Non-Coherent Approach For Multiple Target Through-the-Wall Radar Localization

By

Imran Mohammed

A thesis submitted to Macquarie University

for the degree of Master of Research

Department of Engineering

July 2018

MACQUARIE
University
SYDNEY·AUSTRALIA

Except where acknowledged in the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part for a degree in any university.

Imran Mohammed

Acknowledgements

I would like to thank my supervisor, Prof. Stephen V. Hanly, and my co-supervisor Prof. Iain B. Collings for the patient guidance, encouragement and advice they have provided throughout my time during this research. I have been extremely lucky to have them as supervisors who cared so much about my work, and who responded to my questions and queries so promptly.

Abstract

This thesis considers non-coherent localization using bi-static radar with multiple receivers. In particular we consider the multiple target scenario, where the targets are located behind a wall. We present an algorithm that eliminates ambiguities that arise in this case by iteratively enforcing a set of constraints. We also present a clustering algorithm to further improve performance by making use of multiple sets of measurements. We show that the proposed approach achieves 50% less target identification errors compared to existing algorithms and achieves location accuracy in the order of 10cm.

Contents

List of Figures

Chapter 1

Introduction

RADAR is an acronym for RAdio Detection And Ranging. Radar has developed over a long period since World War II, mainly for the military purpose to track aircraft and ships. Radar is an electrical system that transmits radio frequency electromagnetic waves toward a region and receives reflected electromagnetic waves from the objects in that region [2]. Radars can also be used to detect the shape of target (or target identification), imaging (in imaging radar such as synthetic-aperture-radar (SAR) and inverse-SAR (ISAR)), and now even applications such as looking through walls, which is topic of this thesis.

The distance (range) to a target is measured via time delay and direction via antenna directivity in radar systems. Radars use the Doppler frequency[1] information from received signals to find the radial velocity of a target [3]. Doppler frequency is used to differentiate between stationary and moving targets in the region of interest [4]. Radars send a series of narrow rectangular pulses in the region of interest [5]. Those narrow rectangular pulses hit the target if it present in the area of interest and get reflected back to the radar. Based on the time taken to return the signal, the distance to the target is measured.

Range resolution is defined as the ability of radar to detect targets near each other as distinct objects. Range resolution is inversely related to the width of the pulse of the transmitted signal. The narrower the width of the pulse, the higher the range resolution [4]. These are called pulsed radar systems [6]. In pulsed radar, the detection range depends directly on the energy per pulse, i.e., on the ability of a radar to detect the targets at a range [4]. The more the energy in the signal, the more the detection range. Radar systems have to make a compromise between range resolution and detection range because they are limited by peak power. For example, to improve range resolution for a fixed detection range, the width of the pulse should be reduced, but due to hardware limitations, the width of the pulse cannot

[1] change in frequency of reflected signal due to moving target- Doppler effect

be infinitesimally small. To improve detection range for a fixed range resolution, the energy (or amplitude) of the signal should be increased. There are practical hardware limitations on signal power (for example, when a chirp signal is transmitted, output of the matched filter of the received signal is proportional to the duration of the signal) [4]. Later Woodward pointed out that range resolution is proportional to signal bandwidth, not of the transmitted pulse width [4].

In continuous wave radar, a continuous waveform is transmitted. A continuous waveform with the match-filter could be used to achieve the same range resolution as a small width pulse. This continuous waveform could bypass the problem of peak power problem mentioned above [4]. In this thesis, we are concerned with time of flights only, so either pulsed radar or continuous radar can be used. Time of flight is defined as the time taken for the transmitted signal from the transmitter to the receiver after being reflected by targets of interest.

Radars can be categorized into 1. mono-static radar and 2. bi-static radar based on transmitter and receiver location. In the mono-static scenario, one antenna serves as both transmitter and receiver. In a bi-static configuration, the transmitter and the receiver each have a separate antenna. In the bi-static scenario, the antennas should be considerably far apart. When they are close we call them as mono-static [2]. Continuous waveform radar often uses a bi-static configuration for transmitter and receiver isolation. In pulsed waveform mono-static radar, transmitter and receivers do not operate exactly at the same time. In this thesis, we focus on bi-static radar with multiple receivers but the proposed schemes can be applied to mono-static radars as well.

1.1 Through-the-Wall Radar

A system that can see through visually opaque obstacles could improve capabilities of emergency or defense systems. For example, in fire rescue operations, looking through a wall to find anyone inside a building may be better than having an actual person having to go inside a dangerous scenario looking for any humans trapped in the building. Through-the-wall radar has been developed to detect the targets that are behind a wall (visually opaque objects). Through-the-Wall Radar (TWR) has gained substantial recent interest for its increased civilian and military applications. Applications include rescue operation, anti-terrorist enforcements, surveillance, etc. to name few [7–10]. Current hot topics with unsolved challenges are for

indoor localization and identification.

Target localization in an indoor scenario using radar is a hard problem because of multiple reflections from walls and existing RF signals such as Wi-Fi. One particular scenario is to estimate the location of active devices behind the wall, and receivers are listening to the signal passively. In this specific scenario, when the number of active transmitters inside the room is unknown, there is an ambiguity at the receiver whether the signal received is from a different device or from multi-path signals from same device bouncing off walls. This is one of many challenges when trying to localize active devices.

A similar scenario is trying to locate targets behind a wall by passively listening to signals from active devices. In this problem, the locations of the active devices needs to estimated before the estimation of the target location. Due to multiple reflections from walls and reflection from targets, this is a very hard problem to solve.

A simpler scenario is trying to localize the targets that are behind a wall and receivers are listening to signals from the transmitter which is in our control, as illustrated in Fig. 1.1. This is still a difficult problem to solve due to multiple reflections from the wall. Even with a single target, determining its location is a very hard problem in general.

In this thesis, we are considering a scenario in which there is only one wall between the targets of interest and the transmitter. The targets are assumed to be stationary. The received signal at the receiver contains only the reflected signals from the targets. The number of targets (say N) behind the wall is known.

1.2 Through-the-Wall Localization Techniques

Through-the-Wall Radar (TWR) Localization is classified into two types: 1. Coherent processing 2. Non-coherent processing. Most of the existing through-the-wall radar systems employ coherent processing techniques [7]. Coherent radar imaging is based on beam-forming. This requires arrays with large apertures. Coherent radar imaging achieves high resolutions at the expense of portability, cost and complex system designs. The two-dimensional spatial resolution of a coherent radar imaging system is directly related to the number of antennas [11].

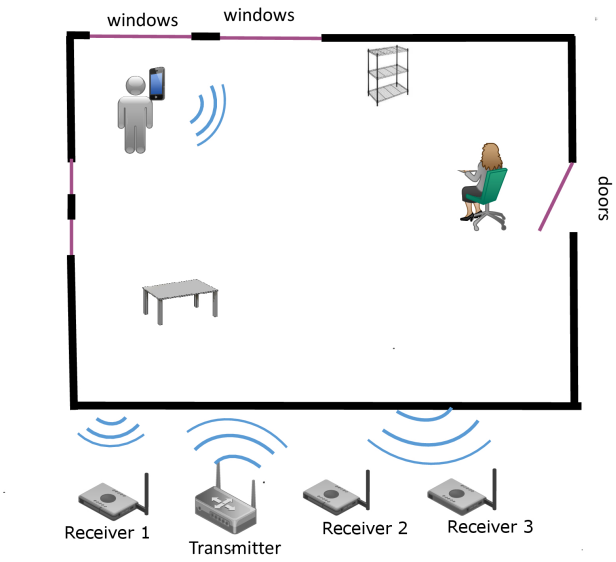

FIGURE 1.1: General Indoor scenario

1.2.1 Coherent Processing

Ferris and Currie [12, 13] have reviewed existing microwave systems that can detect the presence of persons behind walls and track their movement. A co-array based aperture synthesis scheme using sub-arrays and post data acquisition beam-forming is presented for through-the-wall wide-band microwave imaging applications [11].

1.2.2 Non-coherent Processing

A time-of-arrival measurement is defined as the time taken for the signal to reach the receiver from the transmitter. Non-coherent processing based on time-of-arrivals (TOAs) has advantages over coherent processing as it can be achieved using simpler hardware and it allows for a flexibly distributed deployment scenario. Although non-coherent processing is suboptimal, it has practical advantages compared to coherent processing because it significantly relaxes radar positioning and processing requirements [7]. Using the information of TOAs and the position of radars, the target location is estimated using a geometric approach.

The localization can be performed using both mono-static radars and bi-static radars. In the mono-static radar case, the locus of all points that satisfy a time-of-arrival measurement is a circle with the radar location at the center of the circle as shown in Fig. 1.2. The target can be located at any point on the circle. In the bi-static scenario, the locus of all points that satisfy the time-of-arrival measurement is an ellipse, with the transmitter and receiver

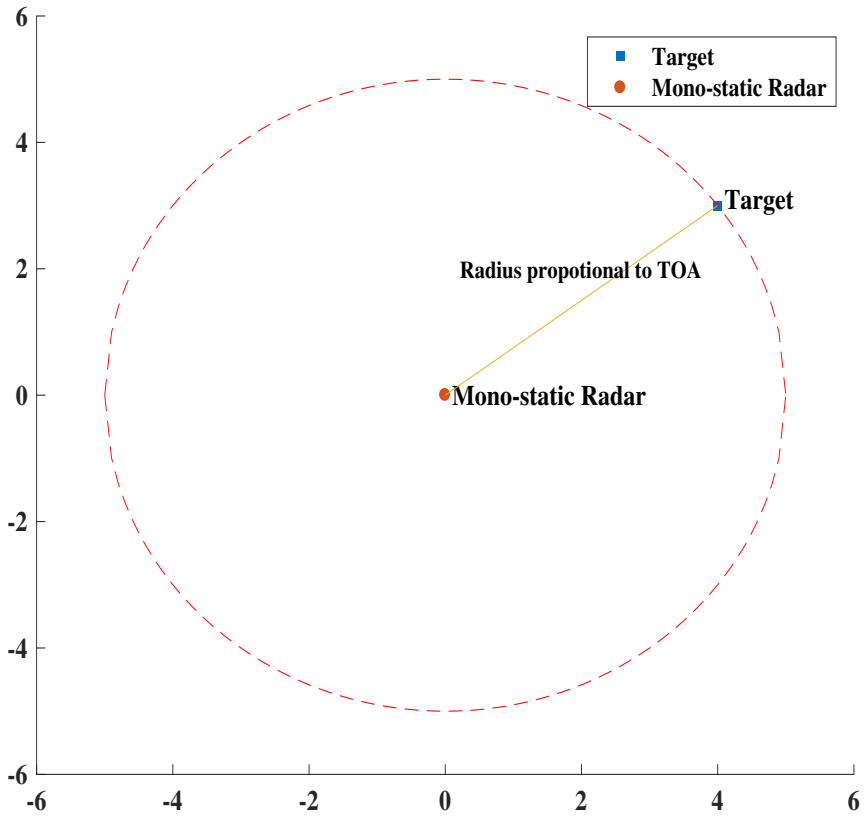

FIGURE 1.2: Locus of all points that satisfy a TOA measurement in mono-static radar scenario-Circle

locations as focal points. The target can be located at any point on the ellipse; this is shown in Fig. 1.3. TOA measurements at different spatial locations can be used to resolve the ambiguity of target location on circle/ellipse.

To measure TOAs at different locations in mono-static case, we need to deploy the radar at multiple locations whereas, in the bi-static scenario, we can to deploy multiple receivers at different locations keeping the transmitter fixed. Using each TOA measurement a circle or an ellipse is plotted. Because all TOA measurements belong to the same target, the target has to lie on every circle or ellipse. So these circles or ellipses will intersect at a common point, the target location. Finding target location using ellipses is called Ellipse-Cross-Localization (ECL). In this thesis, we consider the bi-static scenario with multiple receivers.

The above-discussed approach to localize the target is valid in free space. In the presence of a wall, time-of-arrival measurements differ from time-of-arrival measurements in free space due to the change in the path and velocity of the transmitted signal inside the wall. This is illustrated in Fig. 1.4. TOAs will depend on various factors such as the position of the

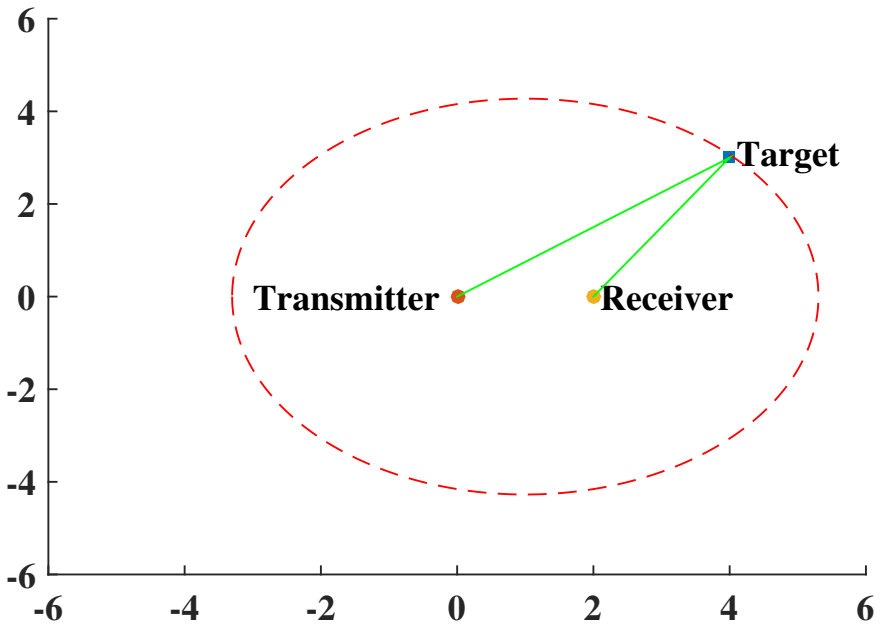

FIGURE 1.3: Locus of all points that satisfy a TOA measurement in bi-static radar scenario-Ellipse

wall, wall parameters such as thickness, the material of which the wall was made, etc. and the position of the receiver. TOA measurements will differ from TOA measurements if there was no wall. If circles or ellipses are plotted using TOA measurements in the presence of the wall, the circles or ellipses will not intersect at common points. This TWR problem has gained recent interest and is solved under various conditions, which are discussed below.

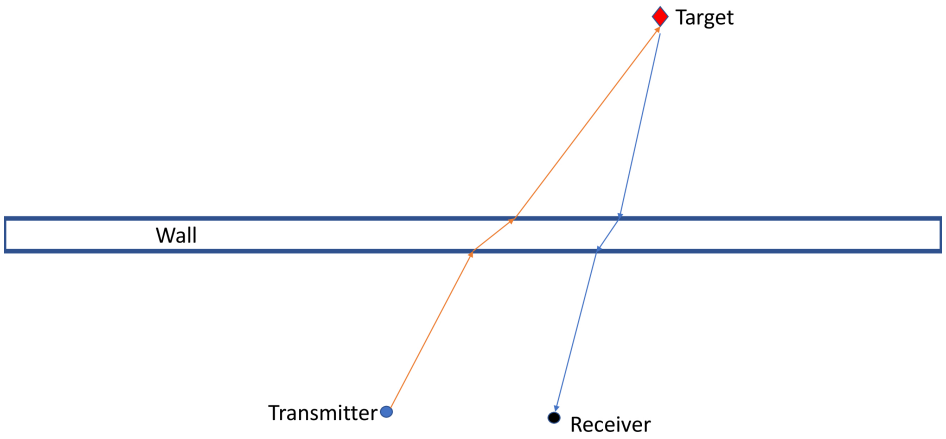

FIGURE 1.4: General Indoor scenario

In [7], non-coherent processing with three mono-static radars based on the trilateration algorithm was proposed. The trilateration algorithm is a procedure to find the intersection

of circles. In [7], wide band signals are used that are synthesized using a stepped-frequency approach. TOAs of received signals measured at three mono-static radars are used to find the target location using the trilateration algorithm. The propagation path of the signal in the presence of a wall is analyzed. Assuming perfect knowledge about the wall, such as the wall thickness, wall location, composition of the wall, i.e., dielectric constant, etc. an error correction scheme was proposed to remove bias. In this approach, the wall is assumed to be homogeneous with uniform thickness.

