does not include the data points where $\Delta f = 0$ or it is very close to zero, as in that case two targets are perceived as one and errors are inevitably increased. The plots are not readable in this case or may mislead the reader, especially when there is a singularity at $\Delta f = 0$. Therefore, the point where $\Delta f = 0$ is removed along with some other closer points to zero.
In this case, the signal length is 64, the frequency of target 1 is set as 0.5, while other target is given a sweep from 0.3 to 0.7 corresponding to Δf of -0.2 to 0.2. Total of 100 frequency pairs are used and 8 points are removed from the graph. Table 4.2 shows the parameters of two targets.

Target	Frequency	Amplitude
1	0.5	0.7071 + 0.7071i
2	0.3-0.7	0.0000 + 0.7500i

Table 4.1: Target Parameters for Data Scenario 1

4.1.1 Discussion: Effect of target proximity on different algorithms

Before testing, it is important to analyze the $\Delta f = 0$ scenario in detail for each algorithm. This analysis will help to detect and fix any anomaly present. A test was conducted with noise variance, $\sigma_v^2 = 0.1$, and RMS alpha errors are recorded for all values of Δf, including zero value. These errors are plotted for CLEAN and OMP in figures 4.1 and 4.2.

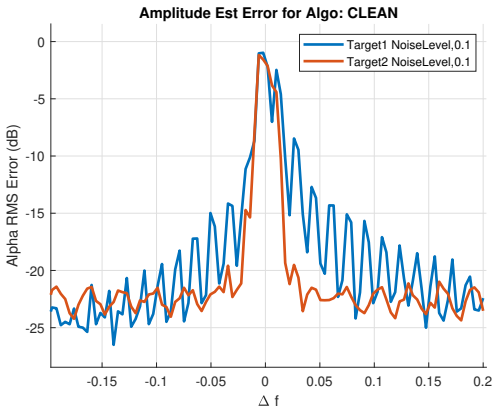


Figure 4.1: RMS alpha Errors CLEAN

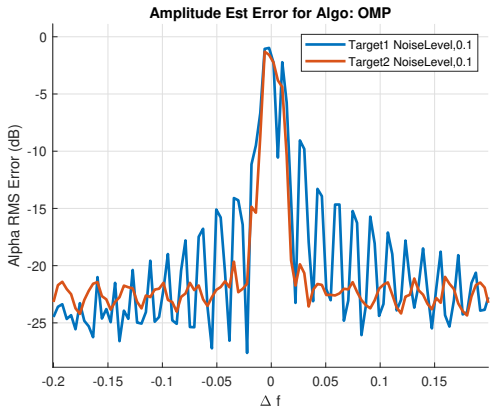


Figure 4.2: RMS alpha Errors OMP

It can be seen that even when two targets are placed very close to each other or even of the same frequency, the alpha errors are contained. Similar test is conducted for ESPRIT and fs-ESPRIT while calculating $\hat{\alpha}$ using (3.42).

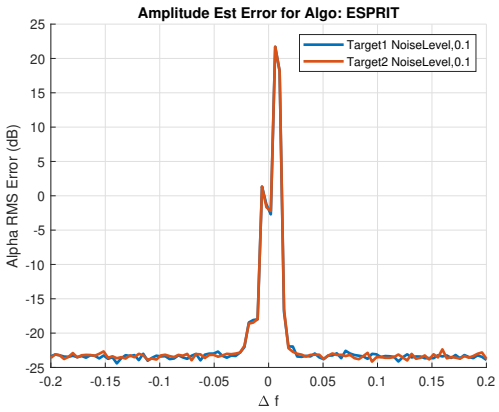


Figure 4.3: RMS alpha Errors ESPRIT

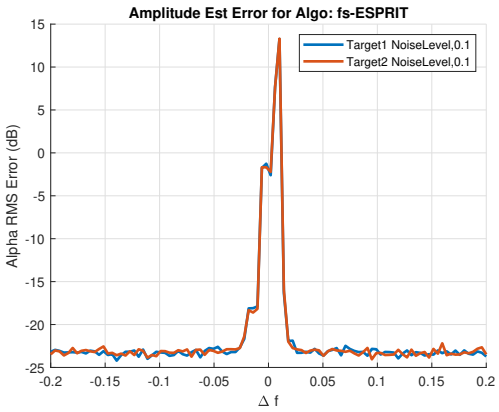


Figure 4.4: RMS alpha Errors fs-ESPRIT

The errors are not contained when targets have same frequency. This problem can be fixed by solving for $\hat{\alpha}$ as an LMMSE problem using (3.43). After some experimentation, the hyper-parameters for LMMSE in ESPRIT and fs-ESPRIT are set as below:

Parameter	Value
$\sigma_{vesprit}$	1
$\sigma_{vfsesprit}$	8

Table 4.2: Parameters for LMMSE in ESPRIT & fs-ESPRIT

The test is repeated and the RMS alpha errors are shown in figures 4.5 and 4.6

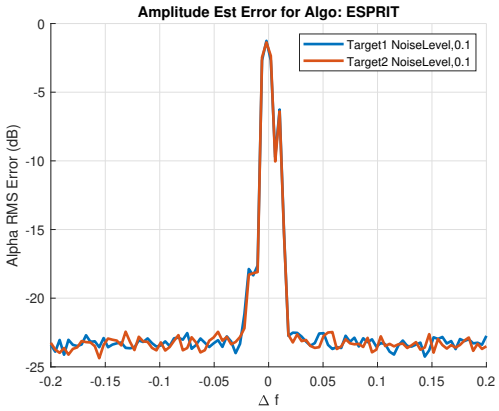


Figure 4.5: RMS alpha Errors ESPRIT after LMMSE

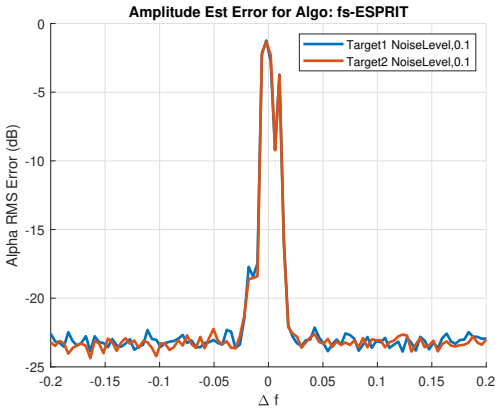


Figure 4.6: RMS alpha Errors fs-ESPRIT after LMMSE

After this modification of these algorithms and selecting appropriate parameters, the proximity test can finally be conducted. This test is performed for three different noise levels, 0.1, 0.5 and 1.

4.1.2 Test Case: Noise Level 0.1

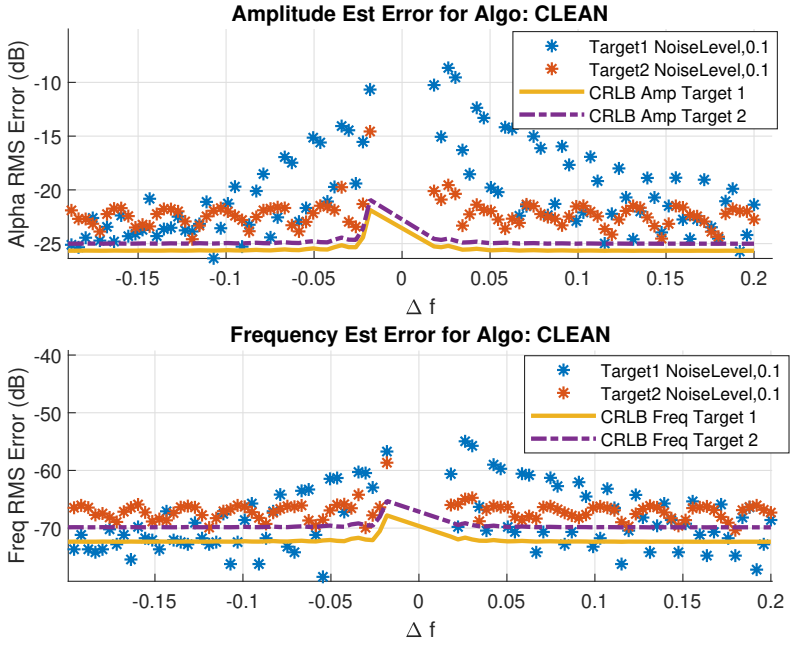


Figure 4.7: Amplitude and Frequency error CLEAN: Noise Level 0.1

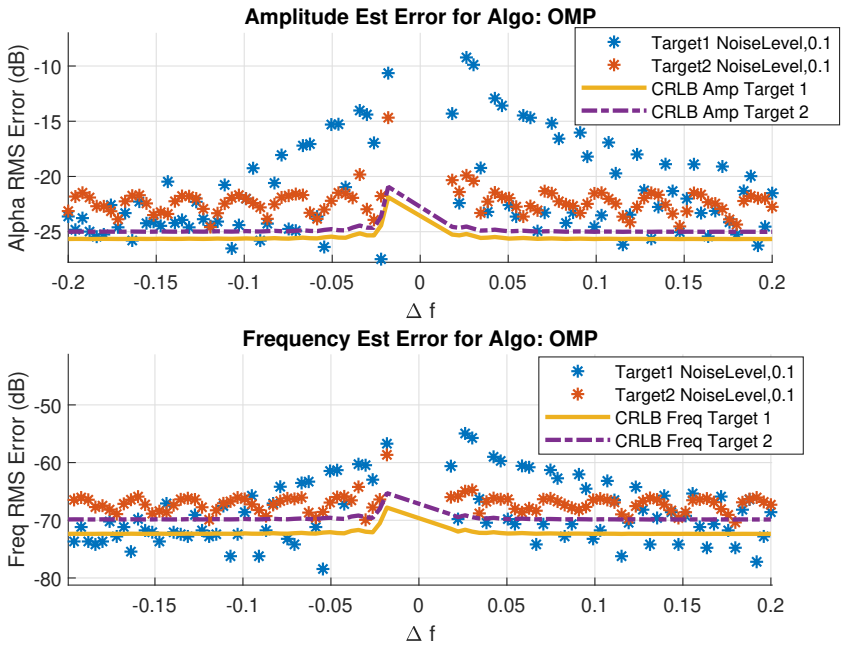


Figure 4.8: Amplitude and Frequency error OMP: Noise Level 0.1

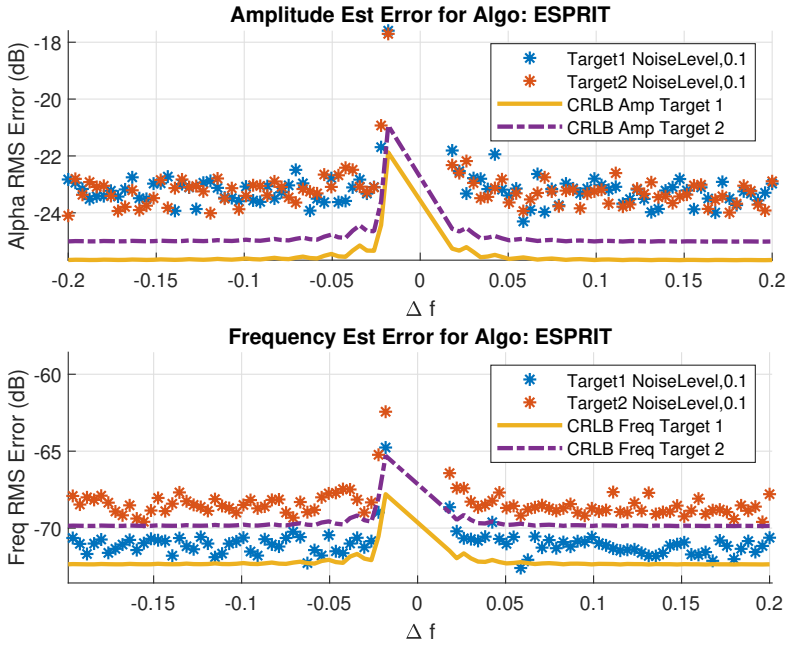


Figure 4.9: Amplitude and Frequency error ESPRIT: Noise Level 0.1

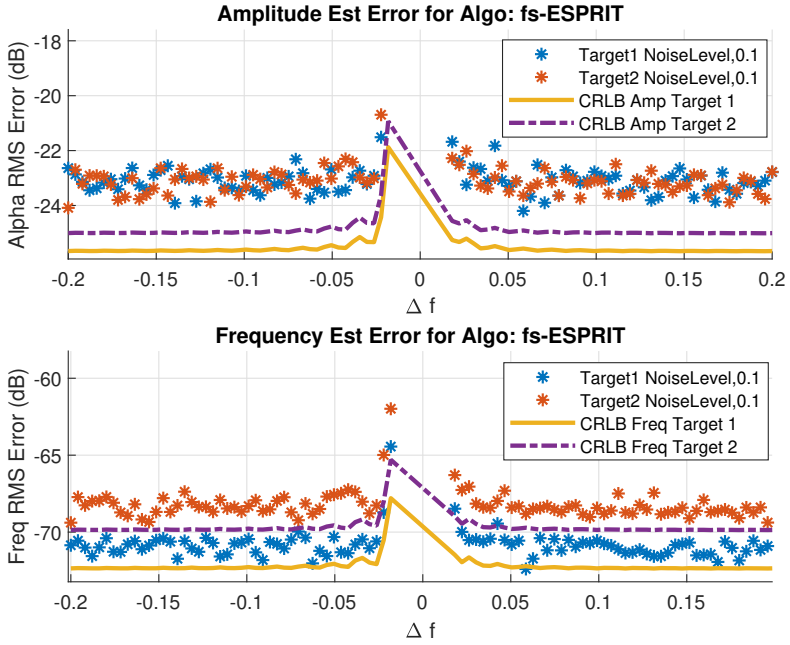


Figure 4.10: Amplitude and Frequency error fs-ESPRIT: Noise Level 0.1

It can be seen that as $\Delta f \rightarrow 0$, the RMS errors increase for CLEAN and OMP, but the errors for ESPRIT and fs-ESPRIT don't increase at the same rate. In fact, these two algorithms prove to have a better performance under high target proximity conditions over classical greedy algorithms.

