# Analyzing Hierarchical Clustering Algorithm for Track-to-Track Association in Diverse Scenarios

By
**Hira Arshad**
00000326114

Supervisor
**Dr. Mian Ilyas Ahmad**

*MASTER OF SCIENCE*
in
*Computational Sciences and Engineering*

School of Interdisciplinary Engineering & Science
National University of Sciences and Technology (NUST)
Islamabad, Pakistan

January, 2024

This thesis is dedicated to *my beloved parents & siblings.*

Annex A to NUST Letter No. 0972/102/Exams/Thesis-Cert dated 11 January, 2024.

## THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis written by Ms **Hira Arshad** Registration No. **326114** of **SINES** has been vetted by undersigned, found complete in all aspects as per NUST Statutes/Regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature with stamp: ______________

Name of Supervisor: Dr. Mian Ilyas Ahmad

Date: 12/01/2024.

Signature of HoD with stamp: ______________ Mian Ilyas Ahmad HoD Engineering Professor SINES - NUST, Sector H-12 Islamabad

Date: 12/01/2024

**Countersign by**

Signature (Dean/Principal): ______________

Date: 15/01/2024

# Declaration

I, *Hira Arshad*, declare that my MS thesis titled ***Analyzing Hierarchical Clustering Algorithm for Track-to-Track Association in Diverse Scenarios*** is my work and has not been submitted previously by me for taking any degree from this University "*National University of Sciences and Technology (NUST), Pakistan*" or anywhere else in the country/world.

At any time, if my statement is found to be incorrect even after my post-graduation, the University has the right to withdraw my MS Degree.

Hira Arshad,
00000326114

# Copyright Notice

# Acknowledgments

*"Read. Read in the name of thy Lord who created; [He] created the human being from a blood clot. Read in the name of thy Lord who taught by the pen: [He] taught the human being what he did not know"*

The importance of education lies in the first Revelation of the Quran, and this proved to be my motivation. All praises to **ALLAH**, who gave me the knowledge, patience, health, and ability to complete this thesis.

First and foremost, I sincerely thank my research supervisor, Dr. Mian Ilyas Ahmad, for his complete support and encouragement throughout my research, particularly in the dubious phases. His constructive advice and suggestions have allowed this research to move in the direction it moved. His support has also instilled confidence in me to become a self-dependent researcher. I would also like to acknowledge my GEC members, Dr. Muhammad Tariq Saeed and Dr. Salma Sherbaz, for their advice and moral support.

I want to offer my special thanks to Mohsin Zubair, a friend and project fellow of mine, for his persistent support, encouragement, and boosts of motivation whenever self-doubt engulfed me. I am grateful to everyone who helped my master's dissertation be completed successfully for their timely discussions and advice, which helped me get through the difficult phases of my research. The faculty and management of SINES, NUST, proved to be extremely resourceful and have helped create a supportive and productive environment within the department.

Last but not least, my deepest gratitude goes to my beloved parents, who are the reason behind all my successes, and my siblings' endless prayers and love.

# List of Abbreviations

| | |
|---|---|
| HAC | Hierarchical Agglomerative Clustering |
| VOS | Validation Optimization System |
| TN | True Negative |
| TP | True Positive |
| FN | False Negative |
| FP | False Positive |
| RADAR | Radio Detection And Ranging |
| LIDAR | Light Detection And Ranging |
| ESM | Electronic Support Measures |
| 3D | Three-Dimensional |
| RCS | Radar Cross Section |
| SNR | Signal-to-Noise Ratio |
| GPS | Global Positioning System |
| APD | Avalanche Photodiode |
| PI | Probability of Intercept |
| UKF | Unscented Kalman Filter |
| EKF | Extended Kalman Filter |
| ECI | Earth-Centered Inertial |
| LLA | Latitude, Longitude, Altitude |
| ECEF | Earth-Centered Earth-Fixed |
| AoA | Angle of Arrival |
| PDAF | Probabilistic Data Association Filter |
| KF | Kalman Filter |
| SMC | Sequential Monte Carlo |
| DBSCAN | Density-Based Spatial Clustering of Applications with Noise |
| HPT | High Priority Target |
| SPT | Secondary Priority Target |
| GHz | Gigahertz |
| ID | Identification / Identifier |

# Contents


**1 Introduction** **1**
- 1.1 Track-to-Track Association . . . . . . 1
- 1.2 Motivation . . . . . . 3
- 1.3 Problem Statement . . . . . . 4
- 1.4 Research Objectives . . . . . . 5
- 1.5 Areas of Application . . . . . . 5
- 1.6 Thesis Layout . . . . . . 6

**2 Literature Review** **7**
- 2.1 Sensors Basics . . . . . . 7
  - 2.1.1 RADAR . . . . . . 8
  - 2.1.2 LIDAR . . . . . . 11
  - 2.1.3 ESM . . . . . . 13
  - 2.1.4 Object Tracking . . . . . . 16
- 2.2 Frames of References . . . . . . 18
  - 2.2.1 Earth Centered Inertial (ECI) . . . . . . 19
  - 2.2.2 Earth Centered Earth Fixed Frame (ECEF) . . . . . . 21
  - 2.2.3 Local Geodetic Frame . . . . . . 23
- 2.3 Filtering for Object Tracking . . . . . . 25
  - 2.3.1 Kalman Filter (KF) . . . . . . 26
  - 2.3.2 Unscented Kalman Filter (UKF) . . . . . . 29
  - 2.3.3 Extended Kalman Filter (EKF) . . . . . . 31
  - 2.3.4 Particle Filter . . . . . . 31
- 2.4 Track-to-Track Association . . . . . . 32
  - 2.4.1 Clustering Based Approaches . . . . . . 34
  - 2.4.2 Statistical Analysis Methods . . . . . . 35
  - 2.4.3 Fusion Methods . . . . . . 36

**3 Methodology** **37**
- 3.1 Preprocessing . . . . . . 38
  - 3.1.1 Data Extraction . . . . . . 38
  - 3.1.2 Data Conversion . . . . . . 38
  - 3.1.3 Local Tracks Formation . . . . . . 39
  - 3.1.4 Error Mitigation Using Standard Deviation . . . . . . 39

3.2 Time Synchronization . . . 40
3.2.1 Kalman Filtering . . . 41
3.3 Cluster Formation using HAC Algorithm . . . 42
3.3.1 Distance Matrix Formulation . . . 43
3.3.2 Global Threshold Calculation . . . 43
3.3.3 Single Linkage Method . . . 44
3.3.4 Cluster Formation . . . 44
3.4 Accuracy Calculation . . . 45
3.4.1 Ground Truth Comparison . . . 46
3.5 Computational Aspects of Association . . . 46

**4 Results and Discussion** **48**
4.1 Two Sensors One Target . . . 48
4.2 Two Sensors Two Targets . . . 50
4.3 Two Sensors Four Targets . . . 53
4.4 Three Sensors Four Targets . . . 55
4.5 Three Sensors Three Targets . . . 58
4.6 Accuracy Computations in HAC Algorithm . . . 60

**5 Conclusion and Future Work** **62**
5.1 Conclusion . . . 62
5.2 Future Work . . . 62

# List of Figures


1.1 Two sensors tracking one object . . . . . . . . . . . . . . . . . . . . 2
1.2 Sensors sharing data via common link . . . . . . . . . . . . . . . . 3
1.3 Growth in Sensor Deployment and Data Generation (2014–2019) [44] . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.1 Block diagram of a RADAR system [32] . . . . . . . . . . . . . . . 9
2.2 Structure of conventional LiDAR sensors [43] . . . . . . . . . . . . 11
2.3 Earth-space slant range of signal below freezing point [38] . . . . 17
2.4 Earth-Centered Inertial (ECI) Frame [22] . . . . . . . . . . . . . . 20
2.5 Earth-Centered Earth-Fixed (ECEF) coordinates [48] . . . . . . 22
2.6 Local Geodic Frame of Reference [25] . . . . . . . . . . . . . . . . 24

3.1 Block Diagram of Methodology . . . . . . . . . . . . . . . . . . . . 37
3.2 Asynchronous and Synchronized Tracks . . . . . . . . . . . . . . . 41
3.3 Computational Aspects of Association . . . . . . . . . . . . . . . . 46

4.1 Scenario 1: Two Sensors One Target . . . . . . . . . . . . . . . . . 48
4.2 VOS scenario 1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
4.3 Dendrogram for Scenario 1 . . . . . . . . . . . . . . . . . . . . . . 50
4.4 Accuracy of 2 sensors tracking 1 object . . . . . . . . . . . . . . . . 50
4.5 Scenario 2: Two Sensors Two Targets . . . . . . . . . . . . . . . . 51
4.6 VOS scenario 2 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51
4.7 Dendrogram for Scenario 2 . . . . . . . . . . . . . . . . . . . . . . 52
4.8 Accuracy of 2 sensors tracking 2 objects . . . . . . . . . . . . . . . 52
4.9 Scenario 3: Two Sensors Four Targets . . . . . . . . . . . . . . . . 53
4.10 VOS scenario 3 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 54
4.11 Dendrogram for Scenario 3 . . . . . . . . . . . . . . . . . . . . . . 54
4.12 Accuracy of 2 sensors tracking 4 objects . . . . . . . . . . . . . . . 55
4.13 Scenario 4: Three Sensors Four Targets . . . . . . . . . . . . . . . 56
4.14 VOS Scenario 4 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 56
4.15 Dendrogram for Scenario 4 . . . . . . . . . . . . . . . . . . . . . . 57
4.16 Accuracy of 3 sensors tracking 4 objects . . . . . . . . . . . . . . . 57
4.17 Scenario 5: Three Sensors Three Targets . . . . . . . . . . . . . . . 58
4.18 VOS scenario 5 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 59
4.19 Dendrogram for Scenario 5 . . . . . . . . . . . . . . . . . . . . . . 59

4.20 Accuracy of 3 sensors tracking 3 objects . . . . . . . . . . . . . . . 60

# Abstract

In applications such as self-driving cars and swarms of aerial vehicles involving a network of sensors, the concept of track-to-track association plays an important role in their accurate surveillance and threat evaluation. The basic idea is to efficiently group together redundant tracks associated with the track of an object in a network of multiple sensors tracking multiple objects. Tracks are either local tracks (observed by the sensor itself) or shared tracks (shared by other sensors in the network), and they record the location of the objects under observation. Due to sensor inaccuracies and network biases, the shared tracks and local tracks are often mismatched, and it is not clear whether the objects being tracked are the same or different. Various researchers have worked on the problem of track-to-track association and have developed multiple association algorithms with the primary focus on increasing the accuracy as the number of sensors and targets increases. Among these approaches, the HAC algorithm has been used for some specific scenarios but not generalized to multiple sensors and targets. In this thesis, an indigenously developed scenario generator has been utilized to create scenarios and perform HAC-based associations on the corresponding network packets. The performance measurement of the HAC algorithm in different scenarios in terms of quantified accuracy calculation using ground truth data is the main contribution of this work. The HAC algorithm starts with the pre-processing of network packets, which involves extraction of tracks; conversion of location from latitude, longitude, and altitude (LLA) to Earth-Centered Earth-Fixed (ECEF) frame and then to sensor local frame; finally, computation of sensor-specific standard deviations. The next step is time synchronization, where each track is estimated at a fixed association time using a Kalman filter or other estimation technique. This means that the track data from shared and local tracks have been estimated/projected at a specific time. The next step is hierarchical top-down cluster formation based on a predefined threshold, allowing the merger of clusters to obtain the resulting associated clusters/tracks. The final step is accuracy calculation, where the associated tracks and ground truth data are compared to validate the association results. Multiple scenarios have been created to validate the performance of track-to-track association, and the clustering algorithm's accuracy has been computed for each scenario. The accuracy varies with estimation and threshold calculations. It is shown that by carefully selecting the threshold, the hierarchical agglomerative clustering (HAC) algorithm results in an accuracy of more than 95%. This is also observed in multiple sensors and multiple target cases.

**Keywords:** *Track-to-Track Association, Multi-Object, Multi-Sensor, Sensor Biases, Hierarchical Agglomerative Clustering (HAC).*

# Chapter 1

# Introduction

In this chapter, the fundamental concepts related to our research work will be introduced, followed by the problem statement, research objectives, and the motivation or applications of the research work. The chapter concludes with an overview of the entire thesis.

## 1.1 Track-to-Track Association

In this section, we introduce the concept of track-to-track association and briefly discuss some related terminologies. We begin with the idea of track formation, which monitors the object of interest. While in motion in specific directions, the sensor and the object under observation can be tracked through measurements. Track formation begins when the object enters the sensor's detection range. Once the object is within this range, the sensor starts to observe and record its location. Multiple objects can be observed, and each object is assigned a unique label (ID) for identification. The sensor continuously records their path as the objects move around as long as they remain in the sensor's range. It's like tracing their journey on a map. Typical dimensions for the location of the objects are the latitude, longitude, and altitude of the objects. This continuous set of observations of the object locations is called the 'track' and is available at each sensor sampling time.

Sensors are designed to detect and observe the objects of interest. **Local tracks** are specific to a single sensor. They capture the movements of objects within that sensor's field of view. It's like the sensor's personal diary of the things it monitors.

On the other hand, **shared tracks** go beyond a single sensor. They represent the combined efforts of multiple sensors, collaborating to track objects that move across their observation areas. Shared tracks provide a more comprehensive view of an object's journey as various sensors contribute to the story.

**Track-to-track Association:** The track-to-track association uses observations and collected data from local and shared tracks to establish meaningful connections. It commences with the initial observations recorded by multiple sensors, such as radars and lidars, collaborating to detect and monitor objects in their respective views. These sensors have unique data processing systems that introduce uncertainties that cause deviations between perceived and actual object positions, which may be influenced by sensor biases or data noise.

These sensors can share information through a process that includes establishing a wireless data link that connects these sensors to facilitate communication, enabling them to share position and object information through data packets. However, challenges arise as these packets of tracks arrive out of sequence due to latency in sensor systems, the data link, and environmental factors. The data collected from these tracks enhances the collective observational capacity of sensors, which helps understand the tracked objects and their journeys on the map. The important task of track-to-track association emerges among sensor biases and asynchronous data complexities. This task aims to determine whether the data from multiple sensors, influenced by biases, represents the same or different objects.

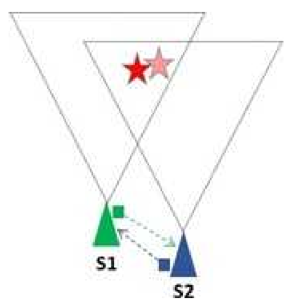



Figure 1.1: Two sensors tracking one object

A fundamental scenario is depicted in Figure 1.1, in which two sensors actively monitor a shared area of interest. Their combined efforts focus on tracking a single object of interest. Once they establish a data link amongst their local and shared tracks at a time instance, they collaborate by sharing information regarding their respective positions and the object they are tracking in their field of view. This sharing of information is facilitated through the transmission of data packets. This approach is essential for effectively associating the data and making sense of the information.

Following this exchanged information, each sensor's display exhibits two separate tracks. These duplications result from the inherent noise introduced by each sensor's processing system, the network, and the surrounding environment. The challenge of track-to-track association arises in this context, aiming to determine whether the objects being tracked by both sensors are the same or distinct entities. Utilizing track-to-track association techniques here can help determine whether the observed data/tracks correspond to the same or different targets by comparing them to the ground truth.

Considering the scenario in Figure 1.2, three sensors are strategically posi-

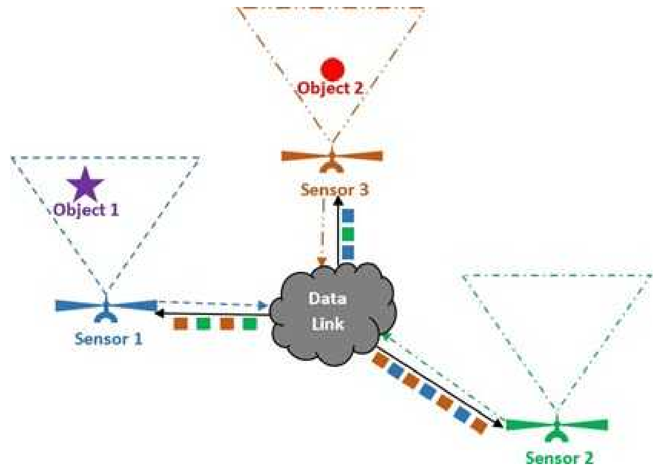



Figure 1.2: Sensors sharing data via common link

tioned in a formation, each overseeing distinct areas of interest. In this scenario, Sensor 1 and Sensor 3 diligently track a single object, while Sensor 2 doesn't track any objects within its field of view. Figure 1.2 portrays the packets of each sensor in distinctive colors corresponding to their respective sensors. Once these sensors establish a data link, they engage in a collaborative exchange of information for their tracks. This shared data encompasses details about their positions and the objects they track within their field of view, conveyed as data packets.

After sharing their track's information, each sensor's display exhibits two distinct tracks. These dual representations arise due to the intrinsic noise introduced by each sensor's processing system, the network, and environmental factors. The challenge of track-to-track association arises at this point. The goal is to determine if the objects tracked by Sensor 1 and Sensor 3 are the same entity or distinct objects. Following the implementation of track-to-track association techniques, the results yield a clear verdict: Sensor 1 and Sensor 3 are indeed tracking different objects. Consequently, it is established that the total number of objects present within the environment is two.

## 1.2 Motivation

This research project's primary motivation is to analyze hierarchical clustering algorithms for track-to-track association in various scenarios. It addresses the challenge of dealing with varying numbers of sensors and objects while dealing with fluctuating thresholds. The study evaluates the HAC algorithm's resilience, particularly in sensor bias scenarios, as understanding and mitigating these biases is crucial for accurate and reliable track-to-track associations. The research also aims to identify and rectify constraints within current track association methodologies, leading to progressive advancements. The study aims to develop more refined and effective track association techniques by implementing optimal thresholds and filters. This aligns with my MS Computational Sciences and Engineering academic background, offering a unique opportunity to explore the intersection of sophisticated algorithms and track association, potentially revolutionizing surveillance and tracking system optimization.

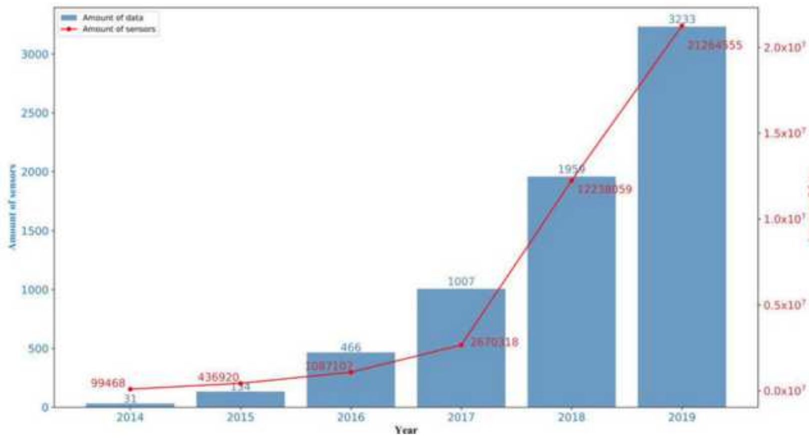



Figure 1.3: Growth in Sensor Deployment and Data Generation (2014–2019) [44]

Figure 1.3 depicts the increase in the number of sensors and the magnitude of data generated within a designated five-year period very well. The provided visual depiction is a compelling illustration of the considerable rise in the deployment of sensors and data generation during the specified time frame.