Based on [7], non-coherent processing with bi-static radars was proposed in [10]. Similar to [7], the propagation path of the signal in the presence of a wall was analyzed for bi-static radar with multiple receivers scenario in [10]. Based on propagation path of the signal and wall parameters information, an approach was proposed to estimate the target location. This approach is detailed in Sec. 2.5. In [10], effects of inaccurate estimation of wall thickness and dielectric constant on TWR localization were analyzed. It claimed that incorrect wall thickness estimation has a more significant effect on TWR localization compared to inaccurate di-electric constant estimation.

Both [7] and [10], assume perfect knowledge about the wall in their respective localization schemes which is not realistic in real time scenarios. Results in [10] shows that incorrect estimation of wall parameters lead to localization errors. They also assume wall to be homogeneous with uniform thickness which is not true in general.

When two TOA measurements are available from a target, a hyperbola can be plotted. Similar to using ellipses or circles, when the set of points are plotted such that difference of distance from two radars (in bi-static case, two receivers) is constant, we obtain a hyperbola. The two radars (in bi-static case, two receivers) act as focal points of the hyperbola. The target can be located at any point on the hyperbola. An approach to localizing a target based on hyperbolas was proposed in [8]. The ambiguity of finding a target location on a hyperbola is resolved by taking more measurements of the target scene, and multiple hyperbolas are plotted. The target has to lie on all those hyperbolas, as they intersect at the target location. This is called Hyperbola-Cross-Localization (HCL). In [8], using HCL and multi-channel cross-correlation detection an approach was proposed. In [8], it is claimed that cross-correlation of two-channel measurements has a higher SNR compared to a single-channel measurement. This is achieved at the expense of more measurements. Although this approach does not assume any knowledge about wall parameters, this requires more measurements.

Based on both Ellipse-Cross-Localization and Hyperbola-Cross-Localization, Ellipse-Hyperbola-Localization (EHL) localization approach was proposed in [9]. EHL approach was proposed taking statistical properties of ellipse (ECL) and hyperbolas (HCL) in the presence of wall into consideration. This approach also assumes no knowledge about the wall or its position.

In most practical scenarios, there are multiple target to be located. For example, a typical scenario as shown in Fig. 1.1 with a chair, bookshelf, table, etc. All the above-proposed localization techniques can localize only one target, none of these techniques can be extended directly to multiple target localization.

When there are N targets and R receivers, each receiver has N TOA measurements. To estimate the target location, TOA measurements need to associated from all the receivers (i.e., for a target, R TOA measurements). But the receiver does not have knowledge about which TOA measurement corresponds to which target. This is a data association problem. For example, in Ellipse-Cross-Localization case with two targets and two receivers, each receiver has 2 TOA measurements. So with respect to each receiver, two ellipses can be plotted. If we pick an ellipse from receiver-1, we need to associate with an ellipse from receiver-2 to estimate the target location. But, with respect to receiver-2 there are two ellipses. Since the receiver doesn't have knowledge about the which range measurement corresponds to which target, there is uncertainty in associating the ellipse. In other words, there is a problem of associating measurements with targets. This leads to multiple potential locations depending on the way association is done.

The first approach to localize multiple targets was proposed [1] using multiple mono-static radars. In [1], all the potential target locations found using the trilateration algorithm are listed. Based on shape of those target locations, each potential target location is assigned an error metric. Based on the error metric an approach is proposed to find the actual target locations among many potential target locations. This approach uses only the error metric as a constraint and performance is highly dependent on time-of-arrivals (TOA) accuracy [7]. In the presence of a wall, the TOA measurements do not match the free space distances and as a result the algorithm proposed in [1] does not perform well in this scenario.

[14] proposed a way to find locations when there is no association problem (i.e., for example, when there are two receivers and two targets, an ellipse from receiver-1 intersects with only one ellipse from receiver-2), but they haven't proposed an approach to find the

target locations when there is an association problem. Similar to [1], the approach is highly dependent on TOA accuracy. In the presence of a wall, we have observed that most of the time there is target association problem. Based on the approach in [14] and position of the potential target locations in space an approach was proposed to reduce the number of potential target locations in some cases. The challenge is to add enough constraints to fully solve the target association problem.

Details of the approaches in [1, 14] will be presented in Chapter 3.

1.3 The Challenge

In this thesis, we are considering a scenario when there is one wall between the area of interest (containing the targets) and the radars. The targets behind the wall are assumed to be stationary. The received signal at the receiver contains only the reflected signals from the targets. The number of targets (say N) behind the wall is known. We are focusing on bi-static radar with 3 receivers, but this can be generalized to multiple mono-static radars. The aim is to find locations of multiple targets behind the wall with non-coherent processing techniques. We do not assume perfect knowledge about the wall, apart from the statistical knowledge. But, when the parameters of homogeneous wall with uniform thickness are available, estimates of target location can be improved using this additional information. We will solve the target association problem to allow us to find the N target locations out of potentially many possible locations, using constraints that we find must be satisfied by the target locations. A more precise problem statement is given in Sec. 2.4

1.4 Thesis outline and Contributions

- We present two different algorithms that finds N target locations among N^3 potential target locations with 3 receivers by iteratively enforcing a set of constraints.
- The proposed algorithm has 50% less target identification errors in a 5 target scenario compared to the algorithm in [1].
- We analyzed different ways to associate multiple set of measurements to improve the target location accuracy.

- We propose a novel clustering algorithm to combine multiple measurements and to estimate the target locations.

- The proposed clustering algorithm achieves location accuracy in the order of $10cm$ when trying to map a $12m^2$ office that was behind a wall (wall is at a distance of $4m$ from radar elements). The office is modeled as a 6 target scenario.

Chapter **2**

Signal Model

This project aims to estimate the location of multiple targets that are behind a wall using non-coherent processing. In general, this is a very challenging problem, but it becomes tractable if we make some simplifying assumptions. The following assumptions are made while localizing the multiple targets.

- There is only one wall between the radars and the targets of interest as shown in Fig. 2.1.
- The number of targets (say N) behind the wall is known.
- The received signal at the receiver contains only the reflected signals from the targets.
- All the targets are stationary.

In Fig. 2.1, the targets are the chair, table, and bookshelf. Knowledge about the wall is not mandatory, but if available this information can be used to improve the estimation of the target locations. We focus on bi-static radars, but the proposed algorithms can be extended to mono-static radar as well. The aim is to estimate the location of multiple targets with bi-static radar with multiple receivers in the through-the-wall scenario under the above assumptions.

Let $s(t)$ be the transmitted signal, then $s_{r,j}(t) = \sum_{i=1}^{N} \alpha_i s(t - t_{i,j}) + n_j(t)$ is the received signal at j^{th} receiver, when there are N targets, where α_i is complex reflectivity of i^{th} target and $n_j(t)$ is system noise at receiver j. We denote $t_{i,j}$ to be the time taken for the signal to travel from transmitter to i^{th} $(i = 1, 2, \ldots N)$ target and from i^{th} target to j^{th} receiver. For Non-coherent processing, time-of-arrivals $(t_{i,j})$ corresponding to targets needs to be estimated $(\hat{t}_{i,j})$. Estimation of TOAs is discussed in the next section.

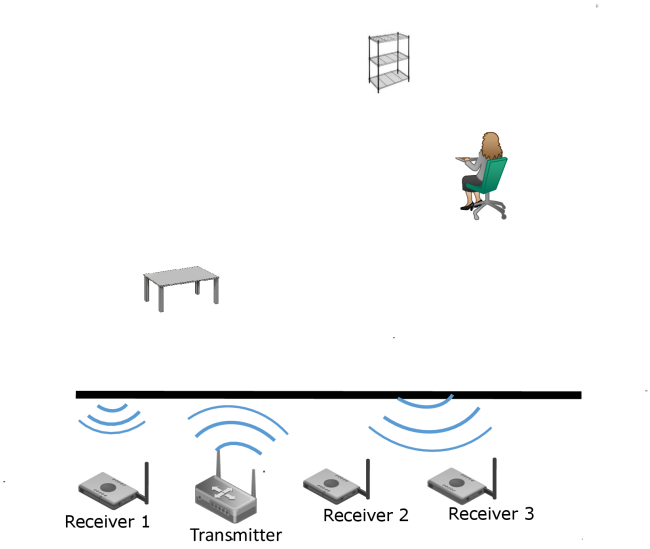

FIGURE 2.1: General Indoor Scenario

2.1 Estimation of TOAs

Time-of-arrivals can be determined using a matched filter [15]. In the matched filter, the received signal at j^{th} receiver, $\left(s_{r,j}(t)\right)$ is passed through a linear time-invarient (LTI) system with impulse response $h(t) = s^*(-t)$ and we find N time instances corresponding N peak values of the output signal $s_{o,j}(t) = s_{r,j}(t) \star h(t)$ as estimates of $t_{i,j}$ $(\hat{t}_{i,j})$ for $i = 1, 2, \ldots, N$.

Consider first the case of one target, which we will call target 1 in what follows, and the transmitted signal is the chirp signal, i.e.,

$$s(t) = rect\left(\frac{t}{T}\right)e^{j\pi(t-t_c)^2} \tag{2.1}$$

where t_c is the time offset at the pulse center referenced to $t = 0$ [15]. The received signal at the j^{th} receiver corresponding to the target 1, $s_{r,j}(t) = \alpha_1 s(t - t_{1,j}) + n_j(t)$, is given by

$$s_{r,j}(t) = \alpha_1 rect\left(\frac{t - t_{1,j}}{T}\right)e^{j\pi(t-t_{1,j}-t_c)^2} + n_j(t) \tag{2.2}$$

and this signal $\left(s_{r,j}(t)\right)$ is passed through matched filter [15] i.e., the LTI system with impulse response $h(t) = s^*(-t) = rect\left(\frac{t}{T}\right)e^{j\pi(t+t_c)^2}$ The output of matched filter is

$$s_{o,j}(t) = T\alpha_1 e^{-j2\pi t_c(t-t_{1,j})} sinc\{T(t - t_{i,j})\} + [h(t) \star n_j(t)] \tag{2.3}$$

The time corresponding to the peak value of $s_{o,j}(t)$ is an estimate of $t_{1,j}$. Similarly when there are N targets, the received signal at j^{th} receiver comprises of N reflected signals from

N targets. The received signal is then as follows

$$s_{r,j}(t) = \sum_{i=1}^{N} \alpha_i rect\left(\frac{t - t_{i,j}}{T}\right) e^{j\pi(t-t_{i,j}-t_c)^2} + n_j(t) \tag{2.4}$$

Then the output of the matched filter is given by

$$s_{o,j}(t) = \sum_{i=1}^{N} T\alpha_i e^{-j2\pi t_c(t-t_{i,j})} sinc\{T(t - t_{i,j})\} + [h(t) \star n_j(t)] \tag{2.5}$$

The time instances corresponding to N peak values of matched filter output signal ($s_{o,j}(t)$) are estimates of $t_{i,j}$ for $i = 1, 2, 3, \ldots, N$. Here, we assume which TOA measurement corresponds to which target by using the subscript "i", but in real time we don't know which TOA measurement corresponds to which target. The receiver receives the reflected signal from which N TOA measurements is extracted. Associating TOA measurements to the targets is the challenge of this thesis. Once the TOAs of all the targets are estimated ($\hat{t}_{i,j}$ for $i = 1, 2, 3, \ldots, N$), the range to the targets ($r_{i,j}$) can be estimated using the formula $\hat{r}_{i,j} = c\hat{t}_{i,j}$ for $i = 1, 2, \ldots, N$ in free space. In the presence of wall, $\hat{r}_{i,j} = c\hat{t}_{i,j}$ is no longer an estimate of $r_{i,j}$ because signal path and speed changes inside the wall (see Fig. 2.7). We assume the location and properties of the wall are unknown. We will use the model in [9], which we will now describe.

In practice, typical frequency ranges are from 1 to 3 GHz. A reasonable assumption of bandwidth is 1GHz. So the range resolution is $\sim 15cm$. The range measurements are effected by the wall. Error due to presence of wall of thickness d is $d_e = (\sqrt{\epsilon} - 1)d$ where ϵ is the dielectric constant of the wall. In general, wall thickness can range from 24 to $36cm$ and dielectric constant ranges from 4 to 9 [9]. So the ranging error due to the presence of the wall is 24 to $72cm$. Considering the error due to the system noise and the presence of the wall and removing the bias ($48cm$), from the range measurements as distance compensation will lead to range error from 12 to $18cm$. So the error in range values after removing the mean is less than $30cm$. In [9], the errors due to the presence of wall is modeled using Gaussian noise.

$$\hat{r}_{i,j} = c\hat{t}_{i,j} \tag{2.6}$$

$$= r_{i,j} + \Delta r_{i,j} \tag{2.7}$$

$\Delta r_{i,j}$ is modeled in [9] as a Gaussian random variable with mean $\mu = 48cm$ and variance σ^2. The error due to the presence of wall and system noise, after taking out the mean, is from 12 to $18cm$. The probability of a Gaussian random variable falling in the interval $[\mu - 3\sigma, \mu - 3\sigma]$ is 99.73%. So $\sigma = 0.1(10cm)$ satisfies the conditions.

Since μ is constant, it can without loss of generality be removed from all measurements resulting in

$$\hat{r}_{i,j} = r_{i,j} + \Delta r_{i,j} \tag{2.8}$$

As in [9], we will simplify the model and assume the measurement errors are independent across the receivers, so we assume that $\Delta r_{i,j}$ are independent and identically distributed Gaussian random variables with distribution $\mathcal{N}(0,\ \sigma^2)$

2.2 Ellipse Cross Localization

In this section, we review Ellipse-cross-localization under the assumption of zero noise. For any transmitter and receiver pair, the set of all points that satisfy the range measurement is an ellipse. The target can be located at any point on the ellipse (as illustrated in Fig. 1.3). For a target, when there are two receivers, two ellipses can be drawn. The target will lie on both ellipses, so they intersect at a point which is the target location. Two ellipses can intersect at multiple points, but in a practical scenario, only one intersection point corresponds to the target of interest.

Finding the target location using ellipses is called Ellipse-Cross-Localization(ECL). In general, two ellipses can intersect at four different locations, but in the radar scenario we are considering, one of the two focal points for each ellipse is same, i.e., the transmitter, so there will be only two intersection points.The ellipses are drawn using the equation $\{\mathbf{e}(e_x, e_y)|\ \|\mathbf{T} - \mathbf{e}\|_2 + \|\mathbf{R}_i - \mathbf{e}\|_2 = \hat{r}_{i,j}\}$, where $\mathbf{T}$ is the target location, $\mathbf{R}_j$ is the j^{th} receiver, and $\hat{r}_{i,j}$ is the estimated range measurement of i^{th} target at j^{th} receiver.

2.2.1 ECL Approach In Free Space For Single Target

When there is only one target, let us denote it as target-1 at location $\mathbf{p}_1(p_{1,x}, p_{1,y})$, each receiver has a single TOA measurement. The transmitter and receiver locations as shown in

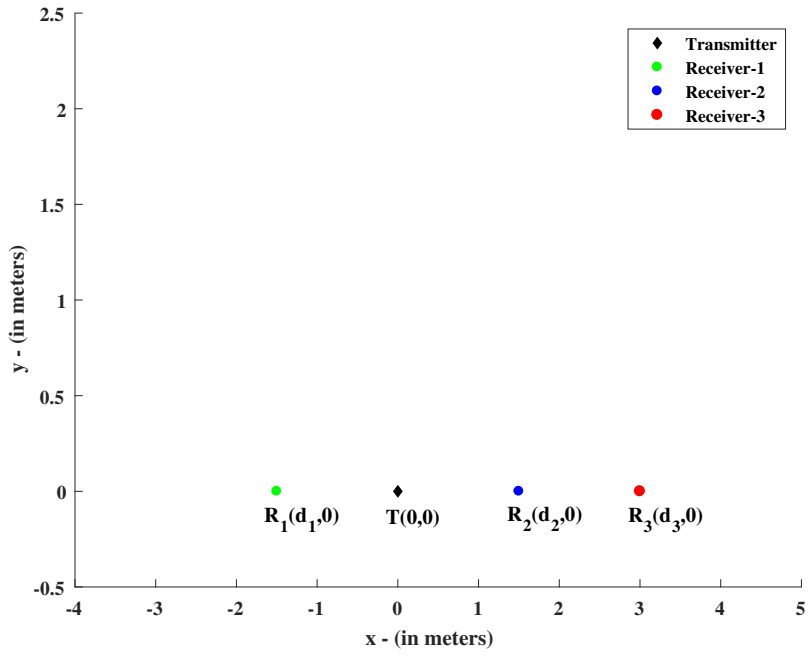

FIGURE 2.2: Transmitter and Receiver location - Bi-static radars with multiple receivers

Fig. 2.2. In free space and without system noise, $\hat{t}_{1,j} = t_{1,j}$ and $\hat{r}_{1,j} = r_{1,j}$. From the Fig. 2.3, it can be seen that all the three ellipses intersect at a single point, corresponding to the target location.