4.1.3 Test Case: Noise Level 0.5

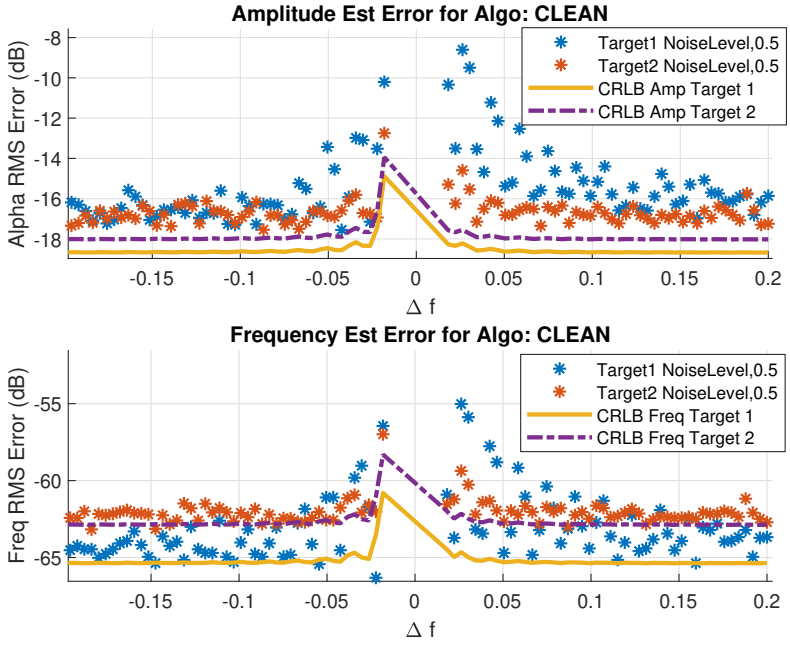


Figure 4.11: Amplitude and Frequency error CLEAN: Noise Level 0.5

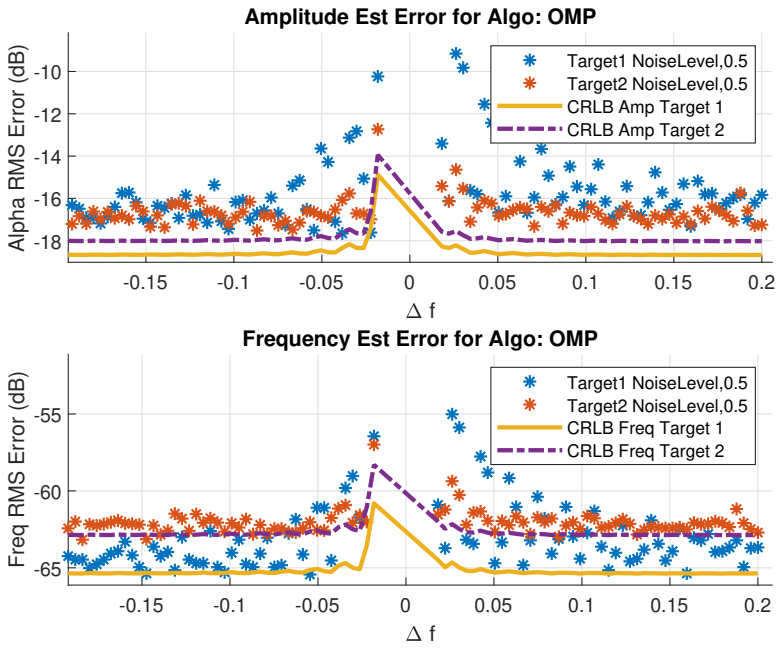


Figure 4.12: Amplitude and Frequency error OMP: Noise Level 0.5

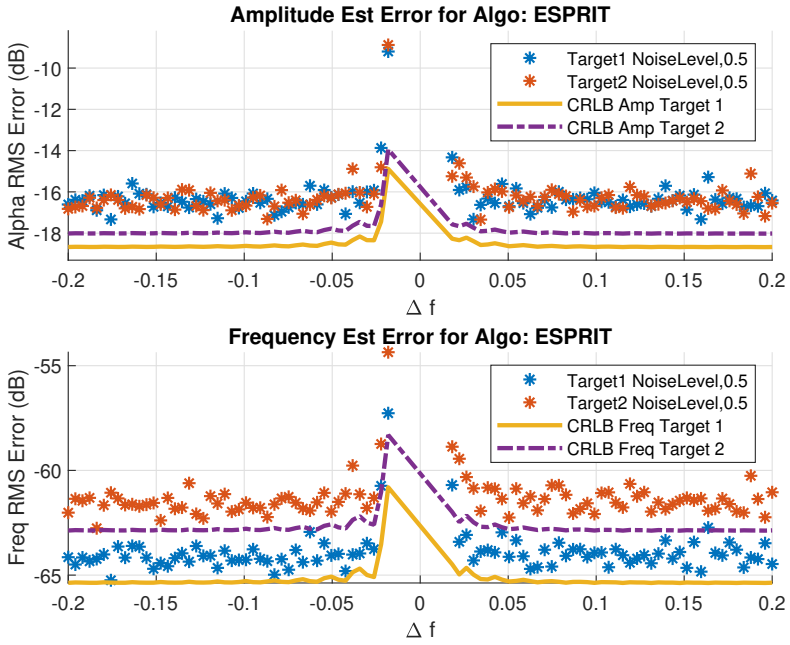


Figure 4.13: Amplitude and Frequency error ESPRIT: Noise Level 0.5

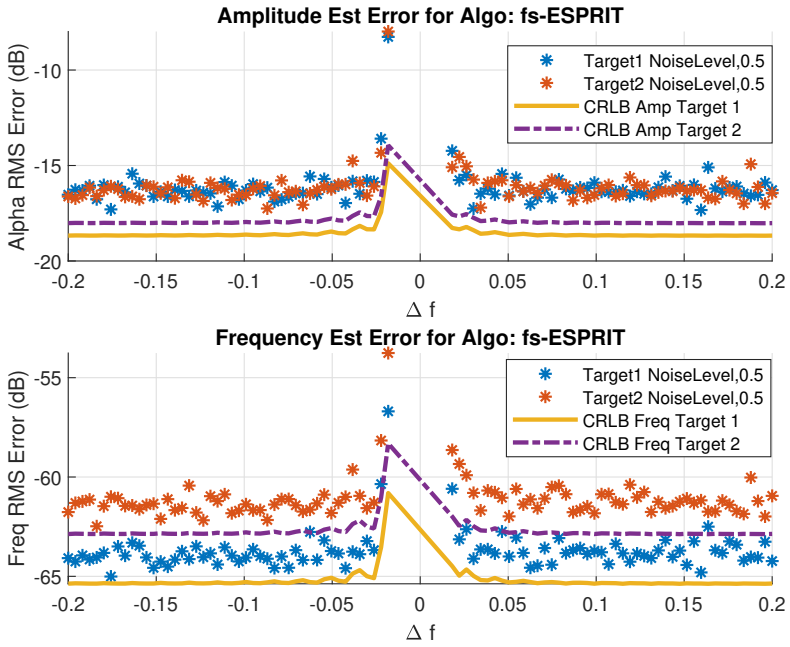


Figure 4.14: Amplitude and Frequency error fs-ESPRIT: Noise Level 0.5

4.1.4 Test Case: Noise Level 1

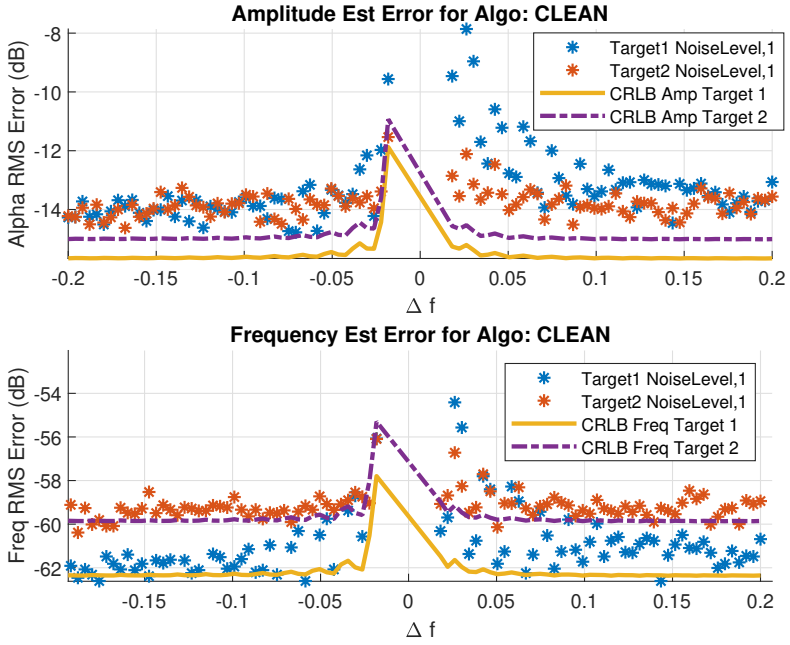


Figure 4.15: Amplitude and Frequency error CLEAN: Noise Level 1

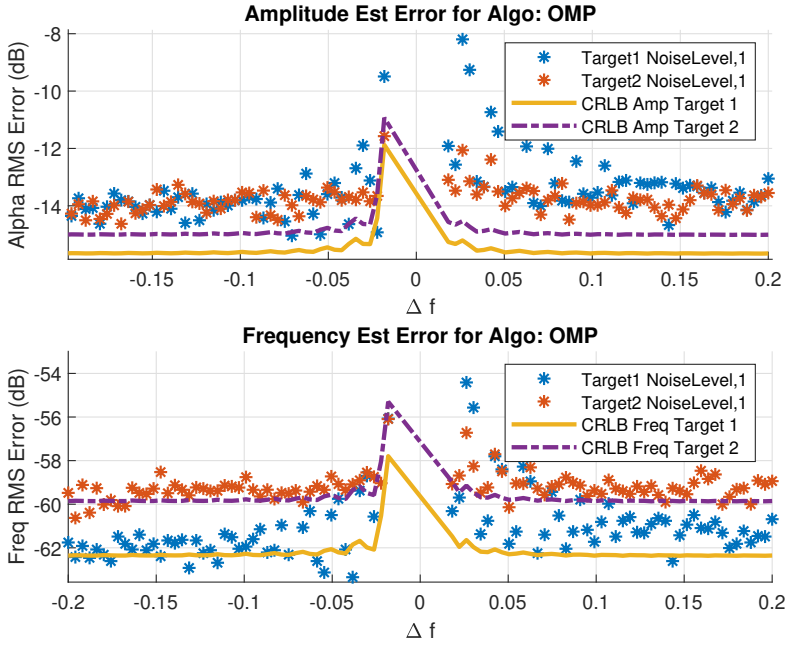


Figure 4.16: Amplitude and Frequency error OMP: Noise Level 1

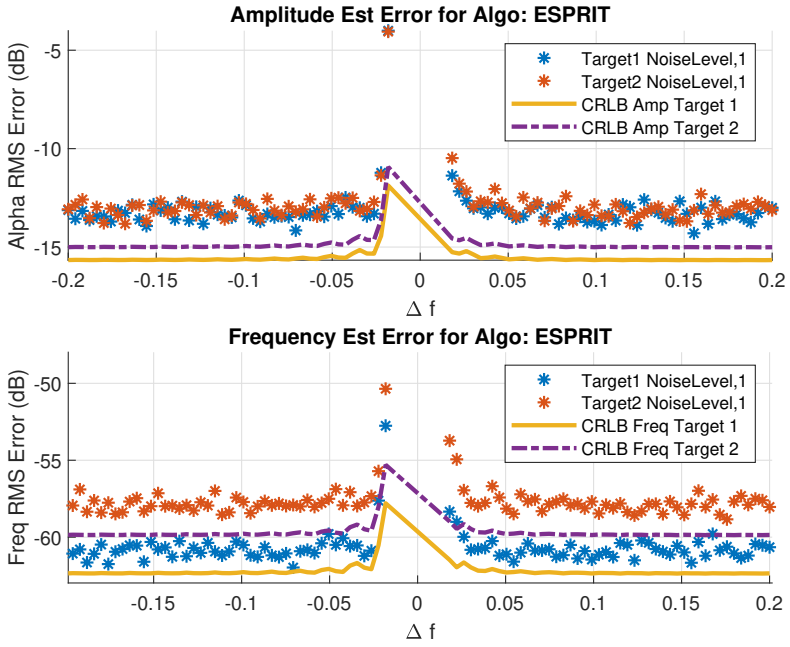


Figure 4.17: Amplitude and Frequency error ESPRIT: Noise Level 1

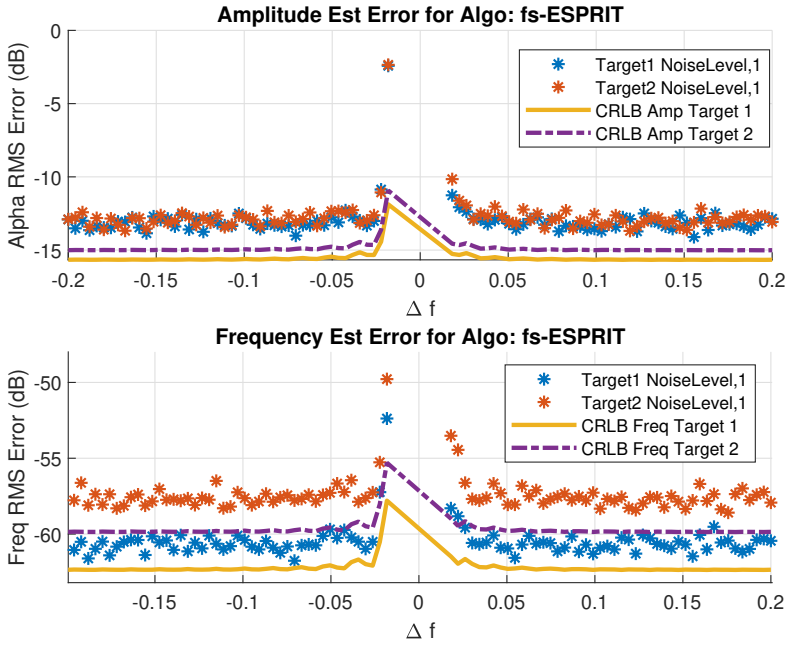


Figure 4.18: Amplitude and Frequency error fs-ESPRIT: Noise Level 1

With the increase in noise variance in signal, the errors for all algorithms are bound to increase too. But it can be concluded that especially under a situation where two targets are very close to each other, the amplitude errors for ESPRIT and fs-ESPRIT are minimum. While under normal conditions, the errors of all algorithms are comparable.