## 1.3 Problem Statement

The rapid advancement of technology has led to the widespread use of sensors for tracking objects in multi-sensor, multi-object environments. With diverse characteristics, these sensors often operate in overlapping roles, increasing the likelihood of multiple sensors tracking the same object. They communicate via wireless data links, exchanging information about their positions and detected objects through data packets. Effective data analysis from these sensors is crucial for determining the total number of objects and distinguishing whether the tracks obtained from different sensors represent the same or different objects within the environment. Research on various variants of HAC was needed, which could increase the accuracy in multi-sensor multi-target scenarios, which has not been explored in the literature.

This scenario has given rise to the challenge of track-to-track association, especially when dealing with systematic biases in the data. To address this complexity in track-to-track association, this research analyses a Hierarchical

Clustering Algorithm to resolve the multi-sensor, multi-object track association problem in diverse scenarios characterized by sensor biases. The goal is to investigate the performance of this algorithm across multiple scenarios and evaluate its robustness in the presence of sensor biases by comparing it with the ground truth. It is important to note that sensor biases can lead to distinct representations of a single target within shared or associated tracks. This research employs hierarchical clustering techniques with a configurable threshold for associated tracks in realistic scenarios. The aim is to develop the accuracy computation framework from the association results of HAC by utilizing ground truth data and identifying multiple scenarios for which the accuracy computations and association results can be validated.

## 1.4 Research Objectives

The primary objectives of this research include:

- Implementation of the HAC algorithm for the problem of track-to-track association on VOS-generated scenarios.
- Modification in VOS to generate ground truth data for each scenario in a specific format
- Development of an accuracy computation framework from the association results of HAC by utilizing the ground truth data.
- Identifying multiple scenarios for which the accuracy computations and association results can be validated.

## 1.5 Areas of Application

Track-to-track association technology is a versatile technology that has applications across various fields. It is crucial in the automotive industry to enhance safety and navigation in intelligent vehicles. It also improves monitoring and threat detection in security and surveillance systems by tracking the movement of people or objects over time. Satellite tracking is essential for accurate monitoring and management of satellites, crucial for space missions. Military aviation enhances air combat and tactical operations, allowing for better situational awareness and decision-making during complex missions. In naval applications, it optimizes underwater navigation and communication, tracking underwater terrain and other vessels for safe and efficient operations. Overall, the track-to-track association is a versatile tool with applications ranging from vehicle safety to military operations and space exploration, each benefiting from improved tracking and monitoring capabilities.

## 1.6 Thesis Layout

The thesis structure offers a logical and thorough examination of the research subject. The Introduction in **Chapter 1** establishes the framework by providing a synopsis of the research question and outlining the particular goals that direct the investigation. Building on the introduction, **Chapter 2**'s Literature Review critically assesses the research already done in the area, providing a comprehensive picture of the academic environment. The core of the thesis is covered in **Chapter 3**, Methodology, which explores the complexities of the system model and scenario generator. This chapter discusses the effects of systematic biases. It clarifies the track-to-track association algorithm and breaks down the intricate flow of track synchronization. Results and Discussions in **Chapter 4** focus on presenting the findings. This chapter illustrates the benefits of the suggested algorithm with simulation and detailed analysis, offering a sophisticated view of its effectiveness and ramifications. **Chapter 5**, Conclusion and Future Work summarizes the research's main conclusions. It provides a summary of the findings and, in looking ahead, considers possible future directions for this work. This well-organized outline guarantees a logical flow of thoughts, making it easier to investigate the research topic thoroughly.

# Chapter 2

# Literature Review

This chapter will start by exploring the basics that form the research foundation from the background and context of what other studies have discovered in the field, followed by key concepts and theories like sensors and how we track moving objects. The previous research and findings will set the stage for understanding the challenges and possibilities we will tackle. This chapter is a roadmap, guiding through what's already known and pointing toward Algorithms and Theoretical frameworks for our research objectives.

## 2.1 Sensors Basics

Tracking systems depend on sensors to convert inputs into actionable data. They convert environmental cues like light, sound, and electromagnetic waves into electrical signals or digital data. Sensors capture vital environmental and object-tracking data. They consistently gather data on position, movement, and other relevant attributes to provide critical insights into monitored objects. Sensors play a foundational role within tracking systems by delivering comprehensive reports on target tracks. These reports include timestamps, identification numbers, state estimates (e.g., position and velocity), and error covariance matrices. Every sensor incorporates an internal tracking module that tailors reports to its specific coordinate system. To achieve comprehensive tracking, spatial and temporal alignment prove indispensable. Spatial alignment establishes a unified coordinate system, while temporal alignment synchronizes sensor timestamps. Data validation filters ensure the transmission of only reliable information. Sensor meta-data, encompassing identification and field of view, significantly contributes to effective data association and synchronization. The track data format typically contains timestamps, sensor meta-data, track ID, state estimates, and estimation error covariance matrices [19].

Using different principles, sensors capture physical properties like light, sound, or electromagnetic waves and convert these into usable data, consistently gathering crucial information about an object's attributes [21]. This continuous monitoring is fundamental for recording essential data about tracked objects. Various sensors employ diverse methodologies for efficient object tracking. For instance, specific sensors leverage clustering algorithms, enabling real-time data processing in dense target environments, thereby enhancing effective object tracking [5]. Hierarchical clustering techniques enhance multi-target tracking accuracy and efficiency within data association [34]. These algorithms optimize object-tracking systems by precisely associating measurements and tracks. In dense target environments, advanced techniques like leader-follower online clustering enhance track association [21].

Sensors in object-tracking systems use many algorithms and methods. These methods enable data collection, processing, and association, enabling precise object tracking in various environments. Many types of sensors detect environmental cues and convert them into understandable data. RADAR, LIDAR, and ESM are examples. RADAR systems emit radio waves and analyze their reflections for location and speed. However, LIDAR measures distances and creates 3D maps using laser pulses. ESM listens to electromagnetic signals from electronic devices to identify and classify radar and communication signals. Sensor types serve different applications in defense, autonomous vehicles, and environmental monitoring.

### 2.1.1 RADAR

Radio Detection and Ranging (RADAR) uses electromagnetic signals, mostly radio waves, to detect objects' presence, location, and movement within their range. Its operational modes make this system versatile. Pulsed and continuous wave RADAR detects objects by intermittently emitting waves and tracking them through uninterrupted signal transmission and reception. The fusion of these modes suits diverse applications and environmental conditions. Advanced signal processing methods like matched filtering and waveform diversity boost RADAR efficiency. Aligning received echoes with transmitted waveforms improves signal-to-noise ratios for more accurate object detection with matched filtering. Waveform diversity broadens the transmitted signal spectrum, improving object-background clutter discrimination. RADAR systems' robustness and precision are attributed to their integration of operational modalities and signal processing methods, enabling rapid, accurate, and detailed object detection and tracking across various domains [14].

According to Ryde and Hillier, RADAR can withstand harsh environmental conditions. Their study compares RADAR and LIDAR ranging devices in adverse conditions, highlighting RADAR's robustness, especially in bad weather. It shows RADAR's accuracy in detecting and tracking targets despite challenging environmental conditions. RADAR excels in challenging weather conditions, demonstrating its accuracy, reliability, and importance in various applications [12]. J. Wang, A. von Trojan, and S. Lourey's study on RADAR technology for active sonar target tracking in anti-submarine warfare uses acoustic waves to locate and monitor submerged objects, particularly submarines. Emitting sound pulses into the water and analyzing the echoes reveals the object's characteristics and movements. Range resolution, target detection, and signal strength for underwater tracking depend on pulse repetition frequency, pulse width, and transmitted power. The study emphasizes the importance of RADAR in defence systems, as it accurately detects and tracks underwater targets in complex military environments [15].

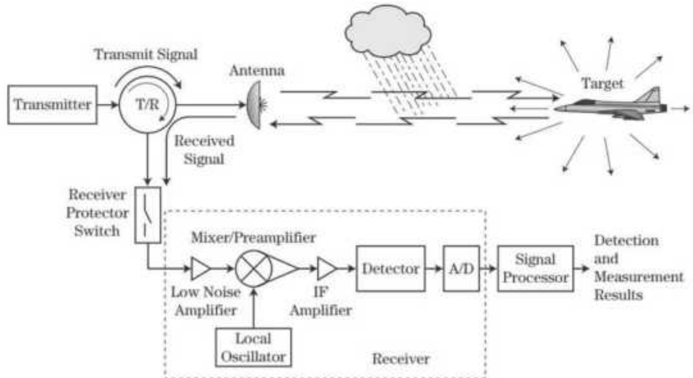



Figure 2.1: Block diagram of a RADAR system [32]

RADAR has simple steps. Transmitters emit radio signals first. Antennas detect object-reflected signals. Deciphering these echoes requires the receiver to eliminate interference and amplify desired signals. A processor then analyzes these signals to determine the object's distance, identity, and motion. For operators to understand the RADAR's environment, processed data is presented or transmitted to other systems. The diagram shows how RADAR systems transmit, receive, and analyze signals to gain a complete understanding of their surroundings[15]. RADAR systems are versatile and widely used. They accurately detect, track, and localize objects in military operations, weather monitoring, and autonomous navigation systems. The technology's adaptability, operational modes, and signal-processing methods make it important in modern sensing and tracking. Several fundamental principles and equations govern RADAR technology's operation and capabilities.

#### 2.1.1.1 RADAR Range Equation

The primary equation for RADAR receiver power from targets is as follows. The result depends on transmitted power, antenna gains, target radar cross-section, and range. This equation is essential to understanding RADAR operation and constraints [49]:

$$P_r = \frac{P_t G_t G_r \lambda^2 \sigma}{(4\pi)^3 R^4}$$

#### 2.1.1.2 Doppler Shift Equation

By comparing the reflected signal's frequency to the transmitted signal, RADAR systems calculate a target's relative velocity using Doppler shift. Defense and weather monitoring require speed detection [14]:

$$f_d = \frac{2v_r f_t}{c}$$

Where:

- $f_d$ is the Doppler frequency shift.
- $v_r$ is the target's radial velocity relative to the RADAR.
- $f_t$ is the transmitted frequency.
- $c$ is the speed of light.

#### 2.1.1.3 RADAR Cross-Section (RCS)

The Radar Cross Section (RCS) measures radar object detection. Intricate value depends on target dimensions, configuration, composition, and texture. The maximum target detection distance depends on the RCS [50]:

$$\sigma = \frac{P_r (4\pi)^3 R^4}{P_t G_t G_r \lambda^2}$$

#### 2.1.1.4 Ambiguity Function

The RADAR ambiguity function analyzes RADAR system resolution and discrimination in range and velocity. It is essential for RADAR system design and analysis [8]:

$$\chi(\tau, f_d) = \int_{-\infty}^{\infty} s(t) s^*(t-\tau) e^{-j2\pi f_d t} dt$$

Where:

- $\chi$ is the ambiguity function.
- $\tau$ is the time delay.
- $f_d$ is the Doppler frequency.
- $s(t)$ is the transmitted signal.
- $s^*(t-\tau)$ is the time-shifted complex conjugate of the transmitted signal.

#### 2.1.1.5 Signal-to-noise ratio (SNR)

RADAR performance depends on signal strength relative to background noise (SNR). It affects RADAR signal detection and processing and is essential for RADAR image and data interpretation [47]:

$$\mathrm{SNR} = \frac{P_{signal}}{P_{noise}}$$

These equations serve as the foundational principles underlying RADAR sensor technology. These devices enable scientists and engineers to create RADAR systems with optimized range, resolution, detection capabilities, and dependability for various applications.

### 2.1.2 LIDAR

LiDAR, or Light Detection and Ranging, is a remote sensing technology that uses laser pulses to measure distances accurately. It evolved after the discovery of lasers in the 1960s and involves the emission of laser beams toward a target and the measurement of the time it takes for these beams to reflect. This method allows for precise distance calculations, which is essential for various applications. The core components of a LiDAR system include a laser source, a scanner or mirror, a photodetector, and a GPS receiver. These components provide a highly accurate and reliable method of measuring distances, facilitating applications in topographic mapping, urban planning, and environmental management. LiDAR is primarily used in geographical mapping, terrain modeling, forestry, urban planning, and environmental monitoring due to its ability to create detailed 3D maps and spatial data.

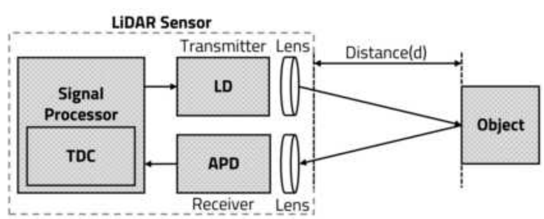



Figure 2.2: Structure of conventional LiDAR sensors [43]

It signifies a crucial leap forward in research, likely discussing essential advancements and applications within LiDAR technology [3]. The focus may delve into radar ESM track-to-track association techniques, enhancing accuracy by linking tracks from various sensors. Specifically, it emphasizes merging radar and LiDAR sensors for applications like highway car-following systems. Diverse methodologies and algorithms have surfaced, aiming to blend sensor data effectively, enhancing automotive sensor fusion capabilities [16]. LiDAR technology, including ground-based and airborne types, has numerous applications in various fields. Ground-based LiDAR systems, using lasers with wavelengths of

500 to 600 nm, are crucial for topographic mapping, environmental monitoring, and forest ecosystem research. These systems are used for precise mapping, environmental surveillance, and the analysis of forest ecosystems, making LiDAR technology a widely documented and widely used tool [1][37]. Airborne LiDAR systems are essential for terrain mapping, surveys, and land development projects because they capture detailed 3D information about the Earth's surface while utilizing wavelengths between 1000 - 1600 nm. These systems use laser pulses to measure the time light returns after hitting objects or the Earth's surface. The near-infrared spectrum provides a comprehensive terrain view, allowing high-resolution mapping while maintaining accuracy [27].

LiDAR sensors are crucial in forest management, disaster management, archaeological surveys, and urban planning. They provide precise tree height estimation, vegetation density analysis, and high-resolution data for assessing terrains, mapping hazards, and planning disaster response strategies. Their ability to penetrate dense foliage and uncover hidden features is essential for archaeological site documentation. Their 3D modeling capabilities also aid infrastructure development, land use planning, environmental impact assessment, and urban environment management [33]. Airborne LiDAR is a special kind of technology where machines on planes or drones use laser beams to see things on the ground. They shoot these lasers down and measure how long the light can bounce back. This helps make detailed maps, check plants and trees, and even helps during disasters or when planning cities. It's famous because it accurately pictures the Earth's surface, helping in forests, city planning, and many other areas. It's all about quickly making 3D pictures, and there are some cool new ways to make it work even better, like using a special laser. This technology is important because it helps us see and understand our world better by taking really good pictures from up high [45].

#### 2.1.2.1 LiDAR Range Equation

LiDAR technology and its modeling principles often involve the LiDAR range equation, which defines the relationship between the flight time of laser pulses and the distance to the target, generally represented as:

$$r = \frac{ct}{2}$$

Where:

- $r$ is the distance between the LiDAR system and the target.
- $c$ is the speed of light.
- $t$ is the time-of-flight of the laser pulse.

LiDAR applications require this equation to calculate precise distances between the system and target objects for mapping and spatial understanding.

#### 2.1.2.2 The Role of Poisson Probability Distribution in LiDAR Detection Reliability

Another equation that might be involved is the Poisson probability distribution equation, which describes the probability of a given number of events occurring in a fixed interval of time or space. In the context of LiDAR and photon detections, it could relate to the probability $p_j$ of the avalanche photodiode (APD) being triggered, given the number of incoming photons.

$$p_j = \frac{\lambda^j \cdot e^{-\lambda}}{j!}$$

Where:

- $p_j$ is the probability of $j$ photons triggering the APD.
- $\lambda$ represents the average number of photons in a given interval.

These equations offer crucial outcomes for LiDAR applications. The LiDAR range equation computes precise distances between the system and target objects, enabling accurate mapping and spatial understanding. Meanwhile, the Poisson probability equation aids in assessing the reliability of photodiode triggers by incoming photons, influencing the quality and fidelity of LiDAR-generated data and ensuring dependable image formation and data collection.

### 2.1.3 ESM

Electronic Support Measures (ESM) sensors track and identify radar signals for surveillance, situational awareness, military operations, and air traffic management. Early remote sensing technologies like laser-based and radar signal interception helped develop ESM sensors. These sensors intercept, identify, and locate radar signals to detect presence, type, and location. Information is crucial for implementing countermeasures, especially in military situations where radar-based threats must be detected and countered for operational success and safety [1][6][49]. LiDAR technology has significantly improved the capabilities of Electronic Support Measures (ESM) sensors by providing accurate 3D imaging and mapping of the environment, enabling them to identify and locate electronic signals. In 2021, 3D imaging models for airborne LiDAR systems were developed, enhancing ESM sensor capabilities in complex and dynamic environments. LiDAR's contribution to object tracking, especially in challenging environments like forests, has highlighted its significant improvement in ESM sensor performance [28][45][36].

Evolution of electromagnetic support measures (ESM) sensors, essential to modern surveillance and defense systems. ESM sensors detect and analyze radar electromagnetic signals to improve object tracking. They also fuse sensor data to improve environmental perception. ESM sensors can capture electromagnetic signals from military radar systems and commercial navigation radars due to their 2 GHz to 40 GHz operating frequency range. Defense systems can distinguish radar emissions for threat assessment and situational understanding [49]. Integrating advanced signal-processing algorithms has advanced ESM. Emitter detection and identification advancements have led to improved track-to-track association methods [8]. In signal-interference-rich environments, the algorithms remove noise and accurately associate radar tracks from sensors. Machine learning techniques improve ESM systems' precision by enabling real-time adaptation to new signals and countermeasures [50]. Advances have made ESM sensors more practical and flexible, making them essential for electronic warfare and intelligence gathering. Combining them with optical or infrared sensor data boosts their effectiveness. This improvement aids strategic decision-making and offers a complete operational view [12].

Algorithmic and statistical improvements have improved ESM sensor functionality. Track initiation methods use statistical analysis to detect and track objects accurately. Track-to-track fusion and association algorithms have revolutionized ESM sensor data interpretation and management. Scholars have used advanced data association techniques to improve precision in recent research [2][8][4][52]. ESM sensors adapt to different environments, showing their versatility. These technologies are used in automotive safety systems, maritime surveillance, and other fields, proving their versatility. ESM technologies' adaptability is crucial for application in diverse sectors, highlighting their importance in various operational contexts [3][17][50]. Despite these advances, sensor biases and data precision in dense target environments remain issues. For ESM technologies to be effective in complex scenarios, future advancements should focus on real-time track association and managing large sensor data sets [12][14][51].

#### 2.1.3.1 Radar Range Equation for ESM

The radar range equation is important in ESM as it connects the received power of an ESM system to the transmitted power, antenna gains, and target distance. The equation is crucial for the ESM operation as it aids in assessing the detectability of radar emitters [49]:

$$P_r = \frac{P_t G_t G_r \lambda^2 \sigma}{(4\pi)^3 R^4}$$

Where:

- $P_r$ is the power received by the ESM receiver.
- $P_t$ is the power transmitted by the radar.
- $G_t$ and $G_r$ are the gains of the transmitting and receiving antennas, respectively.
- $\lambda$ is the wavelength of the radar signal.
- $\sigma$ is the radar cross-section of the target.
- $R$ is the range to the target.