The intersection point (Target location) can be found by solving the following equations

$$r_{1,1} = \sqrt{p_{1,x}^2 + p_{1,y}^2} + \sqrt{(p_{1,x} - d_1)^2 + p_{1,y}^2} \tag{2.9}$$

$$r_{1,2} = \sqrt{p_{1,x}^2 + p_{1,y}^2} + \sqrt{(p_{1,x} - d_2)^2 + p_{1,y}^2} \tag{2.10}$$

$$r_{1,3} = \sqrt{p_{1,x}^2 + p_{1,y}^2} + \sqrt{(p_{1,x} - d_2)^2 + p_{1,y}^2} \tag{2.11}$$

When the above 3 equations are solved for $(p_{1,x}, p_{1,y})$, we will get two intersection points. When the receivers and transmitters are collinear and lie on $y = 0$ axis, as shown in Fig. 2.2, one intersection point lies in the region above the transmitter and receivers (positive y-axis) and the other intersection point lies below (negative y-axis) the transmitter and receiver region. In practical scenarios, we will try to map the targets that are present in one of these regions which is our area of interest (we are focusing on the positive y-axis region). So there will be two intersection points, but only one intersection point will be in the area of interest which is, therefore, the target location.

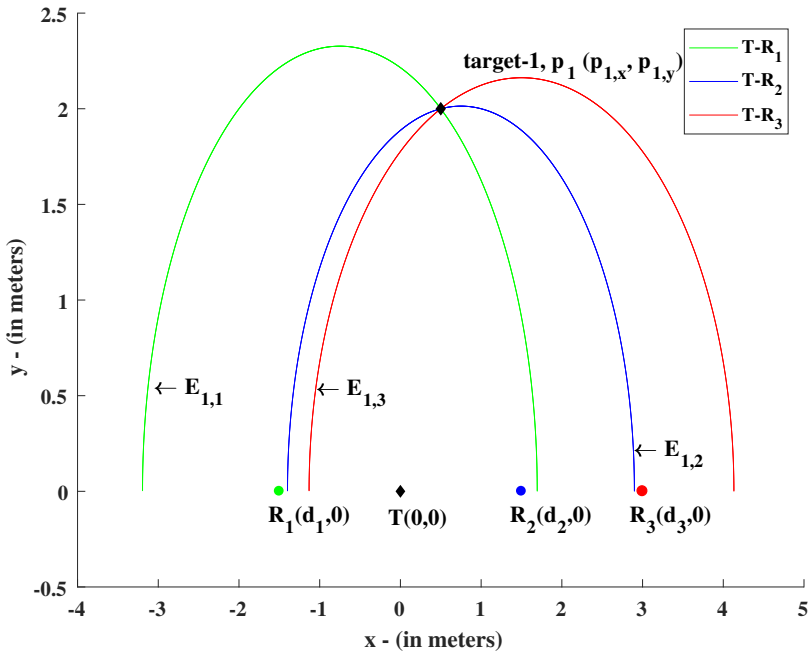

FIGURE 2.3: Estimation of Target location using ECL Approach in free space and with zero system noise. Target is located at intersection all 3 ellipses.

When there is system noise, we find intersection points of ellipse-1 ($r_{i,1}$) with ellipse-2 ($r_{i,2}$), ellipse-2 with ellipse-3 ($r_{i,3}$), and ellipse-3 with ellipse-1. The average of 3 intersection points (that are in the area of interest) can be used as a good estimate of the target location [9].

2.2.2 ECL Approach In The Presence Of Wall For Single Target

The estimated values differ from true range value as follows $\hat{r}_{i,j} = r_{i,j} + \Delta r_{i,j}$ due to the presence of wall and system noise. Here $\Delta r_{i,j}$ is i.i.d Gaussian random variable with mean μ and variance σ^2 [9]. It is claimed in [9] that $\mu = 0.48$ $\sigma = 0.1$ satisfies the most practical cases (see Sec. 2.1). When ellipses are plotted using the corrected measurements (removing μ from all the measurements), the 3 ellipses will not intersect at a common point. Rather, there will be 3 points, where two ellipses intersect and a triangular-like region between them. For example, for a single target, a triangular-like region is shown in Fig. 2.4. In this thesis, we consider these regions as triangles with intersection points as the vertices of the triangle. When the noise values are sufficiently large, the ellipses will not intersect each other, and a triangle is not formed. When the triangles are formed based on range measurements of a particular target, we call this triangle a Single-Target-Triangle (STT). The Centroid of the STT is an estimate of the target location in the ECL approach [9] when there is no information

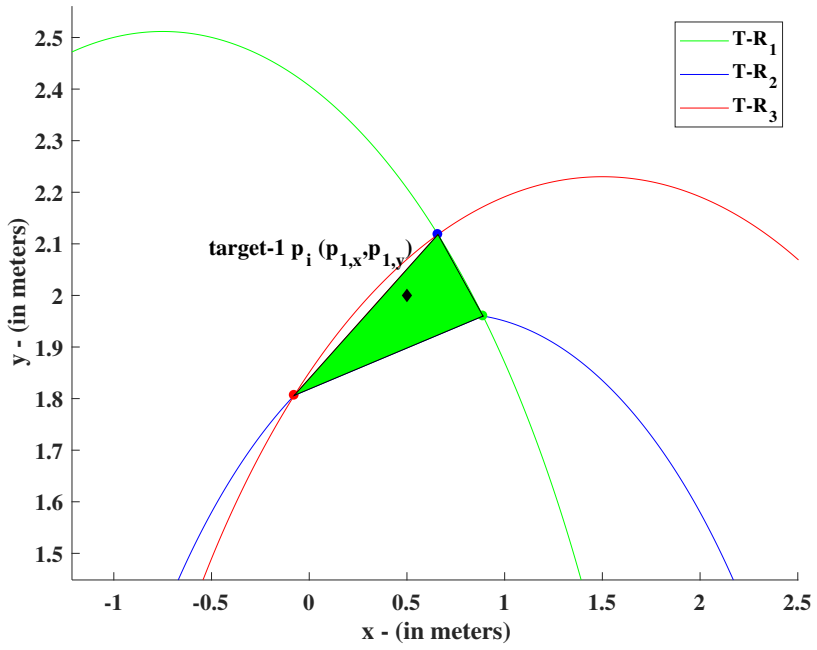

FIGURE 2.4: Formation of triangles due to the presence of wall and system noise ($\sigma = 0.1$). We call this triangle Single-Target-Triangle (STT) because triangle is formed due to estimation errors in range measurements from a particular target.

about the wall parameters.

2.3 Multiple Target Scenario

When there are N targets, each receiver has N range measurements. Using each range value at a receiver, an ellipse can be drawn. So, for each receiver, N ellipses can be plotted. When there are three receivers, $3N$ ellipses will be plotted. Every triangle that is formed by the ellipses from receivers $\mathbf{R}_1, \mathbf{R}_2$ and $\mathbf{R}_3$ is a potential triangle. We denote the ellipse from the range measurement, $\hat{r}_{i,j}$ as $\mathcal{E}_{i,j}$, i.e., the ellipse with respect to j^{th} receiver corresponding to i^{th} target. For any pair of ellipses with respect to receiver $\mathbf{R}_1$ and $\mathbf{R}_2$, there are N ellipses with respect to receiver $\mathbf{R}_3$, which can form a potential triangle. For example, when we select ellipses, $\mathcal{E}_{1,1}$ and $\mathcal{E}_{1,2}$ from receiver $\mathbf{R}_1$ and $\mathbf{R}_2$ respectively, there are N ellipses ($\mathcal{E}_{i,3}$, $i = 1, 2, \ldots, N$) with respect to receiver $\mathbf{R}_3$ that can form a potential triangle. So there can be up to N potential triangles. Similarly, for any pair of ellipses from with respect to receiver $\mathbf{R}_1$ and receiver $\mathbf{R}_3$, there are N ellipses with respect to receiver $\mathbf{R}_2$, which can form a potential triangle. So for every ellipse with respect to receiver $\mathbf{R}_1$, there can be up to N^2 potential triangles. And there are N ellipses with respect to receiver $\mathbf{R}_1$. Not every combination

of range measurement from $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{R}_3$ will form a triangle. So there can be up to (maximum of) N^3 potential triangles.

Fig. 2.5 shows a case with two targets and three receivers in free space and with no estimation error. Because of zero noise, target locations p_1 and p_2 are estimated as the points where all 3 ellipses (from 3 receivers, in Fig. 2.5, 3 different colors) intersect exactly at a particular point. Even with zero noise, triangles (3 sides from 3 different receivers) are formed, green triangles in Fig. 2.5. These triangles are due to target mis-associations, and we call these triangles, "Cross-Target-Triangles (CTT)". For example, the triangle formed by the ellipses $\mathcal{E}_{2,1}, \mathcal{E}_{1,2},$ and $\mathcal{E}_{1,3}$ is a CTT because all the three ellipses (corresponding range measurements) do not belong to a particular target.

When there is zero noise, we know that three ellipses will intersect at a common point which is the target location, but in the presence of noise all the three ellipses will not intersect at a common point, rather they form a triangular-like region with points, where two ellipses intersect (refer to Sec. 2.2.2). For example in Fig. 2.6, ellipses are plotted in the presence of wall and system noise. In Fig. 2.6 red triangles (STT) corresponds to triangles correspond to the target location, and green triangles (CTT) are formed because of target mis-association. In real time, we don't know which triangles corresponds to STT and CTT because both triangles are formed by ellipses from 3 different receivers, and both have a triangular-like shape.

The aim is to find all N STTs among all the triangles. For simplicity, we have shown only two Cross-Target-Triangles (CTT) in Fig. 2.6. Once triangles corresponding to STT are selected, the multi-target problem naturally splits into N single target problems. Based on the knowledge about wall parameters and the locations one the techniques proposed in [7–10] can be used to estimate the target locations.

2.4 Problem Statement

In this thesis, we aim to estimate multiple target locations that are behind a wall using non-coherent processing techniques. Here, we use bi-static radars with three receivers. Because of target association, there can be up to N^3 potential target locations out of which N are actual target locations. We use ellipse-cross-localization and find the N triangles that correspond to single-target-triangle (STT) out of N^3 potential STT triangles.

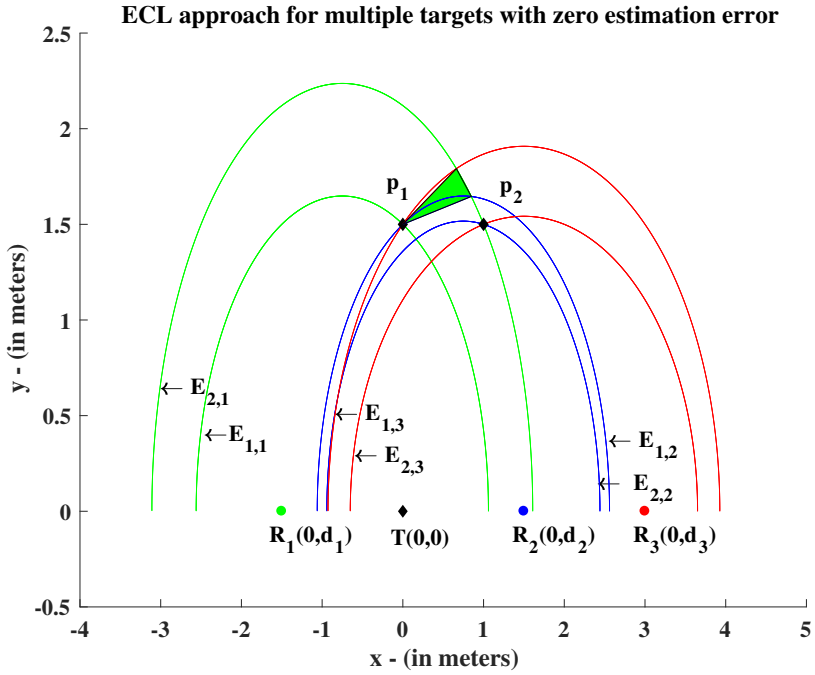

FIGURE 2.5: Estimation of location of multiple targets using ECL approach in free space and zero estimation error ($\sigma = 0$), because of zero estimation error target location is estimated as intersection point of ellipses from 3 receivers. In Fig. same color ellipses are drawn from range measurements of a particular receivers i.e., green ellipses are from $\mathbf{T} - \mathbf{R}_1$ pair. Even with zero noise triangles are formed (triangle from ellipses from 3 different receivers) and we call the triangle as Cross-Target-Triangle (CTT).

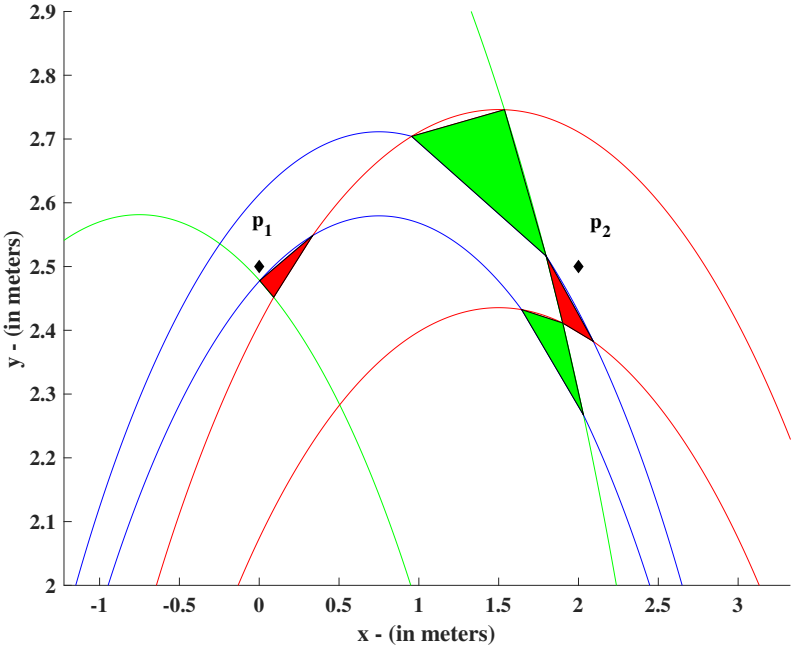

FIGURE 2.6: FECL approach for multiple targets in the presence of noise and system noise ($\sigma = 0.1$) and formation of CTT (green triangles) and STT (red triangles). Only showing two CTT.

Another challenge we address is to associate multiple sets of measurements, when available, for a particular scenario to further improve the target location estimation.

2.5 Estimation of target location when wall parameters are known

In this section, we describe an approach to estimate the target location in the presence of a wall, when complete knowledge about the wall is known. The approach was presented in [7] for mono-static radars, and we extend the approach for a bi-static scenario. When the STT is selected, 3 sides of the triangle represents 3 ellipses. For each ellipse, a range measurement is associated. So the target location needs to be estimated using 3 range measurements and the wall parameters. When the wall is homogeneous with uniform thickness and complete knowledge about the wall is available, the following approach can be used to determine the target location.

In the presence of the wall, the propagation path of the signal is as shown in Fig. 2.7. For simplicity of mathematics, the coordinates are chosen so that the wall is located on $y = 0$ axis and the (y-axis distance either from transmitter or receiver because both are parallel to the x-axis) distance between wall and radars is l. And, the i^{th} target location is at $\mathbf{p}_i(p_{i,x}, p_{i,y})$ By using Snell's law angle of refraction can be calculated as follows

$$\phi_j = sin^{-1}\left(\frac{sin\theta_j}{\sqrt{\epsilon}}\right) \text{ for } j = 1, 2 \text{ and } 3. \tag{2.12}$$

Where ϵ is the di-electric constant of the wall and θ_j is the angle of incidence of the signal on the wall.

In the Fig. 2.7, l_{j3} is the distance traveled by the reflected signal before reaching the wall, l_{j2} is the distance traveled by the reflected signal in the wall, and l_{j1} is the distance traveled by the reflected signal from the wall before reaching the receiver. In this scenario, l_{01} is the distance traveled by the transmitted signal before reaching the wall, l_{02} is the distance traveled by the transmitted signal in the wall, and l_{03} is the distance traveled by the signal before reaching the target from the wall.