4.2 Test Scenario 2: Noise Variance

This test can be performed with any of the Data Scenarios mentioned in section 3.1.2.1 to 3.1.2.4. The noise variance, σ_v^2 is given in logspace from 10^{-3} to 10^0. Total of 25 noise level points are given in this simulation. Naturally, with increase in noise levels, the errors are expected to increase too. But the purpose of this test is to determine:

- Which algorithm is more sensitive to noise?
- Which algorithm performs best at lower noise levels?

This test is conducted for data scenario 0,2 and 3. Data scenarios 2 and 3 comprise of multiple targets. So, the RMS errors for each target are computed for given noise variance for 200 Monte Carlo simulations using (3.67) and (3.68) (for multiple targets).
These errors are then calculated and plotted for all 25 noise variance levels. The RMS erros in amplitude and frequency in dB scale for different algorithms are plotted on y axis and the logspace of noise levels in dB is plotted on x axis. The acceptable noise threshold (See Section 3.5.2) is also indicated on plots.

4.2.1 Test Case: Data Scenario 0

This test is performed on data corresponding to data scenario 0 (3.1.2.1), with a single target. The Cramer Rao lower bound is calculated according to 3.5.3.1 for single frequency. This test provides a simplistic platform to assess the performance of various algorithms. Therefore, ideal target parameters are chosen for this test, so that there is no effect on errors due to these factors. The target frequency is chosen to be on-grid (discussion in section 3.2.1.1). The target amplitude is 1 with zero phase. Tables 4.3, 4.4, 4.5 and 4.6 lists the parameters of test case and hyper-parameters of estimation algorithms selected for this test.

Parameter	**Symbol**	**Value**
Number of Targets	K	1
Target Frequency	f	0.6250
Target Amplitude	α	1 + i0
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^0$

Table 4.3: Parameters for Test Case

Parameter	Range
K	1
N_{FFT}	1024
f_s	0
f_e	1

Table 4.4: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	1
q	32
σ_v	1

Table 4.5: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	1
q	32
$kset$	$[1:64]$
σ_v	1

Table 4.6: Selected Hyper-parameters for fs-ESPRIT

For each noise variance, 200 Monte Carlo iterations are performed and results are recorded in figure 4.19

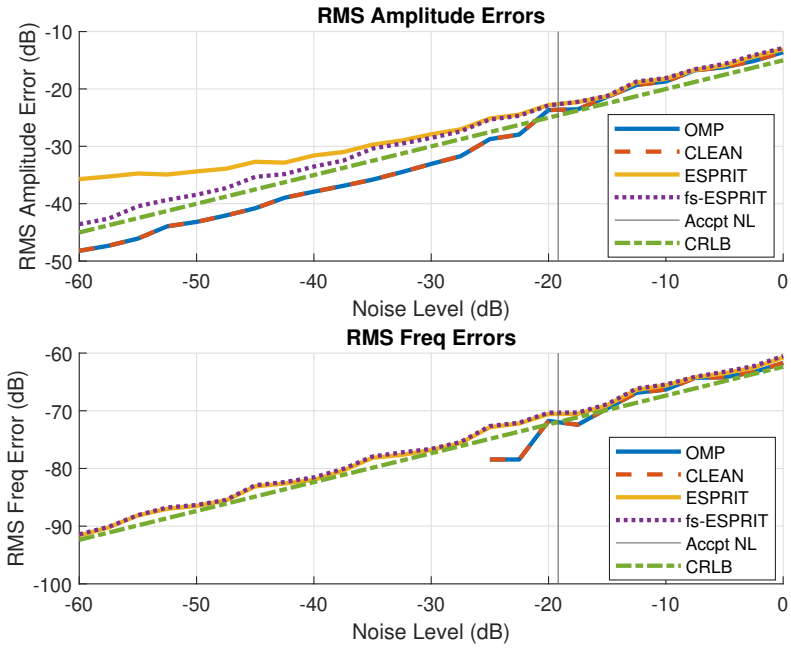


Figure 4.19: RMS Errors: DS0, frequency on-grid

4.2.1.1 Discussion: Effect of Target Frequency being on FFT grid

It is evident from these results that the RMS errors of CLEAN and OMP are lower than ESPRIT and fs-ESPRIT. The reason behind lower errors for OMP and CLEAN is that the target frequency is on the grid,i.e.

$$f * N_{FFT} = 0.6250 * 1024 = 640 \in \mathbb{Z}$$

These FFT based algorithms give very low errors if this is the case. It is already discussed in section 3.2.1.1, that increasing the zero-padding improves the algorithm efficiency, as the distance between target frequency and grid point decreases. So, in this case when target is already on the grid, both of these algorithms produce lower errors.
The test case is repeated for the frequency that is off the grid in order to verify this. The target frequency is now selected as 0.6 instead of 0.6250, which is not on the grid of $N_{FFT} = 1024$:

$$f * N_{FFT} = 0.6 * 1024 = 614.4 \notin \mathbb{Z}$$

The results for this test are shown in figure 4.20.

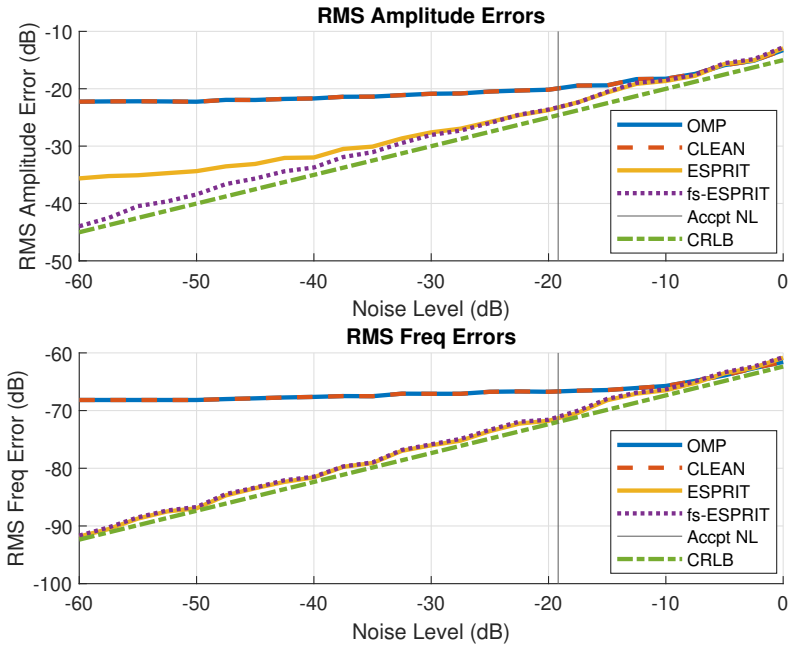


Figure 4.20: RMS Errors: DS0, frequency off-grid

The errors for ESPRIT and fs-ESPRIT remain the same, but errors for OMP and CLEAN have increased significantly.
Similarly, another simulation is performed, in which data is generated with single frequency which is randomly selected between $[0, 1)$ for each Monte Carlo iteration. The results are recorded in figure 4.21.

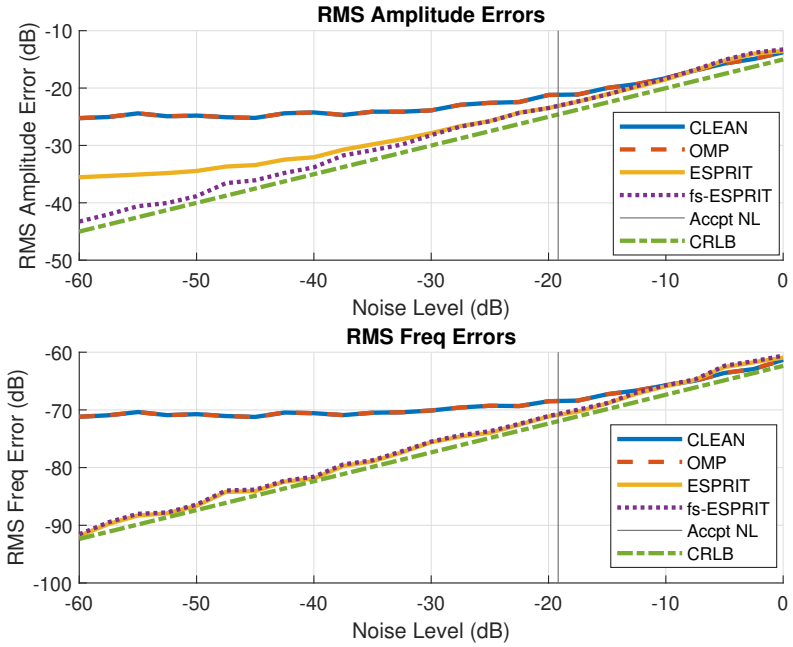


Figure 4.21: RMS Errors: DS0, Random frequency in each MC iteration

The errors for OMP and CLEAN are slightly reduced from the case when the frequency was always off the grid (4.20), but are still larger then ESPRIT and fs-ESPRIT.

This leads to important conclusions:

- The accuracy of OMP and CLEAN is highly dependent upon zero-padding or whether the target frequency is close to the FFT grid point or not.
- The performance of ESPRIT and fs-ESPRIT is independent of the frequency placement on grid.

4.2.2 Test Case: Data Scenario 2

In this test case Data scenario 2 is used (3.1.2.3). There are 5 frequencies that are not necessarily on-grid. The Cramer Rao lower bound is not calculated for this test case as the thesis only discusses the single and dual frequency case. The target frequencies are not closely placed on the grid and have decreasing amplitudes from first to last target. This scenario can be interpreted as a special case where frequency corresponds to target range and further the target is, lower is its amplitude in received signal. Tables 4.7, 4.8, 4.9 and 4.10 lists the parameters of test case and hyper-parameters of estimation algorithms selected for this test.

Parameter	Symbol	Value
Number of Targets	K	5
Target Frequency	f	0.1,0.2,0.5,0.6,0.8
Target Amplitude	α	0.7071 + 0.7071i, 0.0000 + 0.7500i, 0.3400 + 0.5889i, 0.0000 + 0.5200i, 0.3182 + 0.3182i
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^{0}$

Table 4.7: Parameters for Test Case

Parameter	Range
K	5
N_{FFT}	1024
f_s	0
f_e	1

Table 4.8: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	5
q	32
σ_v	1

Table 4.9: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	5
q	32
$kset$	$[1:64]$
σ_v	1

Table 4.10: Selected Hyper-parameters for fs-ESPRIT

The results are shown in figure 4.23.

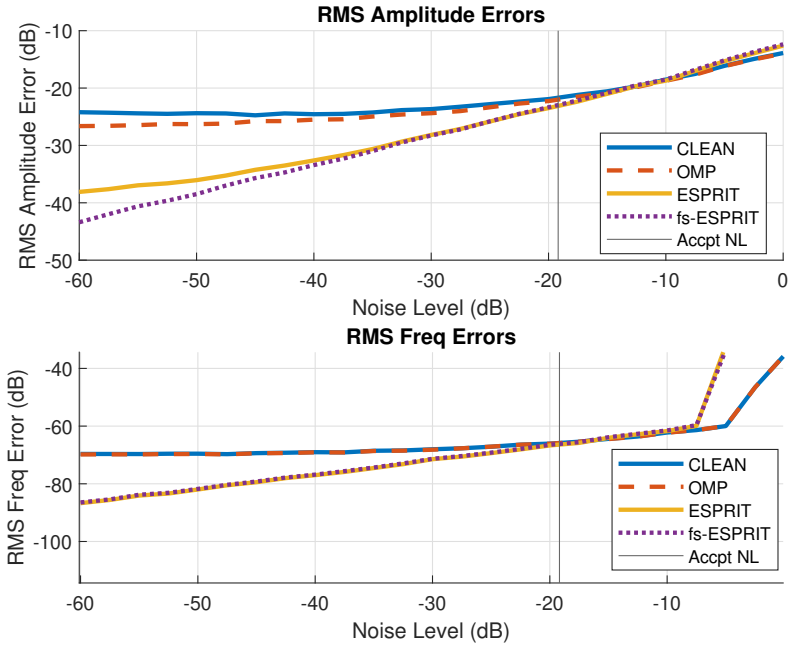


Figure 4.22: RMS Errors: DS2

From the results, it can be seen that both ESPRIT and fs-ESPRIT give better performance than OMP and CLEAN in this case. However, the errors for ESPRIT and fs-ESPRIT slightly increase above the acceptable noise level.