#### 2.1.3.2 Signal-to-noise ratio (SNR) for ESM Detection

Signal processing in ESM systems heavily depends on the signal-to-noise ratio (SNR), which quantifies signal strength compared to background noise. A high signal-to-noise ratio (SNR) is crucial for the effective identification and processing of radar signals by the electronic support measures (ESM) receiver, particularly in the presence of environmental noise [14]:

$$\text{SNR} = \frac{P_{signal}}{P_{noise}}$$

Where:

- $P_{signal}$ is the power of the received signal.
- $P_{noise}$ is the power of the background noise.

#### 2.1.3.3 Angle of Arrival (AoA) Estimation

The angle of arrival (AoA) is a crucial aspect of Electronic Support Measures (ESM) as it helps determine the direction of incoming signals. The AoA estimation can be achieved by measuring the time difference of signal arrival between two spatially separated antennas. This technique has been investigated in various studies on ESM and radar systems [47]:

$$\Delta t = \frac{d \cdot \sin(\theta)}{c}$$

Where:

- - $\Delta t$ is the difference in arrival time of the signal at two antennas.
- $d$ is the distance between the antennas.
- $\theta$ is the signal's arrival angle.
- $c$ is the speed of light (since radar waves travel at the speed of light).

#### 2.1.3.4 Probability of Intercept (PI)

The probability of intercept (PI) is a crucial metric for ESM systems. It quantifies the probability of detecting a specific radar signal by the ESM system over a defined time interval. PI is influenced by various factors such as scan rate, sensitivity, and bandwidth [8].

#### 2.1.3.5 Ambiguity Function

The ambiguity function evaluates ESM systems' time delay and Doppler frequency resolution and performance. ESM systems can differentiate between closely spaced signals in time or frequency [50]. These equations establish the mathematical framework that governs ESM systems' design, development, and operational capabilities. They facilitate the accurate detection, identification, and localization of various signal types essential for surveillance, reconnaissance, and electronic warfare. Theoretical foundations offer a comprehensive understanding of the principles and capabilities of modern ESM systems.

### 2.1.4 Object Tracking

Automotive, military, and aviation industries depend on object tracking, which detects, tracks, and associates objects using radars and lidars. These range- and resolution-specific sensors face biases and noise interference. Sensor performance has improved quantitatively in range, accuracy, and resolution as tracking technologies have evolved. Laser technology improved lidar systems' range, possibly extending detection distance by several kilometers. These systems may have improved accuracy, lowering error margins. Lidar systems may have improved resolution, crucial for detailed mapping and object detection. These improvements in distance measurement, error rates, and spatial resolution marked a major leap in Lidar technology, laying the groundwork for its many applications in the decades that followed [1]. Statistical performance analysis of track initiation methods is used in object tracking. This extensive study likely compared nearest-neighbor tracking, probabilistic data association, and multiple hypothesis tracking. The authors likely assessed these methods using detection probability, false alarm rates, and track accuracy. The paper would have shown how each technique performs under different noise, object density, and movement patterns using simulations. This analysis would have helped determine the best track initiation methods, advancing sensor-based tracking systems [2].

Object tracking requires multiple sensors, making track-to-track association crucial. Radar and ESM integration for track association is documented. ESM systems detect, intercept, and classify radar signals for military use. Correlating radar-detected objects with ESM-detected signals for accurate tracking is the main challenge in this domain. Tracks can be matched using algorithms or statistics using velocity, direction, and electromagnetic signal characteristics. Accurate identification and tracking of airborne or maritime targets are crucial for surveillance and defense applications [3].

Figure 2.3 shows that rain fades can significantly affect the accuracy and reliability of measurements from sensors like radar or lidar, leading to errors or data loss in object-tracking systems. Higher frequency bands like Ka and Ku are more susceptible to rain fade due to their smaller size. Understanding this impact is crucial for designing and optimizing object-tracking systems. Mitigating its impact includes adaptive algorithms or switching to less susceptible frequency bands. Key concepts for object tracking include antenna, datum, antenna height, elevation angle, rain height, slant range, and h_rain [38].

Automotive safety and autonomous vehicle development require multiple sensor fusions to detect and track road obstacles. The integration and association of radar, lidar, and camera data were examined. A complete and accurate vehicle environment representation is needed to detect and track pedestrians,

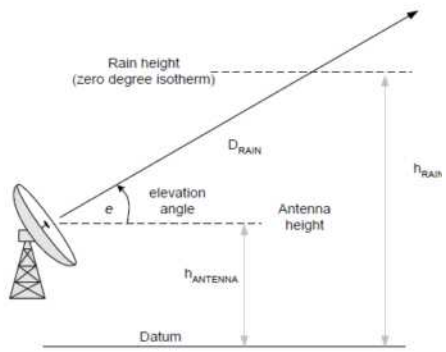



Figure 2.3: Earth-space slant range of signal below freezing point [38]

other vehicles, and road hazards. Sensor data alignment, discrepancy resolution, and real-time processing for dynamic obstacle tracking are proposed [4]. Clustering algorithms are necessary for tracking multiple targets at once. Multiple data points (or 'tracks') are clustered by similarities like spatial proximity or motion patterns by these algorithms. Clustering algorithms simplify tracking by grouping similar tracks. This approach improves object distinction in dense object populations and reduces track mis-association [5]. Sequential nonlinear tracking follows objects using nonlinear methods, which are better at handling real-world targets' unpredictable movements. Sequential nonlinear tracking algorithms are more robust and reliable than linear tracking methods because they can handle sudden trajectories or speed changes. Algorithms update and adjust object paths based on sensor data, ensuring accurate tracking in dynamic environments [8].

Association commonly uses fuzzy logic, a form of many-valued logic that uses approximate rather than fixed logic and exact reasoning for track-to-track association. The fuzzy track-to-track association handles sensor data uncertainties and ambiguities with fuzzy logic. Traditional crisp association methods may fail or produce accurate results with complete or noisy sensor data. Fuzzy logic improves tracking accuracy under uncertain conditions by associating tracks from different sensors, allowing flexibility and tolerance [9]. The fusion of radar and lidar data improves object-tracking systems. Lidar's high resolution and precision in mapping and object detection are enhanced by radar's long-range detection and weather resistance. This combination increases the tracking system's detection range and accuracy in identifying and positioning objects. Lidar may struggle in bad weather, while radar may lack resolution. By integrating sensor data, their limitations can be overcome. A more reliable and comprehensive tracking system is crucial in complex environments like autonomous vehicle navigation, where accurate obstacle detection is crucial [16][35].

Object tracking systems must process massive sensor data. This data is ideal for machine learning algorithms, which find patterns and insights that improve tracking accuracy. Machine learning in object tracking allows systems to learn from data and improve over time. As more scenarios and data are collected, the system improves in predicting and tracking object movements and behaviors. Machine learning algorithms handle object tracking's complexities and variables well. They can adapt to changes in object behavior, environmental conditions, and sensor inputs, making tracking more robust and flexible [34][40]. The fusion of radar and lidar data with machine learning has transformed object tracking. This synergy improves tracking systems' range, accuracy, and ability to learn from data, making them more sophisticated, efficient, and reliable for various applications. Advances in sensor technology, algorithms, and data fusion have transformed object tracking. This evolution has improved tracking systems' accuracy and reliability, making them essential in modern technology.

## 2.2 Frames of References

Reference frames are essential in navigation, space exploration, and environmental monitoring for accurate ranging measurements, safe navigation systems, and precise localization in applications like autonomous vehicles and robotics. They also provide standardized descriptions of positions and movements, enabling accurate communication across platforms. However, challenges persist in accuracy and computational efficiency. ECI, LLA, and ECEF frames are crucial for accurate navigation and coordinates. Researchers have found that accurate frames of reference are crucial for reliable range-finding equipment in autonomous navigation and environmental monitoring, especially in adverse environmental conditions. These frames provide a stable and consistent reference point, preventing errors and inaccuracies from weather and terrain variations. They recommend implementing calibration, correction, and compensation techniques to enhance these devices' performance and reliability [12].

The study emphasizes the significance of frames of reference in interpreting radar and lidar sensing data for localization and mapping. Frames of reference provide a consistent basis for interpreting data, especially in dynamic environments. Radar uses radio waves to detect objects and measure distances, while lidar uses laser beams to create 3D maps. Accurate localization and mapping require alignment of sensor data with a common frame of reference, enabling data fusion and integration [35]. Frames of reference are crucial in airborne LiDAR systems, which use laser beams to create detailed 3D maps of the environment. These frames represent coordinate systems and orientations used to interpret and analyze data collected by LiDAR sensors. A consistent frame of reference ensures accurate interpretation and integration of captured measurements with other sensor data or mapping systems. A reliable frame of reference allows researchers to accurately determine objects' position, orientation, and shape in LiDAR point cloud data, essential for applications like terrain mapping, urban planning, and environmental monitoring [45]. LiDAR (Light Detection and Ranging) uses laser beams to create detailed 3D maps of the environment. A consistent frame of reference is crucial for data processing and analysis, allowing accurate positioning and mapping of objects in the data. Researchers can determine objects' position, orientation, and shape by aligning captured measurements with a common frame of reference. This alignment ensures proper integration with other data sources or mapping systems, enabling the seamless combination of LiDAR data with other geospatial information like satellite imagery or GPS data [1].

The authors propose a sequential nonlinear tracking algorithm using the Unscented Kalman Filter (UKF) and raw range-rate measurements to estimate the position and velocity of an object in a three-dimensional space. They emphasize using different frames of reference, particularly the Earth-Centered Inertial (ECI) frame, to estimate the object's position and motion accurately. The algorithm's mathematical formulation is presented, and its effectiveness is demonstrated through simulations and experiments [42][8]. The literature explores track fusion in a distributed multisensor system, introducing a fuzzy logic-based approach to handle uncertainties and complexities in track association and fusion. It presents mathematically and evaluates the method through simulations and experiments, focusing on distributed multisensor-multitarget tracking challenges and discussing its advantages and limitations [9]. An algorithm was proposed for track association in radar and ESM systems, focusing on hierarchical clustering to address bias. It emphasizes the importance of frames of reference and how a shared frame aligns data from different sensors, enabling accurate track association and improving the overall reliability of the tracking system [31].

The literature discusses clustering techniques in multiple target-tracking algorithms, focusing on grouping targets based on their characteristics. Frames of reference, such as position, velocity, or size, are crucial in defining a consistent frame of reference for accurate target grouping and effective tracking, thereby enhancing the overall algorithm's performance. [5]. The literature also introduces a track-to-track association method for automotive perception systems, addressing the challenges of associating tracks from different sensors in dynamic environments. It emphasizes the importance of frames of reference for accurate track association, establishing a common reference frame like a global coordinate system, thereby ensuring reliable track association and improving the perception system's accuracy in automotive settings [19].

### 2.2.1 Earth Centered Inertial (ECI)

Earth-Centred Inertial (ECI) is a reference frame used in aerospace to describe the position and motion of objects relative to Earth. It is fixed in space, considering Earth's rotation and position in other celestial bodies. ECI allows for accurate tracking and prediction of satellites and spacecraft, providing a stable reference point. It eliminates Earth's rotation effects and allows precise calculations of an object's trajectory. ECI also aids in the accurate estimation of an object's position and velocity in three-dimensional space. The frame of reference, including the celestial sphere, North Pole, Equatorial Plane, and other celestial objects, is crucial for object tracking due to its stable, non-rotating reference system. It simplifies calculations related to celestial mechanics, such as determining gravitational influences on satellites. ECI is a reliable and consistent way to understand space movement.

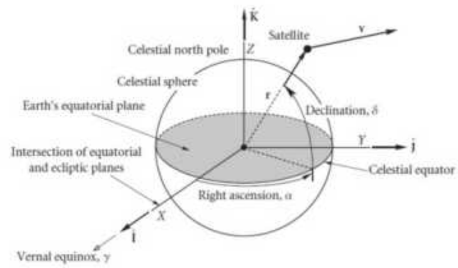



Figure 2.4: Earth-Centered Inertial (ECI) Frame [22]

The Earth-centered inertial (ECI) frame of reference is used for precise spatial measurement and tracking in satellite-based applications and space missions. It provides a stable, non-rotating reference frame in space, offering absolute positioning for precise measurements of satellite positions, velocities, and trajectories. Understanding the performance of laser and radar ranging devices under adverse conditions is crucial for accurate measurements. Satellite tracking is often conducted within the ECI frame due to its stability and non-rotating nature. The accuracy and reliability of spatial measurements within the ECI frame impact activities conducted within the frame [12]. The literature on radar and lidar sensing for localization and mapping highlights the importance of the Earth-Centered Inertial (ECI) frame of reference for satellite-based radar and lidar systems. The ECI frame provides a stable reference system for precise spatial localization and mapping, enhancing the precision of sensing and mapping activities. The choice between radar and lidar sensing methods affects the quality and precision of spatial data [35]. Qiao and Zhao emphasize the significance of the Earth-Centered Inertial (ECI) frame of reference in aligning LiDAR data with global satellite positioning systems. ECI provides a stable, non-rotating reference frame, ensuring spatial accuracy and consistency. The paper focuses on developing a 3D imaging model for airborne LiDAR systems, integrating them with satellite-based positioning systems for accurate alignment and georeferencing [45].

ECI has been used in Lidar systems to calibrate global positions precisely for atmospheric measurements and alignment with satellite observations. It provides a stable and non-rotating reference, ensuring accurate and consistent measurements across different locations. This frame also enables accurate alignment and comparison of Lidar measurements and satellite observations, improving the reliability of atmospheric data. Lidar applications and techniques rely on a stable reference frame like ECI for accurate global positions and alignment with satellite data [1][6]. In satellite tracking and position prediction, ECI provides a stable reference frame at Earth's center, offering precise orbital predictions and consistent navigation for multiple satellites or constellations. ECI is essential for aerospace applications like communication, navigation, and Earth observation. Using UKF and range-rate measurements for sequential tracking requires a stable reference frame like ECI for accurate and reliable satellite position predictions [8]. Ashraf M. Aziz's research explores using ECI parameters in fuzzy track-to-track association and track fusion in multisensor-multitarget environments. ECI provides a stable reference frame for space-based assets, incorporating satellite track information to influence the association process. It also offers a global, inertial reference, enabling consistent sensor data integration. This could enhance global understanding of multisensor data, improve target association and fusion algorithms, and broaden multisensor fusion scope [9].

The fixed coordinate system (ECI frame) is commonly used for tracking satellites and celestial bodies, but these principles are not directly applied to ECI frame applications. Research suggests track-to-track association methods for automotive perception systems, clustering algorithms for space tracking, and an anti-bias track association algorithm using hierarchical clustering. If applied to space surveillance radar systems, these algorithms could be relevant to ECI frame applications. The impact of this research on ECI frame applications could be significant, enhancing orbit determination and prediction, improving situational awareness, and aiding in collision avoidance maneuvers. The analysis underscores the importance of interdisciplinary approaches in scientific research, as innovations in one field can influence practices in another [19][5][31]. The ECI frame is a stable coordinate system unaffected by Earth's rotation or orbital movement, making it crucial for consistent satellite tracking over time. Accurate satellite tracking is essential for communication, weather forecasting, navigation, and other critical space-based applications.

### 2.2.2 Earth Centered Earth Fixed Frame (ECEF)

The Earth-Centered Earth-Fixed (ECEF) frame of reference is a global Cartesian coordinate system used in satellite navigation, aerospace, and geodesy. It has fixed axes relative to the Earth's surface, with the X-axis pointing towards the Prime Meridian and the Equator, the Y-axis at 90° longitude, and the Z-axis aligned with the Earth's rotational axis (North Pole). This frame is crucial in applications such as Lidar technology, track initiation techniques, track-to-track fusion methods, and satellite orbit determination. It accurately represents the position and motion of Lidar sensors and objects, ensuring alignment of position and motion data. ECEF coordinates are used in radar, lidar, and sensor tracking systems to ensure accurate positioning, integration, and data fusion. Radar systems use ECEF coordinates for high precision in determining target positions, while lidar systems use ECEF coordinates for geospatial mapping and coordinate transformation. Sensor tracking systems unify positional data obtained from various sensors, ensuring coherent and accurate tracking. R. T. H. Collis explores integrating Lidar systems with GPS, highlighting the need for a global coordinate system like the Earth-Centered Earth-Fixed (ECEF) framework for precise georeferencing. Lidar systems use GPS technology to assign real-world coordinates to captured points, ensuring latitude, longitude, and altitude accuracy. The Earth-Centered, Earth-Fixed (ECEF) coordinate system is inherent in Lidar systems, providing accurate position data. By integrating Lidar and GPS, Lidar data is guaranteed to be highly accurate worldwide, allowing for consistent and precise placement of points within the ECEF reference frame. This integration benefits applications like mapping, surveying, and navigation systems [1].

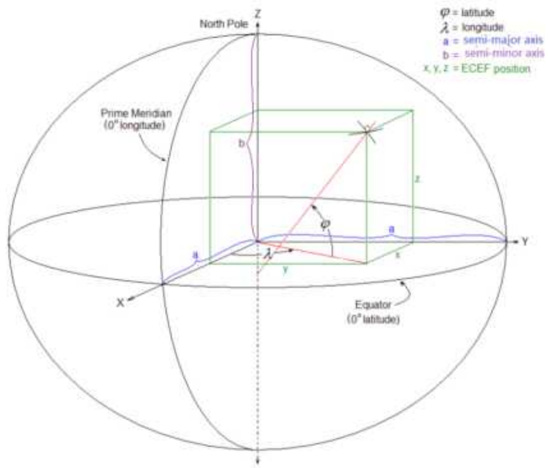



Figure 2.5: Earth-Centered Earth-Fixed (ECEF) coordinates [48]

Figure 2.5 shows an Earth-Centered, Earth-Fixed (ECEF) coordinate system with a geocentric frame of reference that remains constant relative to the Earth's surface despite its rotation. It is represented as an ellipsoid with a semi-major axis (a) and a semi-minor axis (b), accounting for Earth's oblateness. This system is widely used in applications requiring a fixed coordinate relative to Earth, such as GPS satellite positions and navigation systems. It accurately represents positions and movements on the Earth's surface and is essential for geodesy and navigation. The ECEF coordinate system is a crucial tool for understanding the evolution and transformation of objects, providing a visual representation of their evolution and changes over time. The literature also discusses the statistical performance analysis of track initiation techniques in satellite tracking or navigation systems. ECEF coordinates can indirectly benefit these techniques: coordinate handling, algorithm validation, and performance metrics. Coordinate handling is crucial for accurately handling position and velocity data in satellite tracking systems. Algorithm validation involves simulations using known positions and velocities, often defined in an Earth-centered coordinate system like ECEF. Using ECEF coordinates allows for precise measurement comparisons and evaluation of performance metrics across different techniques, leading to improved accuracy, standardization, and enhanced algorithm performance [2]. The Earth-Centered Earth-Fixed (ECEF) frame of reference is used for road obstacle detection in track-to-track fusion. It converts data from various sensors into a common global frame, enabling consistent analysis. Global positioning accuracy is achieved by providing a reference system for accurate georeferencing of obstacles detected by different sensors. Track association helps associate corresponding tracks across different sensor sources, accurately representing the same obstacle. ECEF's potential impact could enhance accuracy and reliability in detecting and localizing road obstacles, aiding in better decision-making for navigation and autonomous driving systems [4].