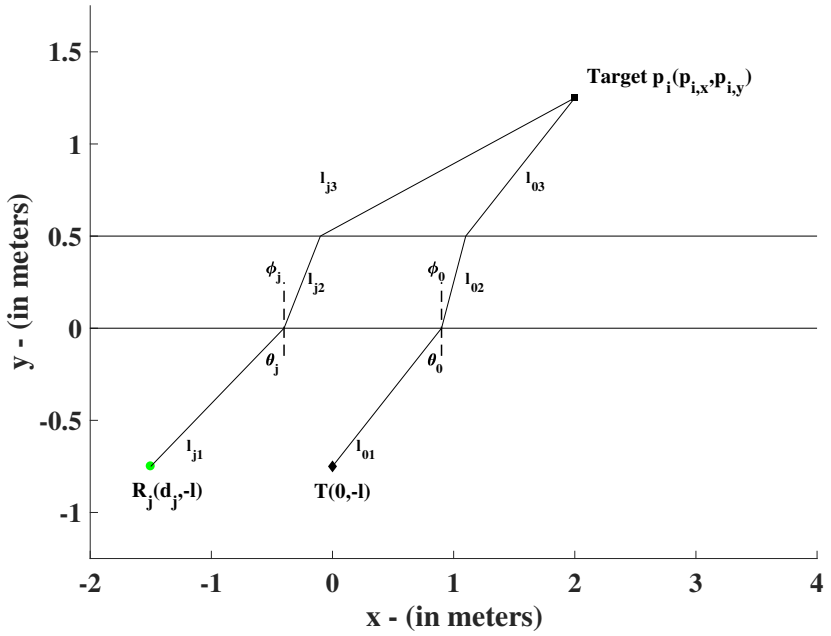

FIGURE 2.7: Propagation of path of the signal in the presence of wall

$$l_{j1} = \frac{l}{cos\theta_j} \quad l_{j2} = \frac{d}{cos\phi_j} \quad l_{j3} = \frac{p_{i,y}}{cos\theta_j} \quad \text{for } j = 0, 1, 2, \text{ and } 3 \tag{2.13}$$

Here l is the distance between radars and wall and d is the wall thickness. By applying the cosine law, the following equation is obtained

$$(p_{i,x} - d_j + (-l)tan(\theta_j))^2 + p_{i,y}^2 = l_{j2}^2 + l_{j3}^2 - 2l_{j2}l_{j3}cos(\pi + \phi_j - \theta_j) \text{ for } j = 1, 2, 3 \tag{2.14}$$

This is a transcendental equation in unknown θ_i (ϕ_i are replaced using equation 2.12). Once θ_i is calculated, l_{i1}, l_{i2}, and l_{i3} can be calculated. Using this information, the range values in the presence of wall can be calculated using the following formula.

$$\hat{r}_{i,j} = \sqrt{\epsilon}(l_{02} + l_{j2}) + l_{03} + l_{01} + l_{j1} + l_{j3} \text{ for } j = 1, 2, \text{and } 3 \tag{2.15}$$

When there is no system noise, only one set of $(p_{i,x}, p_{i,y})$ will satisfy the above equation. When there is system noise, the proposal in [7] is to use least squares, the pair of values $(p_{i,x}, p_{i,y})$ that minimizes the following equation is chosen as the target location:

$$\sum_{j=1}^{3} \left(\hat{r}_{i,j} - (\sqrt{\epsilon}(l_{02} + l_{j2}) + l_{03} + l_{01} + l_{j1} + l_{j3})\right)^2 \tag{2.16}$$

This can be solved via the Newton method, with initial guess of target location $(p_{i,x}, p_{i,y})$ being the location that was obtain by assuming the experiment has done in free space with system noise [7] (see Sec. 2.2.1).

In this thesis, we do not take this approach. Since we do not want to assume perfect knowledge about the wall, we focus on the scenario when the wall parameters are unknown, apart from statistical knowledge of μ and σ as in Eq. 2.7.

Chapter 3

Multiple Target Identification

This chapter investigates identification of multiple targets in the scenario described in Chapter 2, where there is a wall between the radar elements and the targets. As discussed in Chapter 2, Sec. 2.3, there is a data association problem in a multiple targets scenario because the receiver does not have knowledge about which measurement is associated with which target. The problem becomes harder when the number of targets increases.

In this thesis, where we are considering bi-static radar with three receivers as discussed in Chapter 2, the challenge is to identify triangles formed from ellipses that are associated with the targets. As discussed in Sec. 2.3, when there are N targets, each receiver has N range measurements. Any triangle where the 3 sides are formed by ellipses from 3 different receivers is a potential triangle which should be considered (i.e., a triangle can't be formed with two or more sides corresponding to the same receiver). So there can be up to N^3 potential triangles. Recall that the triangles will be of different sizes depending on noise and the placement of the radar elements. The triangles formed by the ellipses corresponding to a particular target and the triangle formed by ellipses corresponding to different targets both look alike in the presence of noise, the challenge is differentiate between these triangles. In this chapter, we have proposed algorithms to find N target locations among up to N^3 potential target locations by assigning confidence to each triangle.

3.1 Existing techniques

In [1], an algorithm was proposed to find N target locations. Note that they use the word "object" instead of "target", but in this chapter we will use the word "target" when describing their algorithm to maintain consistency with our own terminology. The algorithm in [1] is as follows,

1. For every potential target location assign an error metric based upon the shape of the target. In [1], the word “shape" of the target was used because the targets they consider are point targets, line targets, or any practical objects.

2. List out all the potential targets based on the error metric from low to high (i.e., high to low in likelihood order). The error metric for potential target location is not precisely defined, but it was stated that " the metric can be calculated in any reasonable manner." For example, based on how close the intersection points are, or in our particular scenario, based on the shape of the triangles. The reason for lack of precision is likely due to a desire for generality; note that [1] is a patent.

3. Pick the lowest error metric target as an actual target.

4. Remove the above-selected target from the list.

5. Based on the target that was selected as an actual target (in step:4), the list is shortened. Each potential target is given by a set of 3 range measurements (3 receivers). If a potential target is selected as an actual target, there cannot be any other target that is associated with those 3 range measurements, i.e., an ellipse can specify only one target. Potential target locations that are based on the three range measurements are removed from the list.

6. Repeat the process until all the targets are either taken as actual objects or removed from the list.

As indicated in step 3, the algorithm in [1] only uses an error metric as a constraint to find N target locations, although there are many other constraints. Their algorithm makes hard decisions based on the error metric. This a greedy approach, which tries to find N target locations in linear time. One setback of their algorithm is that sometimes it can allocate a big error to a triangle with no association ambiguities (i.e., there are no other triangles on the given ellipse).

In [14], an approach to find STTs that are not competing with other triangles was proposed, i.e., when there is only one triangle on an ellipse, that particular triangle is selected as STT. Although this approach finds STTs that are not competing, they haven’t proposed algorithm to find STTs among the set of triangles that are competing with each other. For example, when there are two triangles on an ellipse one of them must be STT (because each ellipse

corresponds to a target). The algorithm in [14] does not provide a way to resolve this ambiguity. We have observed that in the presence of the wall all the potential triangles are competing with each other, in which the proposed algorithm [14] will not be able to resolve the ambiguity.

Even combining [1] and [14] algorithms i.e filtering the STTs using algorithm in [14] before running the algorithm in [1] does not fully utilize the available constraints. For example, we can consider an ellipse as a group of triangles that has to follow certain rules (i.e., any ellipse can corresponds to only one target), and a triangle on one ellipse can be on another ellipse(i.e., a triangle is formed by 3 ellipses). Using the information based on ellipses, we can update the confidence on triangles i.e., for example, when we conclude a triangle is not STT, we can increase the confidence associated with other triangles on that ellipse. When the confidence of the triangles increases on that ellipse, the same triangles will lie on other ellipses as well. So, this effects the confidence on other triangles. By using these constraints we propose an algorithm.

3.2 Proposed techniques

The following list contains the constraints that we use to resolve the ambiguities that arise in a multiple target scenario, i.e., finding N target locations among N^3 potential target locations.

1. We select the triangle as STT if that is the only triangle on one of its ellipses because each range value (or ellipse), corresponds to only one target, so when there is only one triangle, it has to be a STT [14, 16].

2. We propose to use as the error-metric the area of that triangle, assigning a weight to each triangle based on its area. In the presence of wall and system noise, when the transmitter and receivers are collinear and parallel to the x-axis as in our scenario (as shown in Fig. 2.3). The change in y coordinates of the triangle vertices is lower than the change in x coordinates [9]. The STTs are thin and long. We observed that if we replace the area of the triangle with the perimeter (as an error metric) STTs tend to have larger values than CTTs. However, using triangle areas as the error metric, the STTs do have lower values compared to CTTs because the area is proportional to the product of the change in x-axis and change in the y-axis. Although the change in the

x-axis is high, change in the y-axis is low and less than 1 in general practical scenarios which makes area lower for STT triangles. This is the reason why we choose to assign the area of the triangle as its error metric.

3. When there is more than one triangle on an ellipse, we have observed that the triangle with the lowest area most likely corresponds to a STT. We observed that the larger the area of the triangle, the more likely it is that it corresponds to a CTT. This is because the intersection points are closer to each other when the noise levels are low.

4. We propose to pick a set of triangles as STT such that no two triangles should share a common ellipse because every ellipse corresponds to only one triangle.

Based on these constraints and motivation from coding theory two algorithms have been developed. The key idea is to think of the problem along the lines of a communication decoding theory problem. We have a set of triangles with assigned error metrics (proportional to error probabilities), and a set of constraints given by the ellipses. The key idea is to take an iterative approach to ultimately end up assigning error weights to the triangles with either a zero or one (zero being STT and one to CTT). The number of zero labeled triangles, in the end, should be equal to N (as there are N target, so N STTs). All these triangles should be mutually exclusive, i.e., no two triangles should lie on the same ellipse in the end. Even though the error probability of a particular triangle might initially be high, if it is the only triangle on the three ellipses on which its sides reside, it should be selected as STT, and we should assign a weighting to that particular triangle of zero.

3.2.1 Scheme Based on Triangles

The scheme we first propose is presented in 3 parts. First we pick triangles that are not competing with other triangles on the ellipses that form the sides of the triangle. Second, we introduce an iterative method that uses the constraints of the problem to update weights related to the confidence in each triangle. Third, we select triangles that have high confidence and form a consistent set according to the constraints. We present these three parts in the form of 3 algorithms.

The first part of the scheme, Algorithm 1, is to find ellipses that contain a single triangle. The first step of Algorithm 1 is to select the triangles that are the only triangles on the ellipses

on which one of its sides resides. This is done because each ellipse can correspond to only one target. This first step is the same as was done in [14] and [16]. Once that particular triangle is selected as STT, we propose to remove the triangles on the other ellipses on which the STT resides because there can be only one valid triangle on each ellipse (each ellipse corresponds to one target and one receiver).

The key idea is that we now check for triangles that are unique to ellipses in the reduced set of triangles. Because some triangles are competing with only the above selected STT and the other triangles that are removed in the previous step, such a triangle can now be selected as STT, where they could not previously. Now, if there are any non-competing triangle in the reduced set, they are selected as the STT. We remove the triangles that are on the ellipses on which the STT resides, and we continue this process until all single ellipse triangles are selected, and no further ones can be generated.

Algorithm 1 Single Triangle Ellipses

1: Remove mean ($\mu = 48cm$) from all range measurements.
2: Form an empty set called “STT-Set"
3: Form triangles from ellipse intersection points, such that each side of the triangle corresponds to a different receiver.
4: Put them in a set called “Potential-Triangle-Set"
5: For every triangle in “Potential-Triangle-Set" assign a Triangle-Weight as the area of that triangle.
6: **if** any triangle in Potential-Triangle-Set has one of its sides formed from an ellipse that is unique to that triangle **then**
7: Declare it to be a STT and Move into the “STT-Set".
8: Remove all triangles that have sides formed from the other two ellipses (i.e., remove them from the "POTENTIAL-Triangle-Set")
9: Goto STEP:5
10: **else**
11: **return** “Potential-Triangle-Set" and “STT-Set"
12: **end if**

The Algorithm 2 follows Algorithm 1. In Algorithm 2, we iteratively re-weight the error metrics associated with triangles using the constraints. In Algorithm 2, we compare each triangle with every other triangle in the reduced Potential-Target-Set. When they lie

on a common ellipse, we propose to increase the maximum weight among those triangle weights and decrease the other. This is the key step because by doing this we are increasing the confidence in the triangle that more likely corresponds to a STT and decreasing the confidence of the triangle that more likely corresponds to CTT. We repeat the process until the order of triangles based on triangle weights remains unchanged after comparison and re-weighting. This algorithm is shown pictorially in Fig. 3.1. This is like a decoding algorithm trying to solve for stable probabilities based on constraints. For example, for a triangle that seems to more likely to correspond to a STT when comparing with a particular triangle, the same triangle might seem to be a CTT when comparing with another triangle. The triangle that seems to be a STT in the first iteration can seem to be a CTT in another iteration by comparing with the same triangle. This is because we are comparing each triangle with every other triangle and changing the weights. By doing it this way we are not making any hard decisions until later.

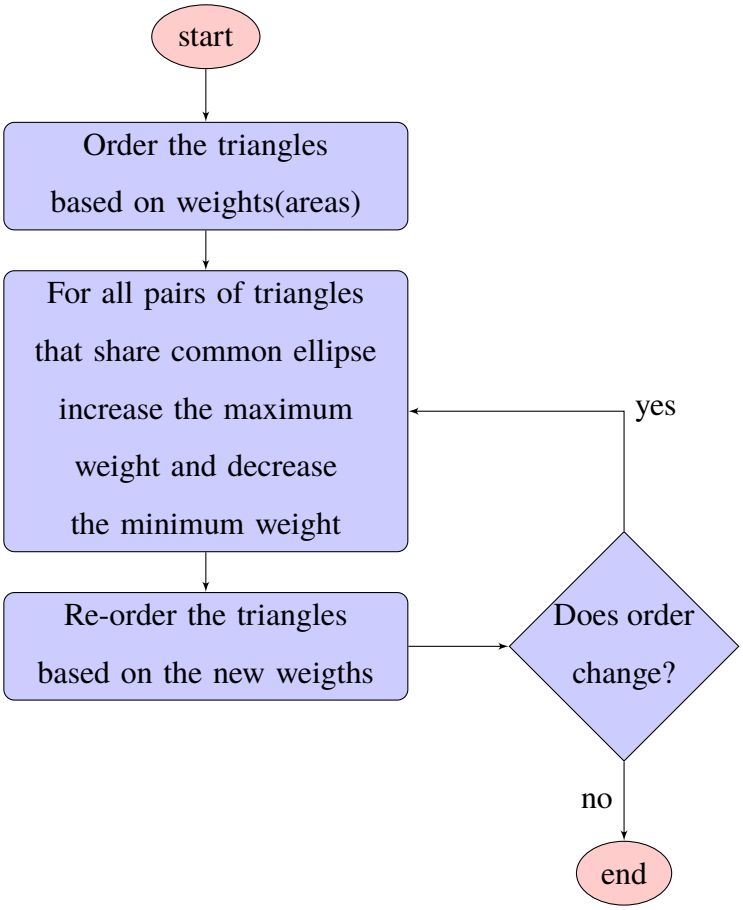

FIGURE 3.1: Pictorial Representation of Algorithm 2

Algorithm 3 follows Algorithm 2. In Algorithm 3, we propose an approach to select a set

Algorithm 2 Iterative constraint propagation

1: Order the triangles in Potential-Triangle-Set based the Triangle-Weights.
2: Assign $v \leftarrow$ number of triangles in Potential-Triangle-Set.
3: Declare $\mathbf{T}_i$ as i^{th} triangle in "Potential-Triangle-Set" and $\mathbf{w}_i$ as it's Triangle-Weight.
4: assign $i \leftarrow 1$
5: **for** $i \leq v$ **do**
6: assign $j \leftarrow i + 1$
7: **for** $j \leq v$ **do**
8: **if** $\mathbf{T}_i$ and $\mathbf{T}_j$ triangles share common ellipse **then**
9: $\min(\mathbf{w}_i, \mathbf{w}_j) \leftarrow \min(\mathbf{w}_i, \mathbf{w}_j)\big(1 - 0.1|\frac{\mathbf{w}_i - \mathbf{w}_j}{\mathbf{w}_i + \mathbf{w}_j}|\big)$
10: $\max(\mathbf{w}_i, \mathbf{w}_j) \leftarrow \max(\mathbf{w}_i, \mathbf{w}_j)\big(1 + 0.1|\frac{\mathbf{w}_i - \mathbf{w}_j}{\mathbf{w}_i + \mathbf{w}_j}|\big)$
11: **end if**
12: assign $j \leftarrow j + 1$
13: **end for**
14: assign $i \leftarrow i + 1$
15: **end for**
16: Re-order the triangles in "Potential-Triangle-Set" based on the new weights.
17: **if** order of triangles changes in STEP:16 **then**
18: GOTO STEP:2
19: **else**
20: **return** "Potential-Triangle-Set" and 'Triangle-Weights'
21: **end if**

of STTs among the potential triangles. Based on new weights after Algorithm 2, order the triangles from lowest (most likely) to highest (least likely). We pick the lowest weight triangle and check whether it fits (we define "fits" as the case when upon adding the triangle to the existing selected triangles no two triangles should lie on a single ellipse) into already existing STT. If it fits we propose to select it as STT because it is not violating the constraint-4 and it is that triangle with the lowest weight after comparing it with every other triangle generated by Algorithm 2.