4.2.3 Test Case: Data Scenario 3

In this test case Data scenario 3 is used (3.1.2.4). In this scenario, there are 5 frequencies that are not necessarily on-grid but are placed very close to each other. This scenario can be interpreted as a special case where multiple targets are flying in formation. They might be located very close to each other in range and Doppler. This is a very difficult but a practical scenario. All the estimation algorithms are expected to deviate from their ideal performance. There might be misdetections or two targets might be detected as one. The test doesn't evaluate the details of each Monte Carlo iteration to check which of five targets were detected correctly, but provides an overall picture of how these algorithms perform in such a tough scenario. The RMS errors for each noise variance is calculated using the same technique as in Data Scenario 2, using (3.67) and (3.68). Moreover, the hyperparameter σ_v in fs-ESPRIT is changed to 8 in order to reduce errors for closely spaced targets as discussed in 4.1.1. Tables 4.11, 4.12, 4.13 and 4.14 lists the parameters of test case and hyper-parameters of estimation algorithms selected for this test.

Parameter	Symbol	Value
Number of Targets	K	5
Target Frequency	f	0.1,0.11,0.131,0.170,0.2
Target Amplitude	α	0.7071 + 0.7071i, 0.0000 + 0.7500i, 0.3400 + 0.5889i, 0.0000 + 0.5200i, 0.3182 + 0.3182i
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^{0}$

Table 4.11: Parameters for Test Case

Parameter	Range
K	5
N_{FFT}	1024
f_s	0
f_e	1

Table 4.12: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	5
q	32
σ_v	1

Table 4.13: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	5
q	32
$kset$	$[1:64]$
σ_v	8

Table 4.14: Selected Hyper-parameters for fs-ESPRIT

The results are shown in figure 4.23.

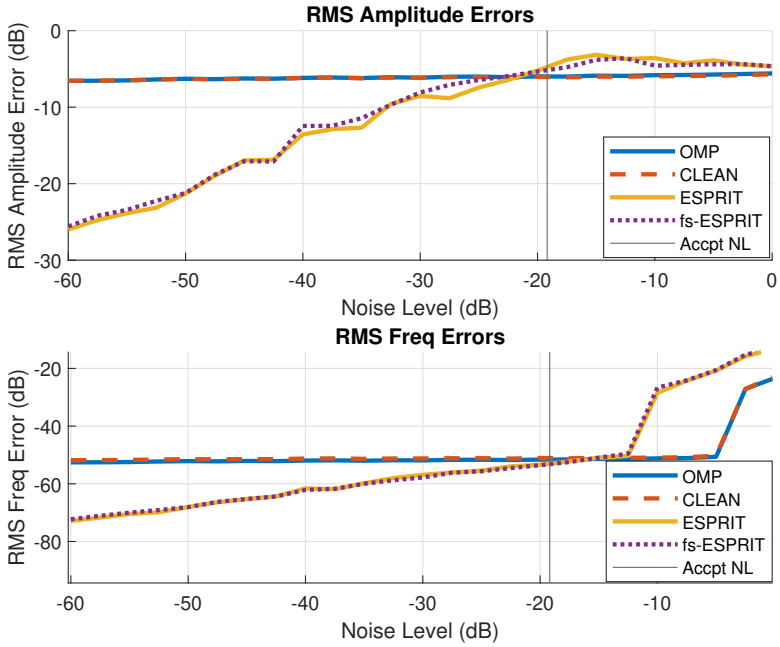


Figure 4.23: RMS Errors: DS3

The errors of OMP and CLEAN are comparable and similarly the errors of ESPRIT and fs-ESPRIT are comparable to each other, but below the acceptable noise level, fs-ESPRIT and ESPRIT have much better performance than OMP and CLEAN. However, for noise variance above the acceptable noise level, errors of ESPRIT and fs-ESPRIT increase.

4.3 Test Scenario 3: Signal Length

One of the most contributing factors affecting the algorithm's performance is the signal length. The error is expected to decrease with longer signal lengths. This is because when signal length increases, resolution also increases which in turn results in better detection of two closely spaced targets.
The errors are calculated in a similar fashion as described in section 4.2. The only difference is that for this test case, noise level, σ_v^2, is fixed and sweep is given along the increasing signal length. Total of 10 signal lengths are used increasing linearly from 64 to 1000. The zero padding for DFT is 2000 instead of 1024. The reason for increasing the zero padding is explained in section 3.2.1.1. As the signal length increases, a higher resolution will be needed on DFT grid to minimize errors due to grid in case of OMP and CLEAN.
Moreover, each test case will be tested for at least two different noise variance levels, σ_v^2, one preferably close to zero and other some higher value. The RMS errors for frequency and amplitudes for different algorithms are plotted on y axis in dB scale while signal length is plotted on x-axis.

4.3.1 Test Case: Data Scenario 0

This test is performed on data corresponding to data scenario 0 (3.1.2.1), with a single target. The Cramer Rao lower bound is calculated according to 3.5.3.1 for single frequency. This is the most simple test case with a single target of frequency on DFT grid, amplitude of maximum value, and phase of zero. Tables 4.15, 4.16, 4.17 and 4.18 lists the parameters of test case and hyper-parameters of estimation algorithms selected for this test.

Parameter	**Symbol**	**Value**
Number of Targets	K	1
Target Frequency	f	0.6250
Target Amplitude	α	1 + i0
Signal Length	N	$\in [64, 168, 272, 376, 480, 584, 688, 792, 896, 1000]$
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$\in [10^{-3}, 0.5]$

Table 4.15: Parameters for Test Case

Parameter	**Range**
K	1
N_{FFT}	2000
f_s	0
f_e	1

Table 4.16: Selected Hyper-parameters for CLEAN & OMP

Parameter	**Range**
$order$	1
q	$\frac{N}{2}$
σ_v	1

Table 4.17: Selected Hyper-parameters for ESPRIT

Parameter	**Range**
$order$	1
q	$\frac{N}{2}$
$kset$	$[1 : N]$
σ_v	1

Table 4.18: Selected Hyper-parameters for fs-ESPRIT

For each noise variance, $\sigma_v^2 = 10^{-3}$, 200 Monte Carlo iterations are performed and results are recorded in figure 4.24

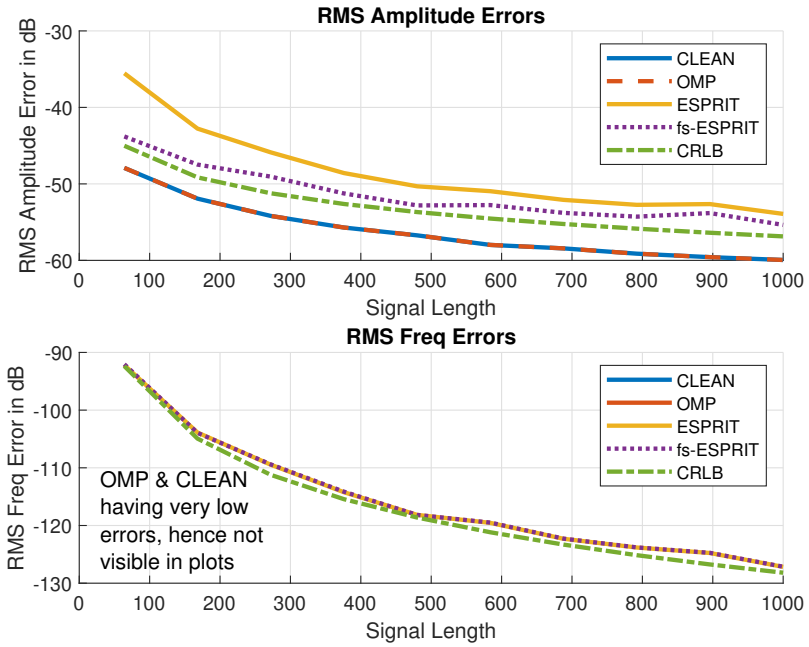


Figure 4.24: RMS Errors: DS0, $\sigma_v^2 = 10^{-3}$

Similarly, the results for $\sigma_v^2 = 0.5$, are presented in figure 4.25

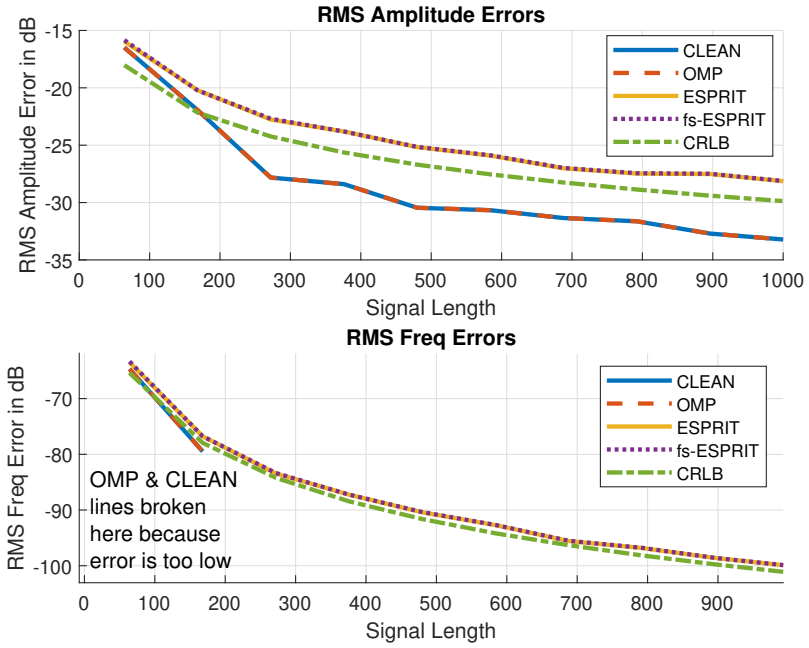


Figure 4.25: RMS Errors: DS0, $\sigma_v^2 = 0.5$

The results show that for noise variance close to zero, fs-ESPRIT has lower errors as

compared to ESPRIT. But OMP and CLEAN produce the lowest errors, the reason behind this is increased zero-padding (2000 instead of 1024). The frequency, 0.6250, is still on the DFT grid ($0.6250 * 2000 = 1250 \in \mathbb{Z}$). Therefore, the RMS errors are minimal, and the RMS frequency errors are so low that they are not even plotted on the dB scale. For higher noise variance, the overall errors for each algorithm is definitely increased, but OMP and CLEAN still prodcue lowest errors, whilst the errors of ESPRIT and fs-ESPRIT are comparable.
The RMS errors for each case and all algorithms decrease as signal length increases, as expected.

4.3.2 Test Case: Data Scenario 2

This test is performed on data corresponding to data scenario 2 (3.1.2.3), with a five targets. This is the test case with a multiple frequencies placed well apart on the DFT grid. Tables 4.19, 4.20, 4.21 and 4.22 lists the parameters of test case and hyper-parameters of estimation algorithms selected for this test.

Parameter	**Symbol**	**Value**
Number of Targets	K	5
Target Frequency	f	0.1,0.2,0.5,0.6,0.8
Target Amplitude	α	0.7071 + 0.7071i, 0.0000 + 0.7500i, 0.3400 + 0.5889i, 0.0000 + 0.5200i, 0.3182 + 0.3182i
Signal Length	N	$\in [64, 168, 272, 376, 480, 584, 688, 792, 896, 1000]$
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$\in [10^{-3}, 0.5]$

Table 4.19: Parameters for Test Case

Parameter	**Range**
K	1
N_{FFT}	2000
f_s	0
f_e	1

Table 4.20: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	1
q	$\frac{N}{2}$
σ_v	1

Table 4.21: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	1
q	$\frac{N}{2}$
$kset$	$[1:N]$
σ_v	1

Table 4.22: Selected Hyper-parameters for fs-ESPRIT

For each noise variance, $\sigma_v^2 = 10^{-3}$, 200 Monte Carlo iterations are performed and results are recorded in figure 4.26

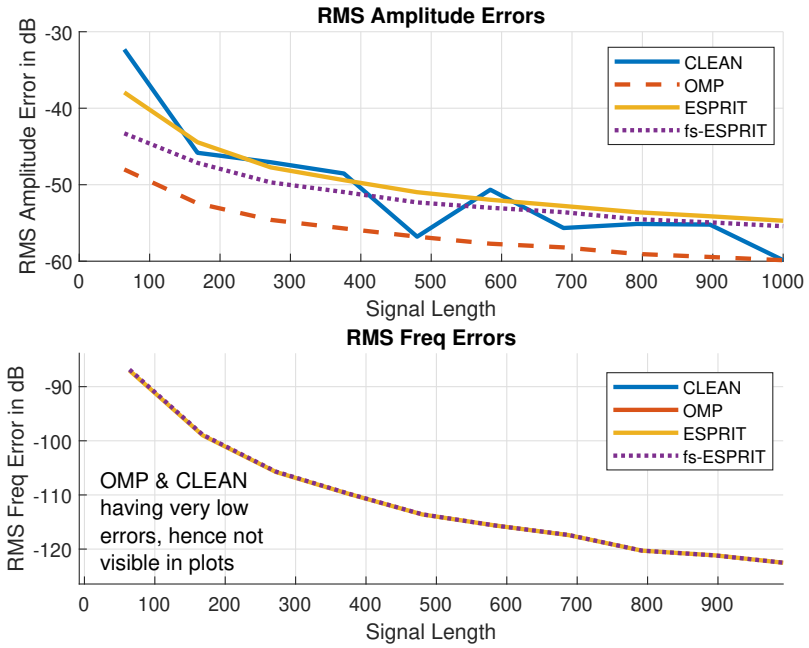


Figure 4.26: RMS Errors: DS2, $\sigma_v^2 = 10^{-3}$

Similarly, the results for $\sigma_v^2 = 0.5$, are presented in figure 4.27

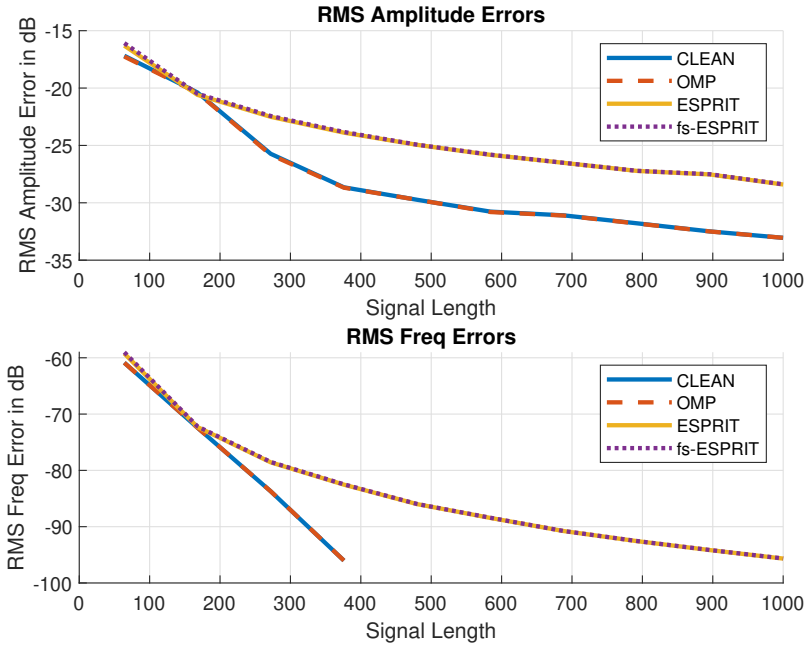


Figure 4.27: RMS Errors: DS2, $\sigma_v^2 = 0.5$

The RMS errors for all algorithms decrease as signal length increases. For noise variance close to zero, this test case has almost similar response as with data scenario 0. The RMS amplitude errors for fs-ESPRIT are lower then that of ESPRIT, and OMP has lowest errors but here it can be seen that CLEAN produce higher errors then that of OMP. For higher noise variance, RMS errors of fs-ESPRIT and ESPRIT are higher and similar, whilst the errors of OMP and CLEAN are lower and similar.