The ECEF frame is crucial in satellite tracking and navigation due to its position representation, sensor fusion, trajectory prediction, and compatibility. Lei and Han's paper suggests transforming raw range-rate measurements into ECEF coordinates, enabling data from multiple satellites or sensors in a consistent reference frame. This helps accurately predict satellite trajectories and orbits, refining orbit calculations and prediction algorithms. The aim is to improve accuracy, consistency, and compatibility in processing and predicting satellite positions, potentially leading to more precise trajectory predictions, enhanced sensor data fusion, and increased overall accuracy in satellite tracking systems [8]. The Earth-Centered Earth-Fixed (ECEF) frame of reference has also been applied in radar systems for precise positioning and tracking of targets. It provides a global reference frame, allowing radar systems to determine target positions relative to the Earth's center, ensuring measurement consistency and accuracy. ECEF coordinates also benefit navigation and surveillance applications by facilitating seamless integration of radar data with other systems or sensors using the same reference frame. Applying ECEF in radar systems leads to enhanced accuracy in determining target position and consistency in positioning across various systems, enhancing their performance and accuracy in tasks like target positioning, navigation, and surveillance [49].

### 2.2.3 Local Geodetic Frame

The Local Geodetic Frame of reference is a coordinate system used for mapping and surveying a specific area on Earth's surface, accounting for Earth's curvature. It provides a local reference frame for measurements and distances, with the X-axis pointing towards the East, the Y-axis pointing towards the North, and the Z-axis pointing towards the Up direction. This coordinate system is crucial in applications like Lidar technology, track initiation techniques, track-to-track fusion methods, radar ESM track-to-track association, and multiple target tracking algorithms, accurately representing the position of the Lidar sensor and objects being scanned. From the literature, the emphasis on the significance of precise positioning in road environments for obstacle detection suggests using a local geodetic frame or local coordinate system. This frame is often used for high-precision positioning in situations like track-to-track fusion. The need for precise localization suggests the potential of incorporating a local geodetic frame of reference for accurate mapping and tracking within the road environment [4]. Dong and Chen emphasize the significance of the Local

Geodetic Frame of Reference in LiDAR remote sensing for precise mapping, point cloud generation, and geospatial applications. They discuss using geodetic frames, including local, global, or projected coordinate systems, to represent and interpret data accurately. They also discuss the transformation between coordinate systems, geoid models, projections, and datums to align the acquired LiDAR data with the Earth's surface [28].

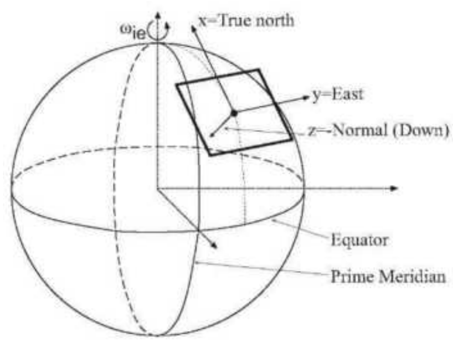



Figure 2.6: Local Geodic Frame of Reference [25]

Figure 2.6 illustrates a local geodetic frame of reference used in geodesy and navigation to define an object's position and orientation on Earth's surface. It consists of a sphere representing Earth, the equator, the prime meridian, three orthogonal axes (x, y, z), and the symbol $\omega_{ie}$. The geodetic frame is likely a local tangent plane, approximating the Earth's surface at a specific location. It is defined by the x-axis (True North), y-axis (East), and z-axis (Normal/Down). The symbol $\omega_{ie}$ represents the Earth's angular velocity concerning inertial space, suggesting rotation with the Earth. This frame is commonly used in applications like GPS, where a clear local coordinate system is crucial for navigation and mapping [25] Mielle et al. emphasize the significance of local geodetic frames or references when comparing radar and LiDAR sensing for localization and mapping. These frames provide a spatial reference system for precise positioning, aligning sensor data, and mapping accurate coordinates onto a given space. The choice of frame significantly impacts data accuracy, affecting localization and mapping precision. Understanding how different sensors align their data to a specific reference frame is crucial for accurate comparison and analysis. Applying local geodetic frames underscores the importance of standardized reference systems for mapping and localization technologies [35].

The study on track-to-track association in Intelligent Transportation Systems suggests using a Local Geodetic Frame of Reference to localize automotive sensors precisely. This frame enhances spatial understanding and aids in better decision-making algorithms for navigation, collision avoidance, and safety measures. It also contributes to more robust track association algorithms, minimizing errors due to sensor data inaccuracies and improving system reliability. The research emphasizes the importance of incorporating a local geodetic frame for accurate sensor fusion [23][24]. The Local Geodetic Frame of reference is crucial in Lidar systems, enabling precise ground measurements and accurate mapping of Earth's surface features. It accounts for Earth's curvature and local variations in the gravitational field, ensuring highly accurate measurements. This frame is essential in terrain mapping, urban planning, forestry, and environmental monitoring applications. The use of local geodetic frames in 3D imaging models of airborne LiDAR systems enhances the precision of LiDAR-based 3D imaging, enabling more accurate representation and measurement of surface features and terrain [1][6][45].

Lei and Han's study on sequential nonlinear tracking using Unscented Kalman Filter(UKF) and raw range-rate measurements suggests using a "Local Geodetic Frame of Reference." This frame of reference allows for precise localization of tracked objects within a specific geographic area, improving the accuracy and reliability of the tracking algorithm. Their study suggests that using a local geodetic frame could significantly impact the precision and reliability of object tracking within a defined geographic region [8]. The Local Geodetic Frame of Reference is crucial for sensor fusion and tracking in multi-object environments. It aligns sensor data to a common reference system, ensuring precise local positioning. This frame can aid in accurate track association between radar and Electronic Support Measures data. The application of the Local Geodetic Frame in this research could enhance track association accuracy by providing a consistent frame of reference for radar and ESM data, mitigating biases and improving reliability in multi-sensor environments, thereby enhancing precise and trustworthy tracking in complex scenarios [9][31]. The Local Geodetic Frame of Reference is a concept that can significantly impact methodologies and algorithms for track-to-track association and tracking in sensor fusion systems. It improves spatial accuracy and precision, especially in urban or confined spaces. Local geodetic frames can also adapt tracking and sensor fusion algorithms to specific local environments, enhancing their robustness and efficiency. This adaptability can significantly impact research related to autonomous navigation, sensor fusion, and local traffic management, enhancing accuracy and efficiency [11][13][24]. Local Geodetic Frames are defined by local authorities or organizations to serve specific regions, aligning with the Earth's surface for accurate measurements. They are crucial for high-precision applications over small areas, such as land parcel measurements and infrastructure construction. They ensure all measurements and derived data are relevant and precise within the local context. To apply the concept, one must examine how local measurements and sensor data are integrated and adapted to create a Local Geodetic Frame for enhanced local precision.

## 2.3 Filtering for Object Tracking

Filtering is a crucial process in object tracking that estimates the position and velocity of moving objects based on noisy sensor data. Filtering algorithms like the Kalman Filter or Particle Filter analyze this data to predict an object's current and future states accurately. These algorithms update predictions with new data, gradually refining the object's estimated trajectory and reducing uncertainty. Filtering is critical in applications like autonomous vehicles, aerospace tracking, and robotics for reliable and safe operation. The literature explores the statistical performance of track initiation techniques for object tracking, focusing on their effectiveness in initial stages like data association and state estimation and filtering in object tracking. Z. Hu, H. Leung, and M. Blanchette discuss using the **Probabilistic Data Association Filter (PDAF)** for object tracking and statistical performance analysis. PDAF is a variant of the Bayesian filter used for multi-object tracking and estimation. It addresses the association problem in object tracking, where sensor measurements must be associated with predicted object states. The authors use PDAF to analyze and evaluate various track initiation techniques, determining their effectiveness in initializing and maintaining tracks under noise, uncertainties, and varying conditions [2].

Wang et al. propose a Consensus-Based Track Association methodology using multi-static sensors in a Nested Probabilistic-Numerical Linguistic Environment. They use an adaptive filtering technique called the **Consensus Filter** for object association and trajectory estimation, which integrates information from multiple sensors while considering uncertainties. The filter dynamically adapts to changing environmental conditions and varying sensor characteristics to improve association accuracy. The Track-to-Track Association Algorithm based on an Adaptive Clustering Threshold uses clustering techniques to group track information and adjust the clustering threshold, optimizing the association process [33][37]. Kalman Filters or variants like the Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF) are used in object tracking and sensor fusion to improve accuracy and reliability. They are particularly useful in car-following scenarios because they can process noisy sensor data, estimate an object's state, and handle uncertainties. These filters iteratively update predictions based on sensor measurements, refining the estimated trajectory of objects being tracked. They are commonly used in radar/Lidar sensor fusion for data preprocessing, measurement fusion, filter-based object tracking, and state prediction and update [16].

### 2.3.1 Kalman Filter (KF)

The mathematical algorithm Kalman Filter estimates dynamic system state from incomplete and noisy measurements. Prediction and update are its two steps. The filter uses a system model to predict the next state in the prediction phase and new measurement data to correct it in the update phase. This process accurately tracks object position and velocity for navigation, robotics, and signal processing. Iterating these steps ensures long-term refinement and tracking. The Kalman Filter is a powerful technique used in object tracking to combine radar and Lidar sensors for car-following applications on highways. It handles noisy measurements and uncertainties in dynamic systems, enabling the estimation of vehicle states. The filter continuously updates and refines state estimates based on sensor measurements, resulting in more accurate and robust estimations of vehicles' positions, velocities, and trajectories. This is crucial for safe and effective car-following maneuvers, especially in highway scenarios where accurate tracking of surrounding vehicles is essential [16].

The literature explored using sensor fusion systems in automotive radar sensors to enhance object tracking accuracy and reliability. It suggests that Kalman Filters are likely employed in this context, as they can process noisy sensor data and accurately estimate an object's state. These filters are suitable for tracking applications, especially in the automotive domain, as they can handle linear dynamic systems affected by Gaussian noise. Integrating data from multiple sensors can improve object tracking accuracy, reduce uncertainties, and enhance safety and decision-making in autonomous or assisted driving systems [27]. The Kalman Filter is a widely used technique for object localization and mapping, predicting dynamic system states despite uncertainties and noise. It estimates object parameters using radar and lidar measurements. The effectiveness of each sensor type in providing data suitable for filtering-based object tracking is evaluated through a comparative analysis, assessing their effectiveness in different scenarios [35].

The Kalman Filter algorithm is utilized in object tracking to improve trajectory estimation and association. It integrates information from various sensors, enhancing accuracy and reliability. This method is used within a multi-sensor track-to-track association framework, ensuring accurate and precise tracking even in incomplete or unreliable measurements. It effectively estimates the state of a linear dynamic system under Gaussian noise [46]. Duraisamy, Schwarz, and Wöhler suggest that filtering techniques like Kalman Filters or its variants are likely best used in automotive sensor fusion and track association. These filters are best for their ability to merge data from multiple sources, account for noise and uncertainties, and provide optimal estimates [23][24]. Kalman Filter has also been used in recursive estimation algorithms to predict and update track state. This helps align and associate tracks, ensuring a consistent representation of the object being tracked [11].

The study focuses on the challenge of associating tracks from different sensors in road environments, emphasizing the importance of filtering for accurate object tracking. Conventional filters like the Kalman Filter can facilitate this process for road obstacle detection or vehicle tracking. The research contributes to understanding the challenges in track association and emphasizes the significance of filtering techniques in handling noisy and uncertain sensor data [10]. The Kalman filter is a widely used algorithm for state estimation in dynamic systems, particularly in noisy input data and measurement uncertainty. It is used to fuse data from radar and Lidar sensors for car-following applications on highways, improving object tracking accuracy and reliability. The Kalman filter considers past measurements and probabilistic models, enhancing precision in car-following scenarios [16].

Kalman Filter predicts, updates, and associates tracks or observations from different sensors or time instances in track-to-track association. State prediction, correction, and association criteria are involved. The state prediction and update steps forecast and the state estimate at k-1. Association criteria link tracks using Mahalanobis distance. This is the mathematical notation: ***Step 1: Prediction State***

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k$$
$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

Here:

- $\hat{x}_{k|k-1}$ is predicted estimate at time that give observations $k$ up to time $k-1$.
- $u_k$ is the control vector.
- $P_{k|k-1}$ is the predicted covariance matrix.
- $B_k$ is the control-input matrix.
- $F_k$ is the state transition matrix.
- $Q_k$ is the process noise covariance.

***Step 2: Update State (Correction)***

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}$$
$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (z_k - H_k \hat{x}_{k|k-1})$$
$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

Where:

- $\hat{x}_{k|k}$ is updated state estimate at time $k$ incorporating the new measurement $z_k$.
- $K_k$ is Kalman gain.
- $R_k$ is measurement noise covariance.
- $H_k$ is measurement matrix.
- $P_{k|k}$ is updated covariance matrix.

***Step 3: Association Criteria*** The association step determines the relationship between tracks using criteria such as the Mahalanobis distance, a common distance metric.

$$D = (\hat{x}_1 - \hat{x}_2)^T (P_1 + P_2)^{-1} (\hat{x}_1 - \hat{x}_2)$$

Where:

- covariance matrices are $P_1$ and $P_2$.
- And estimated states are $\hat{x}_1$ and $\hat{x}_2$.

This mathematical representation of the Kalman Filter for track-to-track association shows prediction, update, and association criteria for linking tracks or observations from different sensors or time instances.

### 2.3.2 Unscented Kalman Filter (UKF)

The Unscented Kalman Filter (UKF) is used in object tracking to estimate state using raw range-rate measurements. It handles nonlinearity and non-Gaussian noise in raw data, allowing for more accurate state estimation in radar tracking scenarios. The UKF approximates nonlinear transformations more accurately than the linear Kalman Filter, enhancing the robustness of object-tracking capabilities. This adaptation is crucial for effective object tracking, as it handles noisy sensor data and uncertainties, making the UKF a valuable tool in object tracking [8]. UKF is a nonlinear filtering method used in object tracking systems to accurately approximate state distributions in nonlinear environments. It is crucial for estimating multiple targets distributed sensors observed in complex environments. UKF handles nonlinearities more effectively than traditional Kalman Filters, making it suitable for nonlinear target dynamics or system uncertainty. Its adaptive nature and ability to capture complex system behavior improve object trajectory estimation accuracy in noisy and uncertain environments [9].

The Unscented Kalman Filter (UKF) improves the accuracy and reliability of tracking systems by addressing the limitations of the standard Kalman Filter in dealing with highly nonlinear systems. It approximates mean and covariance propagation through nonlinear functions using a deterministic sampling approach called unscented transformation. This makes the UKF suitable for systems with nonlinearity and uncertainty, enabling accurate state estimation even in the presence of highly nonlinear dynamics [46] The literature explored the UKF for object tracking that accurately captures the posterior distribution of state variables. The UKF manages nonlinearities and uncertainties introduced by sensor biases, enhancing track-to-track association and reducing ambiguity in trajectories. This approach improves the reliability and accuracy of object tracking systems by effectively managing inherent biases [13].

The study compares radar and Lidar sensing methodologies for localization and mapping using the Extended Kalman Filter (EKF). The EKF fuses data from both sensors, evaluating their accuracy, robustness, and suitability. It enables estimation in non-linear systems and integrates data from multiple sensors, accounting for noise characteristics and uncertainties. The EKF was used to estimate the robot's position and map the environment, providing insights into their strengths and limitations in real-world contexts [35]. The Unscented Kalman Filter (UKF) is also used in the Combi-Tor framework for automotive sensor fusion. It effectively handles nonlinearities and non-Gaussian noise in real-world tracking scenarios. UKF is used to fuse data from various automotive sensors, such as Lidar, radar, and cameras, and perform track-to-track association to estimate object trajectories in a dynamic environment accurately. This enhances object tracking accuracy by providing more reliable state estimates, contributing to the Combi-Tor framework's efficiency in associating tracks from different sensors [23][24].

The Unscented Kalman Filter (UKF) is a modified version of the traditional Kalman Filter, specifically designed to handle non-linearities in system models or measurements effectively.

**Prediction State**

***Step 1: Sigma Point Generation:***

The UKF uses sigma points to represent the probability distribution, deterministically capturing the mean and covariance information of state and error vectors.

$$X_k^{[i]} = \begin{cases} \hat{x}_{k|k-1} & \text{if } i = 0 \\ \hat{x}_{k|k-1} + (\sqrt{(n+\lambda)P_{k|k-1}})_i & \text{if } 1 \leq i \leq n \\ \hat{x}_{k|k-1} - (\sqrt{(n+\lambda)P_{k|k-1}})_{i-n} & \text{if } n < i \leq 2n \end{cases}$$

In these equations, sigma points are $X_k^{[i]}$, $n$ represents the state vector's dimension, and the predicted mean state estimate is $\hat{x}_{k|k-1}$, and $P_{k|k-1}$ is the predicted state covariance matrix. while - $\lambda$ is just a scaling factor [18].

***Step 2: Propagation through Process Model***

The sigma points $X_k^{[i]}$ are propagated through the nonlinear process model to obtain predicted sigma points $X_{k+1|k}^{[i]}$:

$$X_{k+1|k}^{[i]} = f(X_k^{[i]}, u_k)$$

Where $u_k$ is the control vector and $f$ is the process model function.

***Step 3: State Prediction***

Using the predicted sigma points $X_{k+1|k}^{[i]}$, estimate the predicted mean state $\hat{x}_{k+1|k}$ and covariance $P_{k+1|k}$:

$$\hat{x}_{k+1|k} = \sum_{i=0}^{2n} W_i^{[m]} X_{k+1|k}^{[i]}$$

$$P_{k+1|k} = \sum_{i=0}^{2n} W_i^{[c]} (X_{k+1|k}^{[i]} - \hat{x}_{k+1|k})(X_{k+1|k}^{[i]} - \hat{x}_{k+1|k})^T + Q_{k+1}$$

Here, $W_i^{[m]}$ and $W_i^{[c]}$ are the weights used for computing the mean and covariance, respectively, and $Q_{k+1}$ is the process noise at time $k+1$.

**Update State**

The update step corrects the predicted state estimate based on received measurements, using equations that use sigma points to update the state estimate based on the measurement model. The sigma points $Y_{k+1|k}^{[i]}$ are computed using the predicted sigma points $X_{k+1|k}^{[i]}$ and the measurement model:

$$Y_{k+1|k}^{[i]} = h(X_{k+1|k}^{[i]})$$

Where the measurement model function is$h$. The updated mean state estimate $\hat{x}_{k+1|k+1}$ and covariance $P_{k+1|k+1}$ are then computed similarly using the measurement sigma points $Y_{k+1|k}^{[i]}$ and weights [29].

The Unscented Kalman Filter's core steps involve sigma points for capturing state estimation and uncertainty through deterministic sampling, enabling efficient handling of non-linearities in system models or measurements.

### 2.3.3 Extended Kalman Filter (EKF)

The Extended Kalman Filter (EKF) is a filtering approach used in object tracking, specifically for nonlinear system models. It linearizes the system model at each time step, estimating the state of nonlinear systems by approximating nonlinear functions with linear functions. In object tracking, the EKF recursively updates state estimates as new measurements become available, incorporating dynamic model predictions and radar measurement correction. This filtering approach improves track-to-track associations, particularly in surveillance scenarios involving radar observations, even in noisy data [50].

The Extended Kalman Filter (EKF) uses an equation to calculate the Kalman Gain, which is determined by the equation:

$$K = P_{k|k-1} \cdot H^T \cdot (H \cdot P_{k|k-1} \cdot H^T + R)^{-1}$$

Here, Kalman Gain $K$ predicts the covariance of the state $P_{k|k-1}$ at time $k$, where $R$ is noise covariance matrix measurement and $H$ is Jacobian measurement. This equation influences the updated state estimate during the correction step.

The Extended Kalman Filter (EKF) is used in automotive sensor fusion systems to enhance object tracking accuracy. It is a variant of the Kalman Filter, designed to handle nonlinearities in system dynamics and measurement models. EKF is crucial in processing data from next-generation radar sensors, estimating the state of moving objects by predicting their positions and velocities while considering noisy sensor measurements and system dynamics. This refines estimates of object positions and velocities, improving object tracking accuracy, which is crucial for automotive safety and autonomous driving functionalities [27].

### 2.3.4 Particle Filter

The Particle Filter, also known as the Sequential Monte Carlo (SMC) method, is a probabilistic technique used in computer vision and signal processing for object tracking. It uses a set of random particles to represent the probability density function of a system's state. The algorithm starts with an initialization step and then estimates the next state for each particle based on motion models or dynamic equations. The weights of the particles are adjusted based on the likelihood of observed measurements, representing their relative importance in the state estimation process.

The main particle filter weight update equation is:

$$w_t^{(i)} = \frac{p(z_t|x_t^{(i)})}{\sum_{j=1}^{N} p(z_t|x_t^{(j)})}$$

Equation updates or corrects Particle Filter algorithm. To calculate particle weights $w_t^{(i)}$ at time $t$, divide the estimated likelihood of the observed measurement $z_t$ by the sum of all particles' likelihood. These weights indicate how important each particle is in resampling and estimating the system's true state.

The study explores using the particle filter in Maritime Tactical Data Systems to handle large sensor data. The Particle Filter, also known as the Sequential Monte Carlo (SMC) method, is a Bayesian filtering technique for nonlinear and non-Gaussian state estimation problems. It approximates the posterior probability distribution using particles or samples, evolving them through time using the system's dynamics and updating their weights based on observed measurements. This filter is suitable for scenarios where traditional methods like the Kalman Filter struggle due to non-Gaussian noise or complex dynamics [51].

## 2.4 Track-to-Track Association

The track-to-track association is a crucial process in multi-object tracking systems, linking or coordinating observations across different sensors or time frames that correspond to the same physical object or target. It is essential in fields like radar, Lidar, and sensor fusion for applications ranging from surveillance to autonomous vehicles. The core concept of track-to-track association is to create a coherent and consistent trajectory for each detected object. The methodology for this process includes data representation, feature extraction, and association algorithms. Geometric methods use spatial proximity or geometry-based criteria to associate detections, while probabilistic approaches estimate the likelihood of associations based on prediction models and measurement uncertainties. Clustering techniques group detections into clusters, assuming observations within the cluster correspond to the same object. Challenges include ambiguity, noise uncertainty, and complexity as the number of objects or sensors increases. Despite these challenges, the track-to-track association can be effectively used in various fields, such as radar, sonar, object tracking, and surveillance systems.

The study proposes fuzzy logic-based approaches for track-to-track association and fusion in distributed multisensor-multitarget environments, addressing uncertainties from sensor variability. Fuzzy logic allows for flexible decision-making, accommodating imprecise or uncertain information. It reconciles discrepancies caused by sensor differences, aiding in correct track association despite variations in sensor characteristics. The proposed algorithms use hierarchical clustering to group similar observations or tracks together, aiming to identify and associate tracks from different sensors while compensating for biases inherent in sensor data [9][31]. The challenges of handling uncertainties in nontraditional measurements in the track-to-track association are essential to improve the accuracy and reliability of associating uncertain measurements with existing tracks, enabling a more comprehensive understanding of the target's behavior or movement. This need to investigate assignment costs for multiple sensor track-to-track association, focusing on measurement uncertainty in the context of track-to-track association across different sensors. This emphasizes the importance of robustly handling measurement uncertainties in track-to-track association [17][20].

Track-to-track association faces challenges due to data clutter and occlusions in dense target environments. A leader-follower online clustering algorithm was proposed to mitigate data clutter and enhance accuracy. In 2019, the Combi-Tor framework was introduced for automotive sensor fusion to address occlusions caused by surrounding vehicles or objects. This framework combines multiple association methods, leveraging spatial and temporal information or data from various sensor modalities. The study contributes methodologies specifically tailored to address these challenges, aiming to improve the accuracy and reliability of track association despite challenging environmental conditions [21][24]. The track-to-track association is crucial for accurate target tracking but faces challenges like missing or noisy observations. Sensor bias and ambiguity management are key issues, as unreliable data can introduce ambiguity. Strategies to handle these include data fusion techniques, sophisticated filtering, and maintaining track consistency. Predictive modeling, correction mechanisms, historical data, alternative sensors, and signal processing techniques can be used to address signal degradation due to rain fade. Addressing these challenges is essential for maintaining track consistency and preventing signal degradation [13][38].

The literature emphasizes the significance of statistical analysis in track-to-track association, focusing on the performance of track initiation techniques and assignment costs for multiple sensor tracking. The analysis evaluates the accuracy, robustness against noise, sensitivity to conditions, and effectiveness of these techniques. The role of assignment costs in quantifying track association is crucial for improving accuracy and reliability, ultimately improving the effectiveness of multi-sensor tracking systems [2][11]. Track-to-track association links observations from different sensors over time, establishing correspondences with multiple targets or objects. Clustering techniques group similar observations or tracks based on characteristics or spatial proximity. A plot-track association algorithm uses hierarchical and density clustering analysis to associate plots with existing tracks. This is particularly useful for automotive sensor fusion, where tracks from various sensors are linked together, identifying and linking tracks belonging to the same physical object or target [7][23].

The track-to-track association is crucial for understanding environment and object behavior as well. The literature explores sensor data fusion, focusing on ambiguity management and adaptive clustering threshold algorithms to improve the accuracy and reliability of association results by managing biases and ambiguities caused by sensor characteristics. By dynamically adjusting parameters based on sensor data characteristics, these strategies ensure more accurate and reliable track-to-track associations, essential for applications like surveillance, autonomous vehicles, and target tracking systems [16][33]. Track-to-track association links observations from different sensors to the same physical object or target, considering sensor variability and uncertainties. Advanced filtering methods, particularly Kalman filters, are used for estimating and associating tracks in complex scenarios, such as sequential nonlinear tracking and passive multisensor systems [8][47].

### 2.4.1 Clustering Based Approaches

Clustering-based approaches are techniques used to group observations or tracks based on similarities, aiding in track association. These approaches for track-to-track association use algorithms to group tracks or observations with similar spatial, temporal, or feature characteristics.

#### 2.4.1.1 Hierarchical Clustering

Hierarchical clustering is used in various fields, including plot-track association, leader-follower online clustering, and track-to-track data association for automotive sensor fusion. It organizes data points based on spatial characteristics, facilitating track association based on proximity. In dense target environments, it groups tracks exhibiting similar movement patterns, facilitating association in complex and crowded environments. In automotive sensor fusion, it clusters sensor data based on feature similarities, aiding in accurate track association. Hierarchical clustering is crucial in track association, organizing observations, grouping tracks in dense target environments, and simplifying the association or linking of related tracks in various scenarios [7][21][23].

Researchers M. Lei and C. Han have found that Hierarchical Agglomerative Clustering (HAC)-based approaches are effective in improving track association in multi-target tracking systems. HAC's versatility allows for grouping tracks based on factors like spatial proximity or temporal consistency, enhancing the tracking process in multi-target scenarios. This efficient association and fuse of track data aids in more accurate target tracking and estimation. HAC's versatility is particularly beneficial in road obstacle detection, where diverse sensor data must be integrated and associated effectively [17][4]. The hierarchical aggregation clustering (HAC) algorithm is used in applications like automotive sensor fusion and radar-based track association. It uses single linkage and complete linkage methods to merge clusters based on distance measures. Single linkage emphasizes minimum distance, resulting in elongated clusters, while complete linkage prioritizes maximum distance, potentially resulting in compact clusters. These methods are chosen based on application requirements [24][31].

Hierarchical Agglomerative Clustering (HAC) uses distance metrics like Euclidean or Manhattan distances to group tracks with similar spatial or feature characteristics. Euclidean distance, calculated as the straight-line distance between two points in space, is suitable for grouping tracks based on spatial proximity or similarity in three-dimensional space. Manhattan distance, computed along orthogonal axes, can capture spatial relationships between tracks considering their positional differences. In celestial mechanics and dynamical astronomy, Euclidean distance captures overall positional or feature differences, while Manhattan distance emphasizes differences along specific attributes [36][41]. HAC's hierarchical structure enables it to adapt to changing track similarities, making it useful in multi-sensor fusion systems or environments with changing track characteristics. It can dynamically group tracks based on spatial, temporal, or feature similarities, enabling effective track association in dynamic scenarios. HAC's adaptability is particularly important in spatial registration and multi-sensor track association, where it integrates data from multiple sensors, considering evolving similarities or changing spatial relationships [37][46].

HAC's hierarchical structure enables it to adapt to changing track similarities, making it useful in multi-sensor fusion systems or environments with changing track characteristics. It can dynamically group tracks based on spatial, temporal, or feature similarities, enabling effective track association in dynamic scenarios. HAC's adaptability is particularly important in spatial registration and multi-sensor track association, where it integrates data from multiple sensors, considering evolving similarities or changing spatial relationships [34][35][47].

#### 2.4.1.2 Density-based Clustering

Density-based clustering methods like DBSCAN identify clusters based on dense regions of data points, defining them as continuous regions with high-density data points separated by lower-density regions. DBSCAN identifies core points and expands clusters by adding reachable points within their neighborhood. Density clustering is robust against irregular data densities, making it suitable for scenarios where targets might have different data point concentrations [21]. Hierarchical and density-based clustering helps identify similar tracks or observations spatially or feature-wise, aiding in track association in multi-target tracking scenarios. The study examines statistical performance analysis of track initiation techniques, which involve clustering observations or measurements based on spatial or temporal proximity. It explores an algorithm based on hierarchical clustering for multi-target tracking of multi-sensor data fusion, optimizing track association by considering spatial and temporal relationships among measurements [2][5][26].

### 2.4.2 Statistical Analysis Methods

Statistical analysis methods are essential for evaluating the performance, reliability, and efficacy of track initiation techniques in target tracking systems. These methods involve quantitative assessment techniques, using metrics like accuracy, false alarm rate, detection and miss rates, and efficiency measures. They help assess the correctness of associations made by track initiation techniques, reducing false positives and ensuring accurate tracking. Monte Carlo simulations provide robust performance evaluation under various scenarios, such as sensor noise, clutter, or target dynamics. These methods help evaluate the accuracy and robustness of track initiation algorithms in associating tracks correctly, especially in noisy sensor data or complex environments. Researchers can refine these algorithms by identifying weaknesses or inefficiencies, improving reliability, and reducing false associations or missed tracks. These methods provide decision support, providing critical insights for selecting or designing track initiation methods suitable for specific applications [2].

### 2.4.3 Fusion Methods

Fusion methods are essential for improving track-to-track association by combining data from various sensors, such as radar and Lidar, to enhance accuracy and reliability. These methods include radar/Lidar Fusion, which provides robustness in adverse conditions, and track fusion, which merges information from individual tracks into a comprehensive track. Feature-level fusion extracts specific features from each sensor's data to improve association accuracy. Probabilistic fusion uses Bayesian methods to calculate the probability of association between tracks from different sensors. Hierarchical clustering groups are tracked based on similarity or proximity in a multi-dimensional space. Machine learning approaches like Deep Learning optimize the fusion process and improve association accuracy. Fusion methods offer enhanced accuracy, robustness, comprehensive information, and reduced ambiguity. They mitigate the weaknesses of individual sensors and provide a holistic view of target movements [4].

# Chapter 3

# Methodology

This chapter describes the research methodology of our work, including how our study goals are achieved. Our methodology's key steps are the pre-processing of network packets, time synchronization, threshold-based cluster formation, and accuracy calculation. The specific clustering approach used in this research is the hierarchical clustering algorithm. Different scenarios have been developed to demonstrate the performance of the hierarchical clustering-based track-to-track association algorithm in terms of accuracy computation compared to ground truth. The following figure shows the key stages of our methodology, which are discussed individually in this chapter.

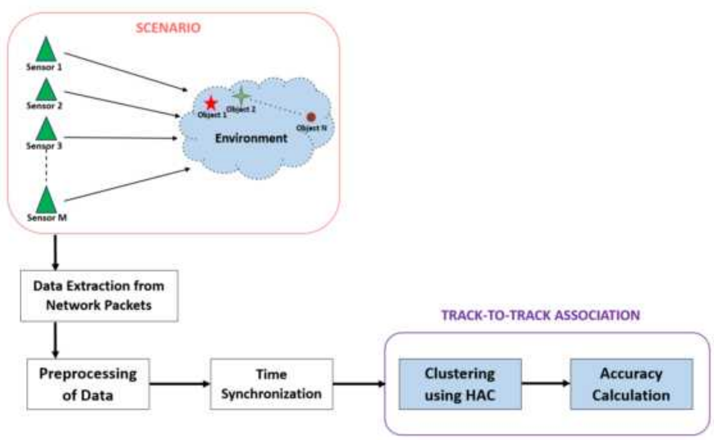



Figure 3.1: Block Diagram of Methodology

## 3.1 Preprocessing

Each scenario will generate a set of packets used to perform track-to-track association. These packets often include multiple attributes that are not necessarily required for the track-to-track association, or they are first modified before their utility in the association algorithm. Extracting the tracking data such as latitude, longitude, and altitude of the target objects at each time instant and converting these tracks in different coordinate systems to incorporate useful information is crucial for processing and analysis. The process also involves identifying and isolating specific information related to the monitored tracks, such as positional, velocity, and trajectory-specific data points. The extracted information is organized in a format suitable for further analysis and track association algorithms, such as creating track histories or profiles from the received packets. Track association algorithms use this stage as a filter to improve data quality and reliability. This preliminary step greatly affects the precision and efficiency of track association.

### 3.1.1 Data Extraction

In an object-tracking sensor network, data processing begins with collecting and packaging sensor data into a High Priority Target (HPT) and Secondary Priority Target (SPT). The initial phase will involve handling network packets containing tracking data. These packets undergo multiple network processing stages. First, sensor data is collected and packaged into High Priority Target (HPT) or Secondary Priority Target (SPT) packets based on target priority. HPT tracks contain critical tracking data like real-time positional, velocity, and trajectory data, which is essential for time-sensitive applications. They are used for applications where rapid decision-making based on accurate, up-to-date tracking data is crucial. SPT tracks are used for less urgent objects or those with less frequent tracking data updates, containing essential tracking information but prioritized differently in the network due to targets' importance or resource allocation constraints. The data processing workflow begins with collecting and packaging raw sensor data, sorted into different types of packets based on the priority level of tracked objects. HPT and SPT tracks are classified and extracted based on their specific positions in the network packets with their priority levels highlighted within the tracking system. In addition to HPT and SPT data, sensor's own position in LLA is also tracked, extracted and in some cases used for track-to-track association to overcome the issue of locking friendly object as potential target.

### 3.1.2 Data Conversion

The extracted data stores the location information of sensors and their tracked object in Latitude, Longitude, and Altitude (LLA). Latitude measures the distance from the equator, Longitude measures the position east or west from the Prime Meridian, and Altitude refers to the height above or below a reference point. LLA coordinates help map targets or objects accurately, improve object tracking through track-to-track fusion methods, and determine satellite position and motion. The data from LLA will then be converted into Earth-Centered, Earth-Fixed (ECEF) coordinates, a common coordinate system used in geodesy and navigation. Converting LLA data to ECEF coordinates is essential to tracking data processing. LLA is a geodetic coordinate system that measures latitude, longitude, and altitude above the Earth's reference ellipsoid in angular and linear units. This format is intuitive for human interpretation and widely used in mapping and navigation. It is not ideal for calculating distances and angles between points, especially when they are not on Earth, or precision tracking is needed over long distances.

In contrast, ECEF coordinates are Cartesian coordinates (X, Y, Z) in three dimensions with the Earth as the origin. This system simplifies the mathematical operations used in association, making it useful for computation. Simple linear algebra can calculate distance, velocity, and angular direction between any two points in ECEF. Using a high-precision application like satellite tracking, ballistic trajectories, and data integration is crucial. The conversion to ECEF creates a uniform data format compatible with global positioning systems and satellite navigation data. The ECEF system is constant, while the LLA system depends on the reference ellipsoid, which varies by region. ECEF grounds all data points in a consistent frame of reference, eliminating ambiguity from multiple geodetic models. Air traffic control, military operations, and global surveillance systems need this consistency for reliability and accuracy. The tracking system will operate reliably, integrate data more seamlessly, and precisely perform complex analysis and decision-making calculations by converting LLA to ECEF.

### 3.1.3 Local Tracks Formation

Local track formation is essential in sensor networks that track objects, especially with multiple sensors, to perform association. Integrating sensor data creates a continuous and coherent track of an object's trajectory over time. The network's sensors detect an object's location, velocity, and direction. These sensor readings may need to be completed or obstructed by environmental factors or object movement. Thus, multiple sensors will be combined to understand the object's behavior. It is crucial because sensors have different capabilities or fields of view as they complement each other's data to fill gaps and reduce uncertainties. The data and sensor performance of the network create a 'local track' for each object that is precise and reliable.

### 3.1.4 Error Mitigation Using Standard Deviation

The standard deviation is crucial for assessing data precision and accuracy in sensor and object tracking.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

Standard deviation ($\sigma$) measures the variability or dispersion of sensor readings around an average value. To calculate, we subtract the mean $\mu$ from each sensor measurement $x_i$, square the deviation to address negative values, and emphasize larger differences. The variance is the average of these squared deviations, and the standard deviation is its square root. This statistical measure is crucial when sensor readings must accurately approximate an object's position. A low standard deviation indicates that sensor readings are close to the mean, indicating reliability and data precision. A high standard deviation indicates a wide range of measurements, suggesting sensor inaccuracies, environmental factors affecting sensor performance, or unpredictable object movements. System quality can be tracked using standard deviation calculations on sensor data. If the standard deviation exceeds acceptable thresholds, sensors, filtering algorithms, or the deployment environment will be calibrated to mitigate external variables. Understanding and minimizing standard deviation improves object tracking system robustness and reliability.

## 3.2 Time Synchronization

Object tracking requires sensor data time synchronization to match sensor readings at a specific time instance to perform association. Kalman Filtering has been used in this research to perform this synchronization. The statistical method minimizes the difference between predicted and actual sensor measurements to estimate a system's state dynamically. Due to latencies, sensors collect data at slightly different times. The Kalman Filter helps align this data by predicting the system state at specific instances and updating these predictions with sensor readings. It creates a coherent and synchronized dataset that accurately tracks the tracked object's position and motion at that specific time instant.

Sensor networks need temporal synchronisation to combine sensor readings for a thorough analysis of the monitored environment. The data fusion module combines intermittent information from three sensors in Figure 3.2 to gain a complete understanding of the observed environment. These sensors are not fully synchronised; thus, the module sometimes has to wait for all sensors to read. Figure 3.2 part b displays the same sensors with state prediction. Assumes sensors took the module's synchronised readings simultaneously. Thus, data flows more easily, which is crucial in real-time sensor-based decision-making systems. Finally, sensor networks need temporal synchronisation to accurately combine sensor readings into a complete image of the environment.

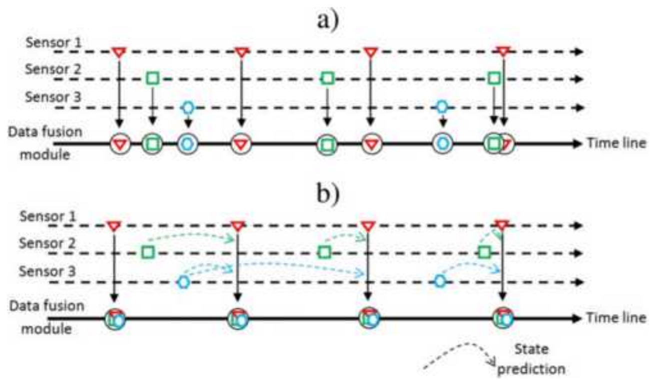



Figure 3.2: Asynchronous and Synchronized Tracks

### 3.2.1 Kalman Filtering

The Kalman Filter is a method that efficiently estimates process states while minimizing mean squared error, estimating past, present, and future states even when the modeled system's nature is unknown, using Predict and Update steps to deal with uncertainty or data noise.

Prediction of estimation state:

$$\hat{x}_{k|k-1} = F_k x_k - 1 + B_k u_k$$

Model the covariance estimate:

$$P_{k|k-1} = F_{k|k-1} F_k^T + Q_k$$

$\hat{x}_{k|k-1}$ is the predicted state estimate, $F_k$ is the state transition model, $\hat{x}_{k-1|k-1}$ is the previous state estimate, $B_k$ is the control-input model, $u_k$ is the control vector, $P_{k|k-1}$ is the predicted covariance estimate, $P_{k-1|k-1}$ is the previous covariance estimate, and $Q_k$ is the process noise covariance. The filter uses a system model to predict its next state in this step. A filter will track a moving object and predict its location in the future in a short time based on its speed and direction. Prediction error or uncertainty exists because this is a prediction.

Calculate Kalman gain:

$$K_k = P_{k|k-1} H_k^T (H_k + R_k)^{-1}$$

State estimate update:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - H_k \hat{x}_{k|k-1})$$

Update covariance estimate:

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

$K_k$ represents the Kalman gain, $H_k$ represents the observation model, $z_k$ represents the actual measurement, $R_k$ represents the measurement noise covariance, and $I$ represents the identity matrix. Here, the filter corrects its predictions with updated measurements, just like a GPS giving the object an updated location. The filter adjusts its prediction to be more accurate using the updated data. A compromise is reached between its prediction and updated data. This step reduces system state estimate uncertainty, improving tracking accuracy.

The Kalman Filter equations are iterated at each time step, updating the predicted state and covariance estimates with updated measurements. This Predict-Update cycle is crucial to improve the accuracy of sensor data in tracking systems. Environmental conditions and technical limitations can cause noisy or incomplete sensor data. The Kalman Filter blends previous state predictions with updated sensor data to address such uncertainties while tracking the objects. Despite data errors, it keeps the system's understanding of the object's state as accurate as possible, which is necessary for this research to calculate and improve the accuracy of our algorithm. Navigation systems, automated vehicles, and aerospace engineering require this process for precision.

## 3.3 Cluster Formation using HAC Algorithm

The problem statement addressed in this research is to analyze the problem of track-to-track association by applying the hierarchical agglomerative clustering algorithm to different scenarios in a network of multiple sensors detecting multiple targets. The reason to implement Hierarchical Agglomerative Clustering (HAC) has been influenced by many reasons, especially in object tracking, where understanding object relationships and groupings is crucial. First, HAC is an easy way to build a cluster hierarchy for this application, where data observation at different granularities is required. This hierarchical structure makes data analysis flexible and shows which objects (or tracks) are similar and how these groups can be clustered at higher levels. We want to see how objects form groups and clusters in tracking applications.

HAC is easy to implement as it doesn't need initial guesses like k-means clustering. In this research, we have real-world tracking scenarios that involve an unknown number of clusters. HAC is flexible and adaptable to different data and similarity measures. Tracking applications requires adaptability because data can take many forms, and similarities can vary depending on the task. HAC clustering is deterministic and simple. The agglomerative process is simple: merge the closest clusters. This determinism is useful to explain how the algorithm clustered a result. HAC requires computing and updating distances between all pairs of clusters in each iteration, making it computationally intensive, especially with large datasets. HAC is still preferred in many practical applications, especially where hierarchical clustering precision outweighs computational constraints. SO HAC's ability to reveal data structure at multiple levels, ease of implementation, adaptability to different data types, deterministic nature, and intuitive understanding of the clustering process make it useful in many applications, including object tracking.

The Hierarchical Agglomerative Clustering (HAC) algorithm requires several steps to form clusters, especially with a single linkage. Each step helps cluster tracks (like moving objects) by similarity. This method groups similar objects, starting with each track as a cluster and then merging these clusters as one rises.

### 3.3.1 Distance Matrix Formulation

Distance Matrix formulation is important to determine how close each track is to others before cluster formation. Creating a distance matrix in Hierarchical Agglomerative Clustering (HAC) is essential to understanding the relationships between each dataset pair of tracks (or data points). Like a table with each cell representing the distance between two tracks, the distance matrix determines how clusters will be merged in HAC. It starts with each track as a cluster with $N$ tracks. An $N \times N$ distance matrix will be created, with $d(i, j)$ representing the distance between tracks $i$ and $j$. This distance will be calculated for each pair of tracks.

The data type and distance matrix determine the distance formula. For geometric data, the Euclidean distance will be calculated between two points $x_i$ and $x_j$ with $m$ dimensions (or features), using the formula:

$$d(i, j) = \sqrt{\sum_{k=1}^{m} (x_{ik} - x_{jk})^2}$$

The $k^{th}$ feature values for tracks $i$ and $j$ are $x_{ik}$ and $x_{jk}$, respectively. If coordinates represent tracks at a given time, the Euclidean distance will measure their distance in space for object tracking. Depending on the data and analysis needs, Manhattan distance, Minkowski distance, or domain-specific metrics can also be used. Distances are calculated and entered into the matrix. This matrix will be symmetric (since $d(i, j) = d(j, i)$) and has zero diagonal elements (representing track distances). The clustering process begins with this matrix. Each HAC iteration merges the clusters with the smallest distance in the matrix and updates it. Thus, the distance matrix formulation is essential for capturing all track pairs' initial similarities or dissimilarities before hierarchical clustering. It contains the dataset's essential spatial or feature-based relationships, which HAC will use to build the cluster hierarchy iteratively.

### 3.3.2 Global Threshold Calculation

After calculating all track distances, a global threshold will be applied. This threshold determines how similar or close tracks should be clustered. Tracks lying within this threshold will be grouped. The HAC algorithm process relies on this global threshold calculation to form clusters. This threshold is based on the distances between each pair of tracks in the data set to group only nearby tracks. To calculate the threshold, we will begin with the distance matrix, each element representing a track separation. Zero values in the matrix are replaced with a large number (the average of all the values in the distance matrix in this research) to remove them from affecting the threshold calculation since a track is zero distance from itself. Importantly, the threshold will only consider different track distances, not track distances to themselves.

Next, the algorithm will find each track's minimum non-zero distance, the closest distance to any other track. Setting the global threshold to the maximum of these minimum distances. The threshold is the largest minimum distance between any track and its nearest neighbor, calculated as $\max_i(\min_{j \neq i}(d_{ij}))$. This method ensures that the threshold will not be too low to merge only similar tracks or too high to merge distinct tracks. This threshold is crucial to clustering. It determines the nearest tracks needed to form a cluster. This mechanism helps in datasets with a non-uniform scale of similarity or proximity or unknown preliminary assumption. The algorithm will adapt to the data structure and create meaningful and representative clusters by dynamically calculating the threshold based on dataset distances. This adaptive threshold setting ensures coherent clusters that reflect the tracks' spatial or feature-based relationships.

### 3.3.3 Single Linkage Method

The single-linkage method, also known as the Nearest Neighbor Technique, forms and merges clusters focusing on the closest pair of points (or tracks) between two clusters; this method defines distance differently. It calculates the distance between two clusters as the shortest distance from any member (or data point) to any other member in the distance matrix. Each cluster is a point group, and this method finds the closest two points from each cluster. These two nearest neighbors will determine the cluster distance. In the single-linkage, the distance between two clusters $C_1$ and $C_2$ is calculated as follows:

$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$

where $d(x, y)$ is the distance between points $x$ and $y$. This formula finds the minimum distance between all possible pairs of points, one from each cluster. Single-linkage clustering is best for detecting natural, linear clusters or well-separated clusters. Since it links clusters along nearest neighbors, it identifies stretched or elongated clusters.

### 3.3.4 Cluster Formation

The clustering started by treating each track as its cluster using Hierarchical Agglomerative Clustering (HAC), especially with the single-linkage approach. The algorithm then merged these clusters iteratively based on proximity using a single linkage. The key was repeatedly combining the closest cluster pairs. The algorithm compared the distances between all pairs of clusters to find the pair with the shortest distance between any of their members at each step. A threshold has been applied to compare this distance. A single cluster has formed where the distance was less than this threshold. And for higher distances, clusters stay separate. This merging continues iteratively, and after each new cluster formation, the algorithm recalculates the distances between the new cluster and all other clusters and searches for the closest pair of clusters to merge. The process continues until no more cluster pairs can be merged without exceeding the distance threshold. This threshold controls cluster combination by ensuring that only clusters that meet the closeness criteria are clustered.

Clusters decrease in number and grow in size as the algorithm progresses. The hierarchy of clusters is formed by combining clusters based on proximity. HAC's hierarchical structure shows how these clusters are formed and combined at each stage, not just the final clustering. The single-linkage method merges clusters based on their closest points, creating a chaining effect. It helps find elongated or linear clusters. It's sensitive to outliers because a single distant point can significantly affect clustering. The clustering process in HAC with single linkage groups tracks or data points based on their similarities, with the flexibility to define 'closeness' through the distance matrix. This systematic approach is ideal for natural data point clustering when the clusters have irregular directions or densities.

## 3.4 Accuracy Calculation

The accuracy of clusters is calculated by comparing predicted classifications to ground truth data using a confusion matrix. This method measures algorithm performance while considering all clustering outcomes. This calculation relies on True Positives (TP), which occur when the predicted classification and ground truth agree that certain objects belong to the same cluster. It's the total count of all pairs of objects; the algorithm correctly classifies it as the same group after comparing it to the ground truth. Conversely, True Negatives (TN) occur when the predicted classification and ground truth agree that certain objects are not in the same cluster. This count shows the algorithm's accuracy in identifying non-group objects. False Positives (FP) occur when the predicted classification groups two objects together when the ground truth places them in separate clusters. These numbers show how the algorithm groups objects incorrectly. False Negatives (FN) occur when the predicted classification places two objects in different clusters when the ground truth places them in the same cluster. The algorithm's missed group members are counted here.

This formula calculates the accuracy of the clustering algorithm:

$$\text{Accuracy} = \frac{\text{Correctly Classified Values}}{\text{Total Predicted Values}} \times 100\%$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100\%$$

This formula calculates the percentage of correct TP and TN classifications out of all classifications. A higher accuracy percentage means the clustering algorithm better identifies object groups according to ground truth. This accuracy calculation method shows the algorithm's performance in grouping objects and identifying non-objects.

### 3.4.1 Ground Truth Comparison

Accurate and reliable measurements or observations determine the ground truth. It comes from precise sensors and well-documented observations in real life. Scenario generators simulate tracking conditions, which can be ground truth in simulations. Ground truth data accurately represents the actual scenario or the most reliable data of an object's positions and movements confirmed by reliable sources or measurements. Each object or track in the clustering results is matched to the ground truth data to compare them. The algorithm's classification (i.e., an object's cluster) is then compared to the ground truth. This process involves checking whether the algorithm placed pairs of objects in the same cluster as the ground truth.

Accuracy is a simple and quantifiable measure of clustering algorithm performance. For refining the algorithm and understanding its applicability in real-world scenarios, the accuracy metric compares clustering results to a known or established ground truth to assess its reliability and effectiveness under different conditions.

## 3.5 Computational Aspects of Association

The computational aspect of the association algorithm is important from a practical implementation point of view. In our setup, we perform a track-to-track association after every 200 milliseconds. This means that the processing time of the association algorithm should be less than 200 milliseconds. If the processing time exceeds 200 milliseconds, the processor will miss some associations and affect the results. This is shown in Figure 3.3, where it is clear that after the first association phase, the processing time is less than 200 milliseconds. Therefore, before the second association time, the algorithm is ready for the second phase of association.

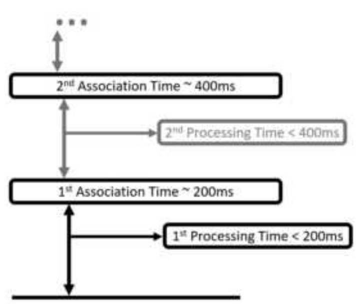



Figure 3.3: Computational Aspects of Association

Similarly, this process will continue as long as the processing time of the association algorithm is less than 200ms. In our HAC algorithm implementation, the computational time/ processing time is always less than 60 ms, which means it isn't missing any associations, and the results generated from our HAC algorithm are accurate and lie within our association time limit.

# Chapter 4

# Results and Discussion

In this chapter, we presented the results of the simulations used to evaluate the HAC algorithm in different scenarios. The scenarios were carefully designed so that the sensors tracked a fixed number of targets throughout their journey. This means that we will already know how the output of our association algorithm will vary. This will help us identify the HAC algorithm's performance in the given scenario. The clusters involved in each scenario are also plotted, and the accuracy computations are shown for each association time.

## 4.1 Two Sensors One Target

We first consider a scenario with two moving sensors and one moving target observed/measured by both sensors throughout its journey. The path followed by both the sensors (green and blue triangles) and the object (red star) is shown in Figure 4.1.

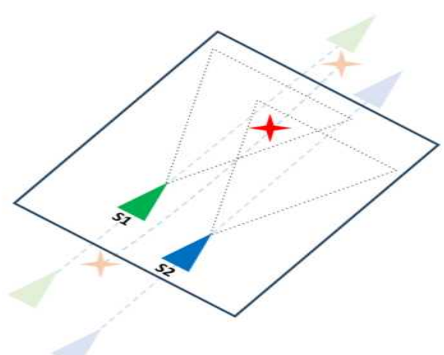



Figure 4.1: Scenario 1: Two Sensors One Target

Since both sensors $S_1$ and $S_2$ observe the object throughout its journey, each sensor will have one local track for that object. When these sensors share their tracks, they will have one local and one shared track. With no association, it is possible that the two sensors falsely show two different objects. The association algorithm should associate the tracks and cluster them in one group. If the association algorithm shows two different objects at any time in the trajectory of the target object, the algorithm will be in error. To observe the accuracy in this case, we developed a scenario similar to Figure 4.1 in the validation optimization system (VOS). The visualization of the VOS-generated scenario is shown in Figure 4.2.

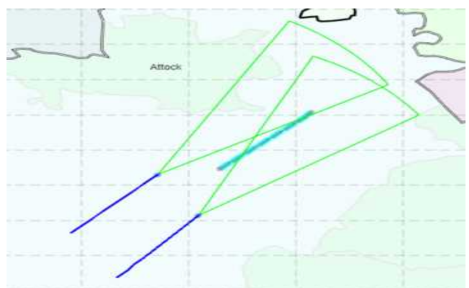



Figure 4.2: VOS scenario 1

Notice that VOS assigns blue colors to the sensors' path and red to the target path. The green cones show the range of the sensors, and the highlighted red path of the target in turquoise shows that the target is measured throughout its journey. The packets associated with the scenario will contain local and shared tracks of the two sensors. These packets will extract the track information and implement the HAC algorithm discussed in Section 3.3. If the association is accurate, the dendrogram for this scenario will involve one cluster of $S_1$ and $S_2$. This is shown in Figure 4.3.

The dendrogram shows that the shared and local tracks of the sensor are on the x-axis and clustered in one group. The y-axis shows the distance between the two tracks in meters (m). Since only one cluster is formed in this scenario, the two tracks with sensors 1 and 2 represent the same object.

On implementation of the HAC algorithm and comparison of the association results with ground truth, the accuracy for this scenario is computed to be 100%. This means that the proposed association algorithm detects the correct number of objects and the sensors observe it throughout its journey without error. The HAC algorithm detects only one object at each time instance, and there's zero error throughout the scenario simulation.

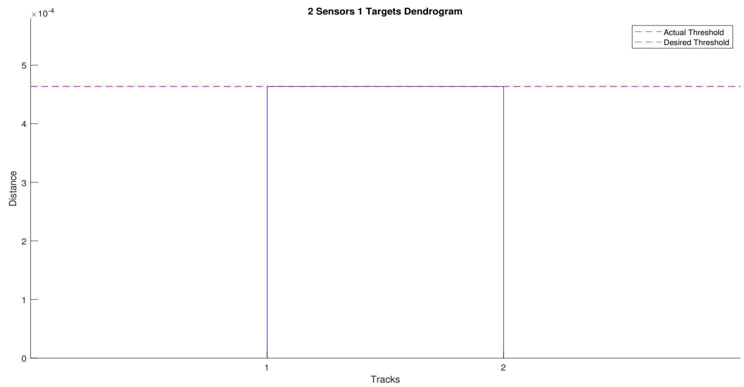



Figure 4.3: Dendrogram for Scenario 1

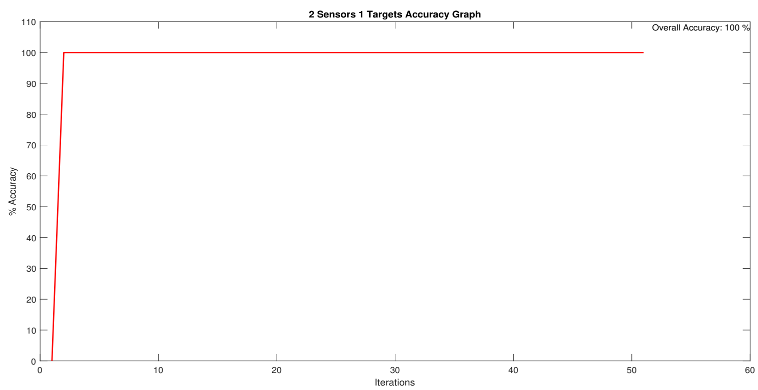



Figure 4.4: Accuracy of 2 sensors tracking 1 object

## 4.2 Two Sensors Two Targets

In this scenario, two moving sensors are tracking two moving objects/targets. The first target is observed by only sensor 1, while the second target is measured by both sensors throughout its journey. The path followed by the two sensors and the two objects is shown in Figure 4.5.

Since $S_1$ observes two objects while $S_2$ observes only one object throughout its journey, $S_1$ will have 2 local tracks for each object while $S_2$ will have only one local track. When these sensors share their tracks, $S_1$ will have two local tracks and one shared track, while $S_2$ will have one local and two shared tracks. Without association, it is possible that the sensors $S_1$ or $S_2$ falsely report three numbers of objects based on the given tracks. The association algorithm should group these tracks into two clusters, one of which will have two associated tracks.

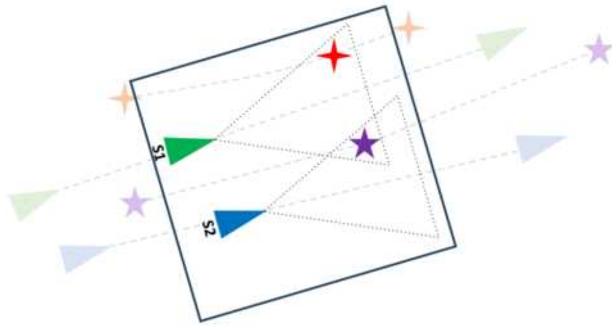



Figure 4.5: Scenario 2: Two Sensors Two Targets

If the association algorithm shows more or less than two different clusters at any time in the complete trajectory, the algorithm will be in error.

To observe the accuracy in this case, we developed a scenario similar to Figure 4.5 in the validation optimization system (VOS). The visualization of the VOS-generated scenario is shown in Figure 4.6.

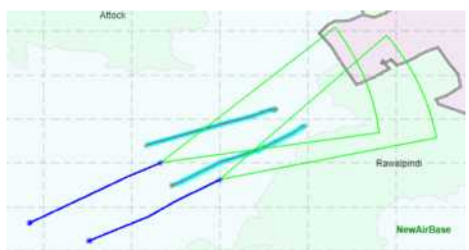



Figure 4.6: VOS scenario 2

The packets associated with this scenario will contain local and shared tracks of the two sensors. These packets will extract the track information and implement the HAC algorithm in Section 3.3. The dendrogram for this scenario will involve two clusters, with one containing the tracks of $S1$ and $S2$ and the other containing a single track of $S_1$ if the association is accurate. This is shown in Figure 4.7.

The dendrogram shows the shared and local tracks of the sensor on the x-axis, and the blue line shows that they are clustered in one group. Since only two clusters are formed in this scenario, sensor 1 and sensor 2 tracks are associated, while the track with sensor 1 is not associated.

On implementation of the HAC algorithm and comparison of the association results with ground truth, the accuracy of this scenario is 66.67%, which shows that the proposed association algorithm is detecting the two objects 66.67% of the association time, while the remaining 33.33% of the time, it is detecting

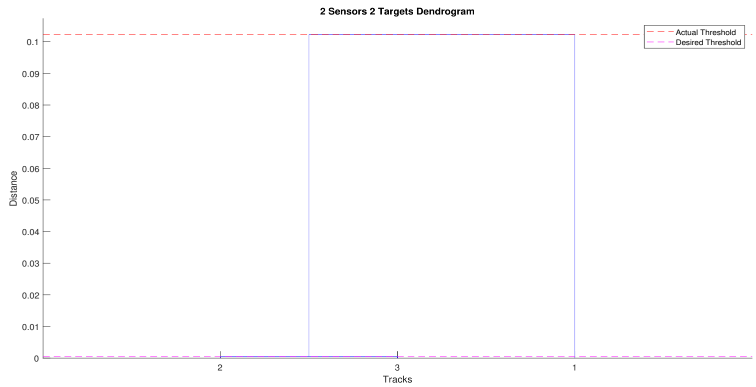



Figure 4.7: Dendrogram for Scenario 2

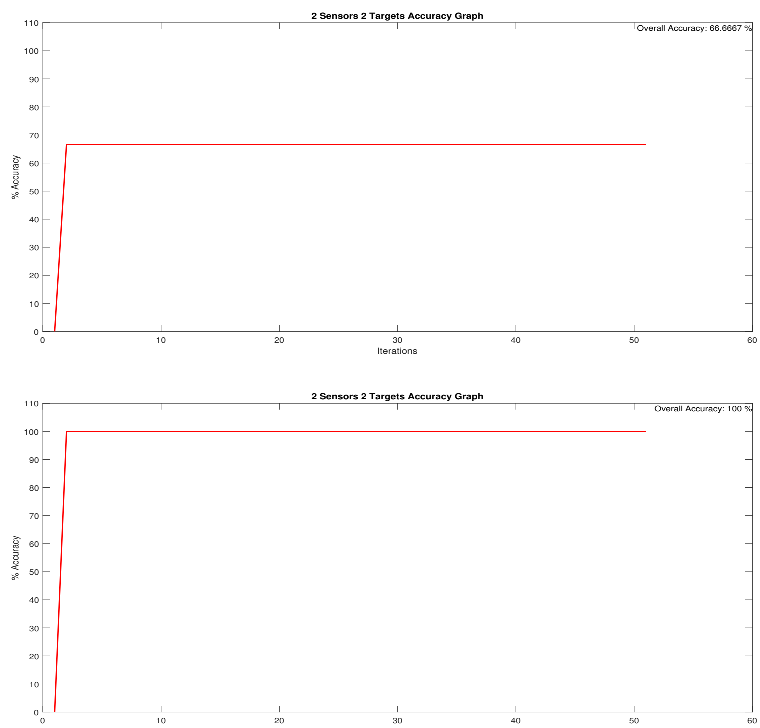



Figure 4.8: Accuracy of 2 sensors tracking 2 objects

3 or 1 object. The convergence of the accuracy to 66.67% is shown against association time in figure 4.8. However, by changing the threshold, we can get

the accuracy of 100%, which is also shown in figure 4.8.

## 4.3 Two Sensors Four Targets

In this scenario, we have two moving sensors tracking four moving objects. The first target is observed by the only sensor 1; both sensors observe the second and third targets, while the fourth target is measured by only sensor 2 throughout its journey. The path followed by the two sensors and the four objects is shown in Figure 4.9.

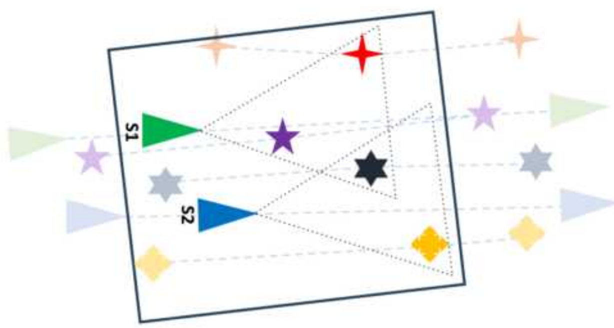



Figure 4.9: Scenario 3: Two Sensors Four Targets

In this case, we have addressed a limitation: one sensor can only observe two targets in one scenario. So, it will ignore any third target detected in its trajectory and continue observing the two targets it has been observing. In this particular scenario, $S_1$ observes three objects while $S_2$ observes two objects throughout its journey. So, $S_1$ will have two local tracks for each object, as it can track a maximum of two objects so that it will ignore the third one. And $S_2$ will also have two local tracks. When these sensors share their tracks, $S_1$ will have two local tracks and two shared tracks, and likewise, $S_2$ will also have two local and two shared tracks. Without association, it is possible that the sensors $S_1$ or $S_2$ falsely report the numbers of objects based on the given tracks. The association algorithm should group these tracks into two clusters, one of which will have two associated tracks. If the association algorithm shows more or less than two different clusters at any time in the complete trajectory, the algorithm will be in error.

To observe the accuracy in this case, we developed a scenario similar to Figure 4.9 in the validation optimization system (VOS). The visualization of the VOS-generated scenario is shown in Figure 4.10

The packets associated with this scenario will contain local and shared tracks of the two sensors. These packets will extract the track information and implement the HAC algorithm in Section 3.3. The dendrogram for this scenario will involve four clusters, with one containing the tracks of $S1$ and $S2$ and the other

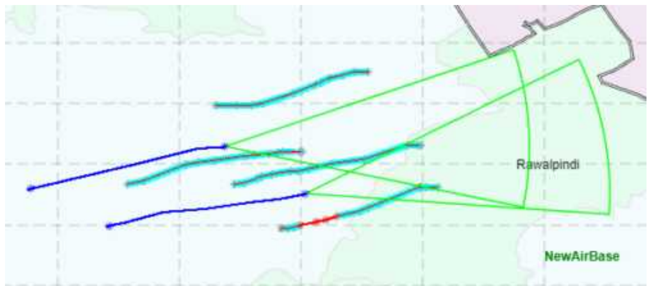



Figure 4.10: VOS scenario 3

three containing single tracks of $S_1$ if the association is accurate. This is shown in Figure 4.11.

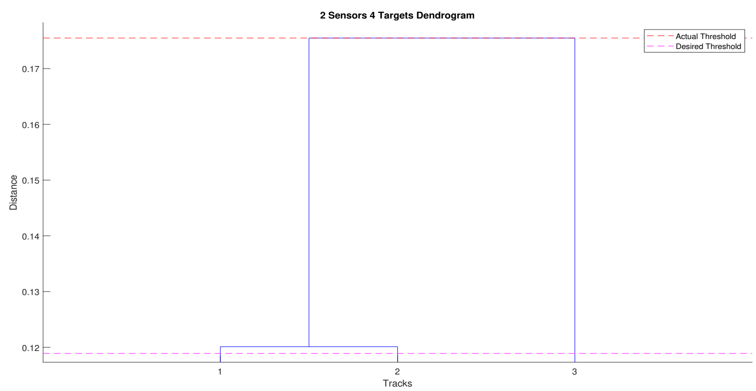



Figure 4.11: Dendrogram for Scenario 3

The dendrogram shows the shared and local tracks of the sensor on the x-axis, and the blue line shows that they are clustered in one group. Since only a cluster is formed in this scenario, only one track of sensor 1 and sensor 2 is associated, while the other tracks from booth sensor 1 and sensor 2 are not.

On implementation of the HAC algorithm and comparison of the association results with ground truth, the overall accuracy of this scenario with a predefined threshold is only 20%, which shows that the proposed association algorithm is detecting four objects only 20% of the association time, while the remaining 80% of the time, it is detecting more or less than 4 objects. The convergence of the accuracy to 20% is shown against association time in figure 4.12. However, by changing the threshold, we can get the optimal accuracy of 100%, which is also shown in figure 4.12.

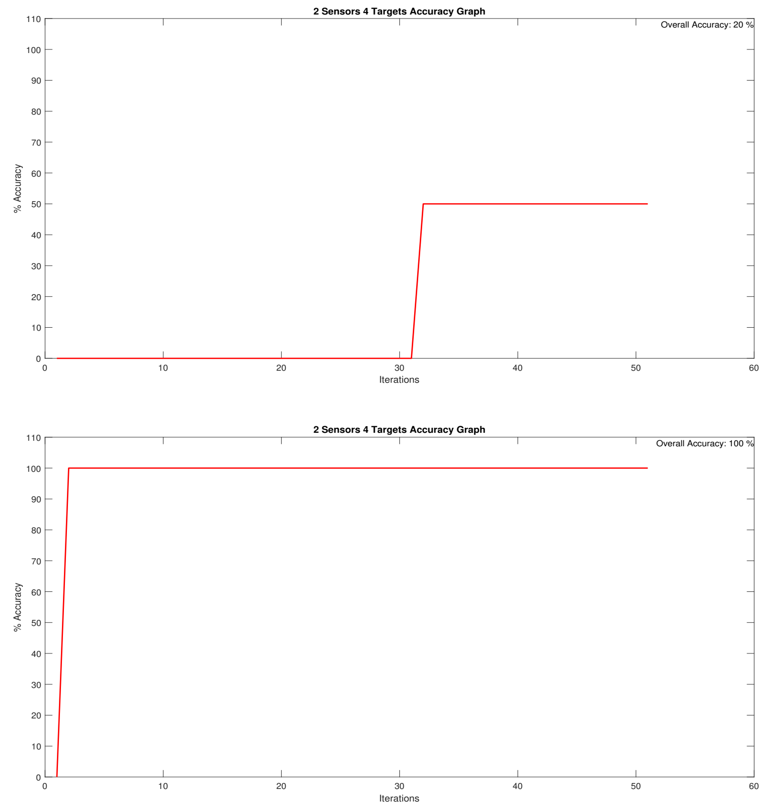



Figure 4.12: Accuracy of 2 sensors tracking 4 objects

## 4.4 Three Sensors Four Targets

In this scenario, we have three moving sensors and four moving targets. The first target is observed by sensors 1 and 2; the second target is measured by only sensor 2; the third target is measured by sensors 2 and 3; and the fourth target is only measured by sensor 3 throughout its journey. The path followed by all three sensors (green, blue, and brown triangles) and four objects (red, purple, black, and yellow) is shown in Figure 4.13.

Sensor $S_1$ observes only one object, while the other sensors $S_2$ and $S_3$ observe two objects each. So, $S_1$ will have one local track for that object, while $S_2$ and $S_3$ will have two local tracks each. When these sensors share their tracks, $S_1$ will have one local and four shared tracks, while $S_2$ and $S_3$ will have two local tracks and three shared tracks. With no association, it is possible that the three sensors falsely show more than four different objects. The association algorithm should associate the tracks and cluster them in groups. If the association algorithm shows more than four different objects at any time in the trajectory of the target

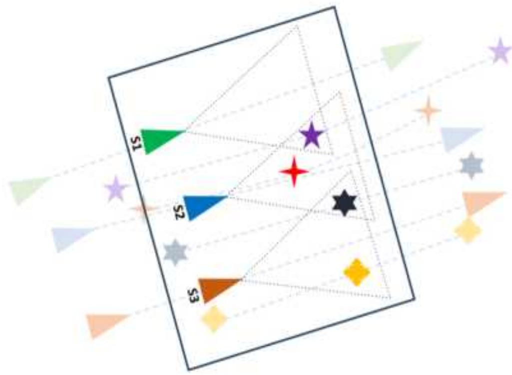



Figure 4.13: Scenario 4: Three Sensors Four Targets

objects, the algorithm will be in error. To observe the accuracy in this case, we developed a scenario similar to Figure 4.13 in the validation optimization system (VOS). The visualization of the VOS-generated scenario is shown in Figure 4.14.

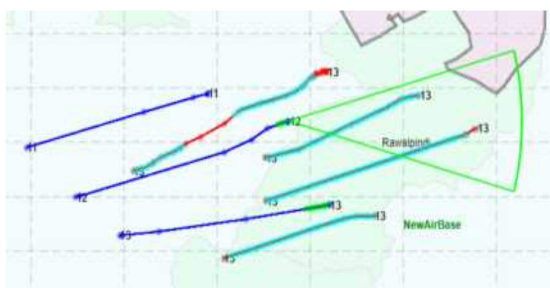



Figure 4.14: VOS Scenario 4

Notice that VOS assigns blue colors to the sensors' paths and red to the targets' paths. The green cones show the range of the sensors, and the highlighted red path of the target in turquoise shows that the target is measured throughout its journey. The packets associated with the scenario will contain the three sensors' local and shared tracks. These packets will extract the track information and implement the HAC algorithm discussed in Section 3.3. The dendrogram for this scenario will involve two clusters, one amongst $S_1$ and $S_2$ and the other amongst $S_2$ and $S_3$ if the association is accurate. This is shown in Figure 4.15.

The dendrogram shows that the shared and local tracks of the sensors are on the x-axis and clustered in a hierarchy. The y-axis shows the distance between the two tracks in meters (m). Since two clusters are formed in this scenario, the

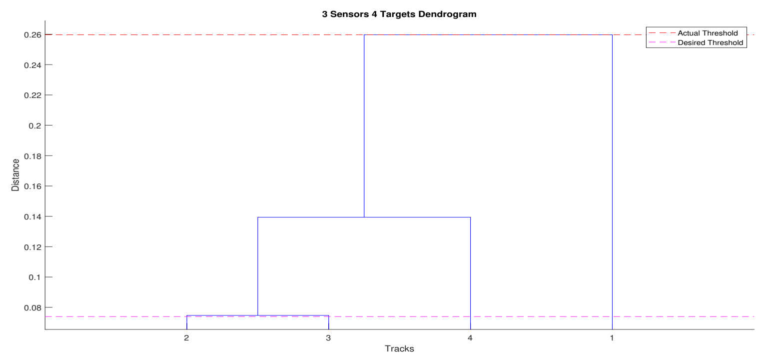
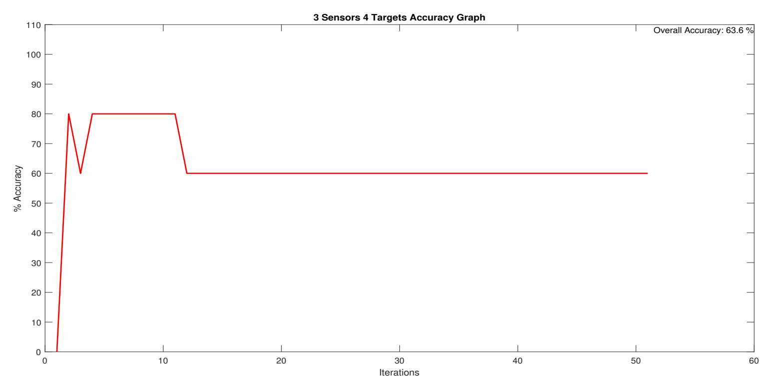



Figure 4.15: Dendrogram for Scenario 4

four tracks, one with sensor 1 and sensor 2, represent the same object, and the other with sensor 2 and sensor 3 represent the same object.



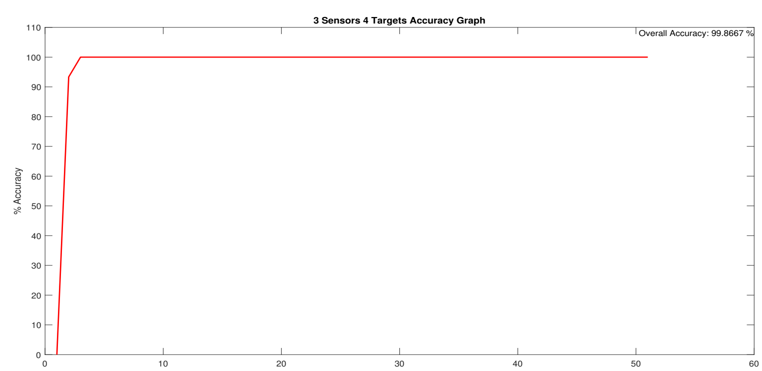




Figure 4.16: Accuracy of 3 sensors tracking 4 objects

On implementation of the HAC algorithm and comparison of the association results with ground truth, the overall accuracy of this scenario with a predefined threshold is 63.6%, which shows that the proposed association algorithm is detecting four objects 63.6% of the association time, while the remaining 36.4% of the time, it is detecting more or less than 4 objects. The convergence of the accuracy to 63.6% is shown against association time in figure 4.16. However, by changing the threshold, we can get the optimal accuracy of 99.8667%, also shown in figure 4.16.

## 4.5 Three Sensors Three Targets

In this scenario, we have three moving sensors and three moving targets. The first target is observed by only sensor 1, the second target is observed by sensors 2 and 3, and the third target is observed by only one sensor throughout its journey. The path followed by the three sensors (green, blue, and brown triangles) and the objects (red, purple, and yellow) is shown in Figure 4.17.

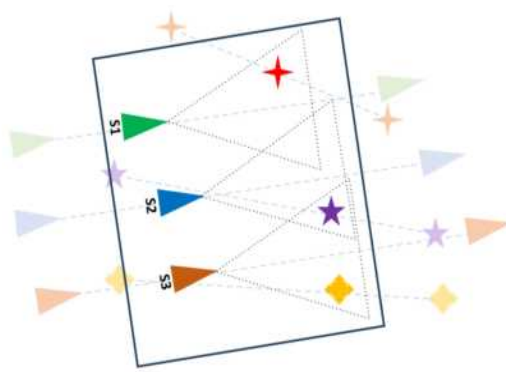



Figure 4.17: Scenario 5: Three Sensors Three Targets

Sensors $S_1$ and $S_2$ observe only one object, while $S_3$ observes two objects. So, $S_1$ and $S_1$ will have one local track each for the objects they observe, while $S_3$ will have two local tracks. When these sensors share their tracks, $S_1$ and $S_2$ will have one local and three shared tracks each, while $S_3$ will have two local tracks and two shared tracks. With no association, it is possible that the three sensors falsely show more than three different objects. The association algorithm should associate the tracks and cluster them in associated groups. If the association algorithm shows more than three different objects at any time in the trajectory of the target objects, the algorithm will be in error. To observe the accuracy in this case, we developed a scenario similar to Figure 4.17 in the validation optimization system (VOS). The visualization of the VOS-generated scenario is shown in Figure 4.18.

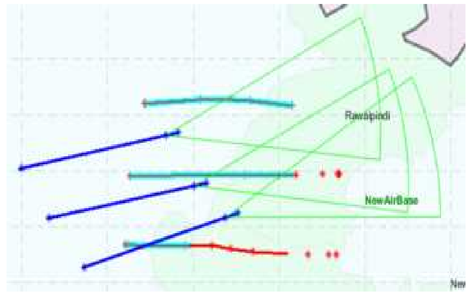



Figure 4.18: VOS scenario 5

Notice that VOS assigns blue colors to the sensors' path and red to the target path. The green cones show the range of the sensors, and the highlighted red path of the target in turquoise shows that the target is measured throughout its journey. The packets associated with the scenario will contain the three sensors' local and shared tracks. These packets will extract the track information and implement the HAC algorithm discussed in Section 3.3. If the association is accurate, the dendrogram for this scenario will involve one cluster of $S_2$ and $S_3$. This is shown in Figure 4.19.

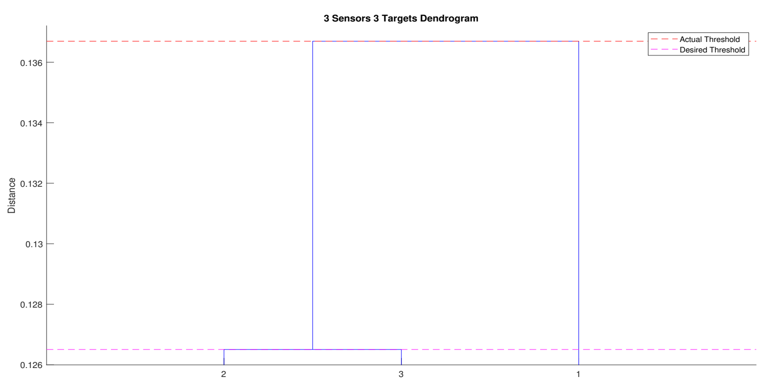



Figure 4.19: Dendrogram for Scenario 5

The dendrogram shows that the shared and local tracks of the sensor are on the x-axis and clustered in one group. The y-axis shows the distance between the two tracks in meters (m). Since only one cluster is formed in this scenario, the two tracks with sensors 2 and 3 represent the same object.

On implementation of the HAC algorithm and comparison of the association results with ground truth, the overall accuracy of this scenario with a predefined threshold is 68%, which shows that the proposed association algorithm is detecting four objects 68% of the association time, while the remaining 32% of

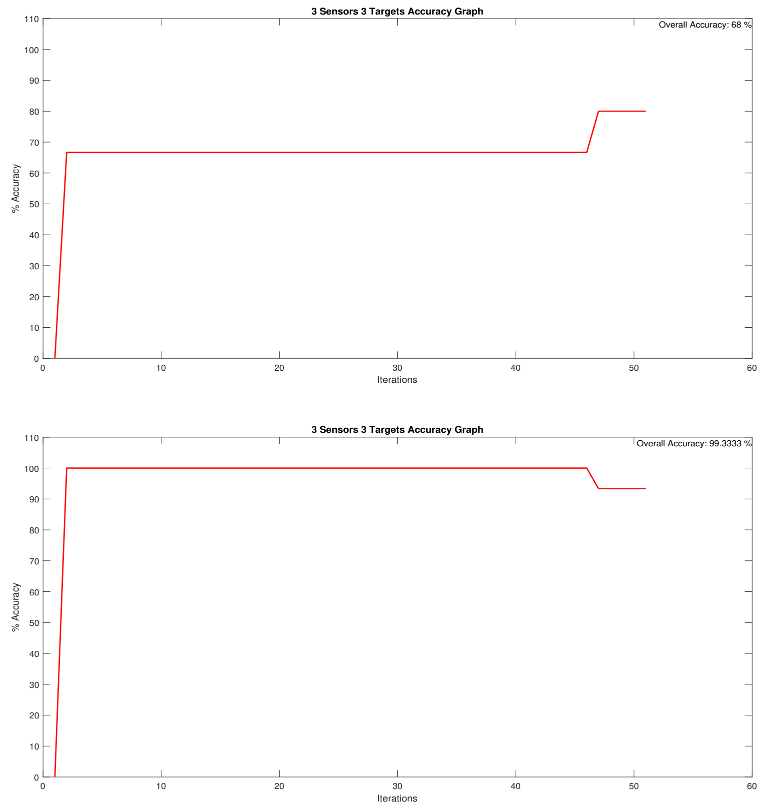



Figure 4.20: Accuracy of 3 sensors tracking 3 objects

the time, it is detecting more or less than 4 objects. The convergence of the accuracy to 68% is shown against association time in figure 4.20. However, by changing the threshold, we can get the optimal accuracy of 99.33%, which is also shown in figure 4.20.

## 4.6 Accuracy Computations in HAC Algorithm

The proposed HAC algorithm has been tested on 5 different scenarios with varying numbers of sensors and targets, and it has demonstrated high accuracy in simple scenarios, but the accuracy is very low in scenarios with increased noise, number of sensors and targets, and complexity. However, this accuracy of complex scenarios can be improved with the tuning of the threshold, and we have tested it on different threshold settings to find the best accuracy and make it suitable for various practical applications in dynamic and multi-object tracking environments. I achieved 100% accuracy in the simplest scenario without threshold adjustment, and this calculation is validated through the ground truth data, showing the accuracy computation is working fine. The accuracy drops to 66.67% in Scenario 2, 20% in Scenario 3, 99.8667% in Scenario 4, and 68% in Scenario 5. On comparison of the clustering data and the ground truth data, it is observed that the variation in accuracy is exactly as per expectations. The issue is related to threshold value as it is giving us wrong clusters, and therefore, on comparison with ground truth, we are getting accuracy drops. On adjusting the threshold value, it is observed that in each of the 4 scenarios, we can improve the accuracy to almost 100%. This means that the accuracy computed by comparison with ground truth is working fine.

# Chapter 5

# Conclusion and Future Work

## 5.1 Conclusion

This thesis explores the challenges of multi-sensor and multi-object track-to-track association problems, focusing on biases. The Hierarchical Clustering Algorithm (HAC) is used as an unsupervised machine learning approach to understand these problems, especially in environments with multiple sensors and objects. The study developed various scenarios to test the HAC algorithm's capabilities, ranging from simple to complex situations. The Validation and Optimization System (VOS) was used to generate these scenarios, providing a controlled yet diverse testing environment. The HAC algorithm demonstrated very diverse accuracies across tested scenarios. However, proper threshold implementation for each scenario demonstrates impressive efficacy in handling these complexities. As the scenarios increased in complexity, which means when the number of sensors and objects increased, the algorithm faced challenges, resulting in varying degrees of accuracy. The study also found that noise levels in tracked objects significantly impacted the algorithm's performance, emphasizing the importance of considering environmental factors and sensor characteristics in the algorithm's deployment. Overall, the thesis successfully navigates the complex domain of multi-sensor and multi-object tracking, providing a thorough and insightful analysis of the HAC algorithm's performance.

## 5.2 Future Work

The thesis explores the future of the Hierarchical Clustering Algorithm (HAC) to improve its accuracy and threshold settings. It suggests areas for further research, including noise management, dynamic threshold optimization, complex scenario analysis, computational efficiency, integration with other machine learning techniques, real-world application and testing, user interface for parameter management, and cross-disciplinary applications. The thesis emphasizes the importance of noise in tracked objects and suggests that future work could focus on improving the algorithm's robustness to noise through advanced noise-filtering techniques or integrating noise-resistant models. Dynamic threshold optimization could involve creating algorithms that automatically adjust thresholds based on real-time data and environmental factors. Complex scenario analysis could involve creating scenarios with a higher number of sensors and objects, varied sensor types, and dynamic object movements. Computational efficiency could be optimized through algorithmic refinements, parallel processing, or efficient data structures. Real-world application and testing could provide valuable feedback on the algorithm's performance and areas for improvement. A user-friendly interface for parameter management could make the algorithm more accessible to users without extensive technical expertise.

# Bibliography


[1] R. T. H. Collis. “Lidar” *Appl. Opt. 9, 1782-1788*, 1970.

[2] Z. Hu, H. Leung, and M. Blanchettee. “Statistical performance analysis of track initiation techniques.” *IEEE Transactions on Signal Processing, vol. 45, no. 2, pp. 445–456*, 1997.

[3] Benameur, Kaouthar. ”Radar ESM Track to Track Association.” *58*, 2001.

[4] C. Blanc, L. Trassoudaine, Y. Le Guilloux, and R. Moreira. “Track to track fusion method applied to road obstacle detection.” *in International Conference on Information Fusion*, 2004.

[5] P. Konstantinova, M. Nikolov, and T. Semerdjiev. “A study of clustering applied to multiple target tracking algorithm.” *in Proceedings of 5th International Conference on Computer Systems and Technologies (CompSysTech’04), pp. 22–26*, 2004.

[6] Wandinger, U. ”Introduction to Lidar.” *In: Weitkamp, C. (eds) Lidar. Springer Series in Optical Sciences, vol 102. Springer, New York, NY*, 2005.

[7] Cong, R., Wang, X.-K, Li, J.-J, Yang, N.-H. ”Plot-track association algorithm based on hierarchical and density clustering analysis.” *17. 841-843.*, 2005.

[8] M. Lei and C. Han. ”Sequential nonlinear tracking using UKF and raw range-rate measurements.” *in IEEE Transactions on Aerospace and Electronic Systems, vol. 43, no. 1, pp. 239-250*, January 2007.

[9] Ashraf M. Aziz. ”Fuzzy track-to-track association and track fusion approach in distributed multisensor-multitarget multiple-attribute environment.” *Signal Process. 87, 1474–1492*, June, 2007.

[10] N. Floudas, P. Lytrivis, A. Polychronopoulos, and A. Amditis. “On the track-to-track association problem in road environments,” *in 2007 10th International Conference on Information Fusion. IEEE, pp. 1–8*, 2007.

[11] L. M. Kaplan, Y. Bar-Shalom, and W. D. Blair. “Assignment costs for multiple sensor track-to-track association.” *IEEE Transactions on Aerospace and Electronic Systems, vol. 44, no. 2, pp. 655–677*, 2008.

[12] Ryde, Julian, and Nick Hillier. ”Performance of Laser and Radar Ranging Devices in Adverse Environmental Conditions.” *Journal of Field Robotics, vol. 26, no. 9, pp. 712–727.*, 2009

[13] D. J. Papageorgiou and M. Holender. “Track-to-track association and ambiguity management in the presence of sensor bias.” *in 12th International Conference on Information Fusion. IEEE, pp. 2012–2019*, 2009.

[14] Richards, Mark A. ”Principles of Modern Radar.” *Volume 1: Basic Principles. Scitech Publishing Inc,*, 2010.

[15] J. Wang, A. von Trojan, and S. Lourey. “Active sonar target tracking for anti-submarine warfare applications.” *in OCEANS’10 IEEE SYDNEY. IEEE, pp. 1–7*, 2010.

[16] D. Göhring, M. Wang, M. Schnürmacher and T. Ganjineh. ”Radar/Lidar sensor fusion for car-following on highways.” *The 5th International Conference on Automation, Robotics and Applications, Wellington, New Zealand, pp. 407-412*, 2011.

[17] R. Mahler. “Measurement-to-track association for nontraditional measurements.” *in 14th International Conference on Information Fusion. IEEE, pp. 1–8*, 2011.

[18] R. Caballero-Aguila, A. Hermoso-Carazo, J. Linares-Perez, ”Unscented Filtering Algorithm for Discrete-Time Systems with Uncertain Observations and State-Dependent Noise.” *Numerical Analysis - Theory and Application, InTech*, 2011.

[19] A. Houenou, P. Bonnifait, V. Cherfaoui and J. -F. Boissou. ”A track-to-track association method for automotive perception systems.” *IEEE Intelligent Vehicles Symposium, Madrid, Spain, 2012, pp. 704-710*, 2012.

[20] Hongyan Zhu, Suying Han. ”Track-to-Track Association Based on Structural Similarity in the Presence of Sensor Biases.” *Journal of Applied Mathematics, vol., Article ID 294657, 8 pages*, 2014.

[21] L. Xu, S. Jin, and G. Yin. “A track association algorithm based on leader-follower on-line clustering in dense target environments.” *Radioengineering, vol. 23, no. 1, p. 259*, 2014.

[22] Maini, Raghav and Guven, Ugur. ”Trajectory Analysis of UPESSAT - A CubeSat Project.” *10.13140/RG.2.1.2156.6962.*, 2014.

[23] B. Duraisamy, T. Schwarz and C. Wöhler. "On track-to-track data association for automotive sensor fusion." *18th International Conference on Information Fusion (Fusion), Washington, DC, USA, pp. 1213-1222*, 2015.

[24] B. Duraisamy and T. Schwarz. "Combi-Tor: Track-to-Track Association Framework for Automotive Sensor Fusion." *IEEE 18th International Conference on Intelligent Transportation Systems, Gran Canaria, Spain, pp. 1616-1623*, 2015.

[25] Mehmood, Rashid, and Bhatti, Umar. "Coarse determination of Attitude using a single GPS receiver." *10.13140/RG.2.2.15665.58721*, 2016.

[26] Wang, Hao, Liu, Tangxing, Bu, Qing, Yang, Bo. "An algorithm based on hierarchical clustering for multi-target tracking of multi-sensor data fusion." *5106-5111. 10.1109/ ChiCC.2016.7554147*, 2016.

[27] J. Steinbaeck, C. Steger, G. Holweg and N. Druml. "Next generation radar sensors in automotive sensor fusion systems." *Sensor Data Fusion: Trends, Solutions, Applications (SDF), Bonn, Germany, 2017, pp. 1-6*, 2017.

[28] Dong, Pinliang, and Qi Chen. "LiDAR Remote Sensing and Applications." *doi:10.4324/9781351233354.*, 2017.

[29] Khalaf, W., Chouaib, I. and Wainakh, M. "Novel adaptive UKF for tightly-coupled INS/GPS integration with experimental validation on an UAV." *Gyroscopy Navig. 8, 259–269*, 2017.

[30] Y. Liu, L. Yao, W. Xiong, and Z. Zhou. "Gf-4 satellite and automatic identification system data fusion for ship tracking," *IEEE Geoscience and Remote Sensing Letters, vol. 16, no. 2, pp. 281–285*, 2018.

[31] S. Sun, C. Xu, L. Oi, J. Li, Y. Liu and K. Dong. "Radar/ESM Anti-Bias Track Association Algorithm Based on Hierarchical Clustering in Formation." *21st International Conference on Information Fusion (FUSION), Cambridge, UK, pp. 2423-2429*, 2018.

[32] Melo, Suzanne. "Interferometric Radar System including a Photonics-based Architecture for Remote Sensing Applications." 2018.

[33] Wang, Y. Zeng, S. Wei, Z. Huang, W. Wang and Z. Wei. "Track-to-Track Association Algorithm Based on Adaptive Clustering Threshold." *IEEE International Conference on Signal, Information and Data Processing (ICSIDP), Chongqing, China, pp. 1-4*, 2019.

[34] Yoon, Kwangjin, Du Yong Kim, Young-Chul Yoon, and Moongu Jeon. "Data Association for Multi-Object Tracking via Deep Neural Networks." *Sensors 19, no. 3: 559*, 2019.

[35] M. Mielle, M. Magnusson and A. J. Lilienthal. "A comparative analysis of radar and lidar sensing for localization and mapping." *European Conference on Mobile Robots (ECMR), Prague, Czech Republic, pp. 1-6*, 2019.

[36] Beland, Martin, et al. "On Promoting the Use of Lidar Systems in Forest Ecosystem Research." *Forest Ecology and Management, vol. 450, p. 117484*, 2019.

[37] Wang, Xinxin, Zeshui Xu, and Xunjie Gou. "Consensus-Based Track Association with Multistatic Sensors under a Nested Probabilistic-Numerical Linguistic Environment." *Sensors 19, no. 6: 1381*, 2019.

[38] Modupe, Sanyaolu. "Rain Fade Analysis at C, Ka and Ku Bands in Nigeria." *10.7176/JEES/9-3-16*, 2020.

[39] Zou K, Zhu H, De Freitas A, Li Y, Najafabadi HE. "Track-to-Track Association for Intelligent Vehicles by Preserving Local Track Geometry." *Sensors (Basel).20(5):1412. PMID: 32143498; PMCID: PMC7085508*, Mar 4, 2020.

[40] Patrick Emami, Panos M. Pardalos, Lily Elefteriadou, and Sanjay Ranka. "Machine Learning Methods for Data Association in Multi-Object Tracking." *ACM Comput. Surv. 53, 4, Article 69 (July 2021), 34 pages*, 2020.

[41] L. Pirovano, G. Principe, and R. Armellin. "Data association and uncertainty pruning for tracks determined on short arcs," *Celestial Mechanics and Dynamical Astronomy, vol. 132, no. 1, pp. 1–23*, 2020.

[42] Li. Song, Cheng. Yongmei, Gou. Bin, Gao. Shibo. "Sequential Unbiased Converted Measurement Nonlinear Filter with Range Rate in ECEF Coordinates." *IET Radar, Sonar and Navigation. 14*, 2020.

[43] Lee, S., Lee, D., Choi, P., and Park, D. "Accuracy–Power Controllable LiDAR Sensor System with 3D Object Recognition for Autonomous Vehicle." *Sensors (Basel, Switzerland), 20.*, 2020.

[44] Bai. Dongxin, Tang. Jingtian, Lu. Guangyin, Zhu. Ziqiang, Liu. Taoying, Fang. Ji. "The design and application of landslide monitoring and early warning system based on microservice architecture." *Geomatics, Natural Hazards and Risk. 11. 928-948.*, 2020.

[45] Qiao, Jun, and Yaohong Zhao. "3D imaging model of airborne LiDAR system." *Seventh Symposium on Novel Photoelectronic Detection Technology and Applications. Vol. 11763. SPIE*, 2021.

[46] J. Wang, Y. Zeng, S. Wei, Z. Wei, Q. Wu and Y. Savaria. "Multi-Sensor Track-to-Track Association and Spatial Registration Algorithm Under Incomplete Measurements." *in IEEE Transactions on Signal Processing, vol. 69, pp. 3337-3350*, 2021.

[47] Ma, M., Wang, D., Zhang, T. and Sun, H. "Track-to-track association algorithm for passive multisensor system based on trajectory parameter." *IET Radar Sonar Navig, 15: 348-358*, 2021.

[48] An, Q.; Cai, Y.; Zhu, J.; Wang, S.; Han, F. "Multi-Target Tracking Algorithm Combined with High-Precision Map." *Sensors 22, 9371.*, 2022.

[49] Skolnik, Merrill I. "Radar." *Encyclopedia Britannica*, 8 Sep, 2023.

[50] Pastor, A., Sanjurjo-Rivo, M. Escobar, D. "Track-to-track association methodology for operational surveillance scenarios with radar observations." *CEAS Space J 15, 535–551*, 2023.

[51] Adiyasa Nurfalah, Suhono Harso Supangkat, Eueung Mulyana. "Effective and near real-time track-to-track association for large sensor data in Maritime Tactical Data System" *ICT Express, ISSN 2405-9595*, 2023.

[52] Xu, Y., Chen, Y., Zhang, Y., Zhu, Q., He, Y., Sheng, H.. "Bilateral association tracking with parzen window density estimation." *IET Image Processing 17, 274–290*, 2023.