Continue the above process until all the triangles are compared. After all the potential triangles are compared, we end the algorithm if there are N STT triangles.

Otherwise, when less than N triangles have been selected as STT, then we know that

either an error has been made in selection or an STT has failed to to form. Nothing can be done for the failed scenario. In error selection, at least one CTT has been selected as a STT. In this scenario we remove a combination of triangles starting from the first triangle that was selected in the initial run of Algorithm 3 from the list of potential triangles and repeat the above steps. This step is explained in Alg. 3 step. 21 - 39. We remove the initial triangle added at the beginning because we have observed that the first triangles selected have a larger impact than those selected later. We repeat the process, i.e., remove a set of triangles and check whether it then results in a set of N triangles. We stop the process when it outputs N triangles. After all the combinations are done and none of them resulted in N triangles, we output the set of triangles that are selected at the initial run Algorithm 3 (Steps 36-39 in Algorithm 3) because this might be the scenario in which one of the Single-Target-Triangles has not formed because of a large error that arose by chance from the random model we are using to model the wall.

Once the set of N triangles are selected, the problem becomes N single target problems. Based on the information about whether knowledge about the wall parameters is available or not, one of the existing techniques [7–10] can be used. When we have exact knowledge of the wall parameters, and if the wall is homogeneous with uniform thickness the proposed algorithms in [7, 10] (see Sec. 2.5) can be used to estimate the target location i.e., given transmitter and receiver locations and range measurements to particular targets (3 sides of a triangle represents 3 ellipses from 3 different receivers, each ellipse corresponds to a range value) along with wall parameters and its location can be used to estimate the target location. If we don't have knowledge about the wall parameters, the proposed algorithms in [8, 9] can be used.

In this thesis, we assume no knowledge about the wall and we use the centroid of the triangle as an estimate of the target location [9].

3.2.2 Scheme Based on Ellipses

In this section, we describe an alternative approach to find the STTs which we will then compare with the previous one. This second scheme is also presented in 3 parts. The first and third part of the algorithm are the same as that of the previous scheme based on triangles, i.e., Algorithm 1 and Algorithm 3 respectively. In the second part, we change the weights of the triangles by comparing triangles on an ellipse instead of comparing all triangles pair

wise. This is computationally less expensive. The second part of this scheme is presented in Algorithm 4.

Algorithm 4 follows Algorithm 1. In Algorithm 4, we iteratively re-weight the error metrics associated with triangles by using the constraints based on ellipses. In Algorithm 4 we compare triangles on every ellipse. When there are multiple triangles on an ellipse, we propose to decrease the triangle weight of the lowest one and increase the triangle weights of the rest. This is detailed in step 13 of Algorithm 4. This way we are increasing the confidence of the triangle which is more likely to be a STT and decreasing confidence assigned to others, as there can be only one triangle on an ellipse that is a STT. We will do this for every ellipse. We propose to continue this process until the order of triangles based on triangle weights stops changing. This algorithm is shown pictorially in the Fig. 3.2. This is also a similar way of solving a decoding problem and finding stable probabilities based on constraints because we get set of elements, out of those elements only one can be true and the rest should be false (only one STT in a set of triangles is on an ellipse). Some of the elements in a set can be common to another set (same triangle shared by two ellipses). Another set should also follow the same rules, i.e., only one element can be true and the rest should be false. By taking this approach we are not making any hard decision.

3.3 Results

In this section, we compare the simulation results of the existing algorithms with the proposed new schemes. Here, we are considering a practical scenario as shown in Fig. 3.3, with an $18m^2$ office, which is located behind the wall. Unlike in many ray-tracing studies, we do not assume internal walls can be modeled as straight line reflectors. Rather, we model the practical situation where these walls are made of plasterboard, and the RF reflectors come from the vertical metal studs that support the walls and doors. Here the targets are metal studs. In this case the aim is to estimate the target location of metal studs. By estimating the location of metal studs, we can map the office using wireless self localization and mapping (SLAM) techniques [17].

The estimated range values ($\hat{r}_{i,j}$) differ from true range values ($r_{i,j}$) i.e.,$\hat{r}_{i,j} = r_{i,j} + \Delta r_{i,j}$ due to presence of wall and system noise. Here $\Delta r_{i,j}$ is i.i.d Gaussian noise with mean ($\mu = 0.48m$) and variance σ^2. From Fig. 3.4 it can be seen that for all values σ the proposed

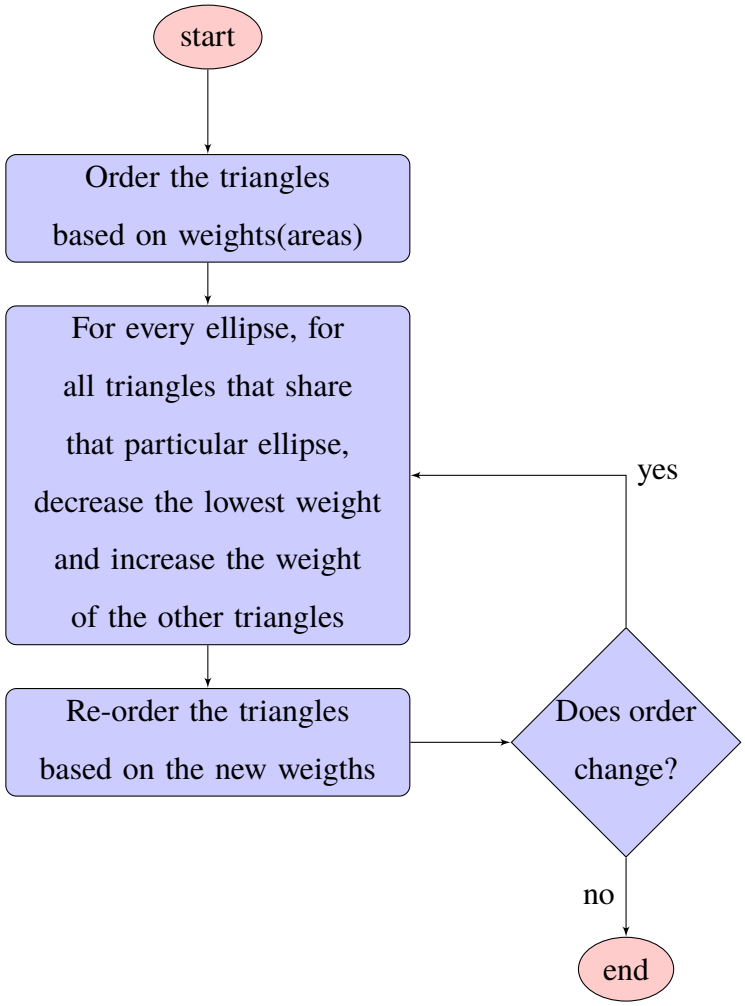

FIGURE 3.2: Pictorial Representation of Algorithm 4

schemes perform better than the algorithm in [1] with the error metric taken to be the area of the triangle. For $\sigma = 0.1$ (which is the condition for most of practical scenario in the presence of wall [9]), the proposed scheme based on triangles has error performance of 6.9% and the proposed scheme based on ellipses has error performance of 6.2%, where as the algorithm in [1] has error performance of 14.2% which shows the schemes give 50% less target identification errors than the algorithm in [1] using triangle area as the error metric. Fig. 3.5 shows the proportion of realizations for which N STTs were not found. Not all times the algorithm produces set of N triangles. In Fig. 3.5 it can be seen that the proposed schemes based on triangles and ellipses produce a set of N triangles almost all the time and perform much better than the algorithm proposed in [1]. In this simulation results, we have targets remain in the same location for all the simulations. The results tend not to be sensitive to the target locations as long as the target locations are not close each other. In the simulations

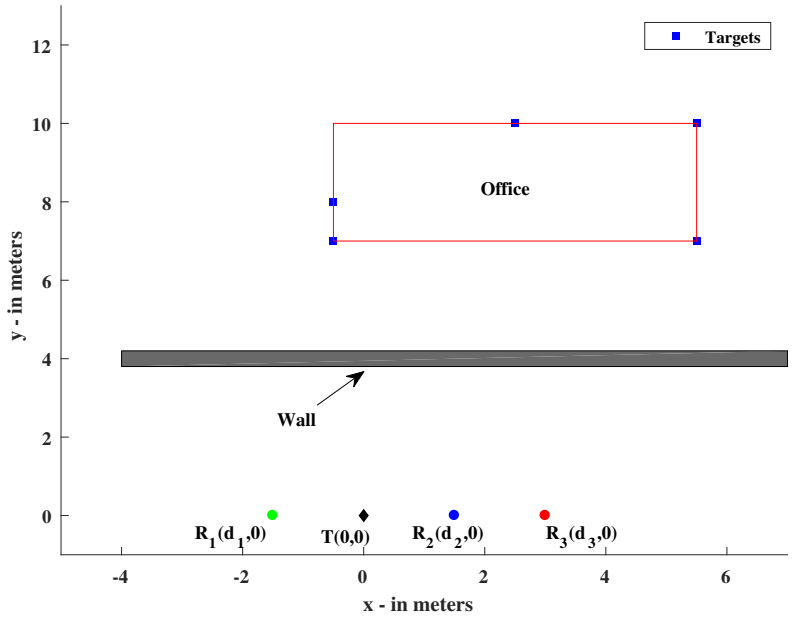

FIGURE 3.3: Target scene

with target location placed with uniform distribution over a box $6 \times 4m^2$ the result seem indistinguishable.

From plots Fig. 3.4 and Fig. 3.5, it can be seen that the two new schemes based on triangles and ellipses have similar performance. As pointed out the algorithm in [14] will not be able to resolve the ambiguity as most of the triangles are competing with each other, however the proposed schemes able to resolve the issue which is a novel contribution.

The proposed schemes in this chapter works on a single set of measurements. As the noise level increases, the proposed schemes might make an error in selecting STTs. To overcome this problem, an approach is proposed in Chapter 4, when multiple sets of measurements are available.

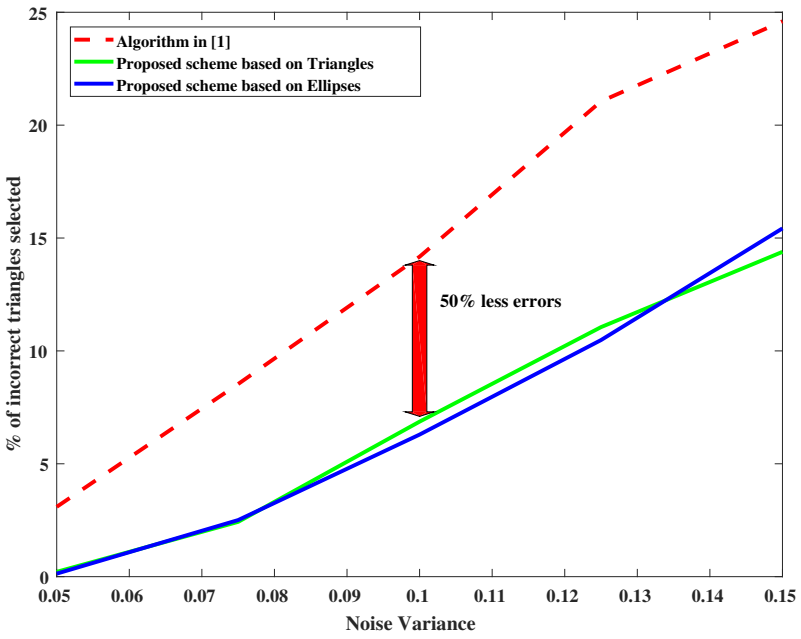

FIGURE 3.4: Error performance proposed schemes (Sec. 3.2.1 and Sec. 3.2.2) and algorithm in [1] for target scene in Fig 3.3($N = 5$ targets) for various noise levels for single set of measurements.

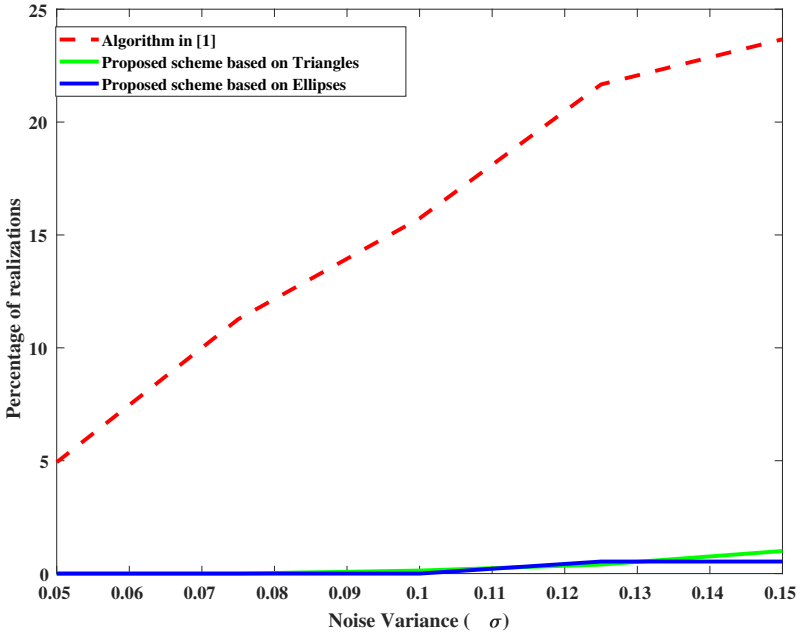

FIGURE 3.5: Percentage of realizations that do not lead to set of N triangles

Algorithm 3 Triangle Selection

Form an empty set called “Test-STT-Set"
Assign $i \leftarrow 1$
for $i \leq v$ **do**
  if $i = 1$ **then**
    Copy T_1 to “Test-STT-Set"
  else
    if T_i “fits" in Test-STT-Set **then**
      Copy T_i to Test-STT-Set
      (for definition of “fits", see Sec. 3.2.1, paragraph 5)
    end if
  end if
  $i \leftarrow i + 1$
end for
if number of triangles in {STT-Set, Test-STT-Set} = N **then**
  Move Test-STT-Set into STT-Set
  return STT-Set
else
  Move triangles from Test-STT-Set into a new set “Reference-Test-STT-Set".
  Move triangles from Potential-Triangle-Set into a new set “Reference-Potential-Triangle-Set".
  Assign $m \leftarrow$ number of triangles in Reference-Test-STT-Set.
  Create an “ordered combination list" from Reference-Test-STT-Set and call it as “CombList". (For eg. if $m = 3$, CombList = $\big[\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{1, 2, 3\}\big]$, where $\{r, s\}$ denotes r^{th} and s^{th} triangles from the ‘Reference-Test-STT-Set)
  Assign $j \leftarrow 1$
  for $j \leq 2^m - 1$ **do**
    Select j^{th} element in CombList, $\mathbf{C}_j$
    Copy all the triangles in Reference-Potential-Triangle-Set to Potential-Triangle-Set except the triangles in set $\mathbf{C}_j$.
    Run STEPS : 2 to 12
    if number of triangles in {STT-Set, Test-STT-Set} = N **then**
      Move Test-STT-Set into STT-Set
      return STT-Set
    else
      Empty Potential-Triangle-Set
      Empty Test-STT-Set
    end if
    $j \leftarrow j + 1$
  end for
  if $j = 2^m$ **then**
    Move Test-STT-Set into STT-Set
    return STT-Set
  end if
end if

Algorithm 4 Ellipses

1: Order the triangles in Potential-Triangle-Set based the Triangle-Weights.
2: Assign $v \leftarrow$ number of triangles in Potential-Triangle-Set.
3: Assign $s \leftarrow$ number of triangles in STT-Set.
4: Declare $\mathbf{T}_i$ as i^{th} triangle in “Potential-Triangle-Set" and $\mathbf{w}_i$ as it’s Triangle-Weight.
5: Number the remaining ellipses in Potential-Triangle-Set from 1 to $3(N-s)$ in no particular order.
6: Assign $i \leftarrow 1$
7: **for** $i \leq 3(N-s)$ **do**
8: pick i^{th} ellipse, $\mathbf{E}_i$.
9: Assign $p \leftarrow$ number of triangles based on $\mathbf{E}_i$.
10: Weights corresponding to those triangles in ascending order as $[\mathbf{ow}_1, \mathbf{ow}_2, \ldots, \mathbf{ow}_p]$
11: Create a vector using MatLab as follows
12: Weights-to-Assign = $[-100\ 100 : -5 : (100 - 5(p-2))]$
13: Weighting-Vector $\leftarrow 1 + \big(1./\text{Weights-to-Assign}\big)$
14: Multiply Weighting-Vector to the weights as follows
$[\mathbf{ow}_1, \mathbf{ow}_2, \ldots, \mathbf{ow}_p] \leftarrow [\mathbf{ow}_1, \mathbf{ow}_2, \ldots, \mathbf{ow}_p] \circ$ (Weighting-Vector)
15: $i \leftarrow i + 1$
16: **end for**
17: Re-order the triangles in “Potential-Triangle-Set" based on the new weights.
18: **if** order of triangles changes in STEP:16 **then**
19: GOTO STEP:4
20: **else**
21: **return** “Potential-Triangle-Set" and ‘Triangle-Weights’
22: **end if**

Chapter **4**

Multiple Realizations

In this chapter, we investigate different ways to associate measurements from multiple realizations of a particular scene. The schemes presented in Chapter 3 work on a single set of TOA measurements. In many practical situations, multiple sets of measurements will be available. For example in Wi-Fi signaling when many packets are sent at a high rate, significantly faster than any movement of the targets, it is possible to take many measurements of one scenario.

In zero noise, as discussed in Chapter 2, target location can be estimated exactly with a single set of measurements. Recall that in the presence of noise; there is a data association problem. As the noise level increases, the proposed schemes in Chapter 3 might make an error in associating range measurements to a target. When multiple sets of TOA measurements are available, triangles corresponding to a target (STT) will form around the target location, and at the same time, some of the CTT will form as a group in a particular region. The number of groups formed by CTT might be high because for every realization, up to $N^3 - N$ triangles corresponds to CTT, but only N are STT. The challenge is to find the group of STTs. The challenges while grouping is how many triangles should be considered and how much area should it be taken into consideration. In this chapter, we examine different ways to group the triangles.

In this case, with multiple realizations, we model the Gaussian measurement noise as independent between realizations. This has been observed in measurements in an anechoic chamber at CSIRO, largely due to the estimation process itself. This simple model gives us a framework to develop a measurement association algorithm that may in future be applied to scenarios in which there is significant measurement noise, perhaps due to low cost receivers, or signal changes due to moving targets.

4.1 Scheme Based On Partitioning The Area Of Interest

In this scheme, we partition the area of interest into mutually exclusive smaller regions. For every triangle, a confidence metric is associated. In the end, we present center points of the N most confident regions as the estimates of target locations. The confidence of a particular region is based on how many triangles are associated with that region and the confidence of the associated triangles (metric of the triangle). This scheme is presented in Algorithm. 5, which will be discussed now.

In Algorithm 5, we partition the area of interest in the form of a grid as shown in Fig. 4.1. For every triangle from M realizations, we associate $\frac{1}{\text{Area of the triangle}}$ as the confidence of that triangle. For every grid square, we find all of the triangles whose centroids fall into that square. We assign confidence to the square as the sum of all those triangle confidences (i.e., triangles whose centroid fall into the square). For a $N = 5$ target scenario, and noise standard deviation $\sigma = 0.1$, for 10 realizations, all the grids with confidences are shown in Fig. 4.2. At the end of the algorithm, we output the N most confident grid centers as target locations. This is shown in Fig. 4.3. We have observed that when we look at N high confident target locations, roughly one-third of actual target locations are in those N confident positions. The main drawback of this scheme is that when two targets fall with one grid square, the two targets are considered as a single target with high confidence. Even when the targets are sufficiently far apart, when the noise level is high, STT may not concentrate in a small enough region where they can be grouped. In that scenario, they might group in two separate grid squares with high confidence.

4.2 Grouping based on Circles

Although the previous scheme based on grouping has drawbacks, that scheme has given insights. When we look at the Fig. 4.2, we can see that around the target location (black dashed lines) most of the grid points have high confidence. Because of that reason we propose to group them. The challenge is how much area should we take in to consideration while grouping. To tackle the problem of how much area should we consider while grouping, we propose to use circles instead of bars. The scheme is presented in Algorithm 6.

In algorithm 6, for every triangle centroid, we assign a confidence metric proportional to the inverse of the area of that triangle as in step 9 of the algorithm. Here, the proportionality

Algorithm 5 Algorithm based on portioning the area of interest

1: List all the triangles from M realizations in “Potential-Triangles-Set".
2: Assign $q \leftarrow$ total number of triangles in set Potential-Triangles-Set.
3: For every triangle in Potential-Triangles-Set, assign $\frac{1}{\text{area of the triangle}}$ as Triangle Weight (Confidence in that particular triangle).
4: Partition the area of interest in the form gird as shown in Fig. 4.1.
5: List all center points of every grid square (center of each small square) in a set “Grid-Point-Set".
6: Initialize weight of every element in Grid-Point-Set as zero.
7: Assign $i \leftarrow 1$
8: **for** $i \leq q$ **do**
9: Pick i^{th} triangle in set Potential-Triangles-Set, $\mathbf{T}_i$ and it's weight, $\mathbf{w}_i$
10: Find the nearest point to centroid of triangle, $\mathbf{T}_i$ from the set of points in set Grid-Point-Set.
11: Let the nearest point in Grid-Point-Set be $\mathbf{G}_h$ and its corresponding weight be $\mathbf{g}_h$
12: $\mathbf{g}_h \leftarrow \mathbf{g}_h + \mathbf{w}_i$
13: $i \leftarrow i + 1$
14: **end for**
15: Output highest N confident grids with center of grids as target estimation.

constant is denoted by ζ. With centroid as a center and confidence as a radius, the circle is plotted for a triangle. In similar fashion we plot all circles corresponding to all triangles from M realizations. For a scenario of $N = 6$ targets, and noise standard deviation $\sigma = 0.1$, for $M = 10$ realizations, the circles are plotted in Fig. 4.4. In this ζ is taken as $\frac{1}{6}$. Now, the aim is to group the circles. We propose to group two circles if they intersect each other (i.e., distance between the centers is less than the sum of the radiuses). We start with most confident circle i.e., circle with larger radius. We find all the circles that intersect with that circle and group them. Now, we look at the other circles in that group and see if any other circle intersects (apart those already in the group). If there is any such circle, we add that circle in to the group. We do this until no other circle intersects with the circles in the group. We repeat this process for next highest confident circle (which is not in the group) and group them. We repeat this process until all the circles are assigned to groups. Confidence of a group is defined as sum of the confidences of the circles in the group. From the outputs we have

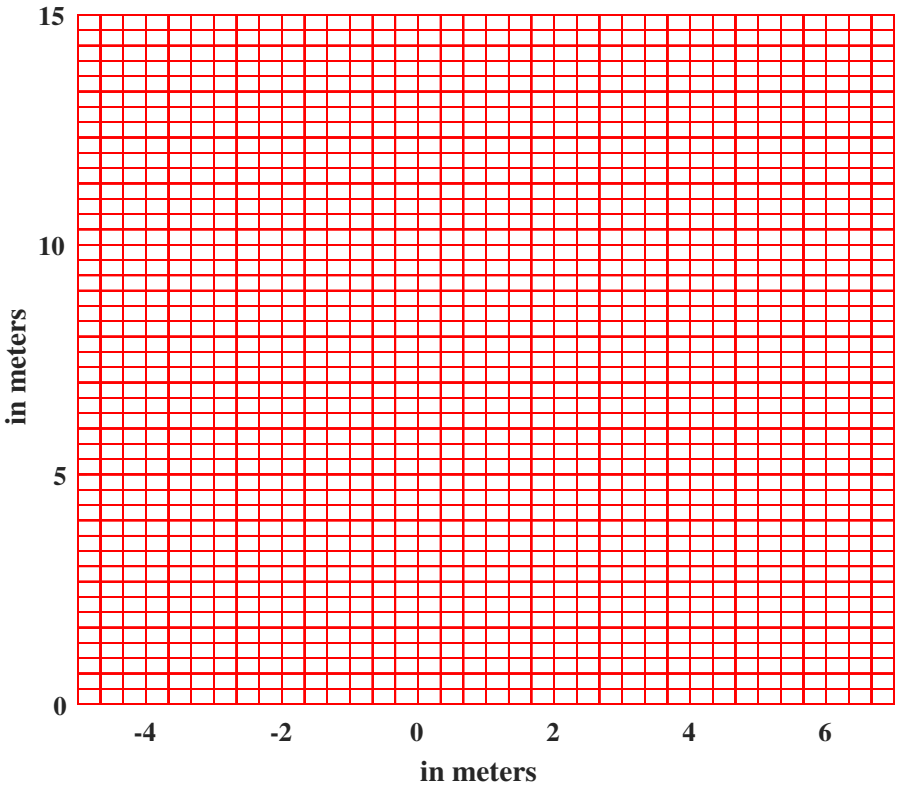

FIGURE 4.1: Partitioning the area of interest in the form of a grid

observed that if we select $2N$ high confidence groups which contains at least $M/2$ circles, the target locations seem to be always in those groups. Because of this we propose to output a maximum of $2N$ high confidence groups such that each group contains at least $M/2$ circles. The output for the scenario in Fig. 4.4 is shown in Fig. 4.5. Out of $2N$ high confidence groups N groups contains the target locations. The main drawback of this scheme is it requires us to choose a proportionality constant ζ. We have observed that the proportionality constant plays a crucial factor in the formation of clusters. If ζ is chosen to be too small, circles might not intersect each other which leads no clusters (or each cluster with with only one circle). If ζ is chosen to be too large, all circles intersect each other and they might form a single largest cluster. From the Fig. 4.5 it can be seen that on the left side of the graph instead of forming a single cluster it split into two clusters. If we pick ζ a bit larger they might form a single cluster. Another set back of the proposed scheme is when the targets are closer they might form a cluster.

Algorithm 6 Circle Grouping

List all the triangles from M realizations in a set called “Potential-Triangle-Set".
Assign weight as area of that particular triangle in Potential-Triangle-Set.
Assign $p \leftarrow$ number of elements in Potential-Triangle-Set.
Create an empty list “Circles-List".
Assign $m \leftarrow 1$
for $m \leq p$ **do**
    Calculate centroid of m^{th} triangle in Potential-Triangle-Set, $\mathbf{T}_m$ as $\mathbf{C}_m$
    Add $\mathbf{C}_m$ to Circles-List.
    Assign radius to $\mathbf{C}_m$ as $\frac{\zeta}{\mathbf{w}_m}$; where $\mathbf{w}_m$ is weight associated with $\mathbf{T}_m$.
    $m \leftarrow m + 1$
end for
Order the elements in Circles-List based on radius from high to low.
Assign $i \leftarrow 1$
Create an empty ordered list with index - i as Group_i
Select first element of Circles-List as $\mathbf{C}_{cmp}$ and radius r_{cmp}
Assign $k \leftarrow 1$
move $\mathbf{C}_{cmp}, r_{cmp}$ to Group_i
Cricles-List-Copy $\leftarrow$ Cricles-List
Assign $u \leftarrow$ size(Circles-List)
Assign $j \leftarrow 1$
for $j \leq$ u **do**
    Pick j^{th} element of Circles-List-Copy as $\mathbf{C}_j$ and radius r_j
    if $\left\|\mathbf{C}_j - \mathbf{C}_{cmp}\right\|_2 \leq r_{cmp} + r_j$ **then**
        move $\mathbf{C}_j, r_j$ from Circles-List to Group_i
    end if
end for
if $k \neq$ **size**(Group_i) **then**
    $k = k + 1$
    Pick k^{th} element in Group_i as $\mathbf{C}_{cmp}, r_{cmp}$
    if size(Cricles-List) = 0 **then**
        GOTO: STEP 42
    else
        GOTO: STEP 18
    end if
else
    if size(Cricles-List) = 0 **then**
        GOTO: STEP 42
    else
        $i = i + 1$
        GOTO: STEP 14
    end if
end if
Define confidence of a group as sum of the confidences of the circles in the group.
$r \leftarrow$ number of groups with at least $M/2$ circles.
$q \leftarrow \max(r, 2N)$
Output q most confident groups.

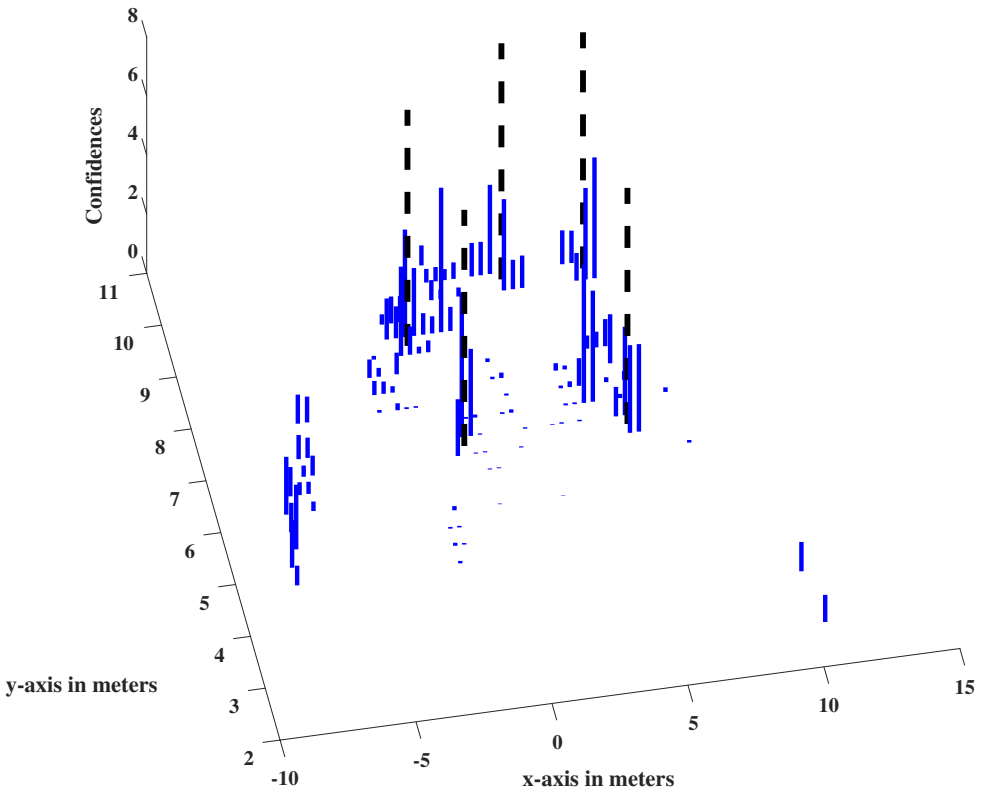

FIGURE 4.2: In this figure, black dashed lines correspond to target locations and blue lines are centers of grid squares with confidence. Height of black dashed lines have no importance as they are plotted just to show the target locations.

4.3 Clustering based on selected triangles

In this section we present the clustering technique that is based on schemes presented in Chapter 3. A cluster is defined as a group of centroids. In this section, we propose to cluster in such a way that in the end there are only N clusters. If we cluster all the triangle centroids from M realizations, the centroids in a cluster are dominated by CTT centroids as there can be up to $N^3 - N$ CTTs and only N STTs. So we propose to cluster centroids of selected STTs from the proposed schemes based on triangles or ellipses from Chapter 3, where most CTTs were eliminated by the target finding algorithm.

There are many different ways to cluster. Most generic clustering algorithms such as k-means will result in different numbers of elements (centroids) in each cluster, but the number of elements in each cluster need not be the same. The k-means algorithm will be outlined

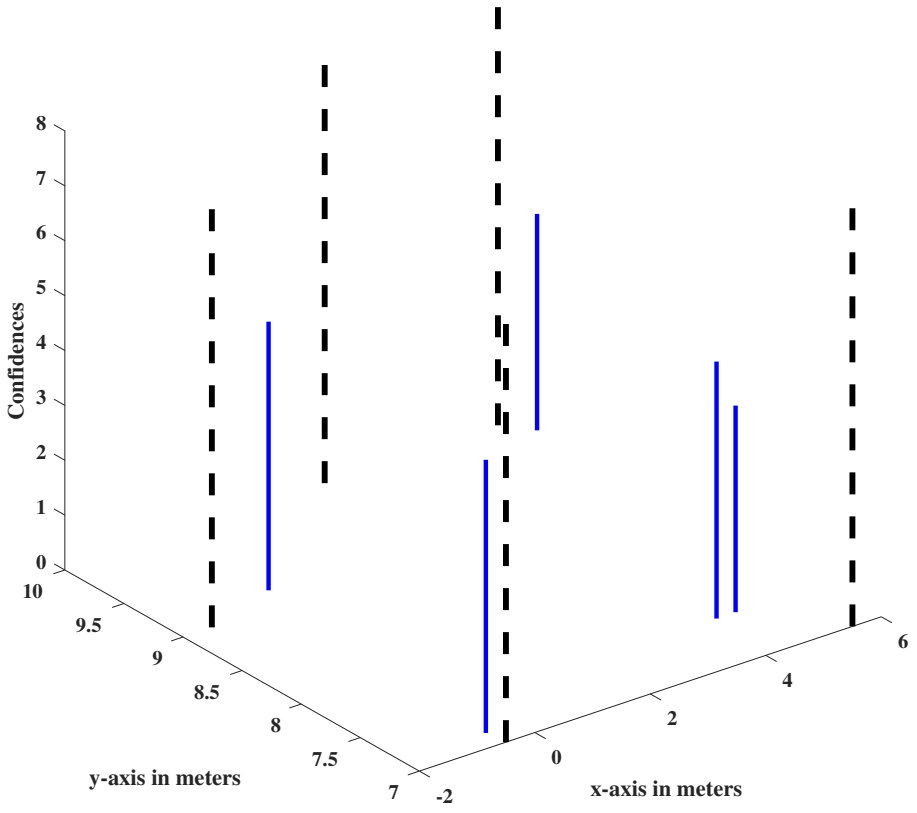

FIGURE 4.3: In this figure, black lines correspond to target locations and blue lines are centers of grids with high confidence. Height of black lines has no importance as they have plotted just to show the target locations.

below

k-means:

1. Initialize means of clusters $\mu_1, \mu_2, \mu_3, \ldots, \mu_N$ randomly. Mean of a cluster is defined as average of all the centroids in the cluster.

2. For every centroid, find the nearest point among $\mu_1, \mu_2, \mu_3, \ldots, \mu_N$. Label that point with that cluster number.

3. Calculate means of clusters $(\mu_1, \mu_2, \mu_3, \ldots, \mu_N)$ based on the labels from the step 2.

4. If means of cluster changes, go to step 2, else output $\mu_1, \mu_2, \mu_3, \ldots, \mu_N$ as target locations.

The above discussed k-means algorithm produces N clusters. Here, we include all the

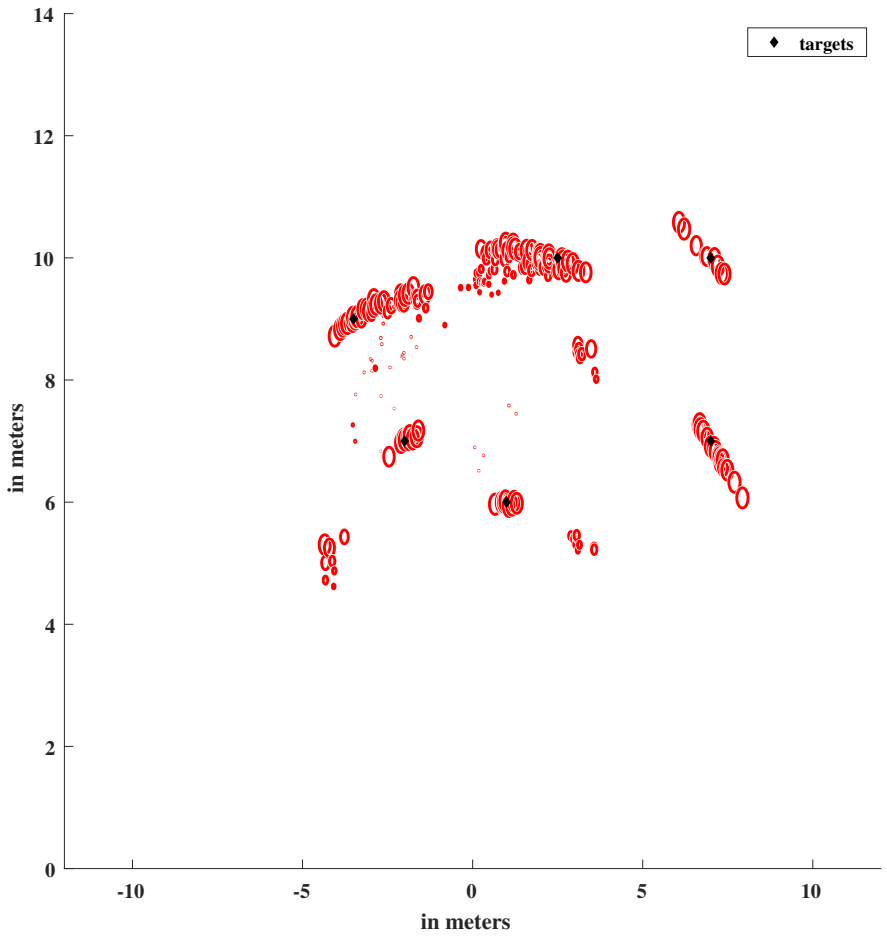

FIGURE 4.4: Output of all circles from 10 realizations

triangles from all the realizations produced by one of the schemes based on triangle or ellipses from Chapter 3, provided that scheme produced N triangles. This is a reasonable assumption because with the proposed schemes in Sec. 3.2 the output contains set of N triangles a high proportion ot the time, see Fig. 3.5. Target location can be estimated as the mean of centroids in the cluster.

Although the k-means algorithm produces N clusters, it does not guarantee each cluster has an equal number of elements (centroids) and its performance is very sensitive to initial choices made in the k-means clustering algorithm. For example, in Fig. 4.7, centroid points are plotted for 20 realizations and $\sigma = 0.1$ (which is equivalent of having wall and system noise) for the scenario of Fig. 4.6 i.e., for $N = 6$ targets. If we perform the k-means clustering to this set of data points (centroids) and if the initial points are not picked appropriately, there

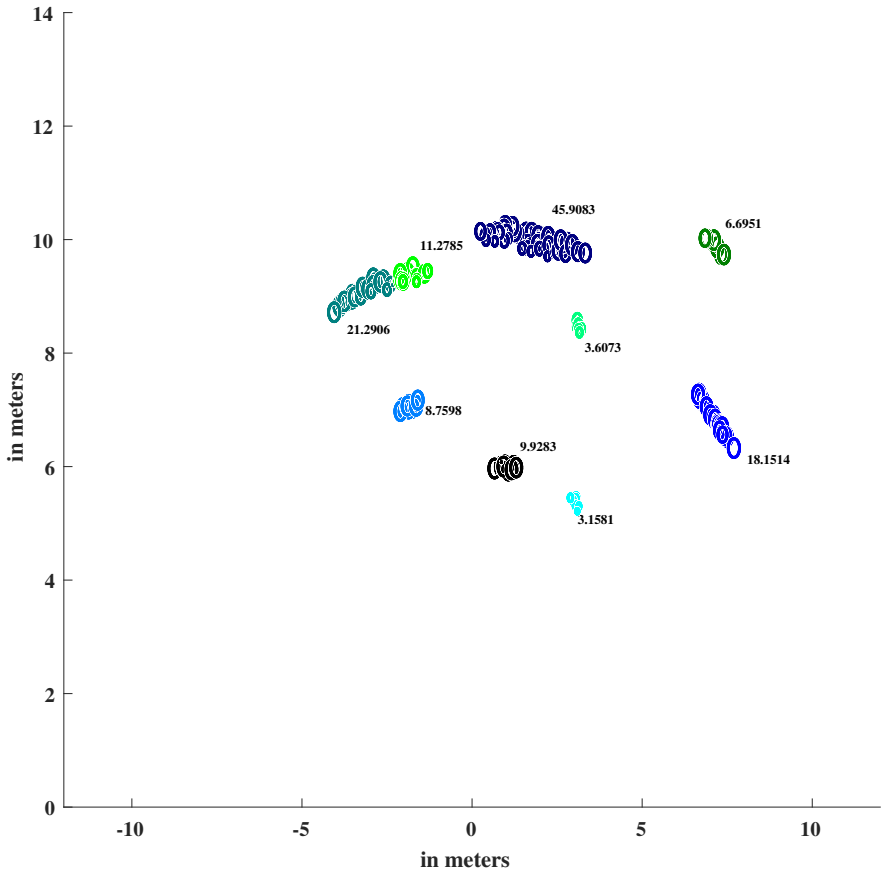

FIGURE 4.5: Output of all circle clusters that contain at least 5 circles. In this figure different colors denote different clusters. The numbers denote the confidence of each group

is high chance that outliers (i.e, left side of the graph) might form a cluster and the rest of the centroids forms 5 clusters. For the scenario of Fig. 4.6 and $\sigma = 0.1$, for 20 realizations output of k-means clustering is plotted in Fig. 4.8. In Fig. 4.8, centroids with same color belong to same cluster. From Fig. 4.8, it can be seen that two targets at the top is labeled with same. This is the main drawback of k-means clustering.

We propose a novel clustering technique taking the number of realizations M into account, i.e., every cluster must have M centroids. In this approach we will also only include the realizations that produces a set of N triangles in the schemes based on triangle or ellipses

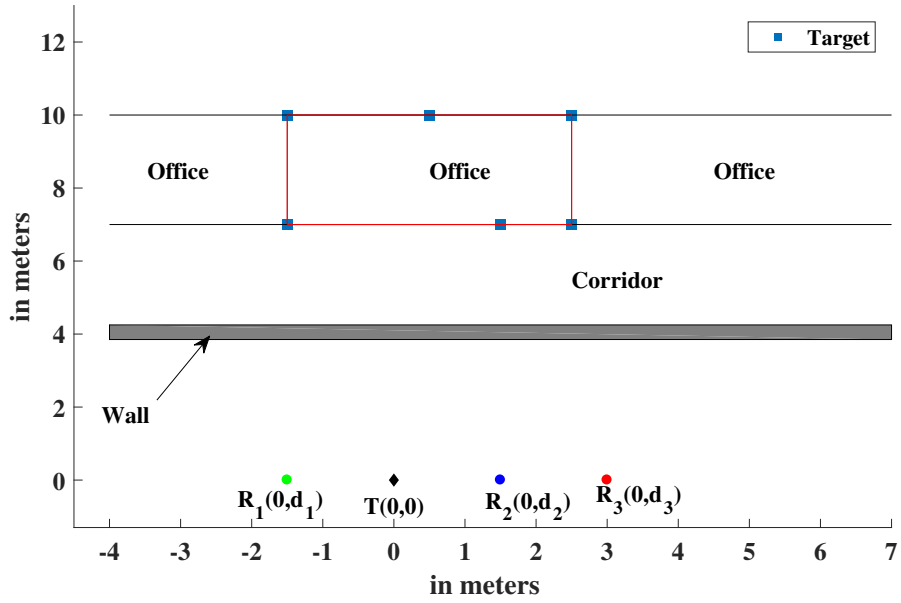

FIGURE 4.6: Target scene

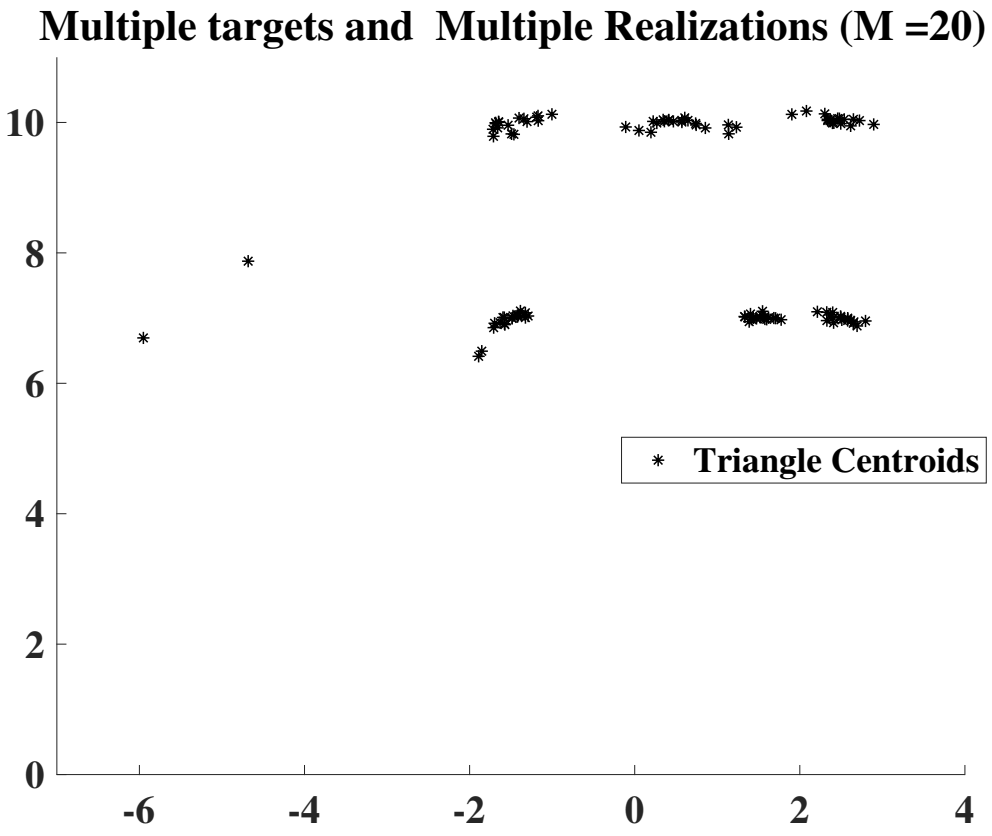

FIGURE 4.7: Triangle Centroid of selected STT for Multiple realizations for a particular target scene in Fig. 4.6 and $\sigma = 0.1$

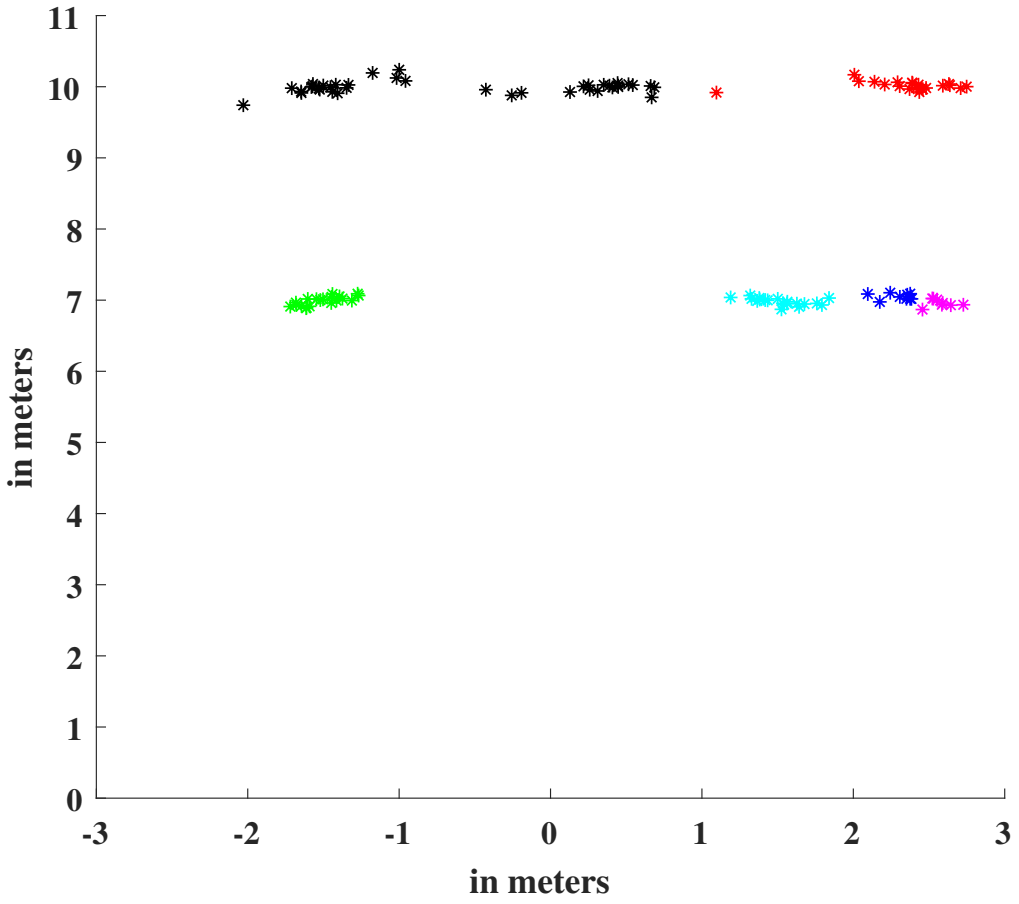

FIGURE 4.8: Output of k-means clustering algorithm for target scene in Fig. 4.6, $\sigma = 0.1$, and 20 realizations. Centroids with same color belongs to same cluster.

from Chapter 3 (so M is the number of included realizations). This scheme is presented in Algorithm 7.

In the Algorithm 7, we first find the centroids of triangles that are obtained using the schemes based on triangles or ellipses for every realization. At the first realization, we pick N triangle centroids as mean of the N clusters. From here, whenever we have a new set of centroids from a new realization, we update the means of the clusters using step 10 in Algorithm 7. For example, let the means of clusters after r realizations be as shown in Fig. 4.9 and the centroids from a new realization is as shown in Fig. 4.10. The three new centroids can be assigned to 3 mean points in 3! ways i.e., let mean points be $\mu_1, \mu_2,$ andμ_3 and new centroids point be $d_1, d_2,$ and d_3 the possible assigning are

Algorithm 7 Multi-Realizations Clustering algorithm

1: assign $M \leftarrow$ number of realizations
2: assign $i \leftarrow 1$
3: **for** $i \leq M$ **do**
4: Find N triangles using the schemes based on triangles or ellipses.
5: Find Centroids of each triangle and store in matrix newCentroid.
6: **if** $i = 1$ **then**
7: meanCluster = newCentroid
8: **else**
9: Permutate the elements in matrix newCentorid, such that distance between the matrices meanCluster and newCentroid is minimized.
10: meanCluster $\leftarrow \frac{(i-1)*\text{meanCluster}+\text{newCentroid}}{i}$
11: **end if**
12: assign $i \leftarrow i + 1$
13: **end for**
14: **return** meanCluster

1. $(\mu_1, d_1), (\mu_2, d_2),$ and (μ_3, d_3)
2. $(\mu_1, d_2), (\mu_2, d_3),$ and (μ_3, d_1)
3. $(\mu_1, d_3), (\mu_2, d_1),$ and (μ_3, d_2)
4. $(\mu_1, d_1), (\mu_2, d_3),$ and (μ_3, d_2)
5. $(\mu_1, d_3), (\mu_2, d_2),$ and (μ_3, d_1)
6. $(\mu_1, d_2), (\mu_2, d_1),$ and (μ_3, d_3)

Here we denote (a, b) as assigning centroid b to the cluster whose mean is a. Out of those 6 combinations, we choose the combination such that sum of the euclidean distances between mean points and assigned centroids is minimized. This is the key step because, by doing this, we are ensuring that each element in a cluster is from a different realization and every cluster will have an equal number of centroids. After assigning the centroids to the means, we update the means by giving a weight of $\frac{r}{r+1}$ to the previous mean points and a weight of $\frac{1}{r+1}$ to new centroid. This is detailed in step 10 of the algorithm. We repeat this procedure for

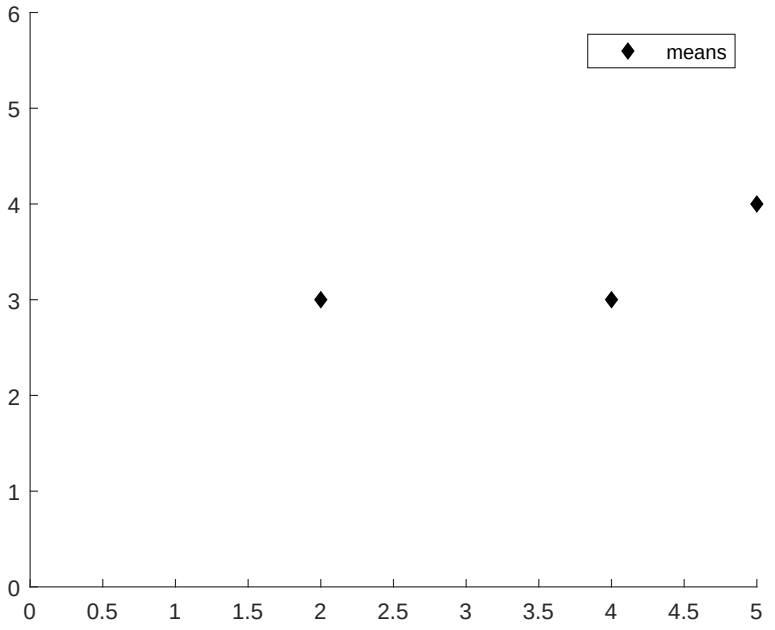

FIGURE 4.9: Means after r-realizations

every realization. In the end, we present the means as the estimated target locations. Output of the proposed clustering technique for the same data as in Fig. 4.8 is plotted in Fig. 4.11.

4.4 Results

Fig. 4.12 is plotted to compare the performance of our proposed clustering technique with the k-means clustering technique for different numbers of realizations. The realizations are taken for the target scene depicted in Fig. 4.6 i.e., with $N = 6$ targets and $\sigma = 0.1$. From Fig. 4.12 it can seen that the proposed algorithm preforms better than the traditional k-means algorithm. For the proposed algorithm, as the number of realizations increases, the average estimation error per target decreases. For 20 realizations the proposed clustering technique has 80% error reduction compared to the k-means clustering technique. We have considered only those realizations that produce a set of N triangles from the scheme based on triangles. The k-means algorithm does not seem to improve with more realizations. In this simulation results, we have targets remain in the same location for all the simulations. However, the results are not sensitive to the target locations as they are dependent on the outcome of the

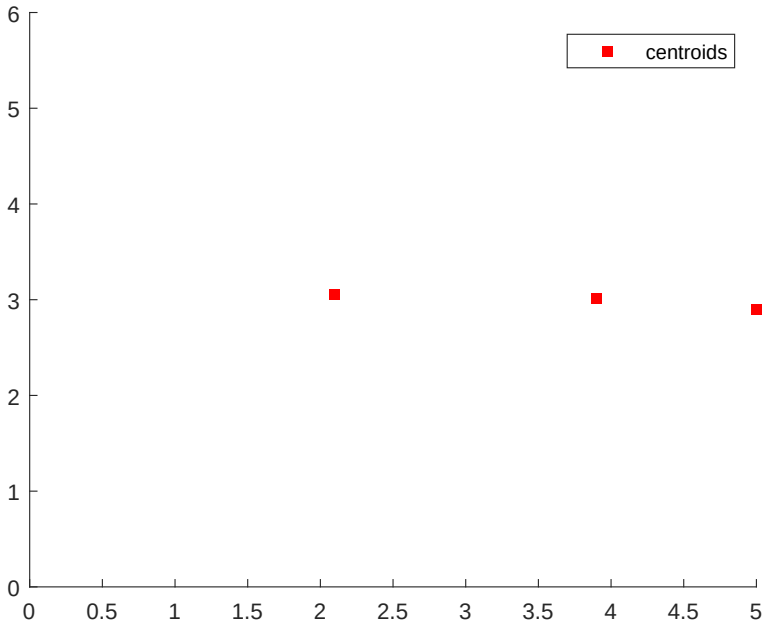

FIGURE 4.10: centroids from the $(r+1)^{th}$ realization

Proposed Schemes based on the Ellipses and the Proposed Scheme Based on Triangles, which are not sensitive to the target locations.

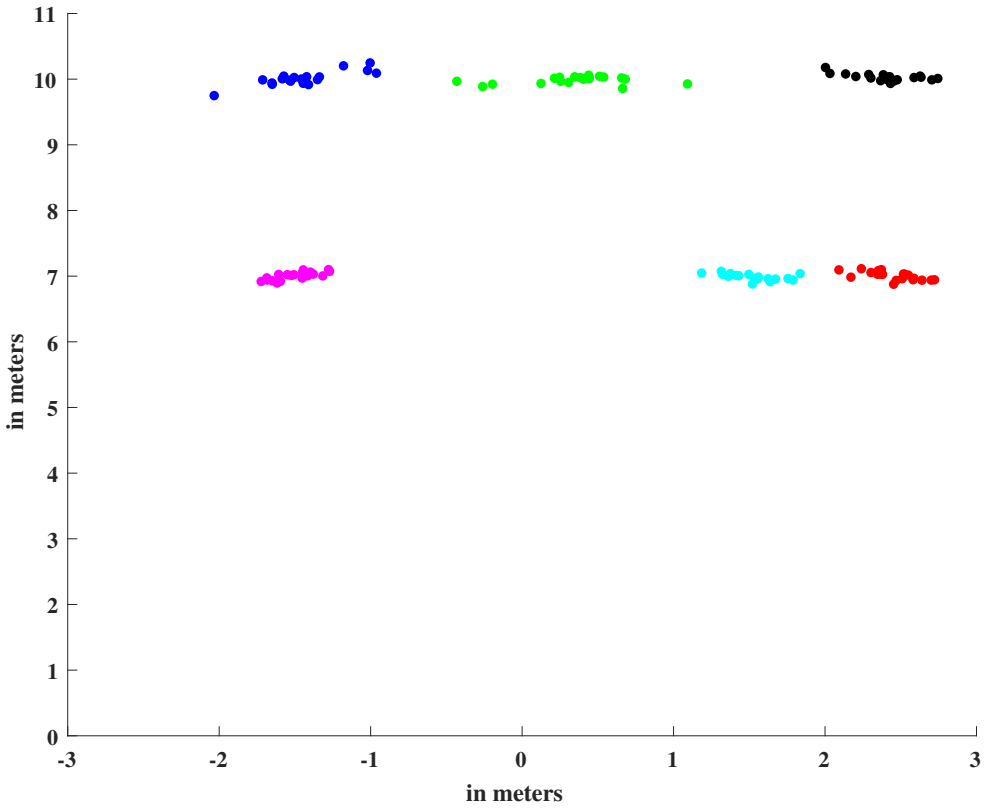

FIGURE 4.11: Output of proposed clustering technique for the same data as in Fig. 4.8. Centroids with same color belongs to same cluster.

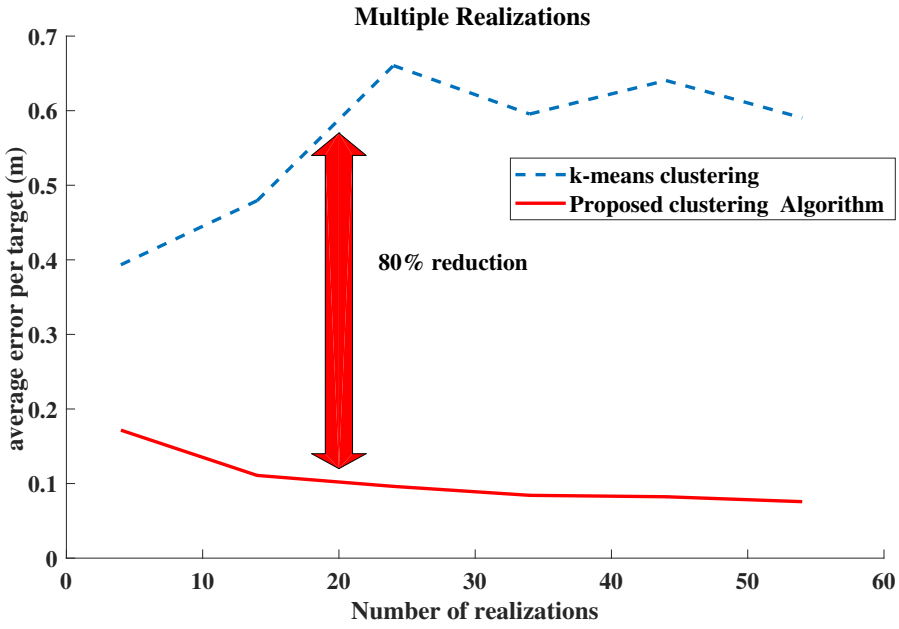

FIGURE 4.12: Error performance for target scene Fig. 4.6 ($N = 6$ targets) for multiple realizations and $\sigma = 0.1$

Chapter 5

Conclusion

In this thesis we have reviewed non-coherent bi-static radar, and related target finding algorithms, and in particular the area of through-the-wall radar. We then focus on identifying and locating multiple targets. We have developed two novel target finding algorithms inspired by the methods of iterative decoding in communications theory to address the uncertainty that arises from the effect of an unknown wall between the targets and the radars. With this uncertainty, a significant part of the problem is associating the measurements with the correct targets. We show that the proposed algorithms have similar performance and both achieve about 50% less target identification errors compared to existing algorithms. We have also addressed the problem of improving the estimates using multiple measurements of the same scene under an assumption of independence of errors across measurements. We have derived a novel clustering algorithm that clusters different estimates together based on Euclidean distance utilizing the constraint that each cluster must have the same number of measurements. In an example, we show that our clustering algorithm can achieve 80% error reduction compared to the k-means clustering technique.

References

[1] M. J. Paradie and B. D. Labitt. *Method and apparatus for identifying complex objects based on range readings from multiple sensors* (2003). US Patent 6,664,918.

[2] M. A. Richards, J. Scheer, W. A. Holm, and W. L. Melvin. *Principles of modern radar* (Citeseer, 2010).

[3] B. R. Mahafza. *Radar Systems Analysis and Design Using MATLAB Second Edition* (Chapman and Hall/CRC, 2005).

[4] C. Cook. *Radar signals: An introduction to theory and application* (Elsevier, 2012).

[5] M. I. Skolnik. *Introduction to radar*. Radar Handbook **2** (1962).

[6] D. R. Wehner. *High resolution radar*. Norwood, MA, Artech House, Inc., 1987, 484 p. (1987).

[7] F. Ahmad and M. G. Amin. *Noncoherent approach to through-the-wall radar localization*. IEEE Transactions on Aerospace and Electronic Systems **42**(4) (2006).

[8] X. Yuan, L. Kong, and Y. Jia. *Through-wall-radar target localization based on random sparse array*. In *Radar (Radar), 2011 IEEE CIE International Conference on*, vol. 2, pp. 1331–1334 (IEEE, 2011).

[9] X. Huang, G. Cui, S. Guo, L. Kong, and X. Yang. *A novel noncoherent localization approach for through-the-wall radar*. In *Radar Conference (RadarConf), 2017 IEEE*, pp. 1245–1249 (IEEE, 2017).

[10] S. Jia, L. Kong, and B. Liu. *Ellipse-cross-localization accuracy analysis of through-the-wall radar*. In *Radar Conference, 2009 IEEE*, pp. 1–4 (IEEE, 2009).

[11] F. Ahmad, M. G. Amin, and S. A. Kassam. *Synthetic aperture beamformer for imaging through a dielectric wall*. IEEE transactions on aerospace and electronic systems **41**(1), 271 (2005).

[12] D. D. Ferris and N. C. Currie. *Survey of current technologies for through-the-wall surveillance (tws)*. In *Sensors, C3I, Information, and Training Technologies for Law Enforcement*, vol. 3577, pp. 62–73 (International Society for Optics and Photonics, 1999).

[13] D. D. Ferris and N. C. Currie. *Microwave and millimeter-wave systems for wall penetration*. In *Targets and Backgrounds: Characterization and Representation IV*, vol. 3375, pp. 269–280 (International Society for Optics and Photonics, 1998).

[14] H. Rabe, E. Denicke, G. Armbrecht, T. Musch, and I. Rolfes. *Considerations on radar localization in multi-target environments*. Advances in Radio Science 7 (2009) **7**, 5 (2009).

[15] I. G. Cumming and F. H. Wong. *Digital processing of synthetic aperture radar data*. Artech house **1**(2), 3 (2005).

[16] A. Samokhin, I. Ivashko, and A. Yarovoy. *Algorithm for multiple targets localization and data association in distributed radar networks*. In *Radar Symposium (IRS), 2014 15th International*, pp. 1–6 (IEEE, 2014).

[17] S. Thrun and M. Montemerlo. *The graph slam algorithm with applications to large-scale mapping of urban structures*. The International Journal of Robotics Research **25**(5-6), 403 (2006).