4.3.3 Test Case: Data Scenario 3

This test is performed on data corresponding to data scenario 3 (3.1.2.4), with five targets. This is the test case with multiple frequencies placed very close to each other on the DFT grid. The purpose of this test is to observe the difference in errors when signal length is increased. The errors are expected to decrease as the distance between two close frequencies increases as the signal length increases. Now, the probability of detecting two targets as one also decreases. Moreover, for this particular case, σ_v in fs-ESPRIT needs to be increased to 8 in order to reduce errors for closely spaced targets as discussed in 4.1.1. Tables 4.23, 4.24, 4.25 and 4.26 lists the parameters of test case and hyper-parameters of estimation algorithms selected for this test.

Parameter	Symbol	Value
Number of Targets	K	5
Target Frequency	f	0.1,0.11,0.131,0.170,0.2
Target Amplitude	α	0.7071 + 0.7071i, 0.0000 + 0.7500i, 0.3400 + 0.5889i, 0.0000 + 0.5200i, 0.3182 + 0.3182i
Signal Length	N	$\in [64, 168, 272, 376, 480, 584, 688, 792, 896, 1000]$
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$\in [10^{-3}, 0.5]$

Table 4.23: Parameters for Test Case

Parameter	Range
K	1
N_{FFT}	2000
f_s	0
f_e	1

Table 4.24: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	1
q	$\frac{N}{2}$
σ_v	1

Table 4.25: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	1
q	$\frac{N}{2}$
$kset$	$[1 : N]$
σ_v	8

Table 4.26: Selected Hyper-parameters for fs-ESPRIT

For each noise variance, $\sigma_v^2 = 10^{-3}$, 200 Monte Carlo iterations are performed and results are recorded in figure 4.28

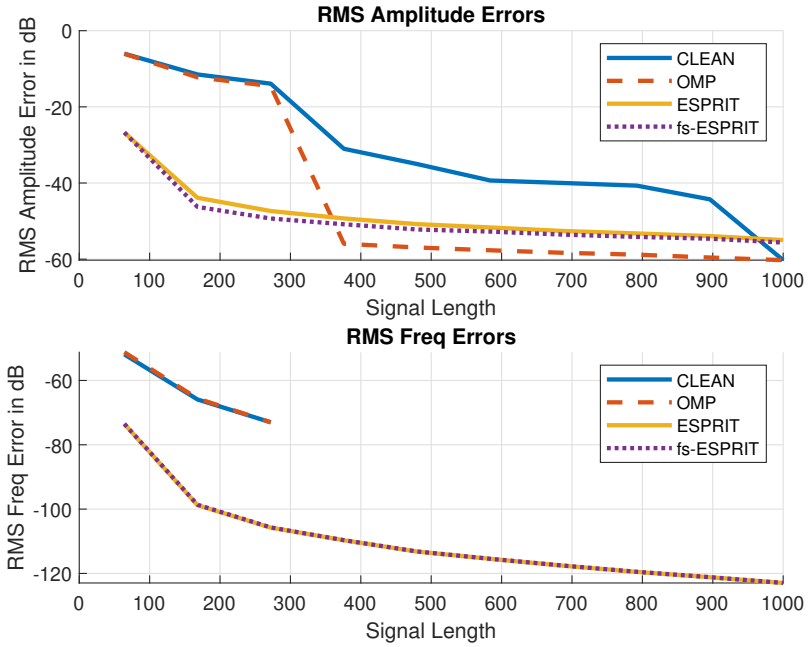


Figure 4.28: RMS Errors: DS3, $\sigma_v^2 = 10^{-3}$

It is evident from results that for lower noise variance, there is a significant difference in errors for CLEAN and OMP, with OMP producing the lowest Amplitude RMS errors for longer signal lengths. For smaller signal lengths, however, the RMS amplitude and frequency errors for ESPRIT and fs-ESPRIT are much lower then that of OMP and CLEAN. This means, that for lower noise levels, and for a case where targets are placed very close to each other, ESPRIT and fs-ESPRIT can give much better performance for smaller signal lengths. On the other hand, much higher signal lengths and therefore more computing power is needed for OMP and CLEAN to give similar results.
The results for $\sigma_v^2 = 0.5$, are presented in figure 4.29

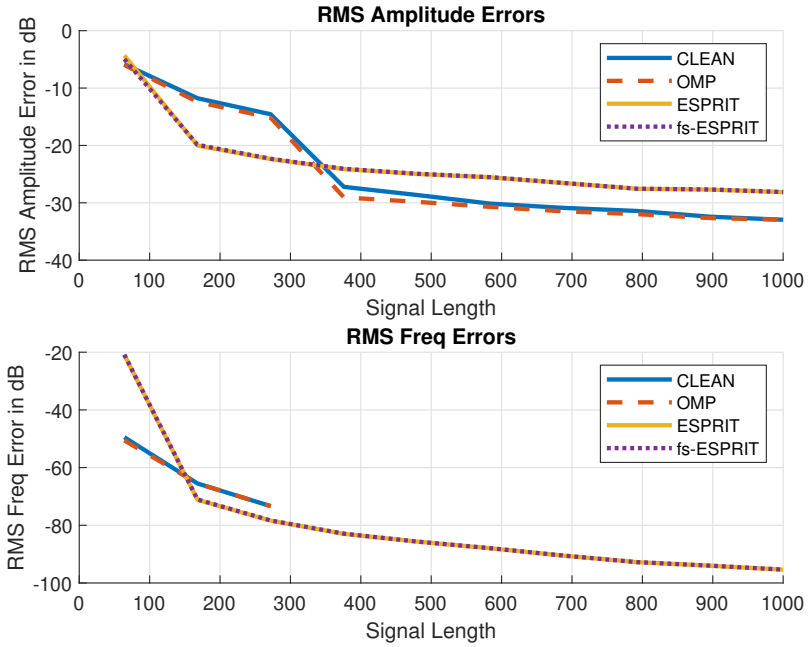


Figure 4.29: RMS Errors: DS3, $\sigma_v^2 = 0.5$

For higher noise levels, the amplitude RMS errors for ESPRIT and fs-ESPRIT are again lower for smaller signal lengths, and the RMS errors for CLEAN and OMP are reduced after N crosses 350. This again leads to same conclusion that it is better to use ESPRIT and fs-ESPRIT if the signal length is limited due to constraints on computation power available.

4.4 Test Scenario 4: Clutter

As already discussed in chapter 2, clutter poses a significant effect on a radar system overall performance. In this section several test cases will be formulated to rigorously test estimation algorithms under mild and heavy clutter. The clutter assumed in these test cases can be linked to ground/*surface* clutter in practical scenarios. *Volume* and *point* clutters are not discussed here. The Cramer Rao Lower Bound for this case is more involved and is not considered.
The clutter range and power is user-configurable. Different data scenarios can be tested under different clutter spreads and strengths. The RMS errors are calculated in the same way as mentioned in (3.67). The purpose of these tests is to evaluate the performance of estimation algorithms under mild and severe clutter. The following three tests will be conducted in this experiment:

1. Single frequency (Data Scenario 0) Clutter spread 5 %, clutter to signal ratio is 1
2. Single frequency (Data Scenario 0) Clutter spread 10 %, clutter to signal ratio is 3
3. Single frequency (Data Scenario 0) Clutter spread 15 %, clutter to signal ratio is 6

4. Multiple frequencies (Data Scenario 2) Clutter spread 5 %, clutter to signal ratio is 1. One of the target lies close to clutter.

4.4.1 Test Case: Single Frequency, clutter Spread 5%, clutter to signal ratio 1

For this test case, Data Scenario 0 (3.1.2.1) is used. The purpose of using this simple data is to observe the effects of clutter on an otherwise perfectly placed single target, with frequency in almost middle of DFT grid and maximum amplitude. In this way it is easier to single out changes in errors caused by clutter alone with no other contributing factors in errors. The clutter is generated as discussed in 3.4.1. The parameters used for clutter generation are mentioned in table 4.27.

Parameter	Range
C	40
f_c	$\in [0, 0.05] \cup [0.95, 1]$
a_c	$\in CN(0, 1)$
σ_c^2	200

Table 4.27: Parameters for clutter generation

Here the clutter power, $\sigma_c^2 = 200$ corresponds to the clutter to signal ratio of 1. This can be seen in the figure 4.30.

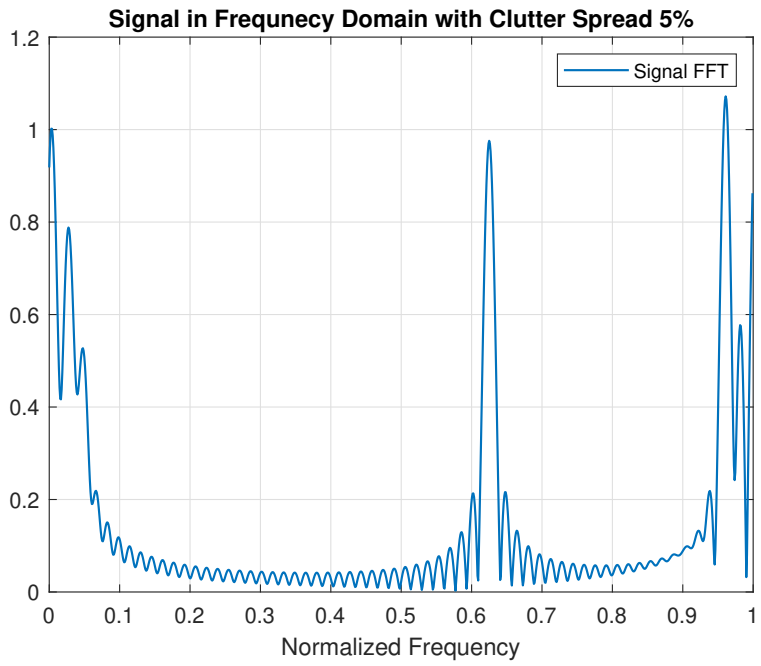


Figure 4.30: Signal in Clutter with 5% spread

The parameters of the test case and that of all the algorithms are listed in tables 4.28, 4.29, 4.30 and 4.31. The hyper-parameters f_s, f_e for CLEAN and OMP, *order*

(ESPRIT) and *kset* (fs-ESPRIT) are set according to the clutter range, f_c mentioned in table 4.27.

Parameter	Symbol	Value
Number of Targets	K	1
Target Frequency	f	0.625
Target Amplitude	α	1 + 0i,
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^0$

Table 4.28: Parameters for Test Case

Parameter	Range
K	1
N_{FFT}	1024
f_s	0.0752
f_e	0.9248

Table 4.29: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	5
q	32
σ_v	1

Table 4.30: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	1
q	32
$kset$	[5 : 60]
σ_v	1

Table 4.31: Selected Hyper-parameters for fs-ESPRIT

The results for 200 Monte Carlo iterations is shown in figure 4.31.

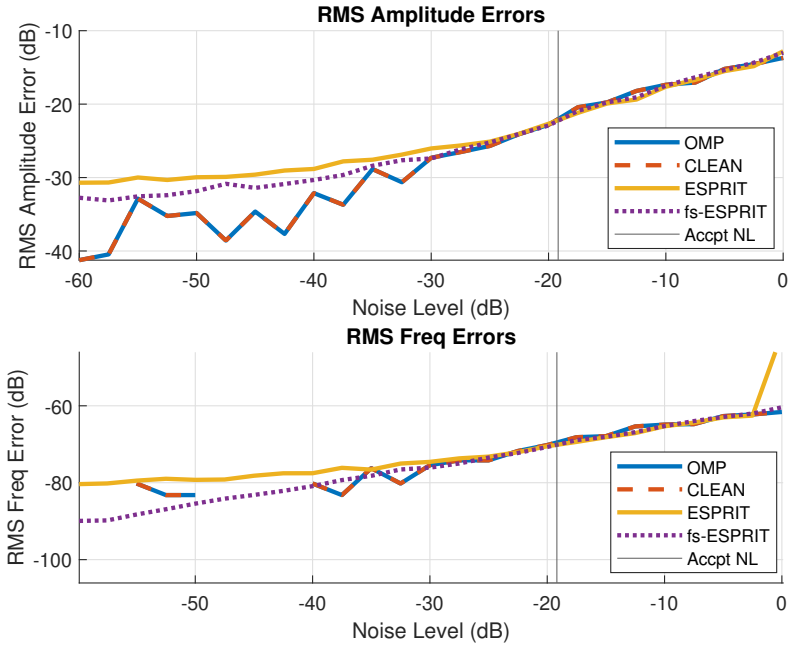


Figure 4.31: RMS Errors Clutter Test Case 1

The results indicate that OMP and CLEAN perform better in this scenario as this was the case in 4.2.1. But the errors when compared to that test case (figure 4.19) are higher and more wobbly. By making this comparison the errors due to clutter can be singled out. When compared to results (4.19) of test case 4.2.1, it can be seen that RMS errors for ESPRIT and fs-ESPRIT have also increased. The noticeable difference here is, however, that RMS frequency errors for fs-ESPRIT are now lower compared to that for ESPRIT. The test is also repeated for 500 Monte Carlo iterations in order to get rid of wobbly errors as in 4.31.

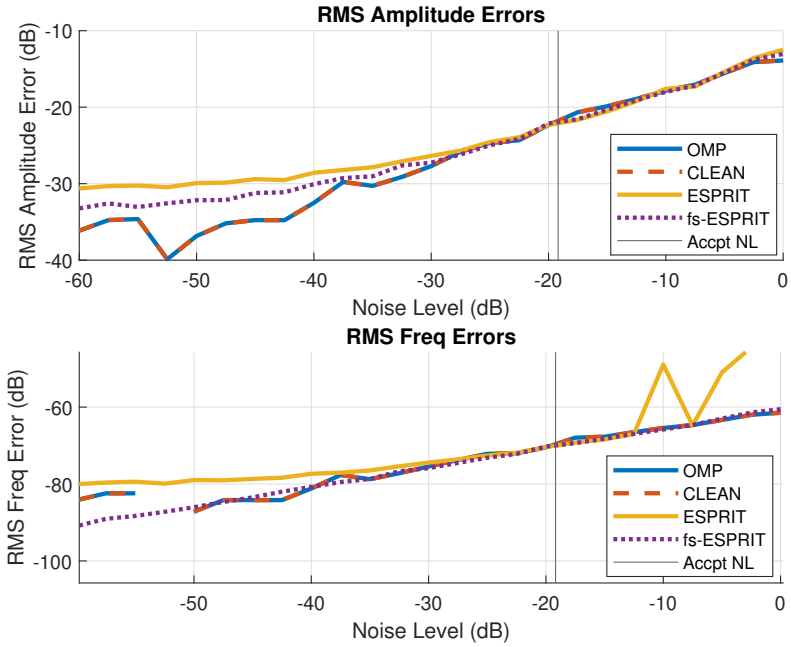


Figure 4.32: RMS Errors Clutter Test Case 1: $N_{MC} = 500$

4.4.2 Test Case: Single Frequency, clutter Spread 10%, clutter to signal ratio 3

Data Scenario 0 (3.1.2.1) is used for this test case too. This is the case of slightly stronger clutter with a large spread. It is indeed quite challenging to recover actual targets in presence of this much clutter. As the clutter spread increases, the probability of it masking the valid targets also increases. In this test case, however, the single target is still in a clutter free frequency zone. The clutter is generated as discussed in 3.4.1. The parameters used for clutter generation are mentioned in table 4.32.

Parameter	Range
C	40
f_c	$\in [0, 0.10] \cup [0.9, 1]$
a_c	$\in CN(0, 1)$
σ_c^2	3500

Table 4.32: Parameters for clutter generation

Here the clutter power, $\sigma_c^2 = 3500$ corresponds to the clutter to signal ratio of 3. This can be seen in the figure 4.33.

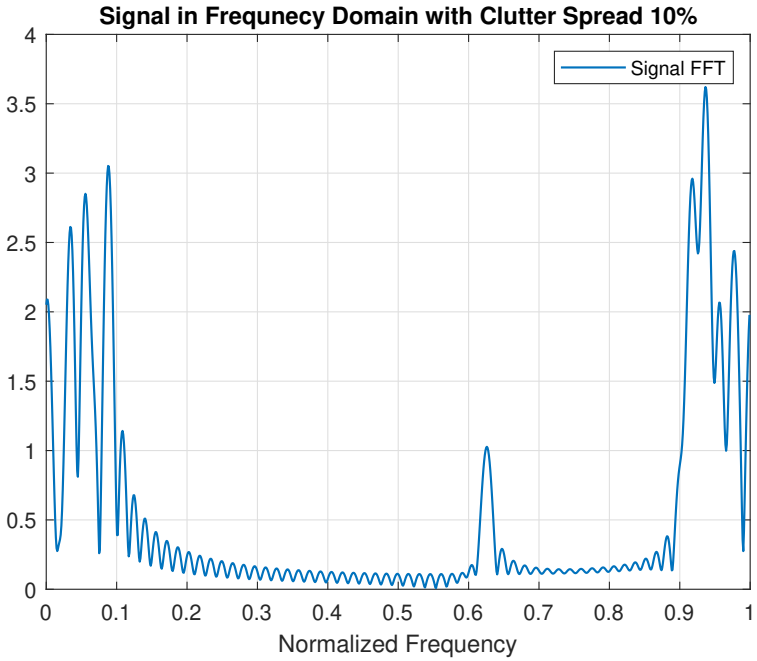


Figure 4.33: Signal in Clutter with 10% spread

The parameters of the test case and that of all the algorithms are listed in tables 4.33, 4.34, 4.35 and 4.36.

Parameter	Symbol	Value
Number of Targets	K	1
Target Frequency	f	0.625
Target Amplitude	α	1 + 0i,
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^{0}$

Table 4.33: Parameters for Test Case

Parameter	Range
K	1
N_{FFT}	1024
f_s	0.1504
f_e	0.8496

Table 4.34: Selected Hyper-parameters for CLEAN & OMP

The results for 200 Monte Carlo iterations is shown in figure 4.34.

Parameter	Range
$order$	5
q	32
σ_v	1

Table 4.35: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	1
q	32
$kset$	$[10:55]$
σ_v	1

Table 4.36: Selected Hyper-parameters for fs-ESPRIT

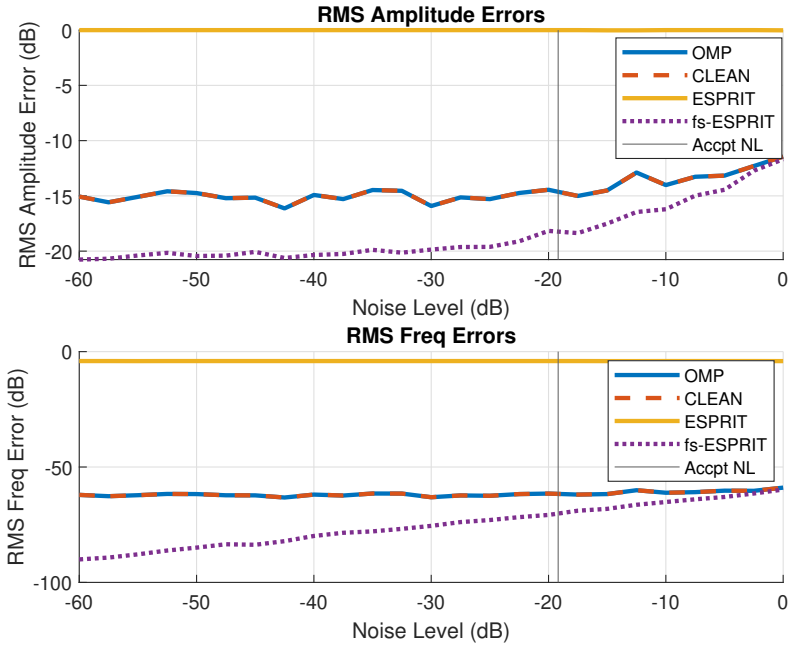


Figure 4.34: RMS Errors Clutter Test Case 2

These results indicate that ESPRIT produces highest RMS errors while fs-ESPRIT produces the lowest. The effect of parameter *order* on performance of ESPRIT is discussed in section 4.4.2.1.

4.4.2.1 Discussion: Effect of *order* on ESPRIT

For ideal cases, with no clutter, the parameter *order* should be set equal to total number of targets, K. But in case of cluttered signal, the order of this algorithm needs to be increased, so that additional targets can be detected in clutter zone, and later can be removed by post processing. But increasing the order too much might also lead to larger errors. So, an exact number is needed to be determined for which

the RMS errors are the lowest.
In order to determine correct *order*, a separate simulation was developed where RMS frequency error and RMS amplitude error for ESPRIT were plotted against the *order* of the algorithm. The order of the algorithms was increased from 1 to 20. The result is shown in figure 4.35

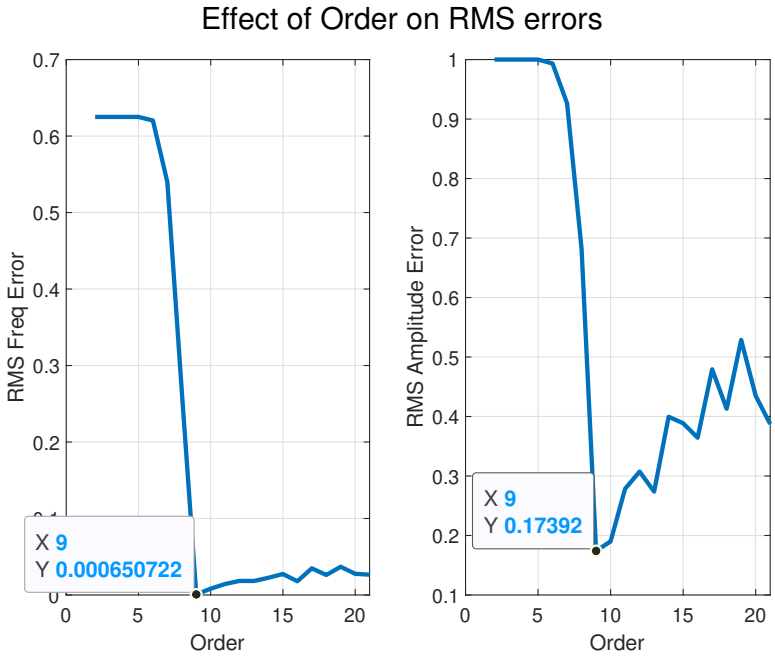


Figure 4.35: Effect of order on RMS Errors

It means that for this particular test case, the *order* needs to be 9, i.e. eight additional frequencies should be demanded from the algorithm. It can also be noted that errors increase again for values greater than 9. The test case is repeated with following parameters for ESPRIT.

Parameter	Range
$order$	9
q	32
σ_v	1

Table 4.37: Revised Hyper-parameters for ESPRIT

And the results are shown in figure 4.36.

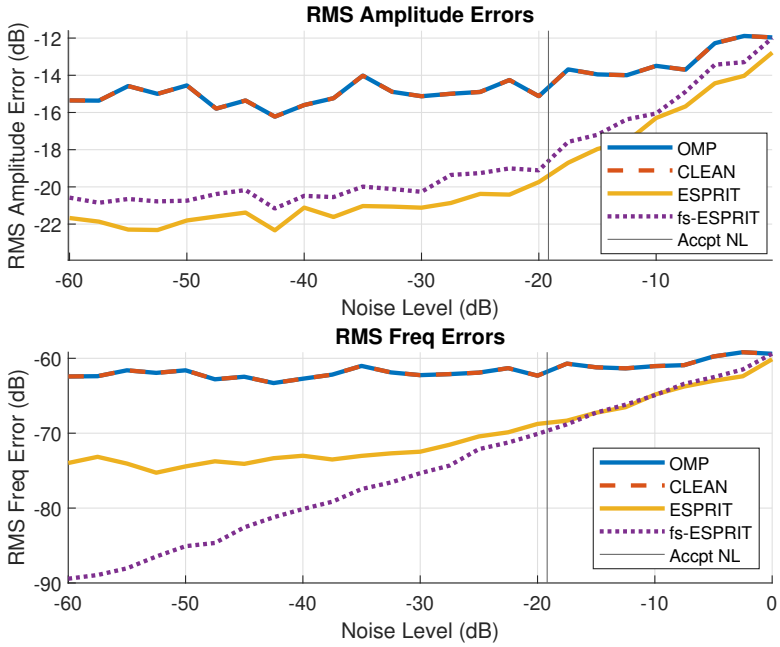


Figure 4.36: RMS Errors Clutter Test Case 2 (Revised)

4.4.3 Test Case: Single Frequency, clutter Spread 15%, clutter to signal ratio 6

This test case involves very high power clutter with huge spread. The Data Scenario 0 (3.1.2.1) is used for this test case too. The clutter is generated as discussed in 3.4.1. The parameters used for clutter generation are mentioned in table 4.38.

Parameter	Range
C	40
f_c	$\in [0, 0.15] \cup [0.85, 1]$
a_c	$\in CN(0, 1)$
σ_c^2	15000

Table 4.38: Parameters for clutter generation

Here the clutter power, $\sigma_c^2 = 15000$ corresponds to the clutter to signal ratio of 6. This can be seen in the figure 4.37.

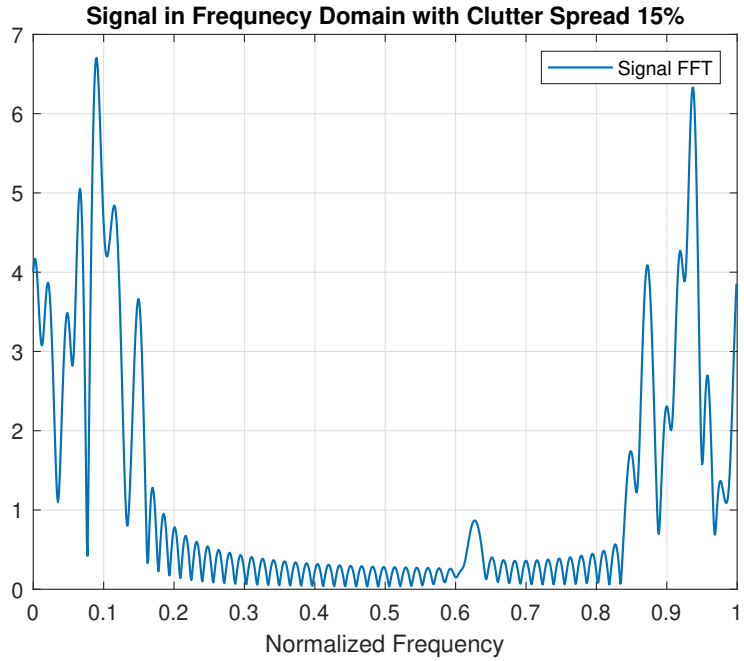


Figure 4.37: Signal in Clutter with 15% spread

The parameters of the test case and that of all the algorithms are listed in tables 4.39, 4.40, 4.41 and 4.42. As discussed in previous section, the order of ESPRIT is set to be 9.

Parameter	Symbol	Value
Number of Targets	K	1
Target Frequency	f	0.625
Target Amplitude	α	1 + 0i,
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^{0}$

Table 4.39: Parameters for Test Case

Parameter	Range
K	1
N_{FFT}	1024
f_s	0.2246
f_e	0.7754

Table 4.40: Selected Hyper-parameters for CLEAN & OMP

The results for 200 Monte Carlo iterations is shown in figure 4.38.

Parameter	Range
$order$	9
q	32
σ_v	1

Table 4.41: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	1
q	32
$kset$	$[15:50]$
σ_v	1

Table 4.42: Selected Hyper-parameters for fs-ESPRIT

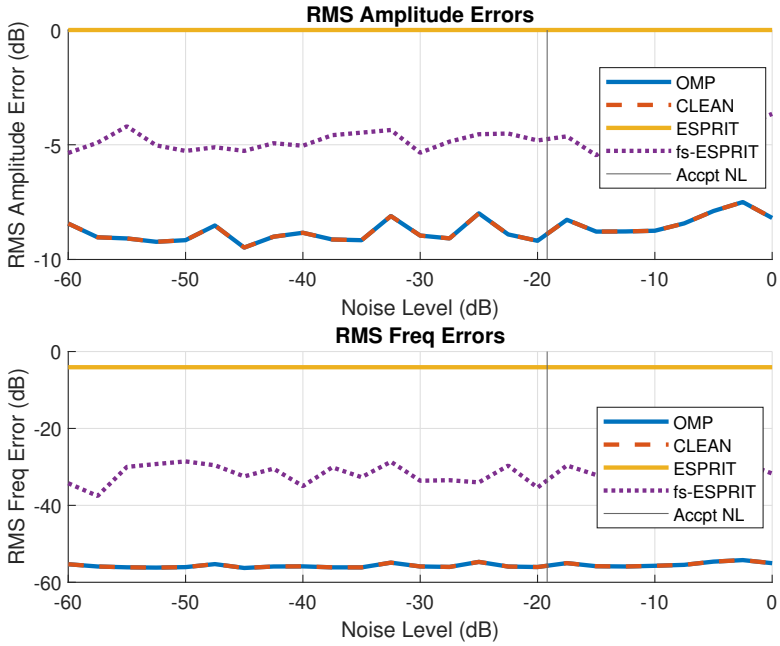


Figure 4.38: RMS Errors Clutter Test Case 3

These test results indicate that ESPRIT produces highest RMS errors, while OMP and CLEAN produce lowest. The errors for fs-ESPRIT increase as length of $kset$ decreases.
The parameter *order* for ESPRIT can be determined by repeating the simulation mentioned in 4.4.2.1. The results of this simulation are shown in figure 4.39.

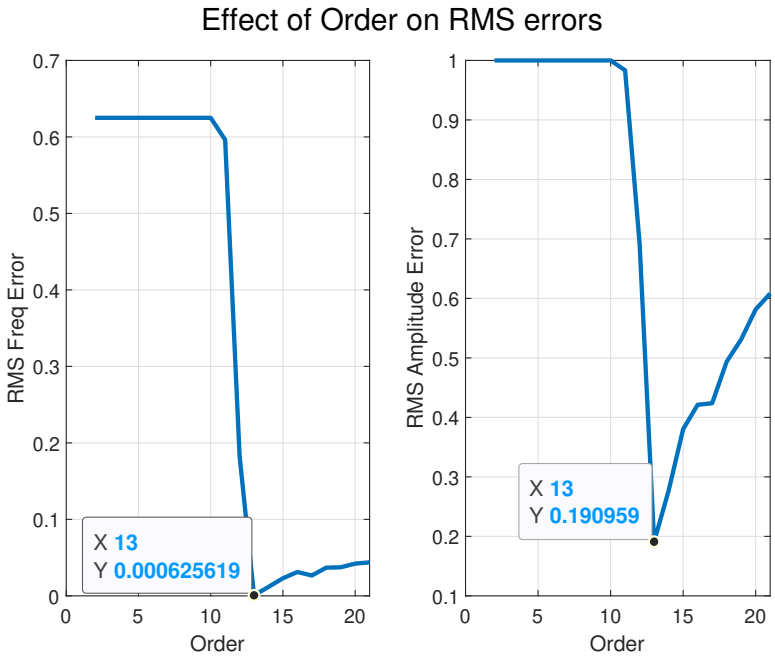


Figure 4.39: RMS Errors Clutter Test Case 3

By setting the hyper-parameter $order = 13$ for ESPRIT in this test case, following RMS errors are recorded.

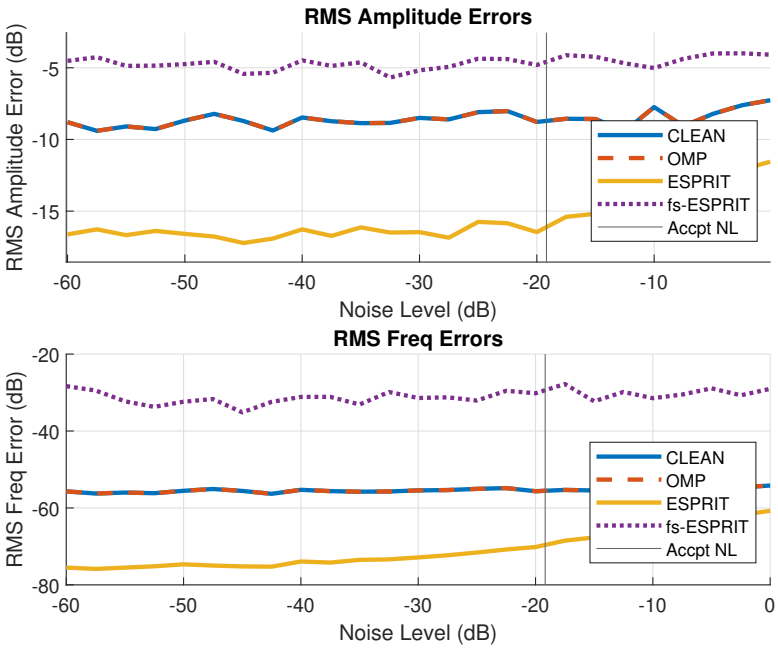


Figure 4.40: RMS Errors Clutter Test Case 3 (Revised)

The errors for ESPRIT are significantly reduced in this case. On the other hand, the RMS errors for fs-ESPRIT are still higher than all algorithms. Another simulation

was done, where the hyper-parameter *kset* was optimized. The results for this simulation are shown in figure 4.41.

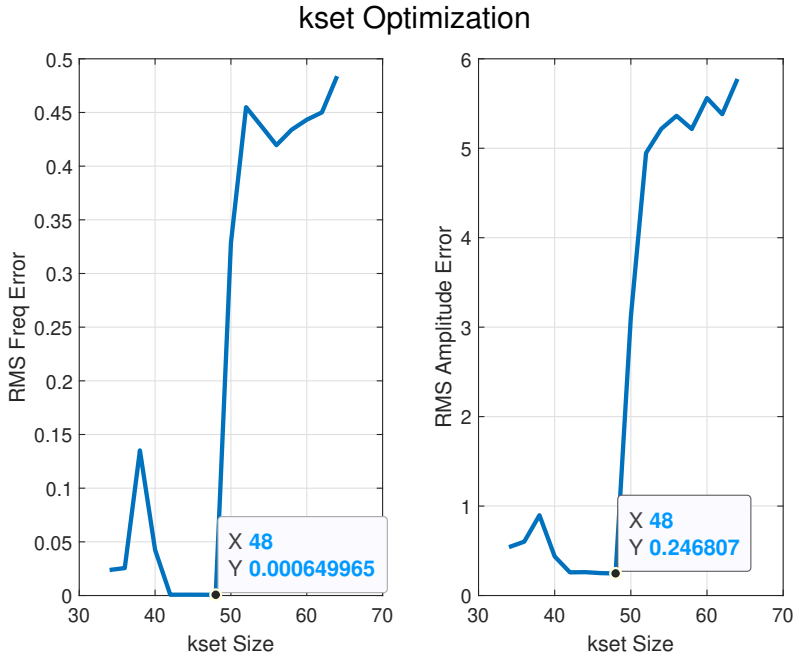


Figure 4.41: kset Optimization

The RMS amplitude and frequency errors for fs-ESPRIT are plotted against the size of *kset*. It can be seen that for the *kset* size between 42 and 48, the errors are lowest. The vector *kset* can then be set according to this size. The results after the optimization are shown in figure 4.42

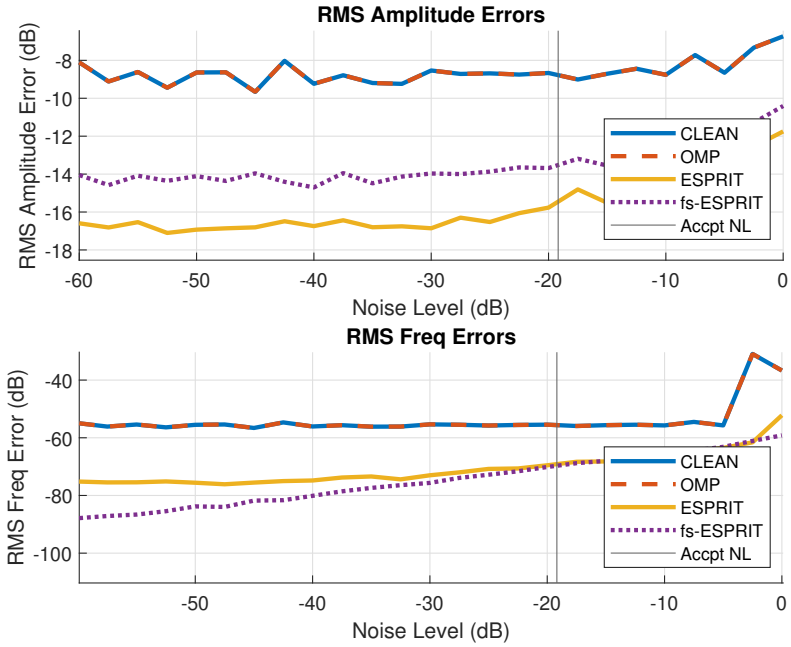


Figure 4.42: RMS Errors Clutter Test Case 3 (with Kset optimization)

It can be seen that now the errors for fs-ESPRIT are significantly reduced.

Some important conclusions can be drawn from these test cases:

- fs-ESPRIT is the most reliable for clutter spreads covering up to 10% of the frequency bandwidth.
- ESPRIT can give good results, but the shortcoming is that exact value of *order* is required. And as in practical scenarios, clutter doesn't necessarily have a consistent spread, this could be a problem.

4.4.4 Test Case: Multiple Frequencies, clutter Spread 5%, clutter to signal ratio 1

For this test case, Data Scenario 2 (3.1.2.3) is used. The purpose of using this simple data is to observe the effects of clutter on a multiple targets signal. And one of the targets lies very close to the clutter region in frequency. This is a highly practical scenario, where slow moving targets may appear in the same Doppler region as clutter, or in the same range bin as clutter. The clutter is generated as discussed in 3.4.1. The parameters used for clutter generation are mentioned in table 4.43.
Here the clutter power, $\sigma_c^2 = 200$ corresponds to the clutter to signal ratio of 1. This can be seen in the figure 4.43.

Parameter	Range
C	40
f_c	$\in [0, 0.05] \cup [0.95, 1]$
a_c	$\in CN(0, 1)$
σ_c^2	200

Table 4.43: Parameters for clutter generation

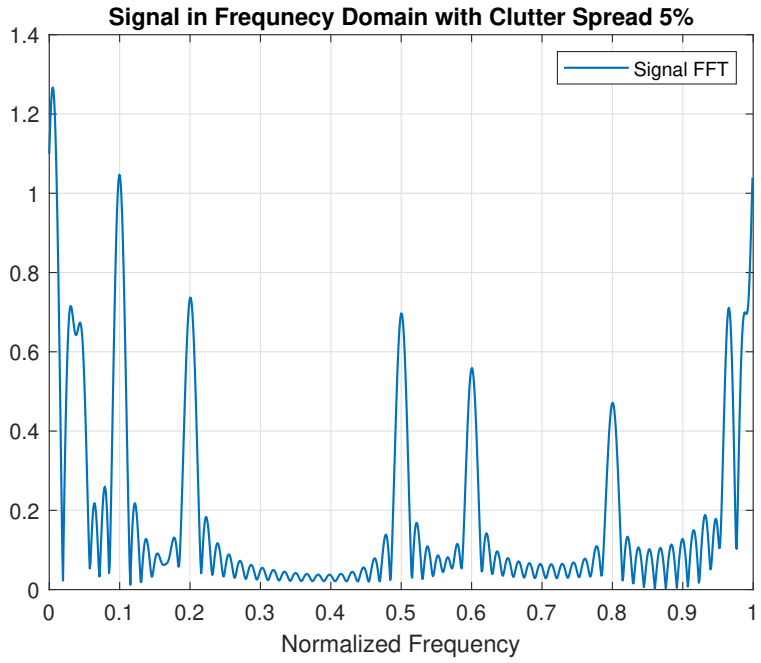


Figure 4.43: Signal in Clutter with 5% spread

The parameters of the test case and that of all the algorithms are listed in tables 4.44, 4.45, 4.46 and 4.47. The hyper-parameter *order* in ESPRIT is set as 9 corresponding to 4 additional frequencies demanded from algorithm.
The results for 200 Monte Carlo iterations is shown in figure 4.44.

Parameter	Symbol	Value
Number of Targets	K	5
Target Frequency	f	0.1,0.2,0.5,0.6,0.8
Target Amplitude	α	0.7071 + 0.7071i, 0.0000 + 0.7500i, 0.3400 + 0.5889i, 0.0000 + 0.5200i, 0.3182 + 0.3182i
Signal Length	N	64
Monte Carlo Iterations	N_{MC}	200
Noise Variance	σ_v^2	$10^{-3} - 10^{0}$

Table 4.44: Parameters for Test Case

Parameter	Range
K	1
N_{FFT}	1024
f_s	0.0752
f_e	0.9248

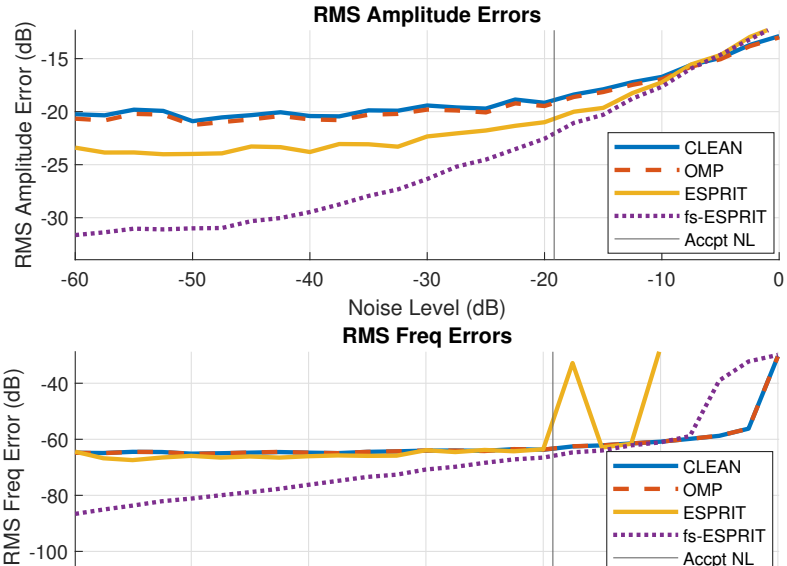
Table 4.45: Selected Hyper-parameters for CLEAN & OMP

Parameter	Range
$order$	9
q	32
σ_v	1

Table 4.46: Selected Hyper-parameters for ESPRIT

Parameter	Range
$order$	5
q	32
$kset$	$[5:60]$
σ_v	1

Table 4.47: Selected Hyper-parameters for fs-ESPRIT



The results indicate in case of multiple frequencies, the performance of fs-ESPRIT is the best.

5
Conclusion

The purpose of this thesis is to build a platform where different grid-less target estimation algorithms can be evaluated and compared. Therefore, a detailed simulation is developed with various data scenarios i.e. variable noise levels, signal lengths and clutter. The simulation includes the single and multiple targets cases, targets very close to each other on DFT grid and targets with different amplitudes. A robust evaluation methodology is devised to suit all these different test cases and to compare the errors, GOSPA metric is effectively used. Lastly, Cramer Rao Lower Bound is also used to establish a benchmark against which all estimators can be measured. The platform is capable of evaluating any estimation algorithm that estimates target frequency and amplitude. However, our work primarily compares two novel algorithms, ESPRIT and fs-ESPRIT with each other and with classic greedy algorithms like OMP and CLEAN. The main conclusion points are listed below:

- ESPRIT and fs-ESPRIT perform better under high target proximity conditions, i.e., when two targets are placed very close to each other on the DFT grid.
- Unlike CLEAN and OMP, ESPRIT and fs-ESPRIT are independent of the zero padding. This property makes them a better choice over classic greedy algorithms especially when computational cost is a limitation.
- ESPRIT and fs-ESPRIT have overall better performance than CLEAN and OMP when the noise in data signal is below acceptable noise level (3.5.2). However, the performance of CLEAN and OMP improves a lot with increase in signal length and zero padding.
- If computational power is not a constraint, OMP and CLEAN can be used with larger signal lengths. But for smaller signal lengths, noise variance below acceptable noise level and in high target proximity conditions, ESPRIT and fs-ESPRIT are better choice.
- For clutter spread up to 10%, fs-ESPRIT is the most reliable estimation algorithm to use.
- The performance of ESPRIT is better for clutter spread higher than 10%, but there is a limitation that exact value of the algorithm's parameter *order* is required, which is not possible in practical scenarios.
- In the presence of clutter, the frequency selective property of fs-ESPRIT makes it easier to use as compared to ESPRIT.

To conclude, the right choice of target estimation algorithm highly depends upon the radar environment. And this thesis provides the necessary tools to ease the process of this decision making.

Bibliography

[1] Maio, A. D., Greco, M. S. (Eds.). (2016). Modern Radar Detection Theory (p. 7). SciTech Publishing, Edison, NJ.

[2] Maio, A. D., Greco, M. S. (Eds.). (2016). Modern Radar Detection Theory (pp. 26-28). SciTech Publishing, Edison, NJ.

[3] Hsueh-Jyh Li, Yean-Woei Kiang, 10 - Radar and Inverse Scattering, (Eds.): WAI-KAI CHEN, The Electrical Engineering Handbook, Academic Press,2005, pp. 680-681, ISBN 9780121709600, https://doi.org/10.1016/B978-012170960-0/50047-5. (https://www.sciencedirect.com/science/article/pii/B9780121709600500475)

[4] https://www.tutorialspoint.com/radar_systems/radar_systems_matched_filter_receiver.htm

[5] Skolnik, M. I. (1981). Introduction to Radar Systems (pp. 392-395). McGraw-Hill, Inc.

[6] F. C. Robey, D. R. Fuhrmann, E. J. Kelly and R. Nitzberg, "A CFAR adaptive matched filter detector," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, no. 1, pp. 208-216, Jan. 1992, doi: 10.1109/7.135446.

[7] E. Conte, M. Lops and G. Ricci, "Adaptive matched filter detection in spherically invariant noise," in *IEEE Signal Processing Letters*, vol. 3, no. 8, pp. 248-250, Aug. 1996, doi: 10.1109/97.511809.

[8] K. Kulpa, "The CLEAN type algorithms for radar signal processing," 2008 *Microwaves, Radar and Remote Sensing Symposium, Kiev, Ukraine*, 2008, pp. 152-157, doi: 10.1109/MRRS.2008.4669567.

[9] Hai Deng, "Effective CLEAN algorithms for performance-enhanced detection of binary coding radar signals," in *IEEE Transactions on Signal Processing*, vol. 52, no. 1, pp. 72-78, Jan. 2004, doi: 10.1109/TSP.2003.820075.

[10] T. T. Cai and L. Wang, "Orthogonal Matching Pursuit for Sparse Signal Recovery With Noise," in *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4680-4688, July 2011, doi: 10.1109/TIT.2011.2146090.

[11] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984-995, July 1989, doi: 10.1109/29.32276.

[12] Daniel Potts and Manfred Tasche "Fast ESPRIT algorithms based on partial singular value decompositions" in *Applied Numerical Mathematics*, vol. 88, pp. 31-45, 2015, doi: 10.1016/j.apnum.2014.10.003

[13] https://www.renesas.com/us/en/blogs/basics-fmcw-radar

[14] S. M. Kay, Fundamentals of Statistical Signal Processing - Estimation Theory (pp. 503). Prentice Hall, 1993.

[15] T. McKelvey and M. Viberg. A Robust Frequency Domain Subspace Algorithm for Multi-Component Harmonic Retrieval. In *Proc. of 35th Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1288–1292, 2001

[16] T McKelvey, T Rylander, and M Viberg. Estimation of damped and undamped sinusoids with application to analysis of Electromagnetic FDTD simulation data. In *Preprints of the 13th IFAC Symposium on System Identification*, Rotterdam, The Netherlands, August 2003

[17] T Rylander, T McKelvey, and M Viberg. Estimation of resonant frequencies and quality factors from time domain computations. *Journal of Computational Physics*, 192:523–545, 2003.

[18] J Gunnarsson and McKelvey T. High SNR Performance Analysis of FESPRIT. In *Proc. of The Thirty-Eighth Annual Asilomar Conference on Signals, Systems, and Computers*, pages TA7–7, Pacific Grove, CA, 2004

[19] J Gunnarson and T McKelvey. Consistency Analysis of a Frequency Domain Subspace Algorithm for Multi-Component Harmonic Retrieval. In *Proc. of the International Conference on Acoustics, Speech and Signal Processing* , Montreal, Cananda, 2004

[20] J Gunnarsson and T McKelvey. Reducing noise sensitivity in F-ESPRIT using weighting matrices. In *Proc. 15th European Signal Processing Conference*, pages 783–787, Poznan, Poland, 2007. EUSIPCO

[21] R A Horn and C R Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, NY, 1985

[22] T. McKelvey, H. Akcay, and L. Ljung. Subspace-Based Multivariable System Identification from Frequency Response Data. *IEEE Trans. on Automatic Control*, 41(7):960–979, July 1996

[23] Richards, M. A., Scheer, J. A., Holm, W. A. (2010). Principles of Modern Radar (p. 555). SciTech Publishing, Edison, NJ.

[24] A.O Steinhardt and C. Bretherton. Threshold in frequency estimation. In Proc. 1985 IEEE International Conference on Acoustics, Speech and Signal Processing, pages 1273–1276, Tampa, FL, March 26-29 1985

[25] Improved ASV Mk. VI discrimination. Air Sea Warf. Dev. Unit (ASWDU) Report 45/20, (1945) (The National Archives AIR 65/182)

[26] M. Beard, B. T. Vo and B. -N. Vo, "OSPA(2): Using the OSPA metric to evaluate multi-target tracking performance," 2017 International Conference on Control, Automation and Information Sciences (ICCAIS), Chiang Mai, Thailand, 2017, pp. 86-91, doi: 10.1109/ICCAIS.2017.8217598.

[27] F. Chang, Z. Chen, W. Wang and L. Wang, "The hausdorff distance template matching algorithm based on kalman filter for target tracking," 2009 IEEE International Conference on Automation and Logistics, Shenyang, China, 2009, pp. 836-840, doi: 10.1109/ICAL.2009.5262808.

[28] K. Da, T. Li, Y. Zhu and Q. Fu, "A Computationally Efficient Approach for Distributed Sensor Localization and Multitarget Tracking," in IEEE Communications Letters, vol. 24, no. 2, pp. 335-338, Feb. 2020, doi: 10.1109/LCOMM.2019.2954888.

[29] A. S. Rahmathullah, Á. F. García-Fernández and L. Svensson, "Generalized optimal sub-pattern assignment metric," 2017 20th International Conference

on Information Fusion (Fusion), Xi'an, China, 2017, pp. 1-8, doi: 10.23919/I-CIF.2017.8009645.
[30] Stoica, P., Moses, R. (2005). Spectral Analysis of Signals (p. 355). Prentice Hall, Inc. Upper Saddle River, New Jersey 07458.
[31] Stoica, P., Moses, R. (2005). Spectral Analysis of Signals (p. 356). Prentice Hall, Inc. Upper Saddle River, New Jersey 07458.
[32] Stoica, P., Moses, R. (2005). Spectral Analysis of Signals (p. 360). Prentice Hall, Inc. Upper Saddle River, New Jersey 07458.
[33] S. M. Kay, Fundamentals of Statistical Signal Processing - Estimation Theory (pp. 524-525). Prentice Hall, 1993.
[34] `https://www.cuemath.com/euclidean-distance-formula/`

A

Appendix 1

The signal model is:

$$\boldsymbol{x} = \boldsymbol{S}\boldsymbol{\alpha} + \boldsymbol{v}$$

And:

$$\begin{aligned}
\boldsymbol{\mu}_{\boldsymbol{x}} &= \mathrm{E}[\boldsymbol{x}] = 0 \\
\boldsymbol{\mu}_{\boldsymbol{\alpha}} &= \mathrm{E}[\boldsymbol{\alpha}] = 0 \\
\mathrm{Cov}\begin{bmatrix}\boldsymbol{\alpha} \\ \boldsymbol{x}\end{bmatrix} &= \begin{bmatrix}\sigma_a^2\boldsymbol{I} & \sigma_a^2\boldsymbol{S}^H \\ \sigma_a^2\boldsymbol{S} & \sigma_a^2\boldsymbol{S}\boldsymbol{S}^H + \sigma_v^2\boldsymbol{I}\end{bmatrix} = \begin{bmatrix}Q_{11} & Q_{12} \\ Q_{21} & Q_{22}\end{bmatrix}
\end{aligned}$$

The LMMSE is calculated as:

$$\begin{aligned}
\boldsymbol{\alpha}_{\boldsymbol{LM\hat{M}SE}} &= \boldsymbol{\mu}_{\boldsymbol{\alpha}} + Q_{12}Q_{22}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}}) \\
&= \sigma_a^2\boldsymbol{S}^H(\sigma_a^2\boldsymbol{S}\boldsymbol{S}^H + \sigma_v^2\boldsymbol{I})^{-1}\boldsymbol{x} \\
&= \left(\boldsymbol{S}^H\boldsymbol{S} + \frac{\sigma_v^2}{\sigma_a^2}\boldsymbol{I}\right)^{-1}\boldsymbol{S}^H\boldsymbol{x}
\end{aligned}$$

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY