TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Robotik, Künstliche Intelligenz und Echtzeitsysteme

Cooperative Localization using Factor Graphs for Autonomous Vehicles

Dhiraj Kumar Gulati

Vollständiger Abdruck der von der Fakultät der Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Prof. Dr. Uwe Baumgarten
Prüfer der Dissertation:	1. Prof. Dr.-Ing. habil. Alois Knoll
	2. Prof. Thomas C. Henderson

Die Dissertation wurde am 28.06.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 30.10.2018 angenommen.

Abstract

Precise localization is one of the critical requirements for the successful realisation of highly assisted or autonomous vehicles. The diminishing cost of hardware and increasing precision, has resulted in a proliferation of sensors in the environment (also known as infrastructure sensors). This is revolutionising the highways into *smart highways* equipped with high-precision sensors. Various studies and experiments from governments and vehicle manufacturers are already under way to evaluate and implement such concepts. Additionally, the communication technology has matured to provide real-time data exchange between senders and receivers.

With all these advancements, cooperative localization using internal ego-vehicle sensors and external high-precision sensors presents itself as a feasible and effective solution for localizing the ego-vehicle and its neighbouring vehicles. This can lower the cost of ownership of such autonomous vehicles. However, there are substantial challenges to fully realize the effective use of sensor infrastructure located outside the ego-vehicle for jointly estimating the state of a vehicle in cooperative localization. They include data association, i.e. measurement-to-track association, coordinate transformation, bandwidth limitations and scalability.

In this thesis, I propose a novel cooperative localization system which not only utilises measurements from external sensors to improve the position estimate of the ego and neighbouring vehicles, but also addresses the above mentioned challenges. Any participating vehicle is assumed to be equipped with an odometer and GPS and a real-time communication mechanism. An external precise radar system is also assumed to be available which can view the targets. The proposed system uses the concept of factor graphs which intuitively lend themselves to express the problem of cooperative localization.

To avoid the overhead of data association and simultaneously address other mentioned challenges, the measurements from an external radar are formulated as a novel *topology factor* between the targets in the field of view of the radar. This topology factor is constructed using the sum of Euclidean distances between the vehicles. I propose another novel factor based on symmetric conversion of radar measurements which not only avoids the step of data association but is also efficient in GPS devoid areas like tunnels and urban canyons. Additionally, I propose a robust topology factor to address the challenges of clutter. The proposed ideas are extended from vehicle-infrastructure cooperative localization to vehicle-vehicle cooperative localization, thus providing a complete system for cooperative localization.

The formulated graph using the factors constructed from internal ego-vehicle sensors and proposed novel factors from external sensors (infrastructure and/or other vehicles) is solved using a non-linear least square optimizer to calculate the state estimate of the vehicles. The simulation and experiments (based on real measurements from an infrastructure radar) show the feasibility of the proposed system.

Zusammenfassung

Präzise Lokalisierung ist eine der wichtigsten Voraussetzungen für den Erfolg hoch-automatisierter oder voll autonom fahrender Fahrzeuge. Die immer sinkend Hardwarekosten und die zunehmende Präzision haben zu einer Proliferation von Sensoren in der Umwelt (auch als Infrastruktursensoren bezeichnet) geführt. Dadurch werden Autobahnen zu *intelligenten Autobahnen (Smart Highways)* die mit hochpräziser Sensorik ausgestattet sind. Verschiedene Studien und Experimente von Regierungen und Autoherstellern sind bereits im Gange, um derartige Konzepte zu evaluieren und umzusetzen. Außerdem hat sich die Kommunikationstechnologie weiter entwickelt und ermöglicht den Datenaustausch zwischen Sendern und Empfängern in Echtzeit.

Mit all diesen Fortschritten bietet sich eine kooperative Lokalisierung mit internen handelsüblichen Ego-Sensoren und externen Hochpräzisions-Sensoren als praktikable und effektive Lösung zur Lokalisierung des Ego-Fahrzeugs und seiner Nachbarfahrzeuge an. Dies kann die Betriebskosten solcher voll-autonom fahrender Autos senken. Es gibt jedoch einige Herausforderungen, um die effektive Nutzung einer Sensorinfrastruktur, die sich außerhalb des Ego-Fahrzeugs befindet, vollständig zu realisieren, und den gemeinsam den Zustand von Fahrzeugen in kooperativer Lokalisierung zu schätzen. Sie umfassen Datenassoziation, d.h. measurement-to-track-Zuordnung, Koordinatentransformation, Bandbreitenbeschränkungen und Skalierbarkeit.

In dieser Arbeit schlage ich ein neuartiges kooperatives Lokalisierungssystem vor, das nicht nur Messungen von externen Sensoren nutzen kann, um die Positionsschätzung des Ego- und benachbarten Fahrzeugs zu verbessern, sondern auch die oben genannten Herausforderungen bewältigt. Es wird davon ausgegangen, dass jedes teilnehmende Fahrzeug mit einem handelsüblichen Odometer und GPS-Empfänger sowie mit einem Echtzeit Kommunikationsmechanismus ausgestattet ist. Das System verwendet das Konzept von der Faktorgraphen, die intuitiv dazu geeignet sind, das Problem der kooperativen Lokalisierung auszudrücken.

Um den Mehraufwand der Datenassoziation zu vermeiden und gleichzeitig die anderen erwähnten Herausforderungen zu adressieren, werden die Messungen von einem externen Radar als ein neuer *Topology Factor* zwischen den Zielen in dem Sichtfeld des Radars formuliert. Dieser Topology Factor wird unter Verwendung der Summe der euklidischen Abstände zwischen den Fahrzeugen erzeugt. Ich schlage einen weiteren neuartigen Faktor vor, der auf der symmetrischen Umwandlung von Radarmessungen beruht und nicht nur den Schritt der Datenassoziation vermeidet, sondern auch in GPS-freien Gebieten wie Tunneln und Häuserschluchten effizient ist. Darüber hinaus schlage ich einen robusten Topology Factor vor, um die Herausforderungen von Störddaten zu bewältigen. Die vorgeschlagenen Ideen sind von der Fahrzeug-

Infrastruktur kooperative Lokalisierung zur Fahrzeug-Fahrzeug kooperative Lokalisierung erweitert und bieten somit einen vollständigen System für kooperative Lokalisierung.

Der Graph, der aus den Faktoren besteht, die aus den Messungen der internen Ego Fahrzeugsensoren erzeugt werden, sowie aus den vorgestellten neuartigen Faktoren aus den Messungen externer Sensoren (Infrastruktur und/oder andere Fahrzeuge) wird mittels eines Algorithmus zur Lösung von nichtlinearen kleinsten Quadrate Problemen berechnet, um die Zustandsvariablen der Fahrzeuge zu schätzen. Die Simulation und Experimente (basierend auf realen Messungen eines Infrastruktur-Radars) zeigen die Funktionstüchtigkeit des vorgestellten Systems.

This thesis is dedicated to my parents.

Acknowledgements

First and foremost, I would like to thank Prof. Dr. Alois Knoll for giving me the opportunity to work under his guidance on this fascinating and challenging topic, as well as for his encouragement and support. I would also like to thank Prof. Dr. Thomas Henderson (University of Utah) for accepting to be my second supervisor and providing feedback for improvements.

I would like to thank Dr. Feihu Zhang for his support in providing the answers for the simplest to the most complex questions and helping me to embark this journey. Thanks to Dr. Daniel Clarke for being my mentor and providing me valuable guidance and support. My colleague, Dr. Vincent Aravantinos developed a keen interest in my work and provided me with the guidance and support till the end of writing of this thesis. A special thanks for many discussions and reviewing my thesis.

I especially thank Daniel Malovetz who not only helped me with project and research work but also in discussing the challenges of life, the universe and everything.

I would also like to thank all my colleagues at fortiss, particularly Uzair Sharif, Annkathrin Kraemmer, Susanne Riemann, Yunis Fanger and Frederik Diehl who supported me during my research endeavours.

Finally, I would like to thank my parents and my friends for the constant support while working on the thesis.

Contents

Chapter 1

Introduction

An entity existing in this world will have a corresponding state, which effectively describes the physical characteristics of the entity. As an example, consider liquid stored in a container. It is characterised by its quantity and its temperature. Therefore, the amount and its temperature are two characteristics of the state of the liquid. There can be other as characteristics, for example amount of salt or sugar present in the liquid, its alkalinity or acidity etc. Similarly, location and speed of a vehicle are the characteristics of the state of the vehicle.

1.1 State and its Estimation

We routinely estimate state of things around us, consciously or unconsciously, to help us make decisions required to perform daily activities. This follows the intelligent agent paradigm of sense, understanding and action. The sensors perceive the environment, the sensed information is understood and based on a rule set or some prior knowledge an action is executed.

For example, we estimate the distance of the next crossing from our current position while guiding somebody to his/her destination. Here somebody had asked for our help (sensing), asking for the direction (understanding) and based on the prior knowledge of the area, we guide the person (execution). Similarly, we estimate the temperature of the water just by slightly touching the surface with our finger and then decide if its of correct temperature to be used. We estimate how much salt or sugar needs to be added to the food by tasting a small amount.

We fuse data acquired using our "biological sensor system" to estimate the state. In the first example if we know the area, we already have a map of the surrounding in our mind. We perform an estimation of the state of ourselves, by the environment perception, in this map and calculate the distance. In the second case, if the liquid in the container is water, we use the internal heat sensor of our body and estimate the state of the water temperature. In the third, we use the taste buds to estimate the state of salt or sugar in the food. Nevertheless, for all the above examples, aided with biological sensors, we can hardly achieve the true value of the state, and even if we do, that generally is a lucky guess.

Therefore, for any intelligent agent, the first step is to correctly sense the state and/or its environment and only then an appropriate decision can be reached.

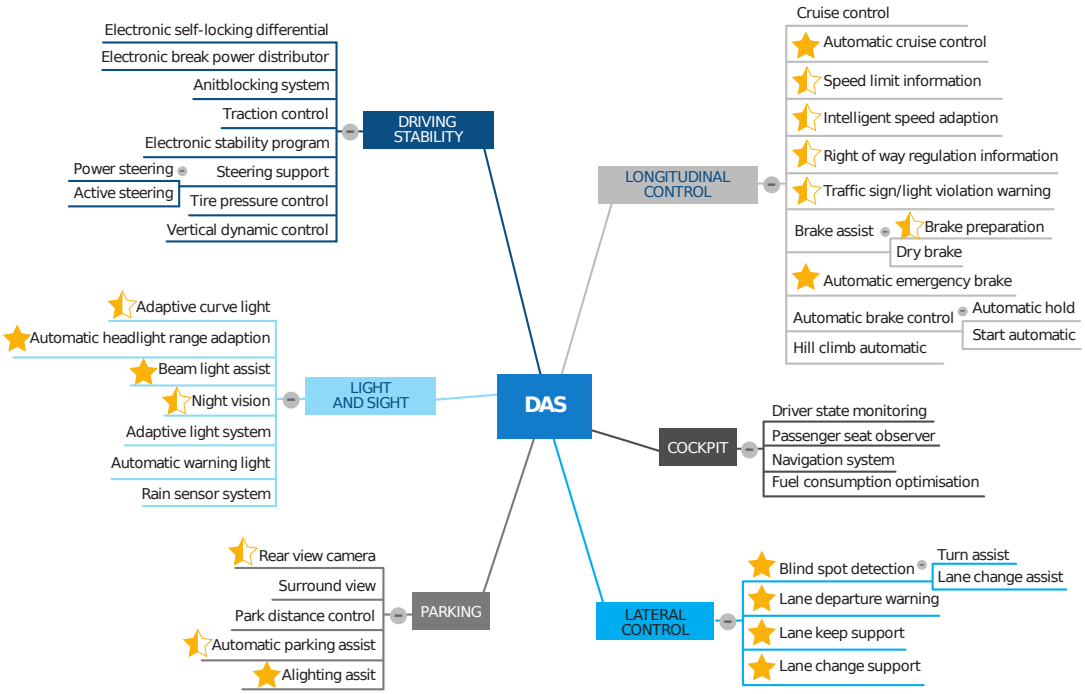


Figure 1.1: Source [2]. ADAS Features available in vehicles. Full stars require intelligence and real time processing capability. Half-coloured stars are rudimentary ADAS cases

1.2 Autonomous Driving

Transportation has been one of the critical factor enabling the development of human civilization, and better transportation system has fuelled the growth of our society. However, with the increasing sophistication in transportation systems, particularly the road transportation, there has also been an increase in the number of casualties. As per WHO 2015 [1] report 1.2 million people die every year because of traffic accidents. This is because, until about a decade ago, the average road user was doing state estimation similarly, as described in previous section, aided with basic physical and/or biological sensor system. However, based on the physical condition (e.g. drunken driver, lack of sleep, distraction because of a cell phone) this state estimate can always be wrong.

Hence, in the last decade, the vehicle manufacturers have increasingly incorporated sophisticated systems for safety for all kinds of road users. These include (1) Driving Assistance Systems (DAS)/Advanced Driving Assistance Systems (ADAS), and (2) Autonomous Driving under certain circumstances. Manufacturers provide vehicles with technologies, where it could also take decisions. For example, if the vehicle detects a pedestrian or a red traffic signal, it could autonomously decide to halt.

Figure 1.1 from [2] gives an overview of various DAS and ADAS functionalities currently available in vehicles. The ADAS functions with the full stars are the ones having required intelligence and real time processing capability. The half-coloured stars represent rudimentary ADAS cases. This is not an exhaustive list. This will vary from one source to another and as even at the time of writing this chapter, newer ADAS functionalities may be thought of, developed and tested.

Autonomous Driving under certain circumstances is a limited capability of a vehicle to drive fully autonomously in simple environments, for example highways. In this kind of scenarios there are, the-

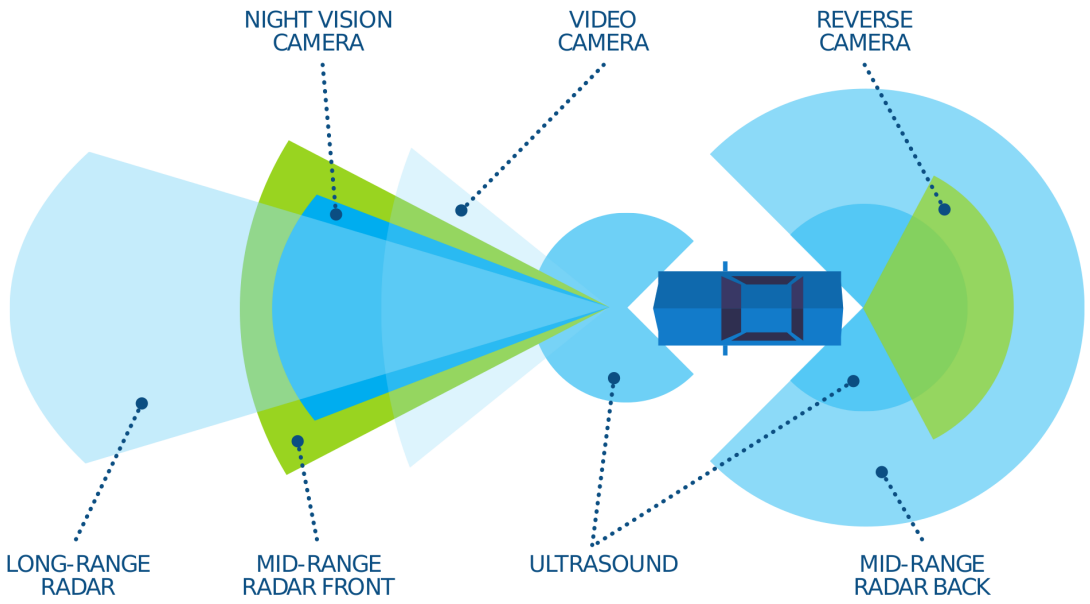


Figure 1.2: Source [2]. Possible sensors for ADAS. It does not mention sensors like odometer, GPS and LiDAR

oretically, no pedestrians crossing the road and no complex manoeuvre at complicated intersections. Additionally, the direction of traffic flow is clearly marked. The system orchestrates all the ADAS functionalities as shown in 1.1. The driver need not drive but should be attentive to take over any time the system sends a signal. This functionality requires a substantial increase in functional safety levels and system redundancies. Currently there is a race to launch a safe, legal and socially acceptable autonomous vehicles which completely removes the human from the loop and hence the uncertain biological sensor system.

In addition to increasing of the safety of road users, autonomous transportation systems can optimize the fuel consumption, thus reduce the environmental pollution. Passengers in autonomous vehicles can relax and avoid stress induced by driving. This is achievable if the autonomous transportation systems is able to make sound decisions, which is dependent on correct state estimates being available to the system.

1.2.1 Autonomous Driving and Sensors

To achieve ADAS functionalities, the vehicle relies on various sensors. These sensors are together used for the final state estimation of the vehicle and its environment, often using data fusion to collate, register, associate and integrate information from various sources.

A possible sensor architecture for such a vehicle is shown in Figure 1.2. Similar to ADAS functionalities, this is one of the possible reference architecture referred in [2]. Figure 1.2 does not include sensors like odometer, Global Positioning System (GPS), LiDAR etc. Various auto manufacturers have built

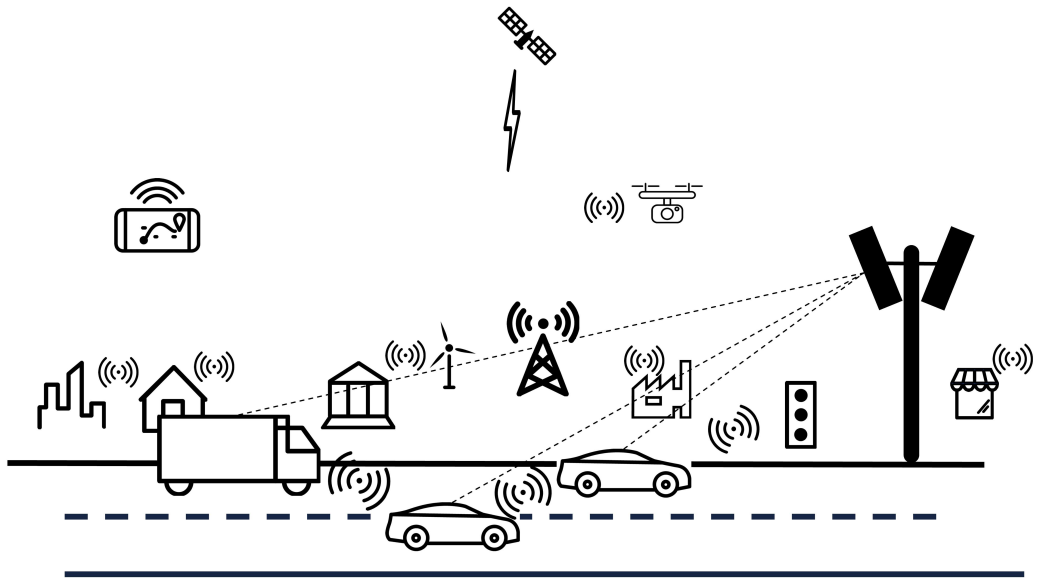


Figure 1.3: Connected World

their own architectures based on their experience with ADAS functionalities.

1.2.2 Localization and Connected World

Various ADAS functionalities for example collision avoidance, location based services like speed limit information and right of way regulation information, and navigation require the position/location of the vehicle. Autonomous Driving under certain circumstances (which is approximately a super set of ADAS functionalities) also needs the location information of the vehicle. In fact, as soon as the vehicle is switched on, its location is an important parameter for ADAS or fully automated driving system. It continues to remain an important parameter during vehicle motion for various other systems like path planning and collision avoidance until it reaches its destination.

One of the primary system for localization is global navigation satellite system (GNSS). GPS is one of the widely used sensor of GNSS. However, a standard off-the-shelf GPS sensor has a measurement error from $7 - 10m$ [3]. This is not precise enough to perform safety critical tasks of Autonomous Driving. It is possible to add higher precision sensors in the Autonomous Vehicle, but that will increase the cost of production many folds.

Furthermore, with a rapid increase in technological innovations, the world is moving towards a perpetual connected state. Because of this, a whole domain of Intelligent Transportation System has emerged [4]. Real time communication between various sensor systems is now possible [5]. Technologies like Dedicated Short Range Radio (DSRC) [6] and Ultra Wide Band Radio [7] make it possible to have a Vehicle-to-Vehicle (V2V) real time connection. Vehicle-to-Infrastructure Sensor (V2I) is also made possible using technologies like $5G$ [5]. The combined technology is known as V2X, where "X" can be Vehicle or Infrastructure Sensor. A connected world can be seen in Figure 1.3. With such an infrastructure available, the question arises:

How can the data from external sensors be used to improve state estimation?

In this thesis, I propose a framework to use the external sensors using V2X technologies to improve the

state estimation of self-driving vehicles. This can mitigate the cost of vehicles as high precision sensors can be deployed in the infrastructure, which can be used by all the nearby vehicles. This also supports and utilizes the Smart Infrastructures [4] where various governments and big corporations are pooling in resource for research and development of such activities.

1.3 Challenges and Motivation

For successful fusion of the data available from external sensors (with focus on the radar) for improved state estimation (as seen in Figure 1.3), following are the challenges which I will address in this thesis:

1. **Data Association**: To fuse data from high precision sensors like radar in a multi-target scenario requires measurement-to-target association. This process is called Data Association. This needs additional solutions, which perform the task of Data Association and thus adding to the latency. A solution that can avoid the task of Data Association would be beneficial.

2. **Coordinate Transformations**: For any data to be fused from a previously unseen sensor, system needs to know the sensors physical configuration in terms of its position and orientation. This is required to perform coordinate transformation to some common coordinate framework before being fused. This poses a challenge as Smart Infrastructures [4] will constitute of multiple sensors and knowing and the physical configuration of all the external sensors for an autonomous vehicle is impossible. This needs a solution where the data from the external sensor can be fused without the overhead of knowing the physical configuration of the sensor.

3. **Bandwidth Limitations**: Multiple vehicles who will utilize some set of external sensors using V2I technology will contend for the available bandwidth. Consequently, keeping the allotted bandwidth utilization to minimum is an important factor.

4. **Scalability**: The solution should scale with the increase in the number of participating vehicles.

5. **Clutter** High precision radar sensor can also miss-detect. In these cases, the measurements do not correspond to any real target. This is called clutter. These false measurements can appear as the valid measurements and cause the degradation of the state estimation. The proposed solution should be robust against such outliers.

6. **Adaptability**: Traditionally, data fusion systems for Autonomous Vehicles using internal sensors have been hard-designed during earlier phases of the development. Various configuration parameters for multi-sensor data fusion can be therefore hard-coded in the system. However, fusing data from a previously unseen external sensor requires a complete framework to be adaptable and support plug and play architecture.

Various researchers have provided novel solutions to handle one or more of the above solutions. We visit some of the interesting solutions in Section 2.3. Despite a tremendous amount of work in the domain, the researched solutions fail in some or other challenges and there lacks a common solution which can address all the challenges together. The motivation of this work is to provide a common framework, which can address maximum number of the above-mentioned challenges.

1.4 Contribution

An innovative **adaptable** framework has been proposed in this thesis which addresses maximum number of the highlighted challenges in the previous section. The framework provides a **scalable** solution for vehicle infrastructure cooperative localization with **minimum bandwidth requirement**, while **avoiding the overhead of data association** and **without coordinate transformation**.

The proposed solution fuses data from off-the-shelf automotive grade sensors like odometer and GPS and from external high precision radar sensor using a Graph-based approach. This addresses the challenges of Data Association and Coordinate Transformation. The framework requires minimal bandwidth, transmitting only the measurements. It remains robust against outliers like a clutter measurement as observed by radar.

The same framework fuses the data from neighbouring vehicles using V2V technologies, further improving the state estimates. An additional fusion strategy for scenarios where GPS is not available is also implemented. This addresses the places like tunnels and underground parking garages.

Thus, the research work proposed in this thesis improved the state estimation for autonomous vehicles utilizing a common framework using V2X communication.

1.4.1 Publications

During the course of the research, I have published, presented and discussed the results in various international peer-reviewed conferences and journals. The complete list of the published articles can found in appendix A.

The research ideas and results presented in this thesis are based on the following published articles:

C1. Gulati et. al.: Vehicle Infrastructure Cooperative Localization using Factor Graphs. 2016 IEEE Intelligent Vehicles Symposium (IV), 2016.

C2. Gulati et. al.: Robust Cooperative Localization in a Dynamic Environment using Factor Graphs and Probability Data Association Filter. 2017 20th International Conference on Information Fusion (Fusion), 2017.

C3. Gulati et. al.: Graph based Cooperative Localization using Symmetric Measurement Equations and Dedicated Short Range Communication. 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2017.

C4. Gulati et. al.: Data association - Solution or Avoidance: Evaluation of a Filter based on RFS Framework and Factor Graphs with SME. 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2017.

C5. Gulati et. al.: Robust Vehicle Infrastructure Cooperative Localization in Presence of Clutter. 21st International Conference on Information Fusion (FUSION), 2018 (accepted)

J1. Gulati et. al.: Graph based Cooperative Localization using Symmetric Measurement Equations. Sensors 17 (6), 2017.

1.5 Thesis Outline

The remainder of the thesis can be organized in three parts. The first is about the mathematical background, localization (the state that we want to estimate) and the state-of-the-art for such state estimation. The second part explains the major contributions. The third discusses these results and concludes the thesis.

The first part constitutes of Chapters 2, 3 and 4. Chapter 2 gives the overview of the problem of localization and discusses the state of the art methodologies, which can be used to solve it. Chapter 3 lays the foundation of the assumptions of our problem statement. Chapter 4 presents an overview of the Simultaneous Localization and Mapping (SLAM) problem and techniques to solve it.

The second part is presented in Chapters 5, 6, 8 and 7. Chapter 5 and 6 gives the overview of the contributions made using V2I technologies but for absence and presence of GPS sensor. 8 discusses the contribution to tackle the clutter. Research results with V2V are discussed in 7.

The third part is explained 9 and 10. Chapter 9 investigates the comparison of one of the state-of-the-art technologies against one of the methods proposed in second part. Chapter 10 concludes the work.

Chapter 2

Cooperative Localization

Three primary questions which any autonomous vehicle has to answer are "Where am I?", "Where am I going?" and "How should I get there?" [8]. The process of localization tries to answer the first one.

Importance of localization can be understood from the fact that since the appearance of current *Homo Sapiens*, humans have used many methods to answer the question. And this expertise is very important otherwise, Columbus would have correctly identified that he had landed in America and not India, and we would have read a different history.

From using celestial sources as references, through to radio-satellites for the centimeter accurate positioning, we have come a long a way in localizing ourselves on the planet. This chapter summarizes some of the modern methods and associated challenges with localization. The first section gives an overview of localization as done by individual robots and later presents cooperative localization. As the focus of this thesis is cooperative localization, subsequently it provides the state-of-the-art methodologies in the domain which attempt to solve the challenges identified in Section 1.3.

Since a modern autonomous vehicle is a robot, the words "robot" and "vehicle" are used interchangeably through the thesis.

2.1 Ego Vehicle Localization

Till about a decade ago, mobile robots were limited to factories and laboratories and also limited in numbers. The environment and the numbers were known in advance and simple sensors were used to localize them in the given environment. This section gives a non-exhaustive overview of various such methodologies which a vehicle can use to localize itself. These methodologies use only the sensors and the resources available within the vehicle.

2.1.1 Odometer

An odometer is one of the most basic sensor, which shows the number of units travelled, be it in miles or kilometers. It is based on the wheel turnings and can estimate the position of vehicle since the last start of the vehicle. One of the primary problem of odometer is that error grows unbounded [9]. This can be understood from Figure 2.1. A vehicle moves forward along the *x-axis*. The ellipse represents the position uncertainty, that is an iso-contour of the error, and visualizes a confidence interval. As seen, this

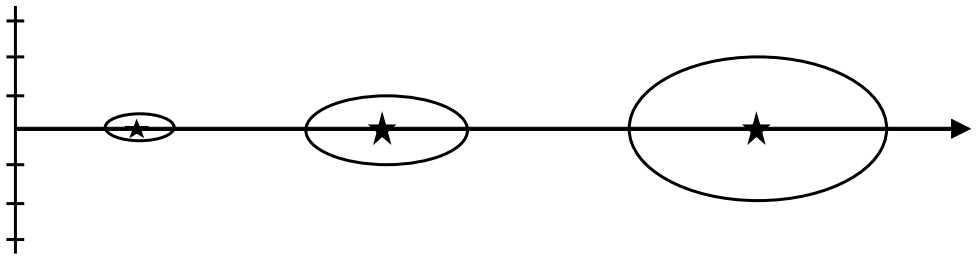


Figure 2.1: A simulation showing the uncertainty in odometer position growing as a function of distance. The circles indicate an iso-contour of the expected error covariance around the true position indicated by the stars.

ellipse grows as the vehicle moves forward along the axis. This signifies that the confidence in its own location reduces. This is because the odometer has an error for every unit of measurement, which keeps on adding as vehicle moves forward.

2.1.2 Inertial Measurement Unit (IMU)

An IMU is primarily a combination of an accelerometer and a gyroscope. The accelerometer gives the acceleration and the gyroscope measures the rate of rotation. The position can be obtained by integrating the measurements, twice for the accelerometer and once for the gyroscope. With advent of the technologies, it has become possible to manufacture cheap and accurate IMU units. Therefore, modern day vehicles are equipped with the IMU unit. Similar to odometer, the IMU suffers an increase in the error as the vehicle moves forward.

2.1.3 Active Triangulation

Using three or more sources of actively transmitting signals, a vehicle can determine its own position. The signal can be radio or light frequencies. To have a continuous estimation of the position, the signal sources should be evenly placed in the desired area. Because minimum three sources of the signal are required, the process is called *triangulation*. It is an old technique [10], but is still actively used. In fact as recently as 2013, Pierlot et. al. [11] presented an improved method for triangulation. The mobile phone towers use the same principle for a mobile phone localization. Also the triangulation method suffers from problems like occlusion and reflection as the signals can be hidden and/or reflected which can increase the uncertainty in the state estimation.

2.1.4 Global Positioning System (GPS)

GPS is a special case of the above-mentioned triangulation where satellites are used to transmit radio signals. Extensive open source literature is available [3]. However, here we provide a basic summary of the technology at the level necessary to understand its implications in the context of this thesis . Based on a minimum of 24 operating satellites consistently transmitting signals giving positions and time, various companies provide various solutions for the consumers. These can be purchased off-the-shelf by the users.

GPS is a U.S.-owned utility providing users with positioning, navigation, and timing services. Parallel to the GPS there are other similar services launched by different countries or organizations. These include

Russian Global Navigation Satellite System (GLONASS) [12], European Union Galileo positioning system [13], China's BeiDou Navigation Satellite System [14], India's NAVIC [15] and Japan's Quasi-Zenith Satellite System [16].

The satellites broadcast their signals in space with certain accuracy, but the reception depends on additional factors, including satellite geometry, signal blockage, atmospheric conditions, and receiver design features/quality. A standard GPS receiver in a smart phone, under open sky with clear visibility to at least a minimum of six satellites has an accuracy of 4.9 m. However, the accuracy worsens near buildings, bridges, and trees and can fall by up to 10 m.

Nevertheless, despite not having a very high accuracy, GPS and similar kind of services have become very common and even a cheap smart phone also comes equipped with a basic GPS sensor.

After obtaining a unique GPS coordinate, further localization is done by combining a pre-built map. For such map-based scenarios, the accuracy not only depends on the GPS but also on the precision of the maps.

2.1.5 Landmark Based Recognition

Any distinct feature in the environment can also be used to localize the vehicle. The features called landmarks can be artificially placed for example Augment-Reality tags (ARTAGS) or already present like trees or buildings.

Localization is done similar to the GPS. However, here instead of the GPS Signals, the vehicle or the robot uses the detected landmarks and after matching them in the map, localizes itself.

The process is often used for accurate indoor localization. One of the earliest work dates back to late 1980's where Tsubouchi et. al. [17] used a camera to do landmark recognition for a mobile robot. The mobile robot localizes itself in an indoor environment using a pre-built map by comparing the images. However, this technique is not very suitable for outdoor localization as pre-existing landmarks like buildings and trees are difficult to detect.

2.1.6 Localization using Multiple Methods

Because none of the methods are 100% reliable, localization is usually carried out as a fusion between two or more methods. For example one can fuse [18] odometer and GPS, or IMU and GPS [19] or Landmark Recognition and GPS [20].

All the methods require multiple sensors to be present in the vehicle. Cheap off-the-shelf sensors are not precise enough to support autonomous driving. General trend from various companies is to use expensive sensors like LiDAR and/or High Definition maps, which help the process of localization. LiDAR is able to construct a three dimensional feature map of the environment using laser beams and matches to the detailed maps to localize. One problem with High Definition maps is that they are not available for all the places and require continuous maintenance because of infrastructure changes in urban scenarios, along with large storage requirements or wide bandwidth requirements. All these reasons make the localization an expensive and challenging task.

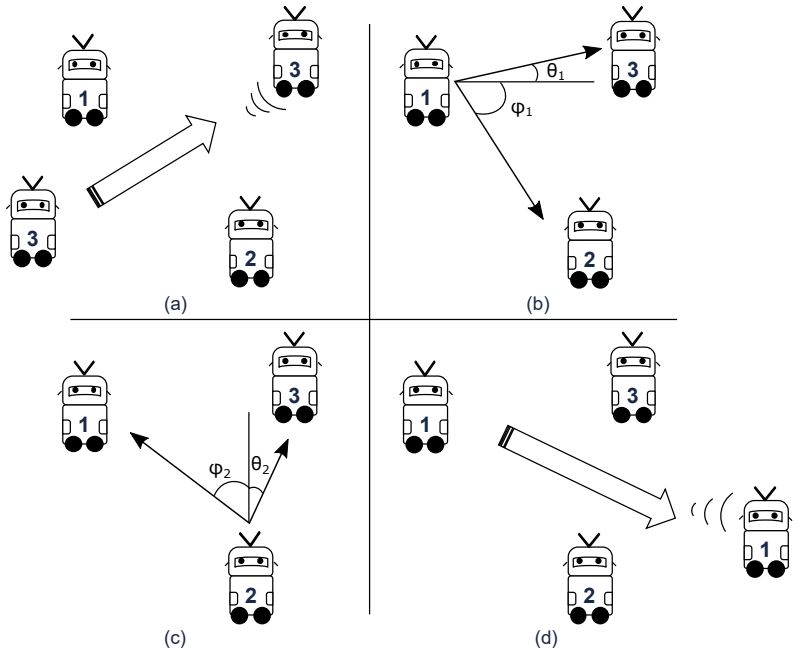


Figure 2.2: Cooperative localization as proposed by Kurazume et. al. [21] (a) Robot 3 moves. (b) After the movement of robot 3, robot 1 measures the azimuth position of robots 2 and 3. (c) Similarly robot 2 measures the azimuth of robots 1 and 3. (d) The new position of robot 3 is estimated and robot 1 moves.

2.2 Need for Cooperative Localization

Self localization for a single autonomous robot is not enough. More often in various environments multiple robots coexist. To achieve the assigned tasks safely and successfully, one common goal for all these robots is knowing where each of them is. This localization is not only valid from an absolute perspective but also from a relative one, i.e., not only know the actual position of the ego robot, but also the position with respect to the neighbouring robot.

The localization helps not only in safe execution, but can also help in execution of cooperative tasks. For such a motivation, the robots need not only to communicate with each other about the common goal they want to achieve but also about their positions. If some of the robots are equipped with high-precision sensors for better absolute localization, and sufficiently high bandwidth is available to facilitate the data exchange between various robots, then the other robots with low precision sensors can localize themselves using the data from the robots with high precision sensors. The same principle can also be applied to some high precision sensors present in the environment and the respective robots present around the communication range of the external sensor.

A localization where robots not only use their own internal sensors, but also utilize the data from external sources to lower their position estimation errors is called **Cooperative Localization**.

Cooperative localization, also known as *cooperative positioning* is not a new concept. In 1994, Kurazume et. al. [21] demonstrated the use of mobile robots to perform cooperative positioning. The authors divided the mobile robots in two sets: when one set moved, the other remained stationary and acted as the landmark for the moving robots. The robots move until a certain distance and stopped. The first stopped set of robots then starts to move using the second set of robots as landmark. This alternate

start and stop of moving robots continues until they reached the target. The authors called this process as the "dance". The approach is visualized in Figure 2.2. Using this technique, they demonstrated that the robots accumulated lower odometer errors. Their method allowed the robots to move forward without the knowledge of any external landmark. Hence it could be used to explore uncharted environments.

The work done by Kurazume et. al. paved the way for an improved position estimation. In addition, advancements in communication mechanisms made it possible to exchange large amount of data in real time. Therefore the vehicles or the robots do not need to be self-contained islands any more: They can communicate and benefit from any other data source inside or outside the autonomous system. Such advancements gave rise to Vehicle-to-Infrastructure (V2I) or Vehicle-to-Vehicle (V2V) communication mechanisms explored for the domain of autonomous driving [22].

In the last 20 years cooperative localization has become an active field of research. This also was triggered by the fact that the localization for a single robot using off-the-shelf sensors was possible only to a certain precision. However, when multiple robots, moving autonomously, are present in the same operating space, this precision in ego-localization is not sufficient enough.

For a small number of robots in a laboratory set up, one can use multiple and/or expensive sensors for each of the participant for ego-localization. However, with the advent of autonomous vehicles, methods to improve the ego-localization without increasing the cost is the preferred way to move ahead. One way to achieve the same is using high precision sensors in the environment acting as an infrastructure to cooperate with the sensors of the ego-vehicle for improved localization. The cost of high precision infrastructure sensor can be amortized over the traffic participants. This can lower the cost of ownership of such autonomous vehicles.

This thesis presents innovative methods for cooperative localization.

2.3 State of The Art

2.3.1 Methods

After Kurazume et. al. [21], many researchers have come up various innovative solutions for cooperative localization. Most of the earlier work uses inter-robot communication to perform the cooperative localization. Spletzer et. al. [23] extended the work of Kurazume et. al. [21] to perform cooperative localization to achieve control for multi-robot manipulation. Along with azimuth and elevation angle, they use the complete pose information based on visual imagery. Their work uses Kalman Filter and require identification and communication of each of the participating robots with each other.

Fenwick et. al. [24] implemented a centralized solution using Kalman Filter where all the states of all the vehicles are combined in huge matrices. This implies that measurement-to-track or the data association is already known.

Roumeliotis et. al [25, 26, 27] proposed an innovative use of a form of Distributed Kalman Filter (KF) to perform cooperative localization which they called as *Collective Localization*. Their proposed solution assumed all robots could move simultaneously, unlike previous solutions where one robot was moving at a given time. A centralized KF fuses the data using the observations from a group of mobile robots. Using both proprioceptive (relative) and exteroceptive (absolute) sensors, the standard KF prediction equations are decentralized and executed on the participating robots. This decentralized approach decouples the

state propagation equations. The state coupling occurs only when relative pose observations become available: that is whenever two robots meet. The relative pose information obtained from a camera tracking system is used to update the pose estimates of the robot team members. The solution could be extended to a system with a fixed robot acting as an external or an infrastructure sensor. However, one requirement for the solution is that the system should be able to identify the participating robots. Additionally, uncertainty for each of the robot is propagated individually at each step, thereby having high bandwidth requirements.

In [28], Rekleitis et. al. compare the various modalities, which can be used for cooperative localization. This includes inter-robot distance, azimuth angle from one robot to the other, a combination of distance and azimuth angle, full pose information etc. However, each of these modalities rely of identification of the individual robots, there by assuming that data association problem is already solved.

In [29], a graph-based approach is proposed for indoor cooperative localization using cameras. The camera scans areas of free space and generates a graph-based description of the environment. This graph is used to guide the exploration process and can be used for subsequent tasks such as place recognition or path planning. This is only possible for line of sight in small environments where the robots are not occluded from each other. The approach is similar to Kurazume et. al. [21] but uses graphs to map and localize.

In [30], Rekleitis et. al. extended [29] and demonstrated the use of a lidar and odometer to perform cooperative localization. They used particle filters to fuse the data. However, the main problem remains the same: they rely on identification of the robots for correct fusion. The work [31] suffers the same problem of robot identification.

Madhavan et. al. [32] use the distributed Extended Kalman Filter (EKF) to tackle non-linearities in the system. They further prove their results for outdoor multi robot localization. Their work can also be extended with one robot acting as a fixed infrastructure sensor. They have similar requirements as Rekleitis et. al. [25]: the solution requires the identification of individual robots.

Fox et. al. perform [33, 34] cooperative localization using Markov Localization: each robot maintains a belief over its position. Every time the robot sees a landmark, its belief is updated based on the likelihood of perceiving the landmark. To perform cooperative localization, belief from each robot is gathered to form a joint distribution. The authors assume that robots pose estimates are independent and use a factorial representation to keep the complexity under control. This approach can be used for centralized fusion where the believes are fused. Their experiments with cooperative Markov Localization clearly show an increased accuracy with usage of multiple robots against just one. This can be extended to a fixed infrastructure sensor scenario, but for fusing any data from external sensor, a measurement-to-track or data association has to be performed.

Das et. al. [35] use ad-hoc networking to perform cooperative localization. Each robot is capable of identifying other participating robots and is capable of estimating the range and bearing of other robots. Each robot is able to estimate the position of its visible neighbours without communication. However, to estimate the relative orientations each robot communicates with each other. All this information is fed into a graph to retrieve the pose of the complete formation. The authors claim is that for the cooperative localization their strategy can work for any number of robots. However, if robot identification is required for robots visible to each other, it can not be used for the purpose of vehicle infrastructure cooperative localization as it requires measurement-to-track association.

Similar to Das et. al. [35], Patwari et. al. [36, 37] use ad-hoc networking using wideband and ultra-wideband radio sensors for cooperative localization. They use various measurement-based statistical models of radio signals useful to describe time-of-arrival, angle-of-arrival, and received-signal-strength measurements in wireless sensor networks. The data can be fused using Kalman Filter or Monte Carlo estimation method that is particle filters. This also depends on node identification.

Yoshida et. al. [38] propose a method to detect automatically the coordinate transformations among a group of robots using a wall. The intention is to perform cooperative localization without configuring various coordinate transformations for participating robots. The approach can not be used for vehicle infrastructure cooperative localization, but highlights the importance of coordinate transformations while fusing sensor data originating outside the ego vehicle.

Howard et. al. [39] used combination of maximum likelihood estimation (MLE) and numerical optimization to infer the relative pose of every robot. They demonstrated the results on a set of four robots. One of the criteria of the approach is that each robot is equipped with a *robot sensor* such that it can measure the relative pose and the identity of nearby robots. The information for the measurements and covariances is maintained at each robot and exchanged whenever a robot is nearby. This makes their method robust to changes in the environment, and robots can use transitive relationships to infer the pose of previously unseen robots. This method can be extended to a stationary sensor observing all the robots. This would require that the stationary sensor performs the track-to-measurement association and sends the covariances on the network which itself can be a problem when the number of participants increase.

In [40], Howard et. al. use Monte Carlo Localization, that is the particle filter to perform cooperative localization. They define five kinds of observations and corresponding update rules similar to the Monte Carlo Localization work done by Fox et. al. [41] for a single robot. Each robot maintains its own set of all other robots distribution lists. The benefit of this approach is the robot has to transmit only the observations. With no new updates, the pose estimates become more uncertain. Since it needs to maintain a list of all other robot poses, it relies heavily on correct coordinate transformations. Each robot broadcasts the measurements of any observed robots and, based on the received measurements, it uses particle filter to judge the true pose. This makes the system computationally expensive, as each robot must maintain separate filters for all the participants. Additionally the system relies on correct identification of robots using the *robot sensor*.

In [42], the authors perform the cooperative localization using bearing of the observed robot by the observer robot (using cameras) and fusing it with odometer. They demonstrate the results using EKF, Particle Filter and a combination of both techniques. They only demonstrate it with two robots and do not encounter the problem of robot identification, but for more than two robots, such measurement-to-identification has to be built-in.

Trawny et. al. [43] try to tackle the problem of cooperative localization in an entirely different way. They try to optimize the entire trajectories for a group of mobile robots that use one another as localization beacons. They minimize the localization uncertainty, thereby obtaining the best localization information possible. They use classic EKF monolithic approach to robot localization. Inspired from Kurazume et. al. [21], the method uses participating robots as beacons, therefore needs measurement-to-track association.

Marco et. al. [44] came up with another interesting formulation of the cooperative localization. This

work is based on their previously proposed work on the single robot SLAM solution [45]. The approach is based on set membership hypotheses on the uncertainties, to simultaneously localize and build a map for a team of cooperating robots. Particularly, it is assumed that the error disturbances are unknown-but-bounded. Since the cardinality of the sets can become large, they propose efficient techniques for computing guaranteed outer approximations of the feasible sets. The methodology can be extended to a fixed sensor in the environment. However, this would require measurement-to-track association to be fused with the robot sensor measurements.

Martinelli et. al. [46] upgraded the approach by Rekleitis et. al. [26] and used distributed Extended Kalman Filter. In addition, they utilized three different relative observations to the fusion system, namely relative bearing, relative distance and relative orientation. Similar set-up has been used by [47] and [48]. All approaches can also be extended to making one of the participants as fixed and then improve the state estimates with vehicle infrastructure cooperative localization. However, they also suffers the same problem that measurement-to-track association is required.

Caglioti et. al. [49] present another interesting method of cooperative localization. Instead of just fusing every available measurement, they formalize the problem of selecting sensor measurements characterized by minimum residual uncertainty about the state of the multi-robot system. That is, they minimize the entropy of the whole system. The globally optimum measurement, i.e. the measurement that minimizes the expected variation of the entropy of the complete state vector (composed on all the robots), is one of the locally optimum measurements. They show that, in addition to EKF (which is used to update the estimate of robot state estimates from sensors), the selection of the optimal measurement can be distributed among the robots. They support their concepts using simulation and two mobile robots. Although an interesting approach, this again uses EKF, and requires measurement-to-track associations.

Karam et. al. [50] propose a novel method of state exchange where state is maintained by each vehicle using its own sensors. This exchange of independent states estimates are shared within the vehicle networks. They avoid the problem of over-convergence. Each participating vehicle maintains separately a state view, called sub-states, of other vehicles. To obtain the global state space all the sub-states are fused together with an Extended Kalman Filter. The algorithm can be extended for vehicle infrastructure cooperative localization by making one of the participant as fixed. In addition, as the number of vehicles will increase the number of maintained sub-states will also increase. However, the data exchange is expensive operation for EKF, as the covariance from the infrastructure will have to be exchanged every time after an update step. Additionally infrastructure sensor should be able to identify the targets to fuse the data.

Karam et. al. [51] proposed another method where instead of individual states; each participant exchanges the group states. Each vehicle maintains its own view of the group and exchanges this big state estimate. They avoid identification of individual vehicles. To fuse two group states they check the Mahalanobis distance to match its group state with the received group state. Additionally the bandwidth requirement increases with whole group state estimates being exchanged.

Many researchers [52, 53, 54, 55, 56] use the inter vehicle distance measurements obtained using real time radio signals fused along with GPS to improve the state estimation of vehicles. The fusion of the data is achieved by Kalman Filter or its extensions: Extended Kalman Filter and semi-extended Kalman filter [54]. Grabowski et. al. [57] use low cost ultrasonic sensor to find range between the robots.

These measurements are used as a position likelihood and all of them are combined to find a global solution that maximizes the position likelihood of each of the participating robot. They also address a unique multipath interference mode that arises as a direct result of the scale of the robot team. They demonstrate their results with localization and control of a small robot team. In [58], Parker et. al. propose the method of vehicle based non-linear least squares approach to estimate the vehicle positions based on the Euclidean distances. Parker et. al. propose a min-max algorithm to solve the problem of cooperative localization in [59]. Based on the distance information shared between the neighbouring nodes, each node solves a min-max problem to create a map of the relative positions of its neighbours in the same cluster. In addition, if a subset of these nodes has prior information about its position with respect to a global coordinate system, then all the nodes within the cluster can determine their global position. Researchers in [60] use robust Cubature Kalman Filter to fuse range measurements using DRSC and GNSS for improved cooperative localization. All these methods work well for vehicular point-to-point networks but for a vehicle infrastructure cooperative localization, measurement-to-target association poses a significant challenge.

Nerurkar et. al. [61] proposed an efficient methodology for cooperative localization. They use a distributed maximum a posteriori approach to solve the problem. Their distributed algorithm reduces the memory and processing requirements by distributing data and computations amongst the robots. To reduce the cost of computing the MAP estimates, a distributed Conjugate Gradient algorithm is used. The approach is quiet efficient and proportional to N^2, where N is the number of robots. Similar to many other approaches, this can also be used for vehicle infrastructure cooperative localization. However, it suffers the same issue like others. The approach requires identification of participants as it uses inter robot distance.

Li et. al. [62], came up with a unique approach to solve cooperative localization. Each robot maintains a decoupled state for the entire participants. Specifically, each robot stores and updates only the 3×3 covariance matrix for every participant for the joint state. On obtaining a relative measurement, the observer robot updates its local state-covariance using Split Covariance Intersection filter (SCIF). Further in [63], authors propose a methodology with decentralized SCIF. The method using SCIF is quiet efficient, and can be extended to a fixed infrastructure sensor. However, it would also apply that the sensor should be able to identify each of the vehicle and hence cannot avoid the challenge of data association.

Ahmed et. al. [64] demonstrate the cooperative localization using graphical representation of poses of robots, and positions and velocities of the neighbouring robots. The observations made by the robots are represented as the edges. These are stacked together as a single non-linear least square error function. They use g^2o [65] framework to represent and solve the graph. They consider the data association for the neighbouring robots is already solved. They show that their proposed solution performs better than EKF. Similar to the approach by Ahmed et. al. [64], Huang et. al. [66] propose the Unscented iSAM using GTSAM [67]. As with many previous other cooperative localization solutions, these can also be extended for vehicle infrastructure cooperative localization. However, both do not resolve the data association problem.

Indelman et. al. [68] propose to utilize vision-based navigation techniques for cooperative navigation. Each vehicle is equipped with an odometer and a camera. The cooperative localization is based on a three-view geometry of a general scene where a measurement is formulated when same scene is observed by different vehicles. Interesting point to be noted is that vehicles are not required to identify each other.

It presents a very good technique, but for the vehicle infrastructure cooperative localization; an external sensor is fixed and cannot use the concept of three-view geometry of a general scene.

In [69], Nerurkar et. al. try to address the bandwidth constraints. The measurement innovation is defined as the difference between the actual and the estimated (by the filter) measurement. Instead of the complete measurements, communicating this measurement innovation can save precious bandwidth. The authors proposed a modified hybrid Sign-of-Innovation Kalman Filter (mhSOI-KF) that addresses the communications bandwidth issue through quantization of measurement data using an asymmetric encoding/decoding scheme. Since the bandwidth availability can vary in realistic scenarios, in [70], they improve their approach by incorporating provision for fluctuating available bandwidth. As the base of their fusion is Kalman Filter, it also requires identification of participating targets.

Savic et. al. [71] use non-parametric belief propagation in sensor networks to perform cooperative localization. The belief propagation method works in scenarios where some *Anchor Nodes* know their position with more accuracy than others. This information is passed across the network to improve the localization of other nodes. They use the particle based non-parametric version and improve it to avoid circular loops in the network. This approach does not easily extend to vehicle infrastructure cooperative localization.

Similar to Martinelli et. al. [46], Xingxi et. al. [72] uses relative measurements between robots for cooperative localization. Instead of EKF, they use distributed Unscented Kalman Filter to perform cooperative localization. This suffers the same problem that it requires identification of participants.

Zhang et. al. [73] propose the use of Probability Hypothesis Density (PHD) Filter to perform the cooperative localization. They use relative measurements between various participants. The data association problem for relative measurement is inherently solved by PHD Filter as it is based on Random Finite Set Statistics. This approach lends easily to the principal of vehicle infrastructure cooperative localization. However, the external measurements still have to be transformed into one common coordinate system. Additionally, covariances have to exchanged at each step, making the bandwidth requirement high.

Sharma et. al. [74] proposed the use Extended Information Filter (EIF), the information form of the EKF, to perform cooperative localization. They use inter-robot bearing measurements to achieve the goal. As with many previous solutions, this can also be extended to perform the vehicle infrastructure cooperative localization, but requires measurement-to-track association of the targets.

Liu et. al. propose a method using GPS in [75] for cooperative localization. They propose ranging technique called weighted least squares double difference (WLS-DD), which is based on the sharing of GPS pseudo range measurements and a weighted least squares method. The carrier-to-noise ratio (CNR) of raw pseudo range measurements are taken into account for noise mitigation. They demonstrate the superiority of their methodology with field experiments. Although this requires only single sensor, but cannot be deployed as presence of GPS signals is not uniform.

Similar to [46, 72], Rohani et. al. [76] use relative measurements and exchanged GPS coordinates fused with combination of MAP and Kalman Filter to perform cooperative localization. Apart from the exchange of measurements and covariance matrices, their techniques also relies on vehicle identification and hence requires additional solution for data association for vehicle infrastructure cooperative localization.

Zhang et. al. [77] proposed a method of using Symmetric Measurement Equation transformation

along with the external sensor measurements incorporated as sum of inter-vehicle distances. The paper uses particle filter to execute the SME transformation with the sum of powers. The result is demonstrated using constant turn and constant velocity simulation. It is an ideal solution, which can help to achieve the goal of avoiding data association, coordinate transformation and low bandwidth requirements. However, the issue of scalability is not clear by the authors and left for further exploration. Additionally, depending on the number of particles, particle filter can prove to be computationally expensive.

2.3.2 State-of-the-Art in a Nutshell

As seen in the previous section, one of the most common requirement of the methods presented in the various methodologies is the identification of the participants for cooperative localization. This introduces the additional steps of data association. Additionally, to fuse the data sensor should transform the coordinates to one common system.

To have a good overview of the various technologies, most frequently used methods have been extracted and summarized in Table 2.1 against the challenges identified in 1.3. This table is a representative of the novel methods analysed in the previous section and there can exist further improvements and variations of these. However, it gives an idea how the current state-of-the-art methods match with the identified challenges. Further details of individual methods and their variations can be referred in the previous section.

With this anaylysis, a **scalable** solution for vehicle infrastructure cooperative localization with **minimum bandwidth requirement**, in process **avoiding the overhead of data association** and **without coordinate transformation** remains a challenging task. These are first four challenges as mentioned in Section 1.3, i.e. points $(1-4)$. Additionally, (a) to make a system usable in realistic scenarios of clutter; and (b) adaptable to the number of sensors and changing configurations; remaining points 5 and 6 also need to be addressed.

2.4 Summary

This chapter explained the problem of localization and cooperative localization. It also highlighted that how the cooperative localization can help in lowering the cost of the ownership of autonomous vehicles. It presented an detailed overview of the State-of-the-Art solutions for cooperative localization and highlighted the fact that a practical solution covering various challenges still requires further investigation. Based on the identified challenges and investigated state-of-the-art methodologies, next chapter constructs the scenario and outlines the assumptions which forms the basis of this thesis.

Table 2.1: State-of-the-Art in a Nutshell

Method	Challenge 1	Challenge 2	Challenge 3	Challenge 4	Challenge 5	Challenge 6	
	Data Association / Target Identification	Coordinate Transformation	Bandwidth Utilization	Scalability	Clutter	Adaptability	Remarks
Kalman Filter and variants (EKF, Distributed (KF, EKF)), Minimum Entropy based approach	Additional solution required	Required	Mostly high	limited to bandwidth	Additional solution required	No	[21], [24], [27], [32], [49], [46]
Particle Filter, Markov Localization	Additional solution required	Required	Mostly high	Difficult	Can be handled	No	[30], [33], [40]
Ad-hoc networking/Inter-vehicle distance	Requires target identification	Required	Low to High	Yes	Required as part of solution	No	Only suitable for V2V ([35], [36],[37], [52], [53])
Maximum Likelihood Estimation / Maximum a Posteriori Estimate	Requires target identification	Required	Low	Yes	Additional Solution Required	No	[39], [61]
Set membership approach on uncertainty	Requires target identification	Required	Not-clear	Yes	Additional Solution Required	No	[44]
Split Covariance Intersection Filter	Requires target identification	Required	Low	Yes	Additional Solution Required	No	[62]
Graph based approach (g2o)	Additional solution required	Solves Internally	Low	Yes	Additional solution required	Yes	[64]

Continued on next page

Table 2.1 – continued from previous page

Method	Challenge 1	Challenge 2	Challenge 3	Challenge 4	Challenge 5	Challenge 6	
	Data Association / Target Identification	Coordinate Transformation	Bandwidth Utilization	Scalability	Clutter	Adaptability	Remarks
Modified hybrid Sign-of-Innovation Kalman Filter	Additional solution required	Required	Low	Yes	Requires additional solution	No	[69]
GLMB Filter / RFS based approaches	Internally solved	Required	High	Yes	Internally Solved	No	[73]
Extended Information Filter	Requires target identification	Required	Low	Yes	Additional Solution Required	No	[74]
SME Transofmration with external sensor measurements incorporated as sum of inter-vehicle distance	Does not requires target identification	Not required	Low	Questionable	Additional Solution Required	No	[77]

Chapter 3

Scenario Definition and Assumptions

Based on the challenges mentioned in Section 1.3 and the analysis of the state-of-the-art methodologies in Section 2.3, the simplest system set-up to address cooperative localization based on Figure 1.3 can be understood from Figure 3.1. The set-up in Figure 3.1 highlights the the external radar and internal ego-vehicle sensors. Although I would be using the word *highway* for most of the discussion but the concepts can be extendend to are valid for the urban scenarios as well. Various components of the scenario can be described as follows:

1. Participating vehicles have two sensors:
   - Odometer: To provide the distance travelled in unit time.
   - GPS: To measure the position of the vehicle in global coordinate system.
2. Radar is mounted across the highway or in an urban scenario in the infrastructure as an external sensor, one for each direction of the traffic. The dotted line represents the measurement of the radar. Using radar measurements for better state estimation can yield various situations:
   (a) Complete radar configuration is available. The precise position of the radar in global coordinate system is available; its measurement in local coordinates can be converted into global coordinate system. It has a data association algorithm, which is able to perform the measurement-to-target matching. This precise information is fused with the odometer and GPS data to perform cooperative localization.
   (b) Partial radar configuration is available. The precise position of the radar in global coordinate system is not available; therefore, its measurement in local coordinates cannot be converted into global coordinate system. However, it can perform the data association. This becomes challenging for the algorithm performing the data fusion, as it is able to associate the right measurement to the right target but is not aware of how the radar measurements can be converted to the coordinate system of the vehicle. Hence, cooperative localization in this scenario is not possible.
   (c) Partial radar configuration is available. The precise position of radar in global coordinate system is available; its measurement in local coordinates can be converted into global coordinate system. However, it has no way to perform the data association. This becomes challenging

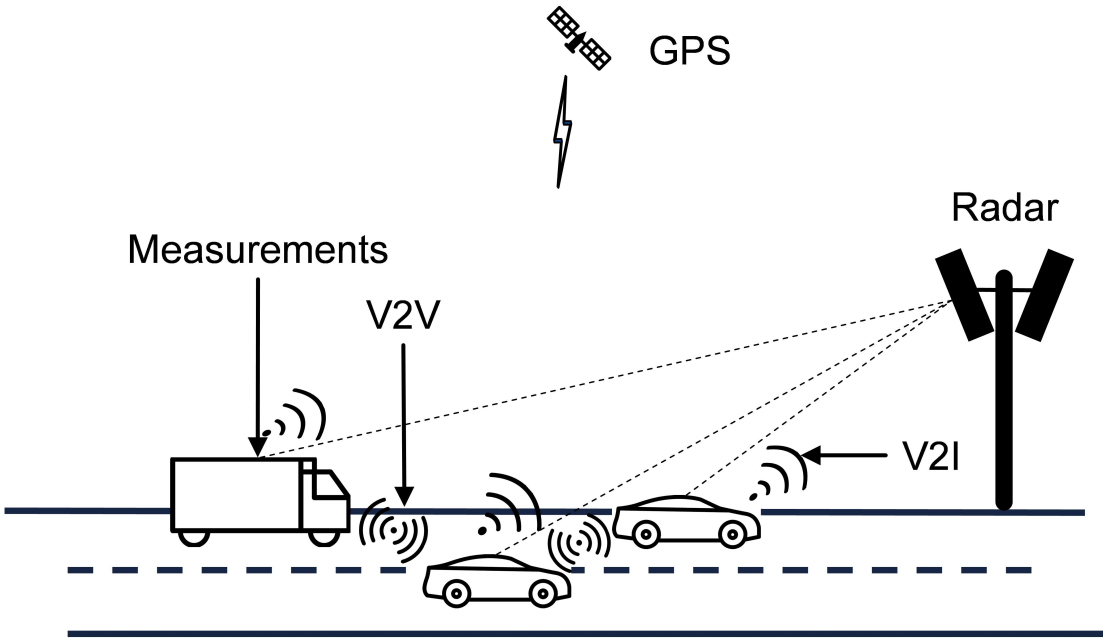


Figure 3.1: Participants in cooperative localization scenario. Vehicles are equipped with off-the-shelf odometer and GPS sensors. High precision radar is mounted above the road on fixed infrastructure on the road where it can measure the positions of vehicles in its range. Vehicle can connect to each other using V2V technologies. The vehicle can connect to the infrastructure radar using V2I technologies.

for the algorithm performing the data fusion to use the measurements for a state estimation, as it is unable to associate the right measurement to the right target. Hence, cooperative localization in this scenario is also not possible.

(d) Lastly no configuration of radar is available. Its position in global coordinate system is not available; hence, it cannot convert the measurement in the local coordinates to the global coordinate system. In addition, it cannot perform the data association; therefore, we do not have the measurement-to-target information. This becomes challenging for the system performing the data fusion to use the radar measurements for an improved state estimation, as not only the system is unable to associate the right measurement to the right target, but also is unable to convert the measurement of radar to the desired coordinate system. Again, cooperative localization in this scenario is also not possible.

Better state estimation using any information from outside the vehicle using V2I is possible, but to make it a success, one should also know the complete configuration. With increasing number of sensors outside the vehicle, knowing the configuration of each of the available sensor is impossible. This is also true for all the sensors deployed on smart highways. Therefore, for this thesis the last case in the above list that is, 2(d), when no configuration is available is one of the primary interest.

3. To perform cooperative localization, the thesis assumes an error free V2X bi-directional communication capability is available. Further, various communication methods are not used to identify the individual vehicles. There are various radio-ranging methodologies like received signal strength, time of arrival, time difference of arrival, Doppler Shift and angle of arrival [78] to identify indi-

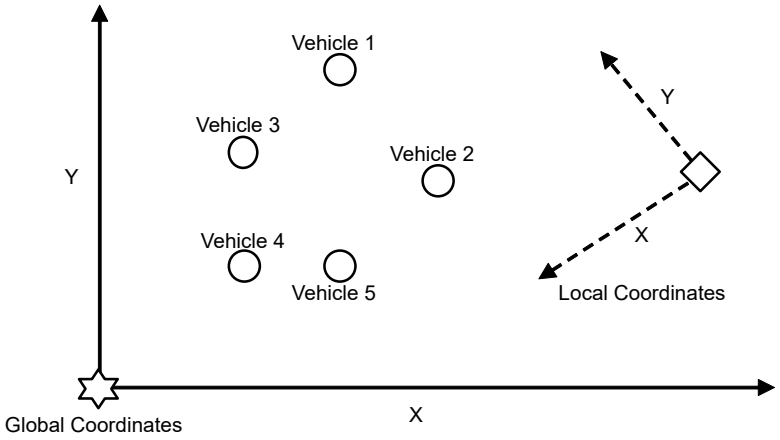


Figure 3.2: Figure shows the relationship between the global coordinate system (in solid lines) and the radar coordinate system (in dashed lines). For a cooperative localization system, dealing with such coordinate transformations for every external sensor is a challenging task.

vidual targets. It is assumed that any external sensor, like infrastructure radar does not undertake any such action to identify the individual targets.

4. Clutter poses a significant challenge as any false measurements in the system negatively effect the state estimation. Skolnik [79] explains various clutter characteristics and its impact on the radar performance. One of the common problems experienced by such dynamic target tracking radars is that of *Ghost Objects*, i.e. measurements which do not correspond to any true object but are presented by radar as valid measurements. In the research work presented in this thesis, first a clutter free environment is assumed. Nevertheless, in Chapter 8, the impact of clutter on the state estimation is considered. It goes further introducing methodology to address the challenges of clutter, when combined with some of the proposed methodologies earlier presented in this Thesis.

The graphical formulation of the problem presented in Figure 3.1 can be expressed as Figure 3.2. In this figure:

1. Circles represent the vehicles.
2. Solid axis represents the global coordinate system.
3. Dashed coordinate system represents the local coordinate system of the radar. Since the configuration of the radar system is unknown, therefore the transformation information between the global and the local coordinate system is also unknown.

3.1 Summary

This chapter constructed a practical scenario along with the assumptions, which would be addressed in this thesis. Next chapter gives an overview of the tools and framework used to address the defined challenges. Subsequent chapters, based on the assumptions presented in this chapter, will propose innovative solutions. In later chapters, the thesis relaxes some of the assumptions, try to model realistic scenarios, and innovate further to tackle these challenges.

Chapter 4

SLAM, Graphs and Non-Linear Least Square Optimization

This work heavily uses the concept of factor graphs. Factor graphs offer an intuitive way to model and represent both the problem of Simultaneous Localization and Mapping (SLAM) and the corresponding solution to build a map and localize itself in it. The constructed map represents the joint probability function and once an adequate graph has been constructed for the state of the problem, it can be *optimised* for the state estimates using various algorithms like Levenberg-Marquardt, iSAM2 etc. The *optimising* of the graph allows to marginalize out the individual probability functions. This chapter gives an overview of the tools used, i.e. concepts of factor graphs in realising the solutions proposed in this thesis.

4.1 Simultaneous Localization and Mapping

Before the concept of factor graph is explained, a brief overview of the kind of problem, which is solved using factor graphs, is provided. Simultaneous Localization and Mapping (SLAM) [8] is one of the oldest but yet one of the most actively researched topic in Robotics.

SLAM is the process undertaken by a mobile robot of building a map of its environment and localizing itself in the constructed map. The mapping and the localization happen at the same time, hence *simultaneous*.

The mobile robot is a multi-sensor system. It can have various kinds of sensors, starting from basic sensors like odometer, GPS, Camera, Ultrasonic, to complex sensors like radar and LiDAR. The mobile robot measures its own movement using sensors like odometer and perceives its environment using sensors like Camera, radar and LiDAR. Additionally, if equipped with GPS, it can also localize itself in a coordinate system, which can uniquely identify its position in on the earth. This coordinate system is called as *World Coordinate System*.

SLAM is a challenging problem. To build the map from various sensor measurements, it needs to know its orientation and position relative to the map. Therefore, it localizes itself in an incomplete map. Hence, localization and mapping go hand-in-hand. If either of them is incorrect, then the future steps will be incorrect too. Hence despite an introduction of the topic in late 1980s [80], it remains an active topic of research.

This Thesis is concerned with the "**L**" of SLAM, that is Localization. The "**M**" part, that is building the map is ignored. To get the best state estimate of the vehicle, the data from all the available sensors has to be fused. Assuming u_i represents the control input applied to the robot at any time t, then the total control over the time can be written as:

$$\mathbb{U} = \{u_1, u_2, \cdots, u_I\} = \{u_i\} \; \forall i = 1 \cdots I \tag{4.1}$$

One of the most common sensor available for any mobile robot is an odometer. Because of the above-defined control input, there is a movement of the robot. Odometer measures this movement between two positions. If a sensor to measure the change in orientation of the robot along with the position is also available, then change in pose can be calculated. Pose of a robot is defined as the combination of position and orientation of the robot.

Assuming Gaussian measurement models, which is a standard in SLAM domain [81], the relationship between two consecutive positions (x_{i-1}, x_i) of a robot, using odometer sensor can be written as:

$$x_i = \mathrm{f}(x_{i-1}, u_{i-1}) + w_{i-1} \tag{4.2}$$

where f denotes a linear or non-linear function which calculates the current position based on the previous position and the control input and w_i represents the Gaussian distributed zero-mean noise of the sensor covariance matrix Σ^o_{i-1} :

$$w_{i-1} \sim \mathcal{N}(0, \Sigma^o_{i-1}) \tag{4.3}$$

and as assumed measurement model is Gaussian:

$$x_i \sim \mathcal{N}(\mathrm{f}((x_{i-1}, u_{i-1}), \Sigma^o_{i-1}) \tag{4.4}$$

It is to be noted that the above function f can mean any kind of motion (constant velocity, constant turn/constant acceleration etc.) undertaken by the robot. A detailed survey of various motions models can be found in [82].

Apart from the control input and the odometer, the mobile robot can be equipped with various other sensors. These can be classified into two parts:

- Proprioceptive Sensors, which can measure the internal state of the system. An example of the same is odometer. Other include Inertial Measurement Unit (IMU), Global Positioning System (GPS) etc.
- Exteroceptive Sensors, which measure and perceive the environment. These include camera, radar, and LiDAR etc. These sensors can measure the distance, pose or orientation of the objects present in the environment.

If $\forall j = 1 \cdots J$, $\mathbb{Z} = \{z_j\}$ represents the sensor measurement (like the distance or orientation) of landmarks $\mathbb{L} = \{l_k\} \; \forall k = 1 \cdots K$, then using a sensor model h, the measurement z_j, between a given landmark l_k and the robot pose can be written as:

$$z_j = \mathrm{h}(x_{ij}, l_{kj}) + \lambda_j \tag{4.5}$$

where λ_j is a zero-mean Gaussian noise with covariance Σ^z_j and is represented as:

$$\lambda_t \sim \mathcal{N}(0, \Sigma^z_j) \tag{4.6}$$

Like f for odometer, h is a (non-linear) function for most practical applications. Similar to x_i, z_j is modelled as Gaussian:

$$z_j \sim \mathcal{N}(\mathrm{h}((x_{ij}, l_{kj}), \Sigma_j^z) \tag{4.7}$$

With this basic introduction, estimating all the position (Localization) and the complete map of the environment based on all the sensor measurements, can be mathematically expressed as:

$$P(\mathbb{X}, \mathbb{M}|\mathbb{U}, \mathbb{Z}, \mathbb{L}) \tag{4.8}$$

where P is the probability distribution over the complete robot state $\mathbb{X}$ in a completely constructed map $\mathbb{M}$. There are different kinds of $\mathbb{M}$ maps possible, like occupancy grid maps and feature maps, depending on the sensor. The details of such maps can be found in [81] and [83]. $\mathbb{U}$ represents all control measurements and $\mathbb{Z}$ other sensor measurements for all the detected landmarks $\mathbb{L}$. This type of solution where the probability distribution over all the available data is used to predict the final state estimation is called *smoothing*.

If x_{t-1} represents the robot position at time $t-1$, M_{t-1} the map constructed until time $t-1$, u_t is the control input at time t, z_t the sensor measurement of landmark l_{kt}, the position x_t at M_t for time t can be written as:

$$P_t(x_t, M_t|x_{t-1}, M_{t-1}, u_t, z_t, l_{kt}) \tag{4.9}$$

Here the next state is calculated just based on the previous state information. This can be expressed recursively and the process is called *filtering*.

4.2 Graphical representations

In this thesis, graphical methods are used to model the SLAM problem. Two such methods are discussed here.

4.2.1 Bayesian Networks

Bayesian networks are directed acyclic graphs that represent the conditional (probabilistic) dependencies between various variables. It can be understood using a widely known simple example [84].

Assume that the grass can be wet because of two events: either it has rained or the sprinkler was on. There is a condition that, when it is raining, the sprinkler is deactivated. If G represents "the grass is wet", S represents "the sprinkler is on" and R represents that "it is raining", then the joint probability $P(G, S, R)$ can be written as:

$$P(G, S, R) = P(G|S, R) \cdot P(S|R) \cdot P(R) \tag{4.10}$$

The above can be graphically represented as Figure 4.1 which shows a simple Bayesian network. Rain ($P(R)$) influences whether the sprinkler is activated ($P(S|R)$). Both rain and the sprinkler influence whether the grass is wet ($P(G|S, R)$).

Using the concept explained above, the SLAM problem in Equation 4.8 can also be modelled. However, the state variables for the SLAM are not only conditionally dependent, but also time dependent. This implies that the subsequent variables are added as the time increases. Therefore, two adjacent variable

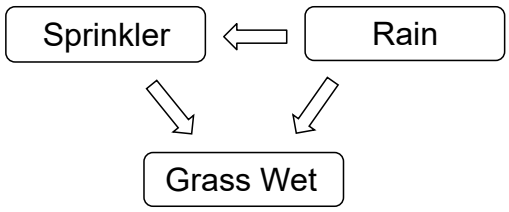


Figure 4.1: Relationship between Sprinkler, Rain and Wet Grass. Rain influences whether the sprinkler is activated, and both rain and the sprinkler influence whether the grass is wet. [84]

.

are connected over adjacent time steps. Since the probability of the initial start state is known, this can also be added. This kind of Bayesian Network having the time dependency between the nodes is called Dynamic Bayesian Network [85].

The joint probability for all the variables in Equation 4.8 is given as:

$$P(\mathbb{X}, \mathbb{M}|\mathbb{U}, \mathbb{Z}, \mathbb{L}) \propto P(x_0) \prod_i P(x_i|x_{i-1}, u_i) \prod_{j,i,k} P(z_j|x_{ij}, l_{kj}) \tag{4.11}$$

where $P(x_0)$ is a prior on the initial state, $P(x_i|x_{i-1}, u_i)$ represents the motion model with u_i as the control input, and $P(z_j|x_{ij}, l_{kj})$ is the sensor model.

Figure 4.2 represents SLAM when expressed as Dynamic Bayes Network.

4.2.2 Factor Graphs

Figure 4.3 is an example of a graph with variables w, x, y and z and functions f_1 and f_2 with factorization: $f(w, x, y, z) = f_1(w, x, y) \cdot f_2(y, z)$. Each of the function f_i is dependent on the variables which are passed as parameters. Kschischang et. al. [86] defined factor graphs as bipartite undirected graphs. As seen in the above example the nodes are either square or circle and no two squares or circles are connected. Therefore, the same definition can be extended to Equation 4.11 which can be seen as the product of

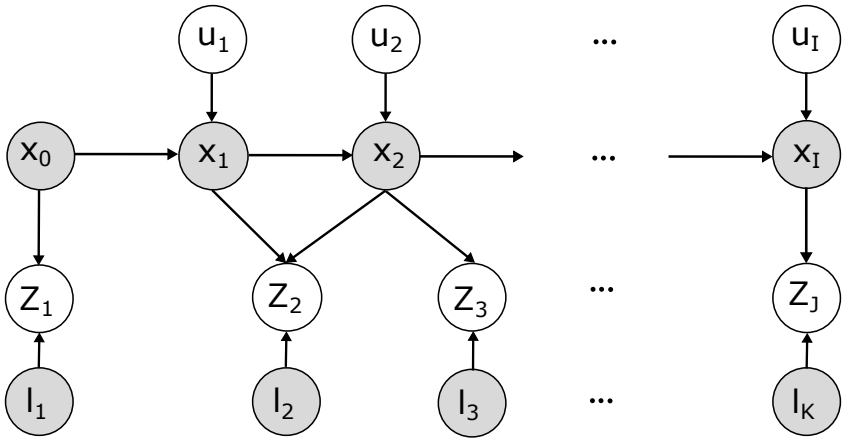


Figure 4.2: The SLAM problem represented as a Dynamic Bayesian Network. Grey nodes are the variables like the robot state x_i and the landmark positions l_k, White nodes represent the observable random variables, e.g. the sensor measurements z_j and control inputs u_i.

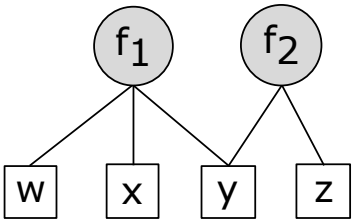


Figure 4.3: Factorization of f using f_1 and f_2. $f(w, x, y, z) = f_1(w, x, y) \cdot f_2(y, z)$

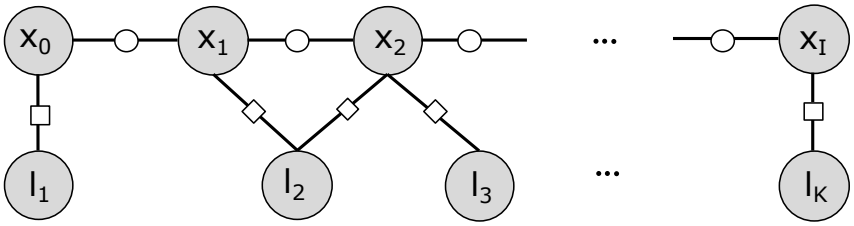


Figure 4.4: Factor graph representation of the SLAM problem with landmarks. The large circles with x_i represent the unknown robot poses and with l_k represent the landmark nodes. The small circle and small square represent probabilistic relationships between them. Odometer factors are represented by small circle and landmark factors are represented by squares.

probabilistic functions which are the *factors* and are dependent on variables which are the parameters to the probabilistic functions.

Mathematically, a bipartite graph $G_k = (F_k, V_k, E_k)$ is defined as a factor graph when: (1) It has two types of nodes: factor nodes $f_i \in F_k$ and variable nodes $v_j \in V_k$; (2) Edges $e_{ij} \in E_k$ can exist only between factor nodes and variable nodes, and are present if and only if the factor f_i involves a variable v_j [87].

Hence, in simpler words, factor graphs provide a natural graphical description of the factorization of a complex probabilistic function into a product of simpler probabilistic functions [86]. Being a bipartite graph, the edges always connect the variable node to the factor node, thus highlighting the probabilistic relationships [86]. Notice that this is similar to the Bayesian Networks as explained in previous section. Hence, factor graphs can also be used to represent the Bayesian Networks[88].

Although a Bayesian Network is similar to factor graph, but it is not the same. The 'arrow' in the Bayesian Network dictates conditional dependence (i.e. the '|' in the probability). This implies that in Bayesian Network the next probability is conditioned on the previous, thereby having a forward flow of the information. On the other hand, in a factor graph the flow of information can be either ways. This is an important property as once we encounter a more certain information, this can influence the uncertainty of the information seen in the past, i.e. both probabilities can change. This property lends itself to represent the SLAM problems.

For SLAM, the robot states and landmarks represent the variables and factor nodes represent the conditional probabilistic relations. Figure 4.4 shows the factor graph of the DBN represented in 4.2. The control input results in motion of the robot and hence has no representation in the factor graph. But the resulting motion is expressed as *Odometer factor* represented as small circles. The sensor measurements to the landmark are expressed as *Landmark factor* represented as square.

4.3 SLAM and Least Squares Optimization Problem

There exist many methodologies to solve the SLAM problem which use various Sensor Fusion Algorithms like Extended Kalman Filter [89], Particle Filter [90] and Random Finite Sets [91]. In this work, Graphical Methods and Non-Linear Least Square Optimization are used to reach the goal. This section describes how SLAM problem is converted into a Non Linear Least Square Optimization problem.

As seen in Equation (4.11), the probability distribution of all the states $\mathbb{X}$ for given $\mathbb{U}$ and $\mathbb{Z}$, is proportional to the likelihood of the measurements. Likelihood of the measurements for the given states can be computed if the observation or the sensor model is available. Since the initial assumption of the noise is Gaussian (Equation 4.6), the likelihood will also be Gaussian. Therefore, Gaussian likelihood for the component of Equation 4.11 can be calculated as:

$$P(z_j|x_{ij}, l_{kj}) \propto \exp(-(\hat{z}_j - z_j)^T(\Sigma_j^z)^{-1}(\hat{z}_j - z_j)) \tag{4.12}$$

where exp is the exponential e, $\hat{z_j}$ is the predicted measurement for a given state x_{ij} (see Equation 4.7) and is the mean of the conditional distribution and (Σ_j^z) is the covariance matrix of the sensor.

Equation 4.12 becomes:

$$P(z_j|x_{ij}, l_{kj}) \propto \exp(\|(\hat{z}_j - z_j)\|^2_{\Sigma_j^z}) \tag{4.13}$$

where,

$$\|(a-b)\|^2_\Sigma = (a-b)^T(\Sigma)^{-1}(a-b) \tag{4.14}$$

is the squared Mahalanobis distance:

Substituting the definition of sensor model from 4.7 in 4.13:

$$P(z_j|x_{ij}, l_{kj}) \propto \exp(\|(\mathrm{h}(x_{ij}, l_{kj}) - z_j)\|^2_{\Sigma_j^z}) \tag{4.15}$$

Similarly from Equation 4.4 we get:

$$P(x_i|x_{i-1}, u_i) \propto \exp(\|(\mathrm{f}(x_{i-1}, u_{i-1}) - x_i)\|^2_{\Sigma_i^o}) \tag{4.16}$$

Therefore, the joint probability in Equation 4.11 can be written as:

$$\begin{aligned} &P(\mathbb{X}, \mathbb{M}|\mathbb{U}, \mathbb{Z}, \mathbb{L}) \propto \\ &\underbrace{P(x_0)}_{\text{Prior}} \cdot \underbrace{\prod_i \exp(\|(\mathrm{f}(x_{i-1}, u_{i-1}) - x_i)\|^2_{\Sigma_i^o})}_{\text{Odometer}} \cdot \underbrace{\prod_{j,i,k} \exp(\|((\mathrm{h}(x_{ij}, l_{kj})) - z_j)\|^2_{\Sigma_j^z})}_{\text{Sensor}} \end{aligned} \tag{4.17}$$

Since the focus of in this thesis is only in the states $\mathbb{X}$ and not the construction of the map $\mathbb{M}$, in further calculation $\mathbb{M}$ is dropped.

For a given observation data $\mathcal{D}$, the parameter Θ can be optimally estimated using Bayes theorem:

$$\begin{aligned} P(\Theta|\mathcal{D}) &= \frac{P(\mathcal{D}|\Theta) \cdot P(\Theta)}{P(\mathcal{D})} \\ \text{posterior} &= \frac{\text{likelihood} \cdot \text{prior}}{\text{evidence}} \end{aligned} \tag{4.18}$$

Therefore, the value of Θ which maximizes the posterior probability, i.e. $P(\Theta|\mathcal{D})$, is the most optimal value. This is denoted as Θ^* and is known as *Maximum a Posteriori* (MAP) estimate. Consequently,

$$\begin{aligned}\Theta^* &= \arg\max_{\Theta} P(\Theta|\mathcal{D}) \\ &= \arg\max_{\Theta} \frac{P(\mathcal{D}|\Theta)\cdot P(\Theta)}{P(\mathcal{D})} \\ &= \arg\max_{\Theta} P(\mathcal{D}|\Theta)\cdot P(\Theta) \\ &= \arg\max_{\Theta} \prod_{d_i\in\mathcal{D}} P(d_i|\Theta)\cdot P(\Theta)\end{aligned} \tag{4.19}$$

The denominator is dropped because it has no direct functional dependence on Θ with respect to which it has to be maximized. Taking negative logarithm of the above equation converts the maximization problem to a minimization problem. That is,

$$\Theta^* = \arg\min_{\Theta} -(\log \sum_{d_i\in\mathcal{D}} P(d_i|\Theta) + \log\, P(\Theta)) \tag{4.20}$$

From the above Equation 4.19 and Equation 4.17, representing $\mathbb{X}^*$ as MAP estimate of $P(\mathbb{X}|\mathbb{U},\mathbb{Z},\mathbb{L})$, it can be seen:

$$\mathbb{X}^* = \arg\max_{X} P(\mathbb{X}|\mathbb{U},\mathbb{Z},\mathbb{L})$$

$$\mathbb{X}^* = \arg\max_{X} \Bigg(\underbrace{P(x_0)}_{\text{Prior}} \cdot \underbrace{\prod_i \exp(\|(\mathrm{f}(x_{i-1},u_{i-1}) - x_i\|^2_{\Sigma^o_i})}_{\text{Odometer}} \cdot \underbrace{\prod_{j,i,k} \exp(\|(\mathrm{h}(x_{ij},l_{kj}) - z_j)\|^2_{\Sigma^z_j})}_{\text{Sensor}} \Bigg) \tag{4.21}$$

Similar to Equation 4.20, this can be simplified by taking negative logarithm and converting it to a minimization problem:

$$\mathbb{X}^* = \arg\min_{X} \left(\sum_i (\|(\mathrm{f}(x_{i-1},u_{i-1}) - x_i)\|^2_{\Sigma^o_i}) + \sum_{j,i,k} (\|(\mathrm{h}(x_{ij},l_{kj}) - z_j)\|^2_{\Sigma^z_j}) \right) \tag{4.22}$$

Note the prior $P(x_0)$ has been dropped for simplicity.

Further, for sensors where an absolute measurements is available, like GPS or a measurement of a vehicle in the field of view of an external radar, there is no landmark involved. Therefore, under such scenarios the terms $\mathbb{L}$ and l_{kj} can also be dropped.

$$\begin{aligned}\mathbb{X}^* &= \arg\max_{X} P(\mathbb{X}|\mathbb{U},\mathbb{Z}) \\ &= \arg\min_{X} \left(\sum_i (\|(\mathrm{f}(x_{i-1},u_{i-1}) - x_i)\|^2_{\Sigma^o_i}) + \sum_{j,i} (\|(\mathrm{h}(x_{ij}) - z_j)\|^2_{\Sigma^z_j}) \right)\end{aligned} \tag{4.23}$$

The terms in the above represent the least square optimization problem and are similar to the normal form which can be expressed as:

$$X^* = \arg\min_{X} \|AX - B\|^2_{\Sigma} \tag{4.24}$$

where $A \in \mathbb{R}^{m \times n}$ is the measurement of the first derivative, also called the Jacobian Matrix, evaluated at the linearisation point, X represents the state in the linearization point, B is the measurement and Σ is the covariance matrix.

Here X^* is calculated by minimizing the sum of squared error. Hence, the SLAM problem has been converted to a least square optimization problem. However, in practice, the system are non-linear, hence next section gives an overview of the non-linear least square algorithms, which can solve the above problem.

4.4 Non Linear Least Square Optimization

We have already reduced a SLAM to the least square problem

$$X^* = \arg\min_X \mathrm{F}(x) \tag{4.25}$$

where

$$\mathrm{F}(x) = \frac{1}{2}\mathrm{f}(x)^T\mathrm{f}(x) \tag{4.26}$$

and $\mathrm{f}(x) = (\mathrm{f}_1(x), \mathrm{f}_2(x), \cdots, \mathrm{f}_\mathrm{n}(x))$ are the function values for various x for which we want to minimize the error. If f_i depends on linear value of x the solution is called Linear Least Square Optimization and if it depends on non-linear values of x then it is called Non Linear Least Square Optimizer.

Here only some methodologies to solve the later scenario are introduced. Solvers for such non-linear scenarios generally use iterative approaches. They start from some initial value say x_0 and try to converge to a, hopefully, global optima x^*. This is called "hopefully" because there may exist more than one minima and the one found may not be necessarily the global minima.

Here only four of the many solver available are introduced.

4.4.1 Gradient Descent

A gradient is defined as the first derivative at the point for any differentiable function $\mathrm{f}(x)$. The first derivative of function f is also known as Jacobian or J_f. To find the minima, the direction of the negative gradient is checked. That is, the possible minima lies in the direction where for a small increment or decrement in x, the value of J_f becomes negative. Although a simple method, its convergence to a solution depends on the step size, that is Δx. Hence, it can converge very slowly. For sake of completeness, it is mentioned that, there are some variation which make the Δx dependent on a changing variable, thereby increasing the step size at run time and converging faster. Interested reader can refer to [92] for further details.

4.4.2 Gauss-Newton

The Gauss-Newton method approximates the Newton method. Instead of directly using the Jacobian of the function $\mathrm{f}(x)$, Gauss-Newton solver depends on the approximation of first order Taylor linearisation of $\mathrm{f}(x)$ around x. It solves this linear function explicitly. While approximating the first derivative of $\mathrm{f}(x)$, Newtons method makes use of second derivative, called Hessian H_f. This represents the curvature of $\mathrm{f}(x)$. This means it has higher requirements on the smoothness of f, but it also means that (by using more information) it often converges faster. An interested reader can refer to [92] for further details.

4.4.3 Levenberg-Marquardt

Previously mentioned gradient descent may converge slowly but will reach a solution. Although Gauss-Newton converges faster, if the optimizer is far away from a valid solution, it may fail to converge. However, this method, Levenberg-Marquardt, overcomes both problems by gradually switching between both strategies. It achieves this by an introduction of a parameter λ, called damping factor. By tuning this parameter λ, the algorithm behaves one or the other way. For smaller values, it behaves like the Gauss-Newton and for large values; it has the effect of the Gradient Descent. This automatic switching makes it an optimized solver.

4.4.4 iSAM

iSAM2 or incremental Smoothing and Mapping 2 [93] is the newest incremental version of square root Smoothing and Mapping ($\sqrt{SAM}$) algorithm developed and implemented by Frank Daellaert et. al. [94]. It is an optimized implementation using the sparse matrix representation of the data. It factorizes the Jacobians or the Information Matrix into square root forms and applies them to SLAM problems. As it uses matrix square roots, it is also called square root SAM.

Square Root SAM performs an internal reordering of the variables in the matrix. It uses the data structure of elimination tree to perform the reordering. It then uses Cholskey or QR factorization to calculate the optimal robot states and maps.

The corresponding incremental version updates the Information Matrix and uses full Cholesky factorization. In the newer version, that is, iSAM2 Kaess et. al. [93] use the data structure of Bayes Tree for incremental reordering of variables and achieve an efficient run time state estimation.

An interested reader should read the [93, 94] for detailed explanation of various square root SAM implementations. As iSAM2 supports a run time linearisation, hence it is a good algorithm to perform online SLAM and is well suited for our proposed solutions.

4.5 Summary

This chapter has provided a brief introduction to the concept of SLAM, and introduced the mathematical framework for solving the SLAM problem. It presented graphical methods and presented a connection between SLAM and graphical methods. It then expressed the SLAM formulation as a least square problem and finally gave a small overview of the algorithms to solve non-linear least square problems.

In next chapters, the above-established mathematical concepts are utilized to address the problems highlighted in the introduction.

Chapter 5

Topology Factor

5.1 Motivation

This chapter considers the constraints defined in Chapter 3, namely 1, 2d, 3 and 4:

- We assume measurements from off-the-shelf odometer and GPS are available.
- An externally deployed radar system, which can monitor the vehicles. The configuration of the radar is unknown.
- V2X bi-directional communication is available.
- Lastly, no clutter originates from the radar sensor.

The scenario depicting the above constraints is presented in Figure 5.1. With the given constraints, as explained in Chapter 3, cooperative localization is not feasible to improve the state estimation of vehicles. In this chapter, we propose a solution called *Topology Factor* which addresses the above constraints. These results were published in the conference paper:

1. Gulati et. al.: Vehicle infrastructure cooperative localization using Factor Graphs. 2016 IEEE Intelligent Vehicles Symposium (IV), 2016.

5.2 SLAM Formulation

For the improved state estimation of the vehicles by cooperative localization, as mentioned in the motivation we have access to measurements from the odometer, GPS and the infrastructure radar sensor.

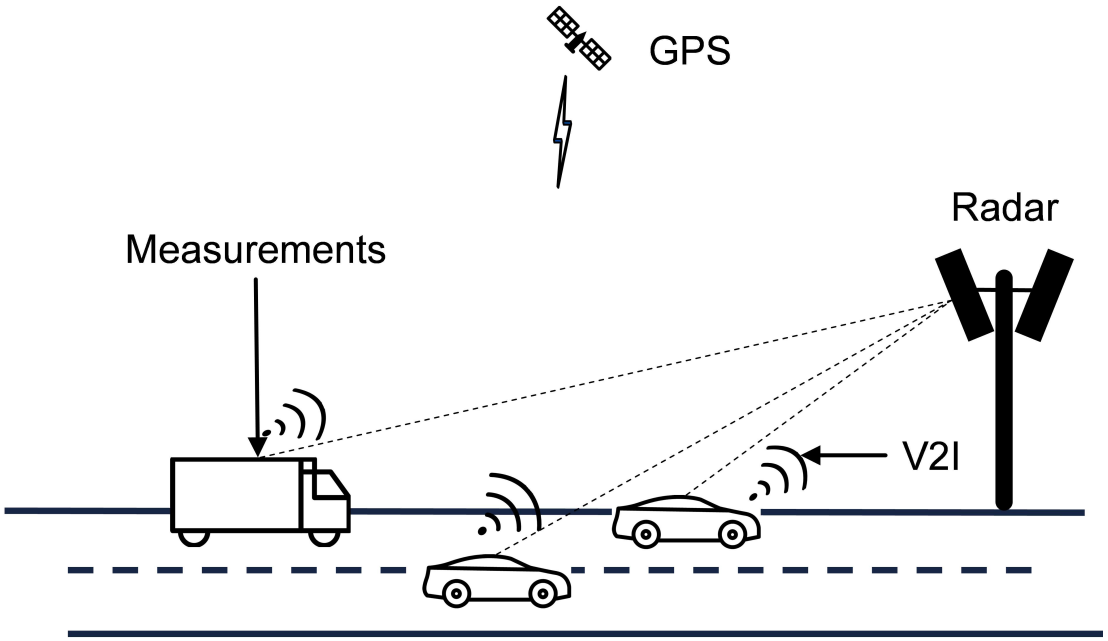


Figure 5.1: The figure highlights the constraints outlined in the motivation. This is further simplification from Figure 3.1 but without considering he V2V communication.

Therefore, Equation 4.23 can be written as:

$$\mathbb{X}^* = \arg\min_X \Bigg(\underbrace{\sum_1^n \|(\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i\|^2_{\Sigma_i^o}}_{\text{Odometer}} + \underbrace{\sum_1^m \|(\mathrm{g}(x_{i_k})) - z_k^g\|^2_{\Sigma_k^g}}_{\text{GPS}} + \underbrace{\sum_1^l \|(\mathrm{r}(x_{i_k})) - z_k^r\|^2_{\Sigma_k^r}}_{\text{Radar}} \Bigg) \tag{5.1}$$

where the mathematical terms describing odometer are same as previously defined. $\mathrm{g}(\cdot)$ and $\mathrm{r}(\cdot)$ are the GPS and radar measurement functions respectively which represent the state of the vehicle which has to be estimated, i.e. the location. z_k^g and z_k^r are the measurements generated from the mapping of true GPS and radar measurements to the state of the vehicle. Σ_k^g and Σ_k^r are the covariance matrices for GPS and radar sensors.

However, since we do not know the configuration of the radar, therefore we do not have the definition $\mathrm{r}(\cdot)$ and Σ_k^r. This chapter proposes a novel method for mapping the radar measurements without the radar configuration, which can be used to improve the state estimation of vehicles in the field of view of the radar using cooperative localization. This new definition of function $\mathrm{r}(\cdot)$ and the corresponding z_k^r can be directly substituted in Equation 5.1.

5.3 Inter-Vehicle Distance

Many researchers (eg. [52, 53, 54, 55, 56, 58, 59]) have used the inter vehicle distance measurements or the Euclidean distances to perform the cooperative localization. The distances were obtained using various radio technologies and fusion depended on identification of the vehicles. An important observation here is that the sensors to measure these Euclidean distance are located on the ego-vehicle. For any three vehicles V_1, V_2, and V_3, denote the inter vehicle distance between any two vehicles V_i and V_j as d_{ij}. If the position of V_1, V_2, and V_3 (in a single dimension) is defined as x_1, x_2 and x_3 respectively, then d_{ij}:

$$\begin{aligned} d_{12} &= \sqrt{(x_1 - x_2)^2} \\ d_{13} &= \sqrt{(x_1 - x_3)^2} \\ d_{23} &= \sqrt{(x_3 - x_2)^2} \end{aligned} \tag{5.2}$$

The d_{ij} is independent of the vehicles involved. Therefore, it can be said:

$$d_{ij} = d_{ji} \tag{5.3}$$

However, to calculate this inter-vehicle distance for any two given vehicles in a group of vehicles, the process of data association needs to be performed. As per our initial assumptions in Section 2, and repeated in the 'Motivation' (5.1) section of this chapter, this is the precise overhead what we would like to avoid.

Therefore, instead of the individual Euclidean distances among the various vehicles, let us consider the sum of inter-vehicle distance and denote it with D. For the above three vehicles it can be written as:

$$D = d_{12} + d_{13} + d_{23} \tag{5.4}$$

As can be seen from the above equation, sum of inter-vehicle distances D is independent of the order in which the distance is calculated. This implies, irrespective of the order of the vehicles considered, as long as all the vehicles are considered, the sum of inter-vehicle distances D will remain the same. Therefore, it is sufficient to have all the measurements of the vehicles to get the correct value of D. **Measurement-to-target association is not required to calculate the sum of inter-vehicle distance.**

The inter-vehicle distance is a scalar value and remains independent irrespective of the coordinate system. That is, the inter-vehicle distance will remain same if calculated in a local coordinate system of a radar or a global coordinate system of a GPS. **The sum of inter-vehicle distance is a scalar value, which is independent of the coordinate system of the measuring sensor**

Therefore, if the radar measurements of multiple vehicles is converted into the sum of inter-vehicle distances and added as a new measurement, the measurement-to-target association and coordinate transformation can be avoided.

If the position of V_i and V_j is defined in two dimensions as (x_i, y_i) and (x_j, y_j), then d_{ij} can be expressed as:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{5.5}$$

Although in Equation 5.2, the *square root* and the *square* can be cancelled, but it has not been done as the process can not be generalized. Additionally, next section presents another reason for not cancelling the same.

5.4 Topology Factor Formulation

The topology factor is defined as a factor between various states of the vehicles which is the function of sum of inter-vehicle distances where the measurements from external radar sensor are used in order to perform the task of cooperative localization.

We call the new factor a *topology factor* because the it originates from the geometric topology which can be a straight line, a triangle, a quadrilateral or any other geometric form.

Using this consideration, Equation 5.4 can be generalized as:

$$D = \sum_{k=1}^{N-1} \sum_{l=k+1}^{N} \sqrt{(x_k - x_l)^2} \tag{5.6}$$

where x_k represents the position of k^{th} vehicle as observed by the infrastructure radar in its local coordinate system.

As explained in Section 4.4 to solve a non-linear equations, we also need to calculate the first derivative, that is the Jacobian of the function. In the above equation, derivative of the function involving the *square root* is more complex as the one without a *square root*. Without loss of generality, instead of d_{12}, d_{13} and d_{23} in the previous section we consider consequently d_{12}^2, d_{13}^2 and d_{23}^2. Therefore, Equation 5.2 becomes:

$$\begin{aligned} d_{12}^2 &= (x_1 - x_2)^2 \\ d_{13}^2 &= (x_1 - x_3)^2 \\ d_{23}^2 &= (x_3 - x_2)^2 \end{aligned} \tag{5.7}$$

Additionally, Equation 5.6 can be then written as:

$$D^2 = \sum_{k=1}^{N-1} \sum_{l=k+1}^{N} (x_k - x_l)^2 \tag{5.8}$$

The properties which we identified in the last section, remain intact for D^2. That is, the task of measurement-to-target association is not required to calculate the sum of **square of** inter-vehicle distance; and the sum of **square of** inter-vehicle distance is a scalar value which remains independent of the coordinate system of the measuring sensor.

Equation 5.8 defines the function $\mathrm{r}(\,\cdot\,)$ in Equation 5.1. Therefore,

$$\begin{aligned} D^2 = \mathrm{r}(\,\cdot\,) &= \mathrm{r}((x_1), (x_2), \cdots, (x_N)) \\ &= \sum_{k=1}^{N-1} \sum_{l=k+1}^{N} (x_k - x_l)^2 \end{aligned} \tag{5.9}$$

Here $r(\,\cdot\,)$ represents the mapping constructed as a "sum of square of inter-vehicle distances" between the true radar measurements and the state of the vehicles observed by the radar.

The solution can be understood from Figure 5.2. The figure highlights the topology factor solution, which forms the geometric topology triangle for three vehicles. Original radar measurements are used to construct the topology factor.

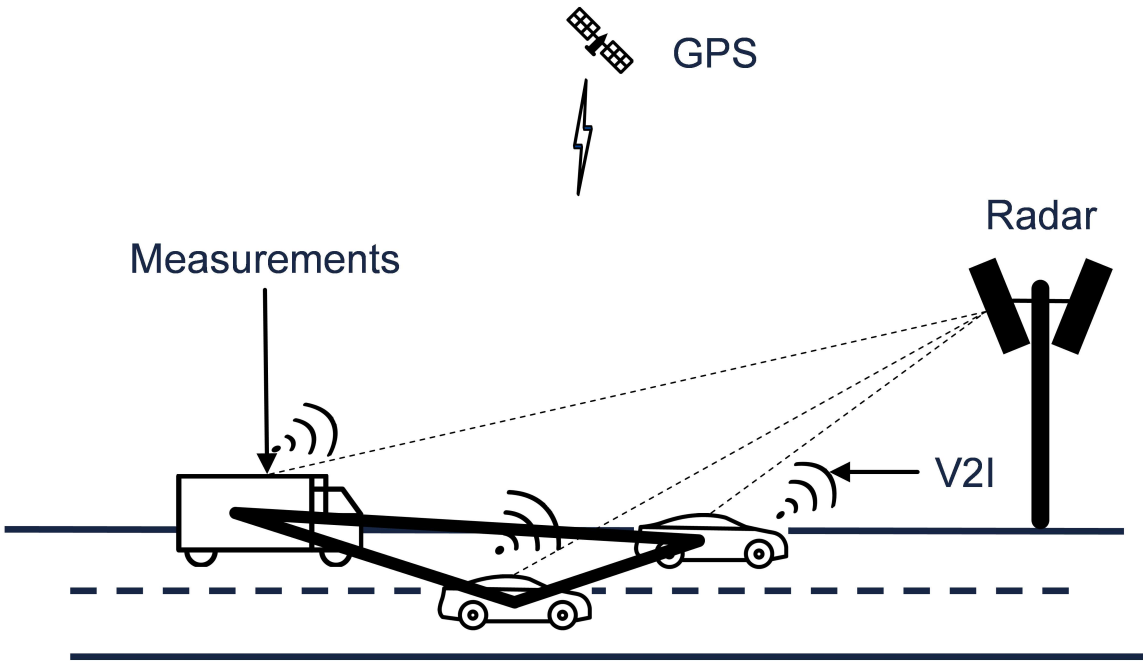


Figure 5.2: The figure highlights the topology factor solution, which forms the geometric topology triangle for three vehicles. The topology factor is constructed using the radar measurements.

5.5 Jacobians for the Non-Linear Least Square Optimization

Section 4.4 highlights the fact that the first derivative of the function is required to numerically solve the non-linear equations, that is the Jacobian of the function. This section derives the Jacobians from the respective equations, required for the optimization process.

5.5.1 Odometer Jacobian

Equation 5.1 defines the mapping of the odometer function to the position as $\mathrm{f}(x_i, u_i)$ at the i^{th} step. Therefore, the Jacobian for the vehicle is obtained by getting a partial derivative with x_i:

$$Jacobian_{odom} = \frac{\partial(\mathrm{f}(x_i, u_i))}{\partial x_i} \tag{5.10}$$

For a constant velocity model, the odometer mapping function $\mathrm{f}(\cdot)$ is a linear function of x. Therefore,

$$Jacobian_{odom} = \frac{\partial(\mathrm{f}(x_i, u_i))}{\partial x_i} = 1 \tag{5.11}$$

For a two dimensional plane, the position is of the form x, y, hence the Jacobian is 2×2 matrix. The first row is calculated with partial derivative with x_i. In the first coloumn only the x terms are used to calculate the partial derivative, treating y as constants. For the second coloumn, only y terms are considered, treating x terms as constants. The steps are repeated for the second row but with partial derivative with y_i. Using these principles, the Jacobian becomes:

$$Jacobian_{odom} = \frac{\partial(\mathrm{f}((x_i, y_i), u_i)}{\partial x_i \partial y_i} = \begin{bmatrix} \frac{\partial(\mathrm{f}((x_i, y_i), u_i))}{\partial x_i} & 0 \\ 0 & \frac{\partial(\mathrm{f}((x_i, y_i), u_i))}{\partial y_i} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.12}$$

5.5.2 GPS Jacobian

Again referring to Equation 5.1, the GPS mapping function from the measurement to the position is defined as $\mathrm{g}(x_i)$ at i^{th} time step. Therefore, like odometer, the Jacobian is obtained by getting the partial derivative with x_i:

$$Jacobian_{gps} = \frac{\partial(\mathrm{g}(x_i))}{\partial x_i} \tag{5.13}$$

A GPS measurement is also a linear function of x. Therefore,

$$\frac{\partial(\mathrm{g}(x_i))}{\partial x_i} = 1 \tag{5.14}$$

Similar to the odometer, the Jacobian for a position (x, y) in a two-dimensional coordinate system can therefore be obtained as:

$$Jacobian_{gps} = \begin{bmatrix} \frac{\partial(\mathrm{g}(x_i,y_i))}{\partial x_i} & 0 \\ 0 & \frac{\partial(\mathrm{g}(x_i,y_i))}{\partial y_i} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.15}$$

5.5.3 Topology Jacobian

Equation 5.9 represents the topology mapping function from the radar measurements. The Jacobian is obtained by getting partial derivative with the position of the all the N participants $x_k,\ \forall k = 1 \cdots N$:

$$Jacobian_{top} = \frac{\partial(\mathrm{r}(x_1, x_2, \cdots, x_N))}{\partial x_1 \partial x_2 \cdots \partial x_N} \tag{5.16}$$

Equation 5.16 is the Jacobian of the whole system. However, the Jacobian for individual participants is required. Therefore, Equation 5.16 for the n^{th} participant can be written as:

$$\begin{aligned} Jacobian^n_{top} &= \frac{\partial(\mathrm{r}(x_1, x_2, \cdots, x_N)))}{\partial x_n} \\ &= \frac{\partial\Big(\sum_{k=1}^{N-1}\sum_{l=k+1}^{N}(x_k - x_l)^2\Big)}{\partial x_n} \\ &= 2 \cdot \sum_{l=1}^{N}(x_n - x_l) \end{aligned} \tag{5.17}$$

Similar to odometer and GPS, for a position x, y in a two-dimensional system, the Jacobian for topology factor is 2×2 matrix. The process of calculating the Jacobian remains same as highlighted for odometer, i.e. treating x and y terms as constants in coloumns 1 and 2 respectively and calculating the partial derivative with x_i for the first row and with y_i for the second. This can be written as:

$$\begin{aligned} Jacobian^n_{top} &= \frac{\partial(\mathrm{r}((x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)))}{\partial x_n \partial y_n} \\ &= \begin{bmatrix} 2 \cdot \sum_{l=1}^{N}(x_n - x_l) & 0 \\ 0 & 2 \cdot \sum_{l=1}^{N}(y_n - y_l) \end{bmatrix} \end{aligned} \tag{5.18}$$

5.6 Measurement Uncertainties / Covariances

A sensor can not measure a 100% accurate value because of the limits of the hardware properties. Therefore, a sensor has a measurement uncertainty. This is represented by a covariance matrix and shows the confidence interval in which the measurement lies. Consider Figure 2.1 where we highlighted how the error in an odometer increases. The covariance for the odometer sensor will specify these confidence intervals and hence approximates the error added to the system as it moves forward. A GPS sensor has an uncertainty on the absolute measurement. Similarly other sensors have their individual uncertainties represented by covariance matrices, often denoted by Σ.

To calculate the optimal state estimate Σs of the participating sensor have to used. As seen, Equation 5.1 requires the covariance matrices Σ_k^o, Σ_k^g and Σ_k^r, for the odometer, GPS and radar factors. This section derives these respective covariances.

5.6.1 Odometer Covariance

An odometer measurement is a direct measurement from the sensor. We assume that the sensor manufacturer, along with the sensor, also provides the technical specifications. These would also contain the covariance matrices for the sensor. Hence, the same can be used as Σ_k^o.

5.6.2 GPS Covariance

Like odometer, a GPS measurement is a direct measurement from the sensor. Therefore, the assumption is that the sensor manufacturer, along with the sensor, also provides the technical specifications. These would also contain the covariance matrices for the sensor. Hence, the same can be used as Σ_k^g.

5.6.3 Topology Covariance

The topology factor uses the measurements from a radar, but unlike odometer and GPS measurements, it is not directly used in the process of cooperative localization. The radar measurement is converted using additional function (see Equation 5.9) and the result is used in the process. Therefore, it should be treated as a new virtual sensor.

For this new virtual sensor, a new covariance matrix is derived using the Error Propagation Law. The law states that, once the measurements from a sensor are subjected to transformations which result in new measurements, the error in the measurements also has to be propagated to the new transformed measurements. Old covariance matrix no longer holds true for the new transformed measurements. As we have transformed the actual radar measurements into topology factors, we also need to transform the radar covariance matrix to the topology covariance matrix. This new error or the covariance matrix can be calculated using this Error Propagation Law. The details for derivation of the Law are present in the Technical Report written by Kai Oliver Arras [95]. In this thesis, we show how the law can be used to calculate the new covariance matrix. The Error Propagation Law states the following:

$$\mathbf{C_{new}} = \mathbf{F} \cdot \mathbf{C_{orig}} \cdot \mathbf{F}^T \tag{5.19}$$

where $\mathbf{C_{orig}}$ is a $n \times n$ and $\mathbf{C_{new}}$ is a $p \times p$ covariance matrix. $\mathbf{F}$ is a matrix of dimension $p \times n$ which consists of elements from the first derivative of the function f($\cdot$) with respect to the participating variables

in the f(·).

This can be understood with an example function f(a, b, c). Let $\mathbf{C_{orig}}$, the original covariance matrix, assuming a, b and c are independent, be defined as follows:

$$\mathbf{C_{orig}} = \begin{bmatrix} \sigma_a^2 & 0 & 0 \\ 0 & \sigma_b^2 & 0 \\ 0 & 0 & \sigma_c^2 \end{bmatrix} \tag{5.20}$$

where σ_a^2, σ_b^2 and σ_c^2 are the original variances of a, b and c respectively.

$\mathbf{F}$ and $\mathbf{F}^T$ for the function f(a, b, c) are then written as:

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} \frac{\partial \mathrm{f}(a,b,c)}{\partial a} & \frac{\partial \mathrm{f}(a,b,c)}{\partial b} & \frac{\partial \mathrm{f}(a,b,c)}{\partial c} \end{bmatrix} \\ \mathbf{F}^T &= \begin{bmatrix} \frac{\partial \mathrm{f}(a,b,c)}{\partial a} \\ \frac{\partial \mathrm{f}(a,b,c)}{\partial b} \\ \frac{\partial \mathrm{f}(a,b,c)}{\partial c} \end{bmatrix} \end{aligned} \tag{5.21}$$

Using Equations 5.19, 5.20 and 5.21, the new covariance $\mathbf{C_{new}}$ of the function f(a, b, c) can then be derived as:

$$\mathbf{C_{new}} = \begin{bmatrix} \frac{\partial \mathrm{f}(a,b,c)}{\partial a} & \frac{\partial \mathrm{f}(a,b,c)}{\partial b} & \frac{\partial \mathrm{f}(a,b,c)}{\partial c} \end{bmatrix} \cdot \begin{bmatrix} \sigma_a^2 & 0 & 0 \\ 0 & \sigma_b^2 & 0 \\ 0 & 0 & \sigma_c^2 \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \mathrm{f}(a,b,c)}{\partial a} \\ \frac{\partial \mathrm{f}(a,b,c)}{\partial b} \\ \frac{\partial \mathrm{f}(a,b,c)}{\partial c} \end{bmatrix} \tag{5.22}$$

Similarly, the covariance matrix for the new topology factor can be calculated using Equations 5.19 and 5.9 . $\mathbf{F}$ and $\mathbf{F}^T$ for the function r$(x_1, x_2, \cdots, x_n))$ can be written as:

$$\begin{aligned} \mathbf{F} =& \begin{bmatrix} \frac{\partial(\mathrm{r}(x_1, x_2, \cdots, x_N))}{\partial x_1} \cdots & \frac{\partial(\mathrm{r}(x_1, x_2, \cdots, x_N))}{\partial x_N} \end{bmatrix} \\ =& \begin{bmatrix} 2 \cdot \sum_{k=1}^{N} (x_1 - x_k) \cdots & 2 \cdot \sum_{k=1}^{N} (x_N - x_k) \end{bmatrix} \end{aligned} \tag{5.23}$$

$$\mathbf{F}^T = \begin{bmatrix} \frac{\partial(\mathrm{r}(x_1, x_2, \cdots, x_N))}{\partial x_1} \\ \vdots \\ \frac{\partial(r(x_1, x_2, \cdots, x_N))}{\partial x_N} \end{bmatrix} = \begin{bmatrix} 2 \cdot \sum_{k=1}^{N} (x_1 - x_k) \\ \vdots \\ 2 \cdot \sum_{k=1}^{N} (x_N - x_k) \end{bmatrix} \tag{5.24}$$

If σ_x^2 is the variance for the radar for x, then the covariance matrix, assuming $\sigma_{x_k}^2$ for all x_k are same and independent, for N participating targets detected by the infrastructure radar can be written as the following $N \times N$ matrix:

$$\mathbf{C_{orig}} = \begin{bmatrix} \sigma_{x_1}^2 & 0 & \cdots & 0 \\ 0 & \sigma_{x_2}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{x_N}^2 \end{bmatrix} = \begin{bmatrix} \sigma_x^2 & 0 & \cdots & 0 \\ 0 & \sigma_x^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_x^2 \end{bmatrix} \tag{5.25}$$

Substituting from Equations 5.23, 5.24 and 5.25 in Equation 5.19, the derived covariance is:

$$\begin{aligned}\Sigma^r =& \Big[2 \cdot \sum_{k=1}^{N} (x_1 - x_k) \quad \cdots \quad 2 \cdot \sum_{k=1}^{N} (x_N - x_k)\Big] \cdot \\ & \begin{bmatrix} \sigma_x^2 & 0 & \cdots & 0 \\ 0 & \sigma_x^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_x^2 \end{bmatrix} \cdot \begin{bmatrix} 2 \cdot \sum_{k=1}^{N} (x_1 - x_k) \\ \vdots \\ 2 \cdot \sum_{k=1}^{N} (x_N - x_k) \end{bmatrix} \\ =& 4 \cdot \Big(\sum_{k=1}^{N} (x_1 - x_k) \Big) \cdot \sigma_x^2 + \cdots + 4 \cdot \Big(\sum_{k=1}^{N} (x_N - x_k) \Big) \cdot \sigma_x^2 \end{aligned} \tag{5.26}$$

For a position (x, y) in a two-dimensional plane, where σ_x^2 and σ_y^2 are the variances for the radar for x and y, assuming x and y are independent, the covariance of the topology factor can be extended as follows:

$$\begin{aligned}\Sigma^r =& 4 \cdot \Big(\sum_{k=1}^{N} (x_1 - x_k) \Big) \cdot \sigma_x^2 + 4 \cdot \Big(\sum_{j=1}^{N} (y_1 - y_k) \Big) \cdot \sigma_y^2 + \cdots \\ & 4 \cdot \Big(\sum_{k=1}^{N} (x_N - x_k) \Big) \cdot \sigma_x^2 + 4 \cdot \Big(\sum_{k=1}^{N} (y_N - y_k) \Big) \cdot \sigma_y^2 \end{aligned} \tag{5.27}$$

When compared to covariance in Equation 5.26, Equation 5.27 adds the y terms and results in a covariance for a topology factor for a two dimensional plane. Note that the covariance here is a scalar value of a one dimension and not a matrix, as the sum of Euclidean distances of the participating targets is also a scalar value.

5.7 Evaluation

5.7.1 System Setup

The approach presented in the previous section has been evaluated on a simulated data set, which is based on real radar observations. The simulation was implemented with up to four vehicles on a highway for 200 steps.

The simulated vehicles are assumed to have an odometer sensor to measure their relative distance travelled per time unit. They are also assumed to have a GPS sensor to measure the location of the simulated vehicle in the global coordinate system. The simulated highway is equipped with an infrastructure radar sensor, which can observe all the vehicles moving in one direction. As said in the initial requirements in Chapter 3, point2d and again in Section 5.1, there is no configuration information of the radar available, i.e. we do not know the location of the radar, therefore we do not have any transformation information for converting the measurements in the radar coordinate system to the vehicle coordinate system.

All the simulated sensors have zero mean Gaussian Noises. The covariances assumed are $diag\,[1.0, 1.0]$, $diag\,[10.0, 10.0]$ and $diag\,[0.1, 0.1]$ for the odometer, GPS and radar respectively. The time step interval is taken as 1 *sec*.

For topology factor two sets of simulations, Constant Velocity Model and Constant Turn Model, are performed. The results for each of them are compared between:

1. the fused trajectory only using odometer and GPS sensors.

2. the fused trajectory using odometer, GPS and topology factor. Topology factor are constructed using radar measurements.

5.7.2 Root Mean Square Error

The performance for the approach is calculated using *Root Mean Square Error* (RMSE) for the complete system. The *Mean Square Error* (MSE) is a measure, which shows how close the predicted line to the measured data points is. It is calculated by adding the square of the vertical distance of the data point to the fitted curve and dividing by n. The result is squared to avoid the influence of negative values. The smaller the MSE is, the closer fit is the estimated line to the data points. Mathematically:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{DataPoint}})^2$$

The Root Mean square Error (RMSE) is square root of the MSE. Hence, RMSE is the distance of the data point from the predicted line, which is measured along a vertical line. It is directly and easily interpretable in terms of measurement units and hence is a good measure to compare the performance between various estimation algorithms. RMSE, when we have the available ground truth, can be written as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{GroundTruth}})^2} \tag{5.28}$$

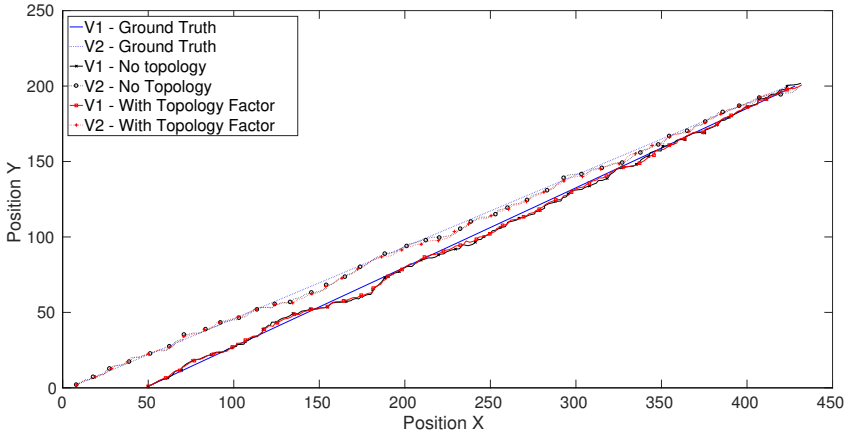
5.7.3 Simulation Results

Figures 5.3(a), 5.3(b) and 5.3(c) show the trajectories for 2, 3 and 4 vehicle system respectively. One can see that trajectories resulting from the use of topology factor are closer to the ground truth than the ones resulting without the topology factor.

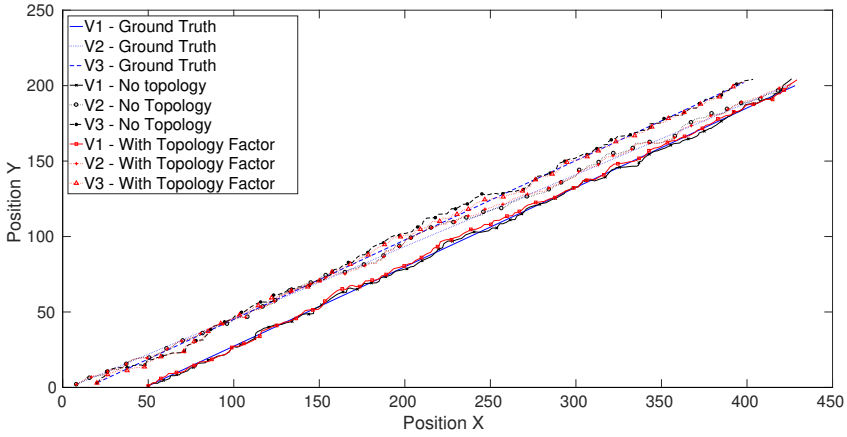
As the number of vehicles increase the trajectories become difficult to distinguish. Therefore, the total error in the system is analysed by studying the RMSE at each step. Figures 5.4(a), 5.4(b) and 5.4(c) show the corresponding RMSEs for 2, 3 and 4 vehicle systems respectively. It can be clearly seen that adding topology factor decreases the error in the system.

To show the stability of the proposed solution, we run a Monte Carlo simulation of 1000 iterations of above mentioned scenario for 2, 3 and 4 vehicles. Table 5.1 shows the average RMSE values for 1000 iterations for simulated trajectories as shown in Figure 5.3. It can be clearly seen that the simulation with the topology factor performs better than the one without the topology factor.

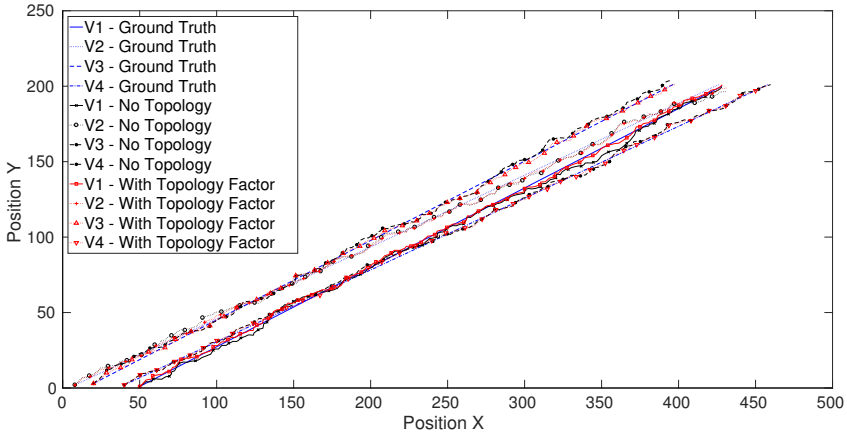
The topology factor is further tested for vehicles having a turning trajectory. Figures 5.5(a), 5.5(b) and 5.5(c) shows the trajectories for 2, 3 and 4 vehicle system travelling on a curve, respectively. Similar to the linear trajectories, trajectories resulting from use of topology factor, for the vehicles travelling on a curve, are closer to the ground truth trajectories than the ones resulting without the topology factor.



(a)

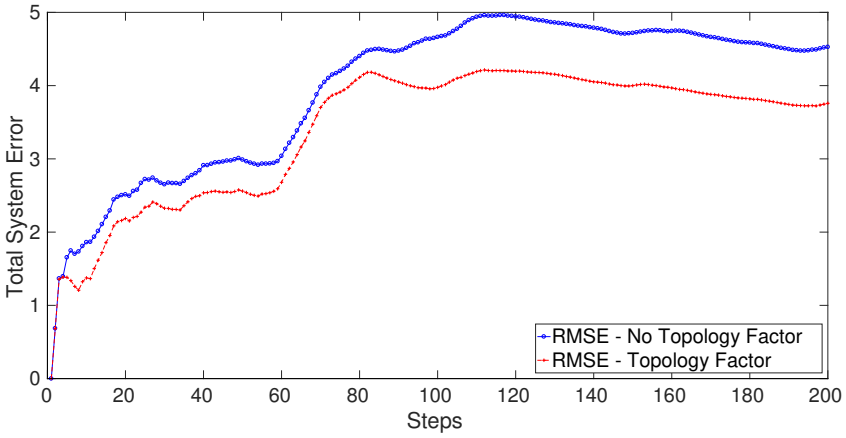


(b)

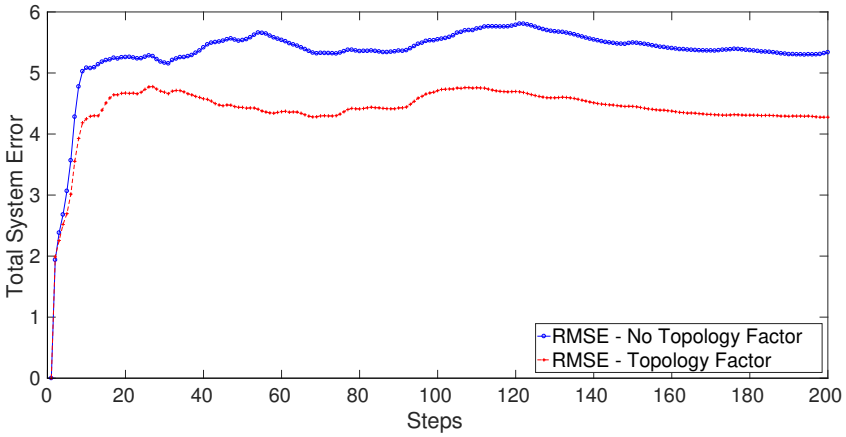


(c)

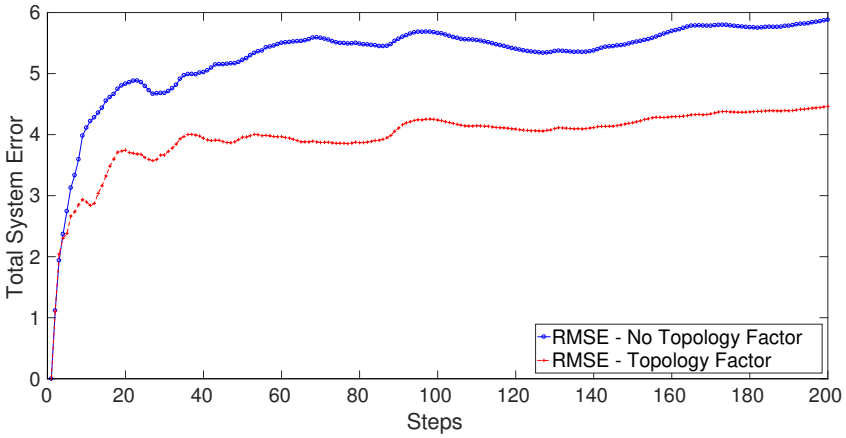
Figure 5.3: Ground truth; Fused trajectories without topology factor and with topology factor for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)

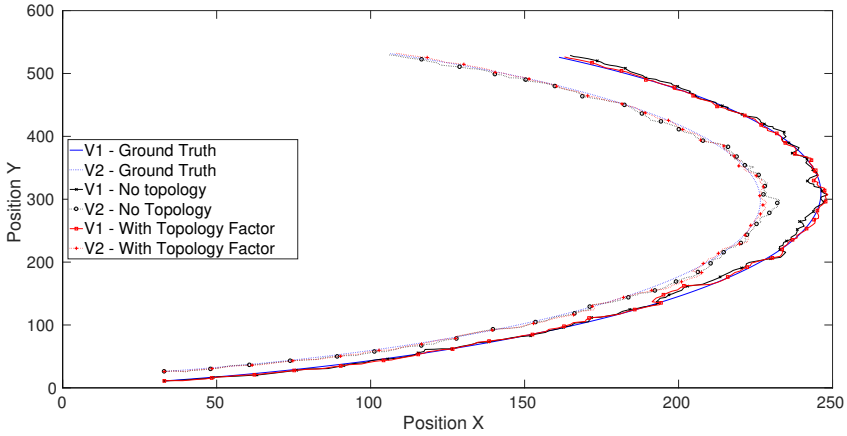


(b)

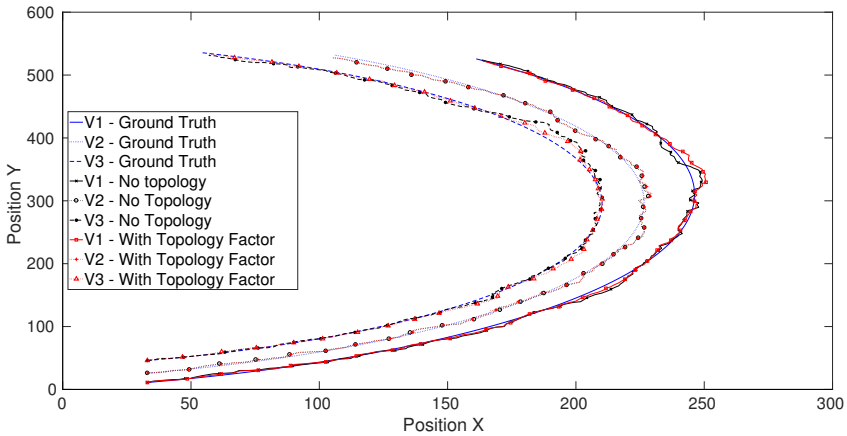


(c)

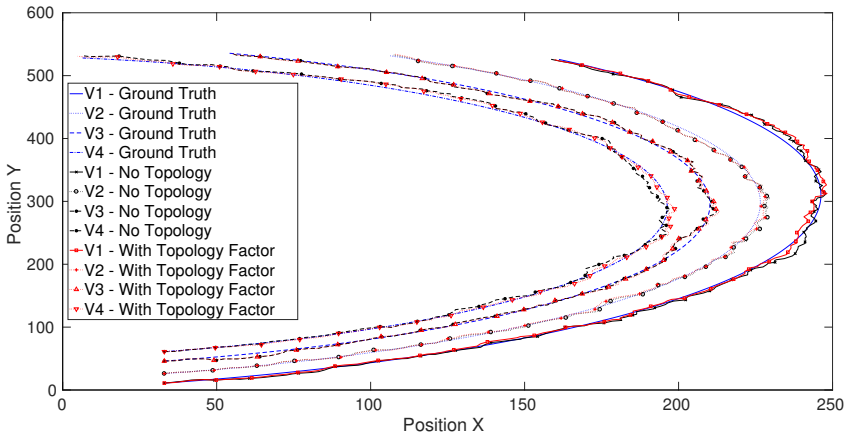
Figure 5.4: Total System RMSE for trajectories shown in Figure 5.3 for (a) 2 vehicles (Figure 5.3(a)). (b) 3 vehicles (Figure 5.3(b)). (c) 4 vehicles (Figure 5.3(c)).



(a)

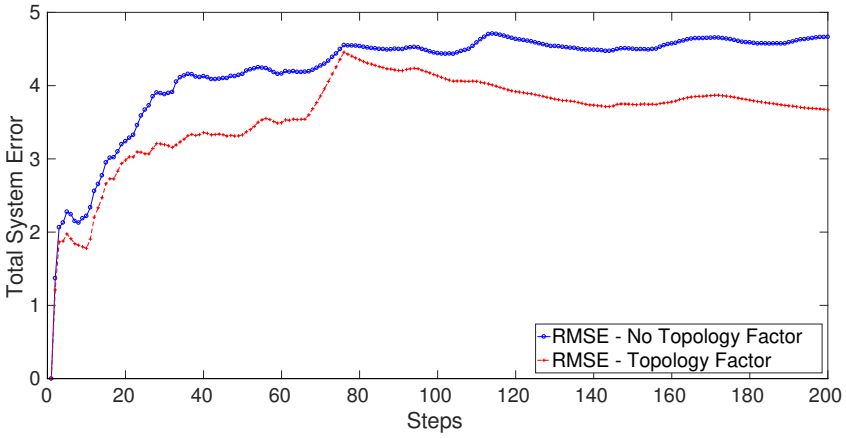


(b)

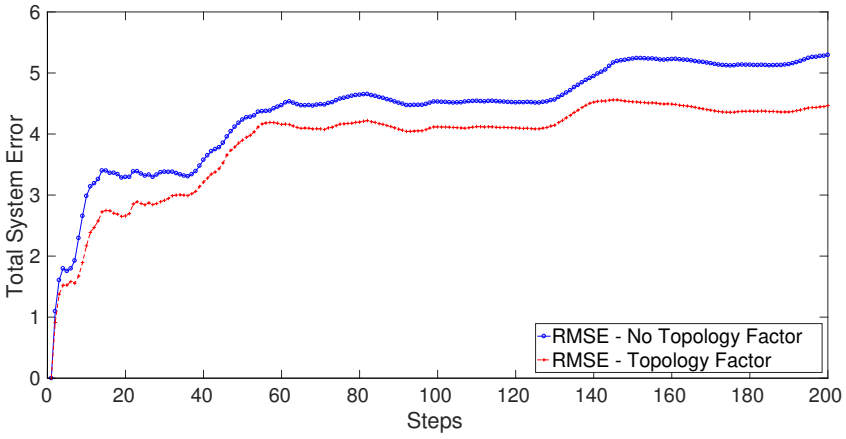


(c)

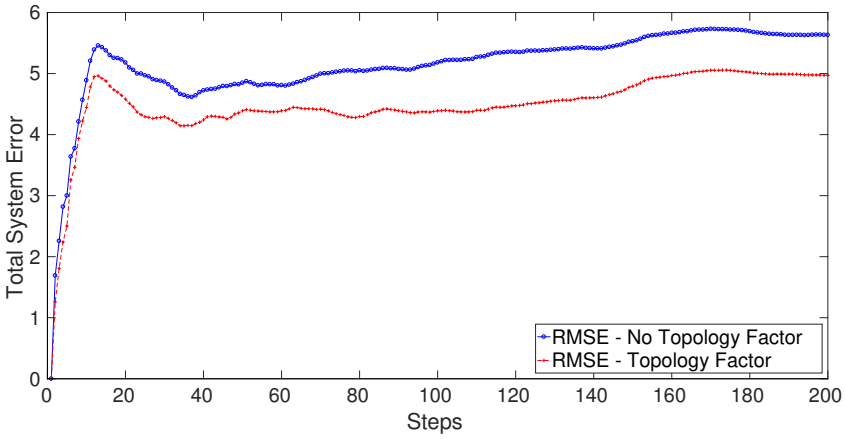
Figure 5.5: Ground truth; Fused trajectories without topology factor, with topology factor for constant turn model for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)



(b)



(c)

Figure 5.6: Total System RMSE for trajectories shown in Figure 5.5 for (a) 2 vehicles (Figure 5.5(a)). (b) 3 vehicles (Figure 5.5(b)). (c) 4 vehicles (Figure 5.5(c)).

Table 5.1: Average total system RMSE for 1000 iterations for 2, 3 and 4 vehicles.

Vehicles	No Topology Factor	With Topology Factor
2	4.257	3.202
3	5.224	4.166
4	5.997	4.750

To overcome any confusion in comparing the trajectories, the respective RMSEs are also plotted in Figures 5.6(a), 5.6(b) and 5.6(c). Similar to the linear trajectories, the RMSEs clearly show the reduction in error with the use of proposed topology factor.

The stability of the proposed solution on a curved trajectory is also evaluated using Monte Carlo simulation of 1000 iterations. The resulting average RMSE are shown in Table 5.2.

Table 5.2: Average total system RMSE for 1000 iterations for 2, 3 and 4 vehicles on a curved trajectory.

Vehicles	No Topology Factor	With Topology Factor
2	4.502	3.259
3	5.532	4.556
4	6.316	5.429

Average RMSE values for curved trajectories in Table 5.2 show similar results as linear trajectory in Table 5.1. Trajectories with topology factors have lower error than the ones without topology factor.

Tables 5.1 and 5.2 show the total system RMSE of the system. The actual RMSE per vehicle is lower. It is not difficult to see that with an increase in the number of participants, the RMSE per vehicle decreases. This is because the combined constraints of all the factors (Odometer, GPS and Topology) (for an increase in the number of participants) is stronger and hence forces the optimizer to strive for the global minima instead of the first found solution.

5.8 Summary

The topology factor presented in this section, provides a novel mechanism to fuse the external radar measurements with internal sensors for multi-target scenario without performing any measurement-to-target association. The measurement in the local radar coordinates can be used to formulate the topology factor. The non-linear least square algorithm, used for state estimation, is able to handle an increase in the number of vehicles. Hence, the solution remains scalable. Further we only need to transmit the new measurements and not the covariances, and the original covariance of the radar is only required to be transmitted only once for the fusion. This keeps the bandwidth requirement to the minimum.

Although we have used the offline algorithm of Levenberg-Marquardt for the final state estimation, we can also use the iSAM2 [93] algorithm of GTSAM to perform an online state estimation of the vehicles.

Therefore, we are able to meet 5 of the 6 challenges highlighted in Section 1.3. Hence, the topology factor addresses the challenges of cooperative localization in an environment where the radar does not report any clutter measurements.

Chapter 6

SME Factor

6.1 Motivation

The previous chapter introduced the novel concept of topology factor, which solves some of the major challenges encountered in cooperative localization. The availability of three sensors was assumed: odometer, GPS from the vehicle, and a high precision infrastructure radar. Improved state estimation was achieved using the topology factor.

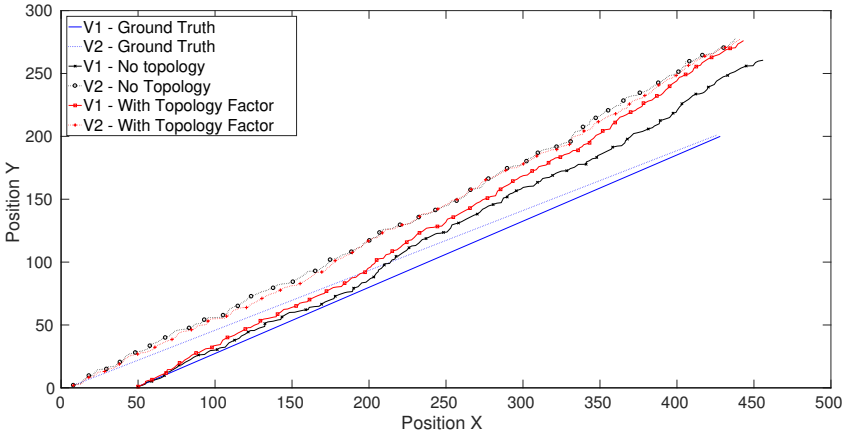
The vehicle odometer data is available as long as the vehicle is moving, but in environments like tunnels and underground parking garages, the direct line of sight to the GPS Satellites is obstructed, thereby the GPS measurements are not available. Further, in city scenarios among high-rise buildings, situation of urban canyons can be created where not enough satellites are available to form a reliable or precise position estimate. Under these circumstances, cooperative localization is reduced to odometer and infrastructure radar. In such cases, the measurements from the radar can be incorporated as topology factor for improved state estimation.

6.1.1 Topology Factor in absence of GPS

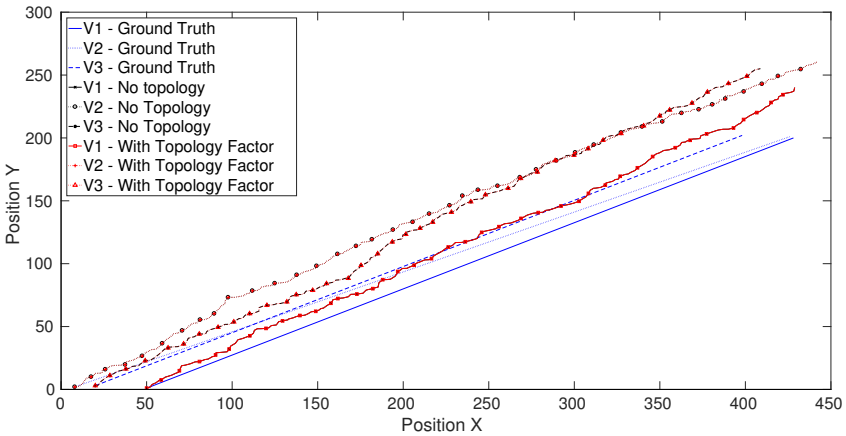
Simulation using cooperative localization (as presented in Figure 5.3) is rerun without the GPS measurements to simulate a GPS devoid scenario. The results are presented in Figure 6.1. Figures 6.1(a), 6.1(b) and 6.1(c) show the trajectories for 2, 3, and 4 vehicles respectively. It can be seen that the trajectories without using topology factor, drift away from the ground truth. With the addition of the topology factor, the fused trajectory still drifts away from the ground truth. "Drifting away" of the calculated trajectories in comparison with the ground truth implies the system accumulates more errors. This is confirmed by the corresponding total system RMSE values plotted in Figure 6.2. Figures 6.2(a), 6.2(b) and 6.2(c) show the RMSE for 2, 3 and 4 vehicles respectively. It can be seen that the error continuously increases and the solution does not converge.

6.1.2 Reasons for Degraded Performance of Topology Factor

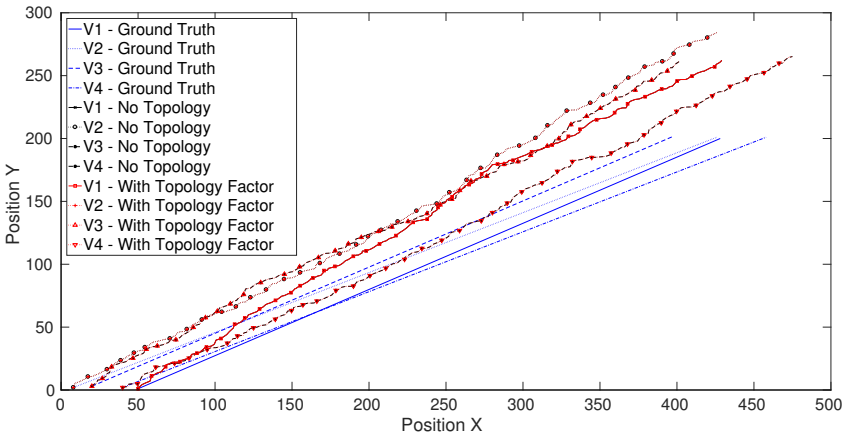
A GPS is an absolute measurement. In the presence of GPS sensor data, the topology factor is constrained by the GPS measurements; hence, it contributes to the information correctly and improves the state



(a)

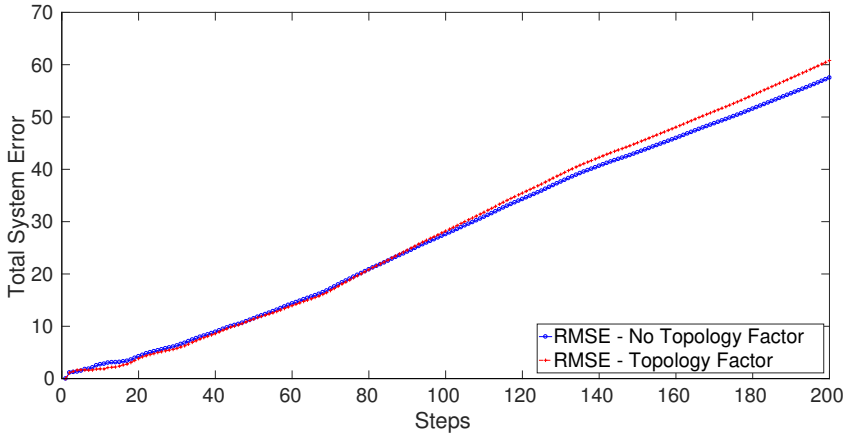


(b)

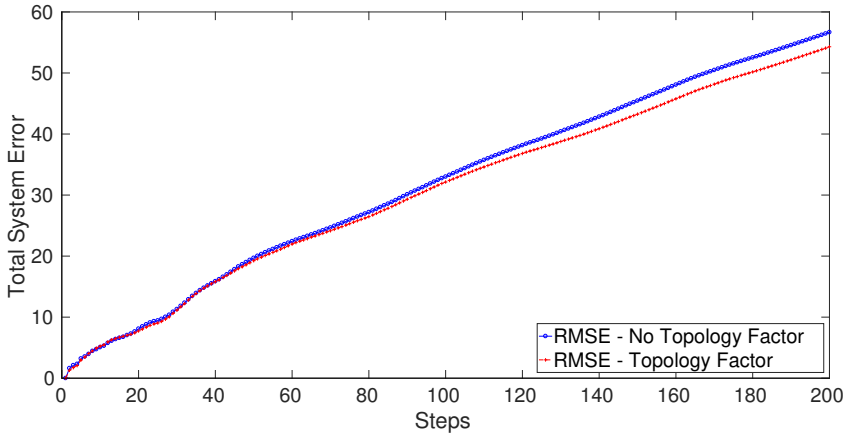


(c)

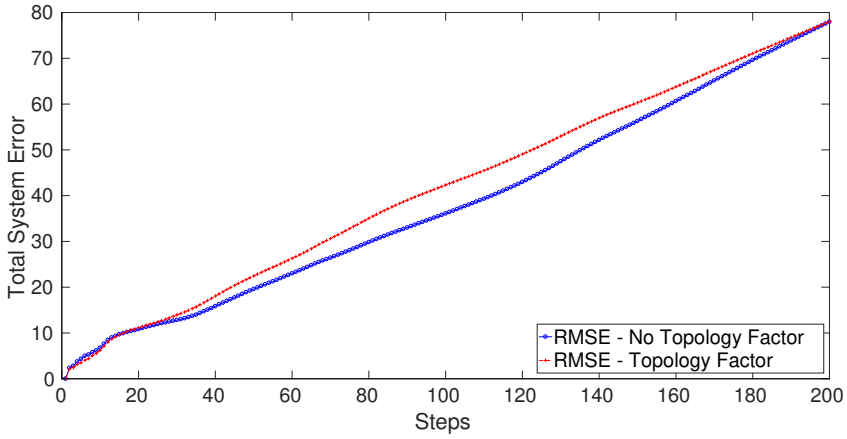
Figure 6.1: Ground truth; Fused trajectories without topology factor and with topology factor in **GPS devoid** areas for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)



(b)



(c)

Figure 6.2: Total System RMSE trajectories shown in figure 6.1 for (a) 2 vehicles (fig 6.1(a)). (b) 3 vehicles (fig 6.1(b)). (c) 4 vehicles (fig 6.1(c)).

estimation.

In absence of GPS measurements, for the case without a topology factor, only the odometer is used. An odometer has an error in relative measurement. Therefore, the error will continue to increase. Under normal circumstances, fusing the data from another high precision sensor (an infrastructure radar) would lower the error in the state estimation. However, when the radar measurements are added as topology factors, it does not achieve the desired results.

The topology factor is formulated using the sum of inter-vehicle distance: the information of all the targets actual locations is lost. Graphically, topology measurement is a 1-dimensional scalar value, without any location information associated with it. Therefore, it can be plotted anywhere in a 2-dimensional plane.

Mathematically, topology calculation is not an injective function, i.e. various sets of target measurements can result in the same sum of Euclidean distances. Therefore, it needs further information to correctly contribute to the final state estimation. GPS measurements provided an anchor which helped it to converge the states to a correct solution, but in GPS devoid areas this support is lost.

Therefore, although the information provided by the topology factor is still of the same precision, but the optimizer fails to utilize the additional information provided by the high precision sensor and we do not see the expected improvement in the state estimates.

From the RMSE plots in Figure 6.2(a), for 2 vehicles, before 20 time steps the topology factor does perform a bit better without any topology factor. However, the performance quickly becomes similar to the case when system does not have any topology factor. At the end after 120 time steps, the system without topology factors performs better than the system with topology factors. Similar fluctuations in performance can be observed for 3 and 4 vehicles as well. Overall a 1-dimensional value of sum of inter-vehicle distance without an anchor point (supplied by GPS measurements) is not sufficient and the optimization process is driven by the odometer measurements and hence we see the RMSE graphs are similar for both: with and without the topology factor.

6.1.3 Scenario Description

The motivation of the research work presented in this chapter is to perform the task of cooperative localization even in GPS devoid areas. Therefore, the constraints presented in Chapter 5 Section 5.1 for topology factor change slightly:

- We assume measurements from off-the-shelf odometer. **GPS measurements may or may not be available**.
- An externally deployed radar system can monitor the vehicles. The configuration of the radar is unknown.
- V2X bi-directional communication is available.
- Lastly, no clutter originates from the radar sensor.

The scenario depicting the above constraints is summarized in Figure 6.3. In absence of GPS measurements, the topology factor is unable to add sufficient information to improve the state estimation. With the above mentioned constraints as explained in Chapter 3, the previously proposed solution of

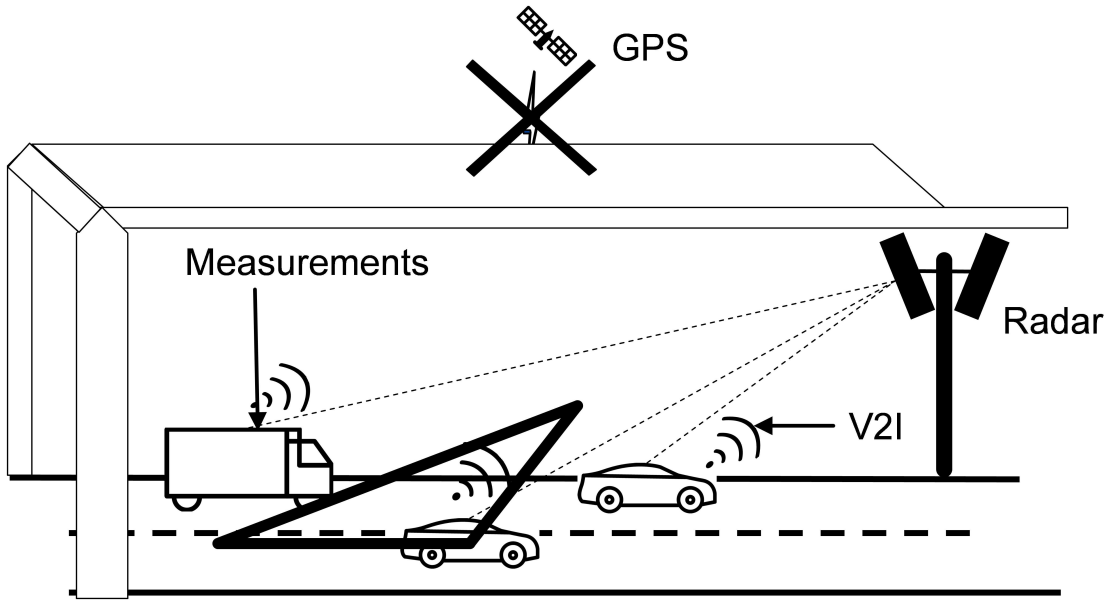


Figure 6.3: The figure highlights the constraints and set up outlined in the motivation. In absence of GPS the topology factor, being a scaler uni-dimensional constraint, is unable to add sufficient information for improved state estimation.

topology factor does not help with cooperative localization. In this chapter a solution called *SME factor* is proposed, which successfully performs cooperative localization even in absence of the GPS sensor data. These results were published in the Sensors Journal:

1. Gulati et. al.: Graph-Based Cooperative Localization Using Symmetric Measurement Equations. 2017 MDPI Sensors, 2017.

The above scenario of GPS devoid areas can be broken in multiple scenarios: (1) In scenarios like urban canyons, the GPS may be available but with lower accuracy (2) In tunnels, GPS is not available. For our experimentation and simulation, for GPS devoid areas we assume no GPS signal is available.

6.2 SLAM Formulation

For the SME factor, when measurement from odometer, GPS and radar sensor are available, the formulation remains same as that of the topology factor in Equation 5.1. However, for GPS devoid areas the SLAM formulation changes as follows:

$$\mathbb{X}^* = \arg\min_X \left(\underbrace{\sum_1^n \|(\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i\|^2_{\Sigma_i^o}}_{\text{Odometer}} + \underbrace{\sum_1^l \|(\mathrm{r}(x_{i_k})) - z_k^r\|^2_{\Sigma_k^r}}_{\text{Radar}} \right) \tag{6.1}$$

the mathematical terms describing the odometer remain the same. $\mathrm{r}(\cdot)$ is the radar functions which maps the state of the vehicle (which has to be estimated, i.e. the location) to the sensor measurements.

z_k^r is the measurement generated from the mapping of the radar measurements to the state of the vehicle. Σ_k^r is the covariance matrix for the radar sensor.

Similar to the topology factor, the configuration of the radar is unknown, therefore we do not have the definition $\mathrm{r}(\cdot)$ and Σ_k^r. This chapter proposes a novel method for mapping the radar measurements without the radar configuration, which can be used to improve the state estimation of vehicles in the field of view of the radar using cooperative localization. This new definition of function $\mathrm{r}(\cdot)$ and the corresponding z_k^r can be directly substituted in Equations 5.1 and 6.1 and which keeps the degradation of state estimation in check in absence of GPS sensor data. Like topology factor, the new solution continues to avoid the problem of Data Association.

6.3 Symmetric Measurement Equations

Kamen (see [96]) introduced a novel filter called Symmetric Measurement Equations (SME) Filter specifically to avoid the measurement-to-track association. The basis of the SME Filter is:

"to convert the measurement data (with associations not known) into a measurement equation by defining new measurements that are symmetric functions of the original measurements. In this way the target states can be estimated without ever considering the associations between targets and measurements." [97]

Therefore, the approach maps a finite set of random measurements (as the target associations is not known) generated by multiple targets into a random vector using a non-linear function. The process generates *pseudo-measurements* from the original measurements. The important aspect of this mapping is that it should be symmetric, in that it is only dependent on the number of measurements and independent of the order of the measurements.

6.3.1 SME Example

The concept of SME can be understood by considering a case for 2 targets, t_1 and t_2 moving in 1 dimension. Assuming no clutter, a radar sensor observing the targets will result in 2 measurements, m_1 and m_2. The measurement-to-target association is unknown, therefore it is not possible to predict which measurement belongs to which target. Using appropriate symmetric functions, the original measurements can be converted to pseudo-measurements pm_1 and pm_2.

One example of such a set of Symmetric Functions which can result in pm_1 and pm_2 is Sum-of-Products [96], which is written as:

$$\begin{aligned} pm_1 &= m_1 + m_2 &&= \mathrm{f}_1(m_1, m_2) \\ pm_2 &= m_1 \cdot m_2 &&= \mathrm{f}_2(m_1, m_2) \end{aligned} \tag{6.2}$$

Similarly for a 3 target system in 1 dimension, Sum-of-Products can be written as:

$$\begin{aligned} pm_1 &= m_1 + m_2 + m_3 &&= \mathrm{f}_1(m_1, m_2, m_3) \\ pm_2 &= m_1 \cdot m_2 + m_2 \cdot m_3 + m_3 \cdot m_1 &&= \mathrm{f}_2(m_1, m_2, m_3) \\ pm_3 &= m_1 \cdot m_2 \cdot m_3 &&= \mathrm{f}_3(m_1, m_2, m_3) \end{aligned} \tag{6.3}$$

Another example of such a set of Symmetric Functions which can result in pm_1 and pm_2 is Sum-of-

Powers [96], which is written as.

$$\begin{aligned} pm_1 = m_1 + m_2 &= \mathrm{f}_1(m_1, m_2) \\ pm_2 = m_1^2 + m_2^2 &= \mathrm{f}_2(m_1, m_2) \end{aligned} \tag{6.4}$$

Similarly for a 3 target system in 1 dimension, Sum-of-Powers can be written as:

$$\begin{aligned} pm_1 &= m_1 + m_2 + m_3 = \mathrm{f}_1(m_1, m_2, m_3) \\ pm_2 &= m_1^2 + m_2^2 + m_3^2 = \mathrm{f}_2(m_1, m_2, m_3) \\ pm_3 &= m_1^3 + m_2^3 + m_3^2 = \mathrm{f}_3(m_1, m_2, m_3) \end{aligned} \tag{6.5}$$

It can be seen that the true measurements are *distributed* over the pseudo-measurements, thereby avoiding the need of data association. Further, since the mapping is real and symmetric, i.e. we have same number of variables as equations/functions, therefore the original measurements can be uniquely recovered.

6.3.2 Generalization

From examples presented in the previous subsection, it can be seen that to convert N measurements into N pseudo-measurements, we need total of N symmetric functions, each of which takes the input of the N measurements and results in 1 pseudo-measurement.

For a N targets system, Sum-of-Products can be generalized as:

$$\begin{aligned} pm_1 &= m_1 + m_2 + \cdots + m_N = \sum_{k=1}^{n} m_k = \mathrm{f}_1(m_1, \cdots, m_N) \\ pm_2 &= m_1 \cdot m_2 + m_2 \cdot m_3 + \cdots + m_{N-1} \cdot m_N = \sum_{\substack{k_1 \\ 1 \le k_1 < k_2 \le N}} \sum_{k_2} m_{k_1} m_{k_2} = \mathrm{f}_2(m_1, \cdots, m_N) \\ &\vdots \\ pm_n &= \sum_{\substack{k_1 \\ 1 \le k_1 < k_2 \cdots k_n \le N}} \sum_{k_2} \cdots \sum_{k_n} m_{k_1} m_{k_2} \cdots m_{k_n} = \mathrm{f}_\mathrm{n}(m_1, \cdots, m_N) \\ &\vdots \\ pm_N &= m_1 \cdot m_2 \cdots m_N = \mathrm{f}_\mathrm{N}(m_1, \cdots, m_N) \end{aligned} \tag{6.6}$$

Likewise for a N targets system, Sum-of-Powers can be generalized as:

$$\begin{aligned}
pm_1 &= m_1 + m_2 + \cdots + m_N = \sum_{k=1}^{n} m_k = \mathrm{f_1}(m_1, \cdots, m_N) \\
pm_2 &= m_1^2 + m_2^2 \cdots + m_n^2 = \sum_{k=1}^{N} m_k^2 = \mathrm{f_2}(m_1, \cdots, m_N) \\
&\vdots \\
pm_n &= \sum_{k=1}^{N} m_k^n = \mathrm{f_n}(m_1, \cdots, m_N) \\
&\vdots \\
pm_N &= \sum_{k=1}^{N} m_k^N = \mathrm{f_N}(m_1, \cdots, m_N)
\end{aligned} \tag{6.7}$$

The above two kinds of symmetric transformations, i.e. Sum-Of-Products and Sum-Of-Power provide a mechanism to convert the actual measurements to the pseudo-measurements. These new pseudo-measurements are independent of the targets and hence allow to avoid data association.

Important point to note here is the degree of equations increase with an increase in the number of participants. In case of moving traffic (i.e. not a traffic jam) the value N may go upto 30 vehicles.

6.4 SME Factor Formulation

The topology factor was formulated as the sum of the inter-vehicle distance, which is a scalar value and hence, independent of any coordinate system. However, as seen in the above equations, absolute radar measurements have to be used as parameters to the Symmetric Function. Therefore, any factor formulation, using the Symmetric Functions is to be in the same coordinate system as other sensor measurements in the factor graph. Therefore, to use the Symmetric Functions, coordinate have to be transformed.

For topology factor, we used the world coordinate system or the global coordinate system to uniquely identify the positions for the vehicles. Odometer are relative measurements, therefore do not influence the coordinate system of the formulated graph. On the other hand, radar has its local coordinate system, but it does not help identify the unique position in the world. Therefore to uniquely identify the location of vehicles in the world we choose the transformation of local radar coordinates into standard global coordinate system. This has additional advantage when GPS is intermittently available, then we do not have to perform another coordinate transformation.

Within the context of the scenarios presented herein, if radar has to transform its measurements in global coordinate system, it also needs to know its own position in global coordinate system. Therefore we have to relax the condition that infrastructure radar does not know its configuration. We still do not require to know its precise deployment configuration in terms of heigh and angle of field of view but only its absolute location in global coordinate system. With this, converting the local radar coordinate system to global coordinate system becomes a simple process, which can be implemented in the hardware. This coordinate transformation does not add significant overhead on the cooperative state estimation process.

Using the transformed radar measurements (from local to global coordinate), a new SME factor is proposed in this chapter. The proposed SME factor uses Sum-of-Power as a Symmetric Function and avoids the measurement-to-track association, as it is included within the symmetric measurement formulation. For N detected targets, there will be N different definitions of $\mathrm{r_n}(\cdot)\ \forall n = 1 \cdots N$. Equation 6.7 define the functions $\mathrm{r_n}(\cdot)$ which is used in Equations 5.1 or 6.1. Mathematically,

$$\begin{aligned} R &= \{\mathrm{r_n}(\cdot) \mid \forall n = 1 \cdots N\} \\ &\text{where,} \\ \mathrm{r_n}(\cdot) &= \mathrm{f_n}(x_1,\ \cdots,\ x_N) = \sum_{k=1}^{N} x_k^n \end{aligned} \tag{6.8}$$

where x_k is the k^{th} measurement by the radar. Here $\mathrm{r_n}(\cdot)$ represents the mapping constructed as a sum-of-power using the actual radar measurements, between the state of the vehicles observed by the radar.

6.5 Jacobians for the Non-Linear Least Square Optimization

Similar to the topology factor, the proposed SME factor also requires the first derivative, i.e. the Jacobians, of the functions to numerically solve the non-linear equations. This section derives the Jacobians from the respective equations required for the optimization process.

Odometer Factor Jacobian: The odometer Jacobian remains the same as Equation 5.12, that is:

$$Jacobian_{odom} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.9}$$

GPS Factor Jacobian: GPS measurements may be intermittently available. Moreover, we simulate a tunnel scenario as part of our experiments to analyse the benefits of SME factor. Therefore, when GPS measurements are available, the corresponding factor is formulated and it requires the same Jacobians as Equation 5.15.

$$Jacobian_{gps} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.10}$$

6.5.1 SME Factor Jacobian

Equation 6.8 is the SME factor. The Jacobian for the n^{th} node is obtained by getting the partial derivative with x_n. As N measurements are mapped to N functions, the Jacobian for each of the functions $\{\mathrm{r_l}(\cdot) \mid \forall\ l = 1 \cdots N\}$ for n^{th} node is calculated as:

$$\begin{aligned} Jacobian^1_{sme} &= \frac{\partial(\mathrm{r_1}(x_1, x_2, \cdots, x_N))}{\partial x_n} = \frac{\partial(\sum_{k=1}^{N} x_k)}{\partial x_n} = 1 \\ Jacobian^2_{sme} &= \frac{\partial(\mathrm{r_2}(x_1, x_2, \cdots, x_N))}{\partial x_n} = \frac{\partial(\sum_{k=1}^{N} x_k^2)}{\partial x_n} = 2 \cdot x_n \\ &\vdots \\ Jacobian^N_{sme} &= \frac{\partial(\mathrm{r_N}(x_1, x_2, \cdots, x_N))}{\partial x_n} = \frac{\partial(\sum_{k=1}^{N} x_k^N)}{\partial x_n} = N \cdot x_n^{N-1} \end{aligned} \tag{6.11}$$

Similar to the topology factor, for a position x, y in a two-dimensional system, the process of calculating the Jacobian remains the same, i.e. treating x and y terms as constants in coloumns 1 and 2 respectively and calculating the partial derivative with x_i for the first row and with y_i for the second. The generalized Jacobian for SME factor for functions $\{\mathrm{r_l}(\,\cdot\,) \mid \forall\ l = 1 \cdots N\}$ can be written as:

$$\begin{aligned} Jacobian^{l}_{sme} &= \frac{\partial(\mathrm{r_l}((x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)))}{\partial x_n \partial y_n} \\ &= \begin{bmatrix} l \cdot x_n^{l-1} & 0 \\ 0 & l \cdot y_n^{l-1} \end{bmatrix} \end{aligned} \tag{6.12}$$

6.6 Measurement Uncertainties / Covariances

As explained in 5.6, to calculate the optimal state estimate Σs of the participating sensor have to used. As seen, Equations 5.1 and 6.1 requires the covariance matrices Σ^o_k, Σ^g_k and Σ^r_k, for the odometer, GPS and radar factors. This section derives these respective covariances.

Odometer Sensor Covariance: Similar to the odometer factor formulation in Chapter 5, the covariance Σ^o_k for the odometer sensor is assumed to be provided by the sensor manufacturer.

GPS Sensor Covariance: GPS measurements may be intermittently available. Moreover, we simulate a tunnel scenario as part of our experiments to analyse the benefits of SME factor. Therefore, when GPS measurements are available, then similar to Chapter 5, the covariance Σ^g_k for the GPS sensor is assumed to be provided by the sensor manufacturer.

6.6.1 SME Covariance

Similar to the topology factor, SME factor is also a virtual sensor measurement and hence direct radar sensor covariances cannot be used. Similar to the topology factor we use the Error Propagation Law in Equation 5.19 to calculate the covariance of SME factor. If $\sigma^2_{x_k}$ is variance of x_k then covariance $\forall\ k = 1 \cdots N$ can be written as:

$$\mathbf{C_{orig}} = \begin{bmatrix} \sigma^2_{x_1} & 0 & \cdots & 0 \\ 0 & \sigma^2_{x_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2_{x_N} \end{bmatrix} = \begin{bmatrix} \sigma^2_{x} & 0 & \cdots & 0 \\ 0 & \sigma^2_{x} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2_{x} \end{bmatrix} \tag{6.13}$$

SME factor is composed of a set of functions $\{\mathrm{r_n}(\,\cdot\,) \mid \forall\ n = 1 \cdots N\}$, therefore $\mathbf{F}$ is calculated for each function $\mathrm{r_n}$.

$$\begin{aligned} \mathbf{F_{r_n}} &= \begin{bmatrix} \frac{\partial(\mathrm{r_n}(x_1, x_2, \cdots, x_N))}{\partial x_1} & \frac{\partial(\mathrm{r_n}(x_1, x_2, \cdots, x_N))}{\partial x_2} & \cdots & \frac{\partial(\mathrm{r_n}(x_1, x_2, \cdots, x_N))}{\partial x_N} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial(\sum_{k=1}^{N} x_k^n)}{\partial x_1} & \frac{\partial(\sum_{k=1}^{N} x_k^n)}{\partial x_2} & \cdots & \frac{\partial(\sum_{k=1}^{N} x_k^k)}{\partial x_N} \end{bmatrix} \\ &= \begin{bmatrix} n \cdot x_1^{n-1} & n \cdot x_2^{n-1} & \cdots & n \cdot x_N^{n-1} \end{bmatrix} \end{aligned} \tag{6.14}$$

As per the Error Propagation Law in Equation 5.19, derived SME covariance can be written as:

$$\begin{aligned}\Sigma^{r_n} =& \mathbf{F_{r_n}} \cdot \mathbf{C_{orig}} \cdot \mathbf{F_{r_n}}^T \\ =& \begin{bmatrix} n \cdot x_1^{n-1} & n \cdot x_2^{n-1} & \cdots & n \cdot x_N^{n-1} \end{bmatrix} \cdot \begin{bmatrix} \sigma_x^2 & 0 & \cdots & 0 \\ 0 & \sigma_x^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_x^2 \end{bmatrix} \cdot \begin{bmatrix} n \cdot x_1^{n-1} \\ n \cdot x_2^{n-1} \\ \vdots \\ n \cdot x_N^{n-1} \end{bmatrix} \\ =& n^2 \cdot [x_1^{2 \cdot n-2} + x_2^{2 \cdot n-2} + \cdots + x_N^{2 \cdot n-2}] \cdot \sigma_x^2 \end{aligned} \tag{6.15}$$

For a position (x, y) in two-dimensional plane, where σ_x^2 and σ_y^2 are the variances for the radar for x and y, assuming x and y are independent, the covariance of the SME factor for function r_n can be written as:

$$\begin{aligned}\Sigma^{r_n} = n^2 \cdot [x_1^{2 \cdot n-2} + x_2^{2 \cdot n-2} + \cdots + x_N^{2 \cdot n-2}] \cdot \sigma_x^2 + \\ n^2 \cdot [y_1^{2 \cdot n-2} + y_2^{2 \cdot n-2} + \cdots + y_N^{2 \cdot n-2}] \cdot \sigma_y^2 \end{aligned} \tag{6.16}$$

When compared to covariance in Equation 6.15, Equation 6.16 adds the y terms and results int a covariance for a SME factor for a two dimensional plane. Note that the covariance here is a scalar value of a one dimension and not a matrix, as each of the symmetric function converts the measurement to a single value.

6.7 Evaluation

6.7.1 System Setup

Similar to the topology factor, the solution proposed in the previous section has been evaluated on a simulated data set based on real radar observations. The simulation is implemented with up to four vehicles on a highway for 200 steps.

The simulated vehicles are assumed to have an odometer and GPS sensors. The GPS measurements are used to show the benefits of the proposed SME factor. An infrastructure radar can measure the positions in its own coordinate system. Radar position is available in global coordinate system and the radar sensor system is configured to transform the measurements in its local coordinate system to the global coordinate system. It is assumed such transformation has negligible overhead and such transformation can be implemented in hardware.

Simulated sensors are assumed to have zero mean Gaussian Noises, with covariances $diag\,[1.0, 1.0]$, $diag\,[10.0, 10.0]$ and $diag\,[0.1, 0.1]$ for odometer, GPS and radar respectively. The time step interval is 1.

For SME factor, simulations are run for linear and curved trajectories. The results for each of them are compared as follows:

1. The fused trajectory using odometer and GPS sensor.

2. The fused trajectory using odometer, GPS sensor and an external radar. The external infrastructure radar sensor is fused using topology factor.

3. The fused trajectory using odometer, GPS sensor and an external radar. The external infrastructure radar sensor is fused using new SME factor.

The above comparison is also performed without the GPS sensor measurement to simulate the GPS devoid areas. That is:

1. The fused trajectory using only odometer sensor.
2. The fused trajectory using odometer and an external radar. The external infrastructure radar sensor is fused using topology factor.
3. The fused trajectory using odometer and an external radar. The external infrastructure radar sensor is fused using new SME factor.

Root Mean Square Error: Similar to the topology factor, we compare the performance using Root Mean Square Error. That is,

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{GroundTruth}})^2} \tag{6.17}$$

For details of RMSE and the above equation refer to Section 5.7.2.

6.7.2 Simulation Results

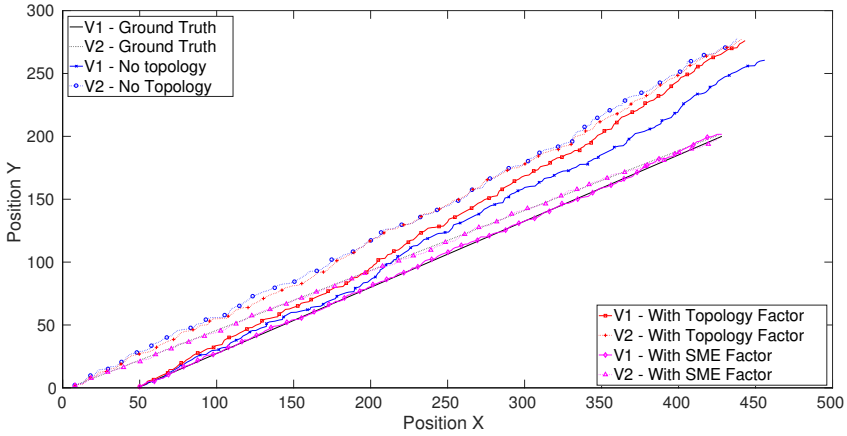
In Figure 6.1 (see Section 6.1), we show the results of the state estimation in absence of GPS measurements. The simulation is rerun with the newly proposed SME factor. The results for 2, 3 and 4 vehicles are shown in Figures 6.4(a), 6.4(b) and 6.4(c). Since with an increase in the number of vehicles, it becomes difficult to quantify the performance by just observing the fused trajectories, total system RMSE for the same are also plotted in Figure 6.5. The corresponding RMSE values for 2, 3 and 4 vehicles, are shown in Figures 6.5(a), 6.5(b) and 6.5(c).

It is very clear from the total RMSE of the system that the absence of GPS is well handled by the SME factor. The reason for the better performance for SME factor is that fully transformed coordinates are added using SME transformations and there is no loss of information due to the the introduction of the SME formulation. Hence, even in the absence of GPS, the state estimation converges to a solution and the error remains bounded. Additionally the data association is implicitly handled during the SME factor formulations.

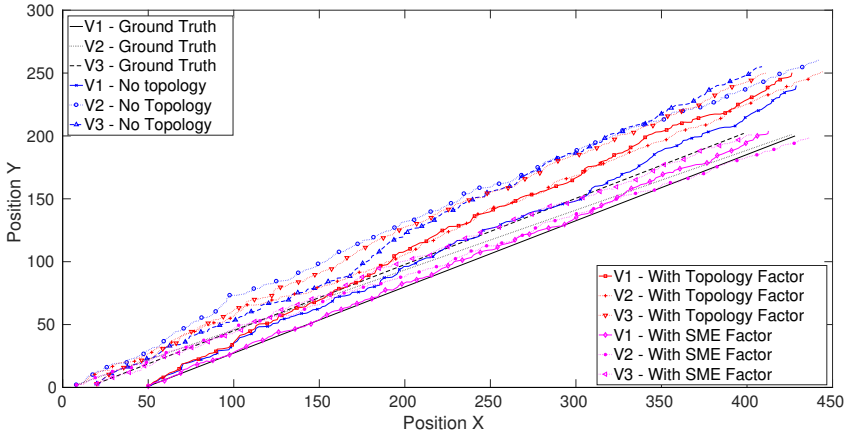
Next, we run the simulation of the system when the GPS measurements are available. Figures 6.6(a), 6.6(b) and 6.6(c) show the trajectories for 2, 3 and 4 vehicles and Figures 6.7(a), 6.7(b) and 6.7(c) show the corresponding total system RMSEs.

The SME factor again is better than the topology factor in presence of GPS measurements. The reason for the same is that using the 'precise' radar sensor with lower covariance, we add more information to the system. Therefore, in presence of all the three sensors with full information the optimizer is able to perform optimally, resulting in improved final state estimates and thus a lower total system RMSE error for all the cases.

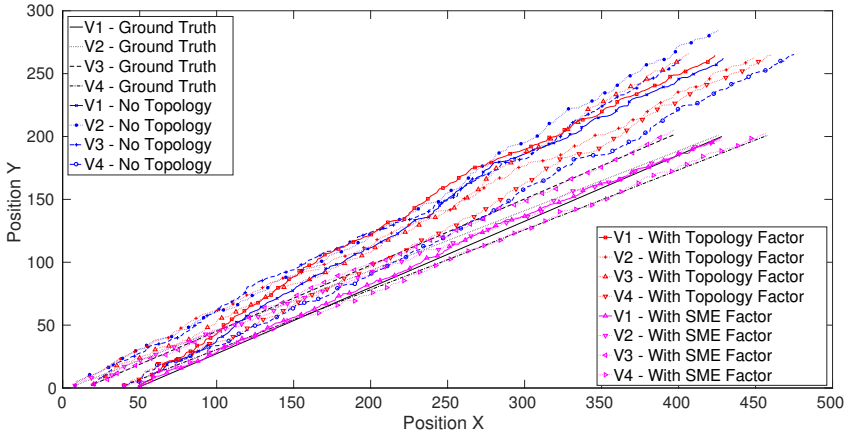
To further test the stability of the solution, a Monte-Carlo simulation is run for 1000 iterations and the average RMSE values are analysed. Table 6.1 presents the average RMSE values for 2, 3 and 4 vehicles. The average results reflect the same behaviour as seen in graphs in Figures 6.5 and 6.7 .



(a)

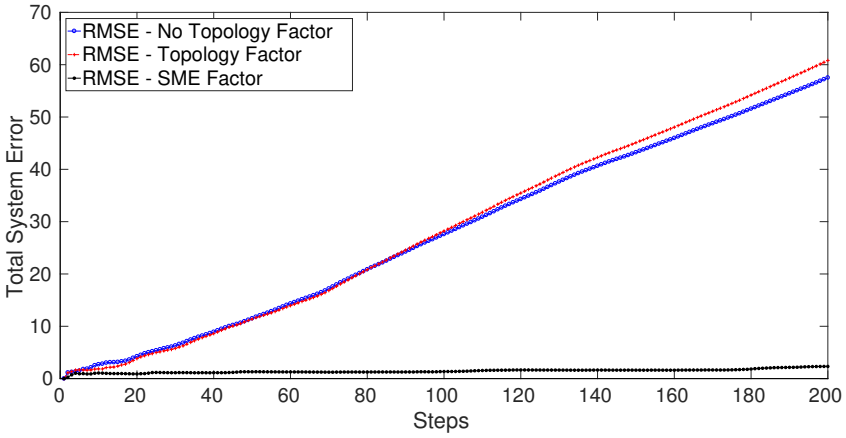


(b)

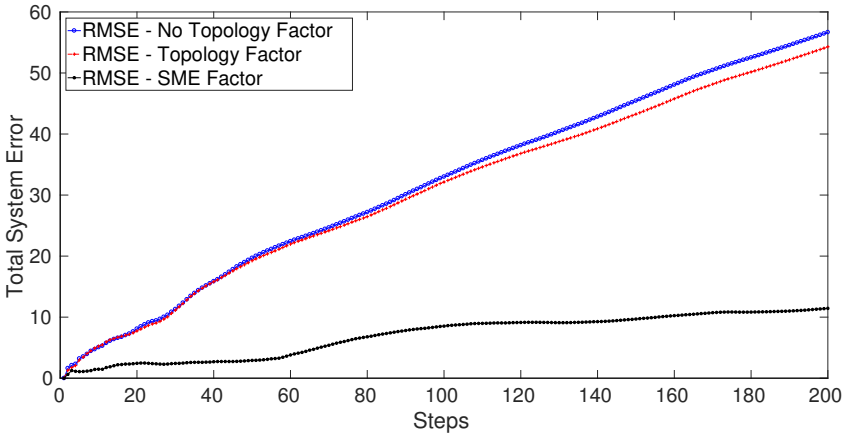


(c)

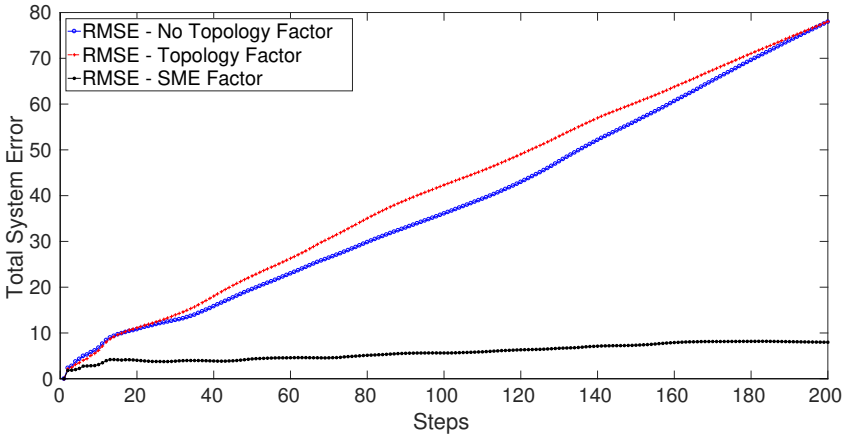
Figure 6.4: Ground truth; Fused trajectories without topology factor, with topology factor and with SME factor in **GPS devoid** areas for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)

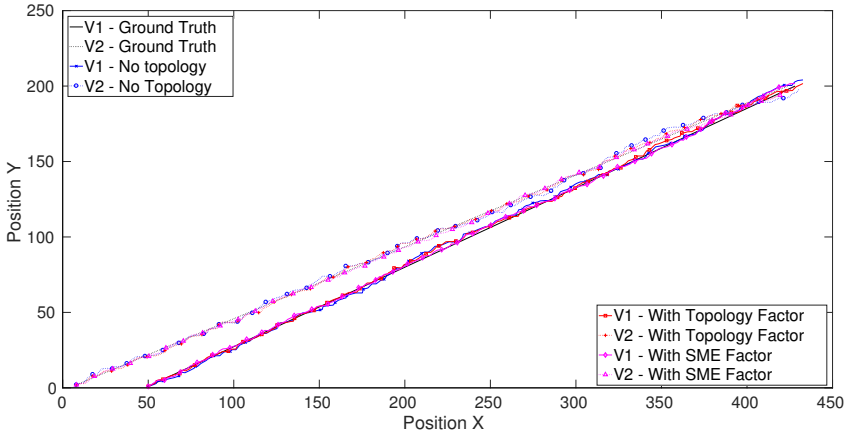


(b)

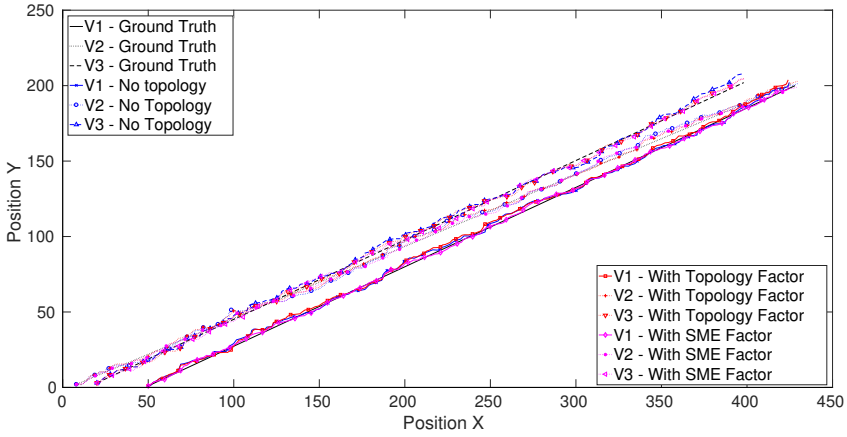


(c)

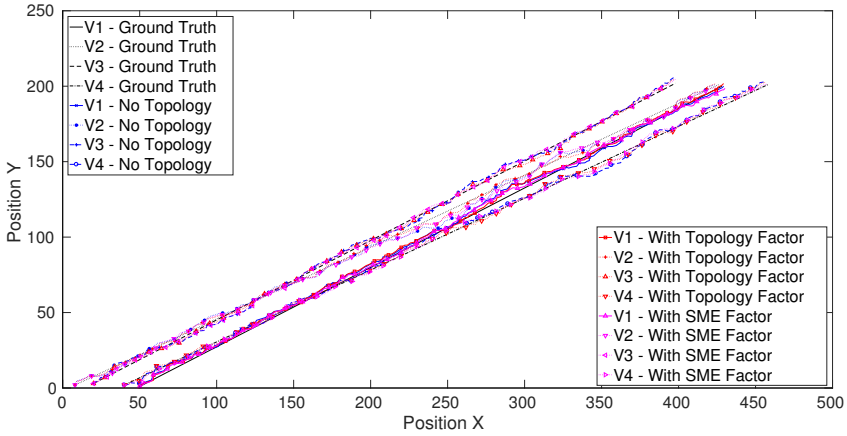
Figure 6.5: Total System RMSE for trajectories shown in Figure 6.4 for (a) 2 vehicle systems (Figure 6.4(a)). (b) 3 vehicle system (Figure 6.4(b)). (c) 4 vehicle systems (Figure 6.4(c)).



(a)

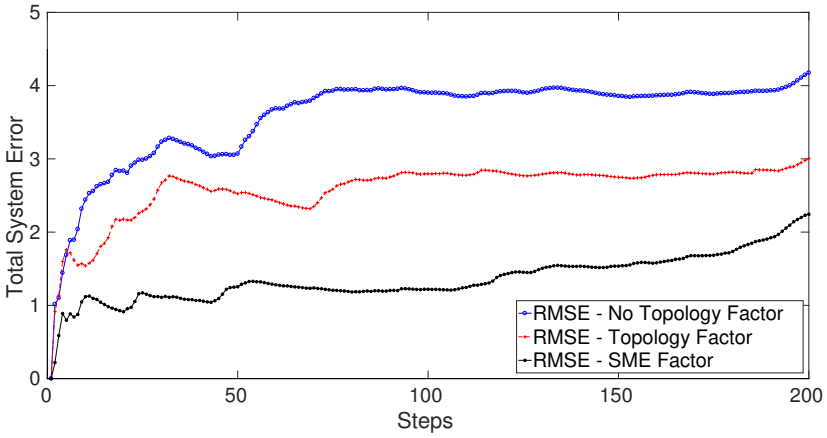


(b)

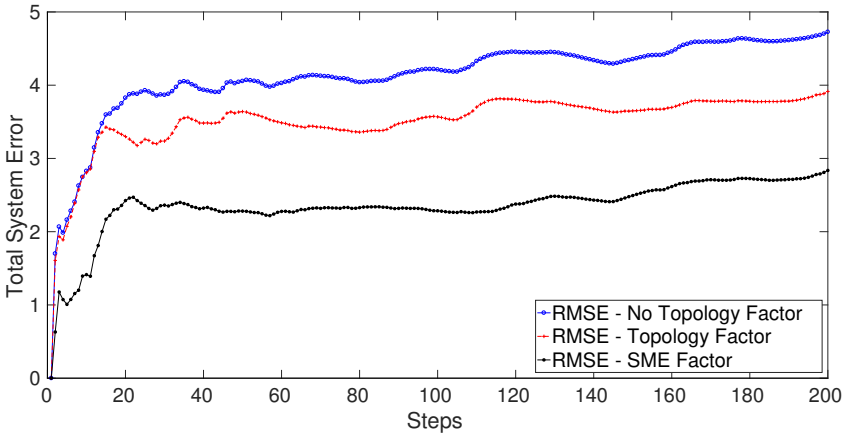


(c)

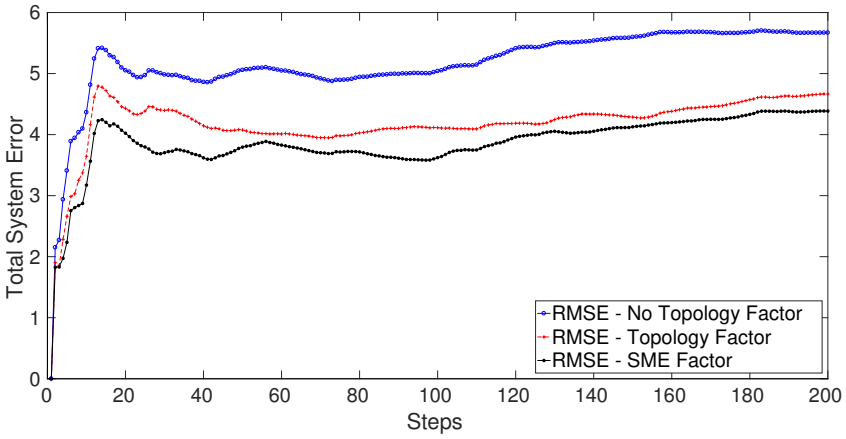
Figure 6.6: Ground truth; Fused trajectories without topology factor, with topology factor and with SME factor when **GPS is available** for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)



(b)



(c)

Figure 6.7: Total System RMSE for trajectories shown in Figure 6.6 for (a) 2 vehicles (Figure 6.6(a)). (b) 3 vehicles (Figure 6.6(b)). (c) 4 vehicles (Figure 6.6(c)).

Table 6.1: Average total system RMSE for 1000 iterations for 2, 3 and 4 vehicles.

Number of Vehicles	No Additional Factor		Topology Factor		SME Factor	
	Without GPS	With GPS	Without GPS	With GPS	Without GPS	With GPS
2	50.405	4.211	53.212	3.183	2.626	2.146
3	62.146	5.180	60.484	4.166	9.492	3.230
4	71.379	5.955	71.619	4.743	13.994	4.153

The proposed solution of SME factor is further tested with simulation of vehicles moving with a constant turn model in curved trajectories. Figures 6.8(a), 6.8(b) and 6.8(c) show the trajectories of 2, 3 and 4 vehicles moving in a curved path when no GPS measurements are present. Figures 6.8(a), 6.8(b) and 6.8(c) show the corresponding total system RMSEs. Similar to the linear trajectories, it is clear that SME factor is able to handle the absence of GPS for curved trajectories as well.

The SME factor is also analysed for constant turn model in the presence of GPS measurements. Figure 6.10(a), 6.10(b) and 6.10(c) show the trajectories and Figures 6.11(a), 6.11(b) and 6.11(c) show the corresponding total system RMSE values of 2, 3 and 4 vehicles for curved paths. Here also SME factor contributes more information for the optimization process and hence results in improved state estimates.

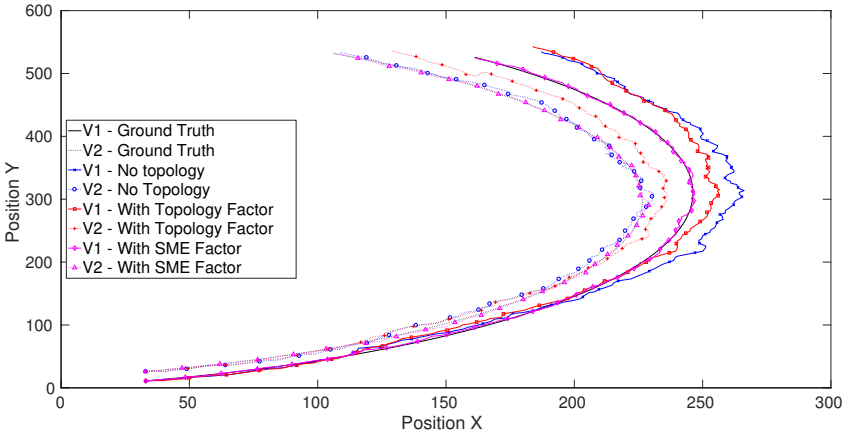
Table 6.2: Average total system RMSE for 1000 iterations for 2, 3 and 4 vehicles for Constant Turn Model.

Number of Vehicles	No Additional Factor		Topology Factor		SME Factor	
	Without GPS	With GPS	Without GPS	With GPS	Without GPS	With GPS
2	18.423	4.487	28.724	3.249	3.585	1.902
3	23.963	5.508	29.805	4.559	8.938	3.234
4	27.873	6.295	32.172	5.430	20.099	4.184

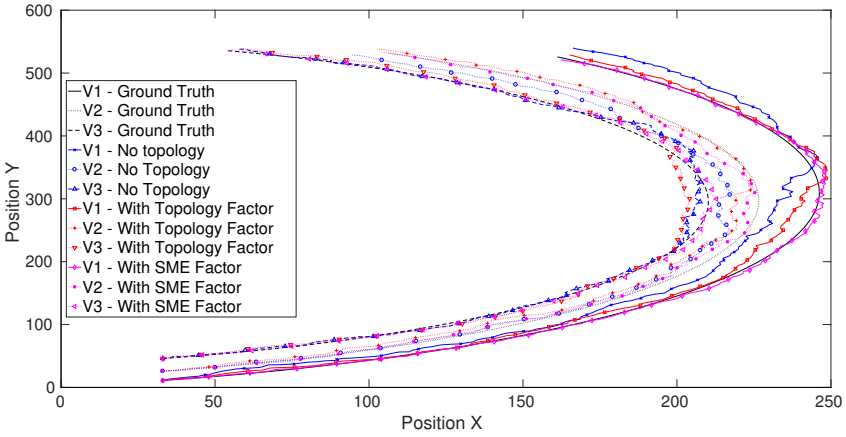
To further analyse the stability of the proposed SME factor for a constant model, a Monte-Carlo simulation of 1000 iterations is run and the average total system RMSE of 2, 3 and 4 vehicles is calculated. Table 6.2 shows the results for both the presence and absence of GPS. From the table it can be seen that in absence of GPS the SME factor performs better than the topology factor. Although in case of 4 vehicles the average RMSE error is a little high but it still is lower than both the other cases of no additional factor and the topology factor in absence of GPS.

6.7.3 Plug and Play and Online Scenario

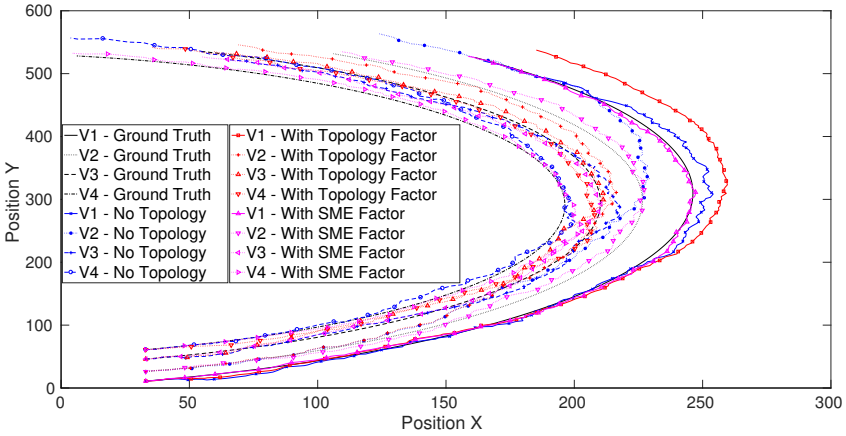
All the previous tests have shown the effectiveness and stability of the proposed SME factor formulation. The scenarios assumed that all the sensor measurements were available/unavailable all the time. The graphs were fully formulated and offline batch optimization was performed using Levenberg-Marquardt algorithm.



(a)

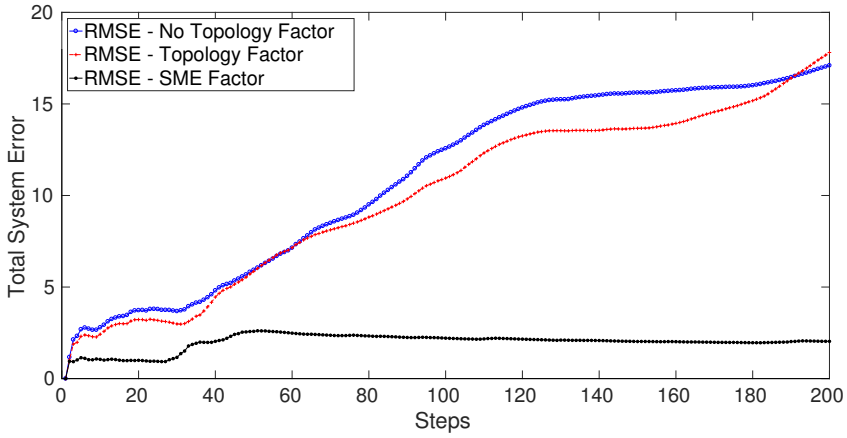


(b)

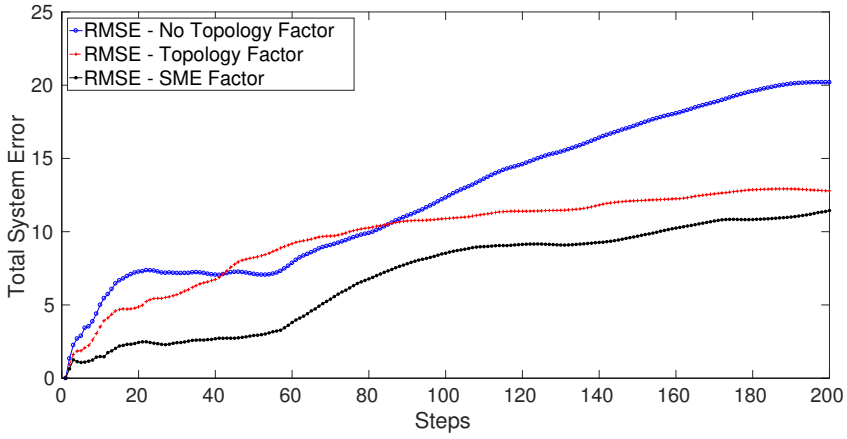


(c)

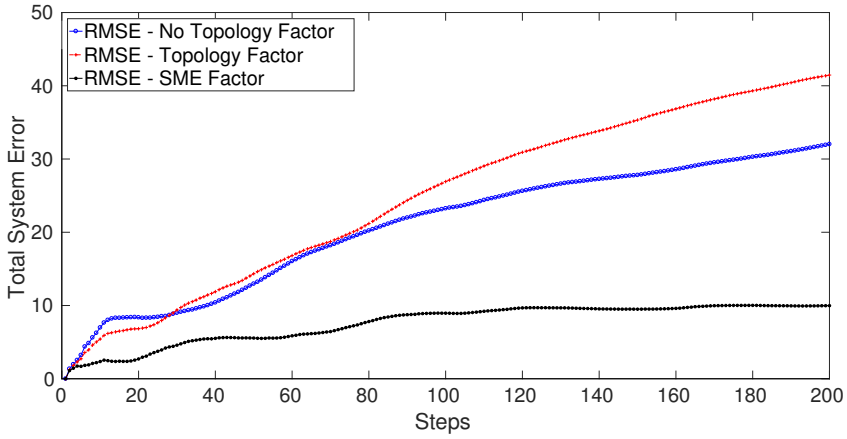
Figure 6.8: Ground truth; Fused trajectories without topology factor, with topology factor and with SME factor for constant turn model in **GPS devoid** areas for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)

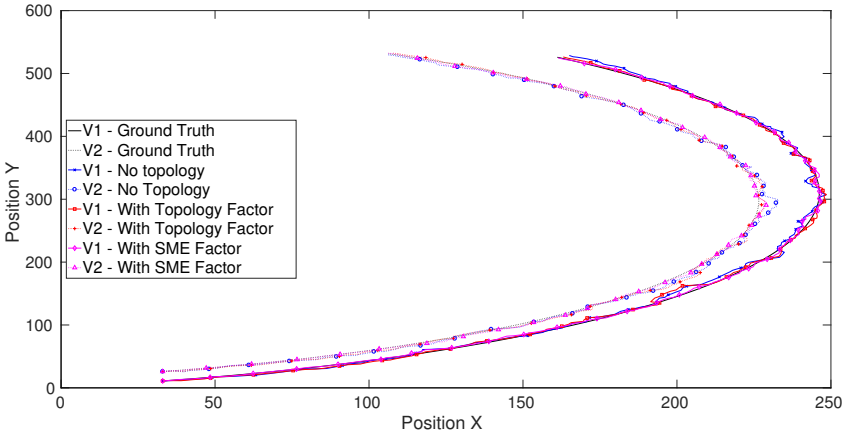


(b)

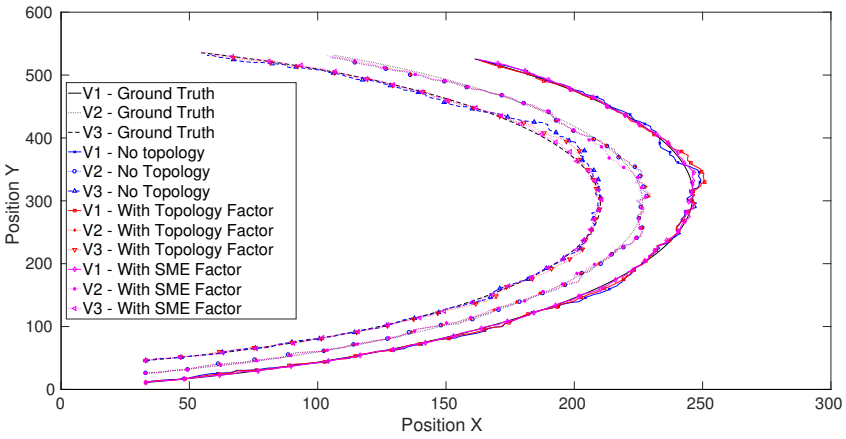


(c)

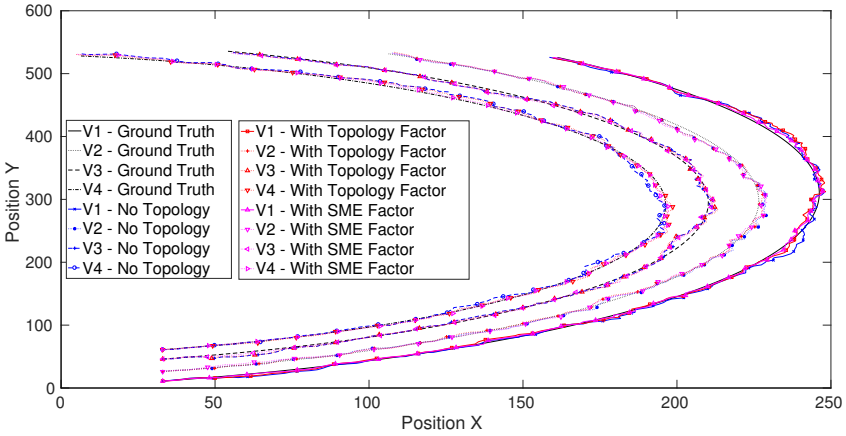
Figure 6.9: Total System RMSE for trajectories in Figure 6.8 for (a) for 2 vehicles (Figure 6.8(a)). (b) 3 vehicles (Figure 6.8(b)). (c) 4 vehicles (Figure 6.8(c)).



(a)

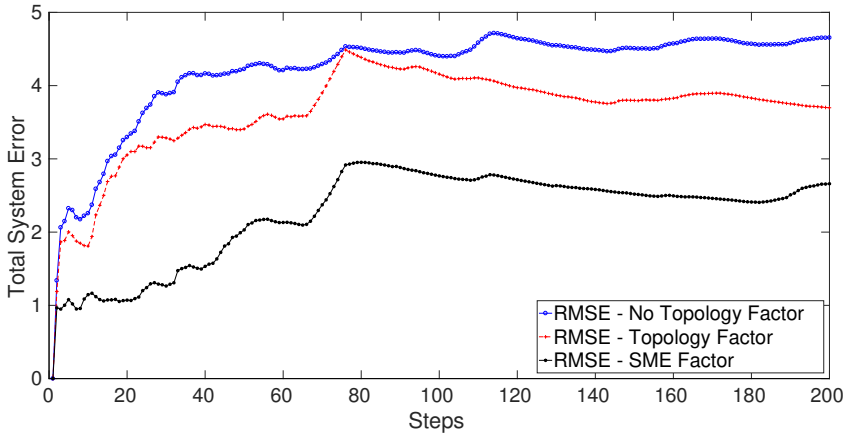


(b)

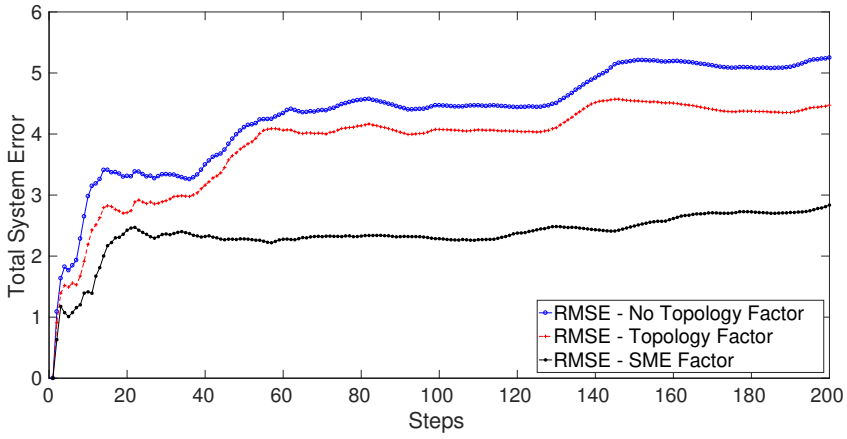


(c)

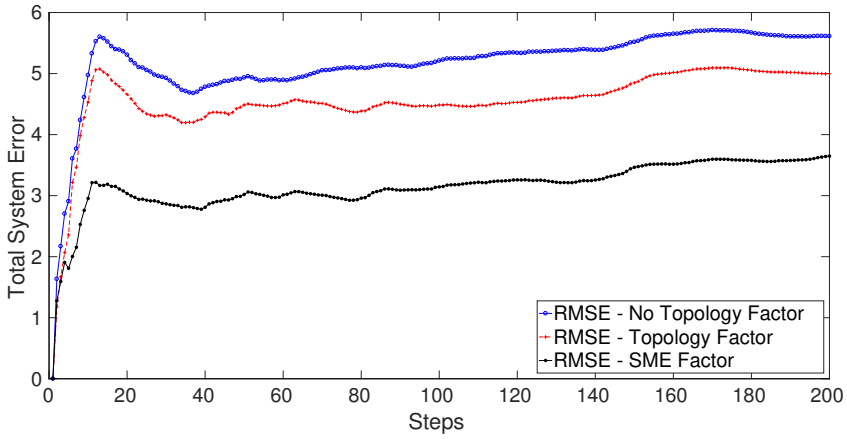
Figure 6.10: Ground truth; Fused trajectories without topology factor, with topology factor and with SME factor for constant turn when **GPS is available** for (a) 2 vehicles. (b) 3 vehicles. (c) 4 vehicles.



(a)



(b)



(c)

Figure 6.11: Total System RMSE for trajectories shown in Figure 6.10 for (a) 2 vehicles (Figure 6.10(a)). (b) 3 vehicles (Figure 6.10(b)). (c) 4 vehicles (Figure 6.10(c)).

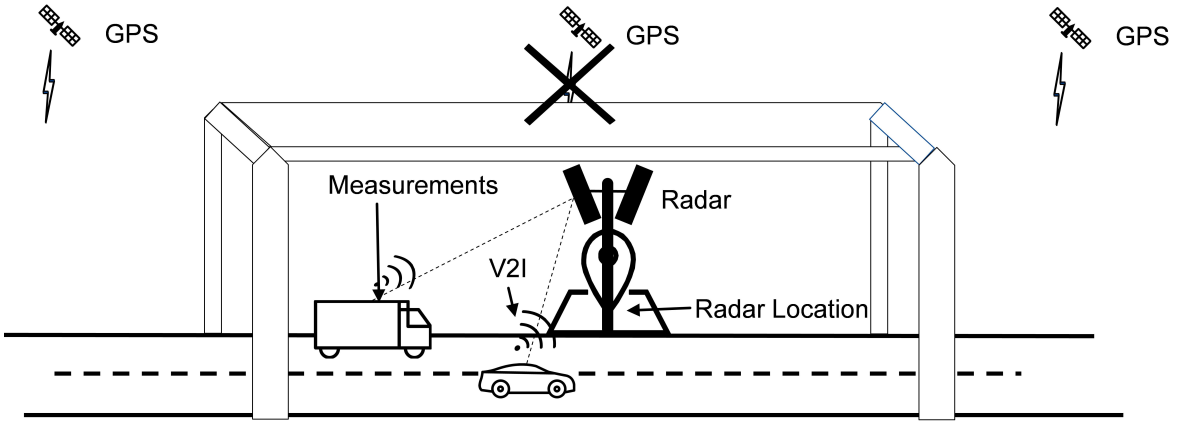


Figure 6.12: The figure highlights the constraints and set up for a tunnel. The tunnel is equipped with a radar sensor whose location information is known. Inside the tunnel, GPS is not available and the radar measurements are incorporates as SME factor.

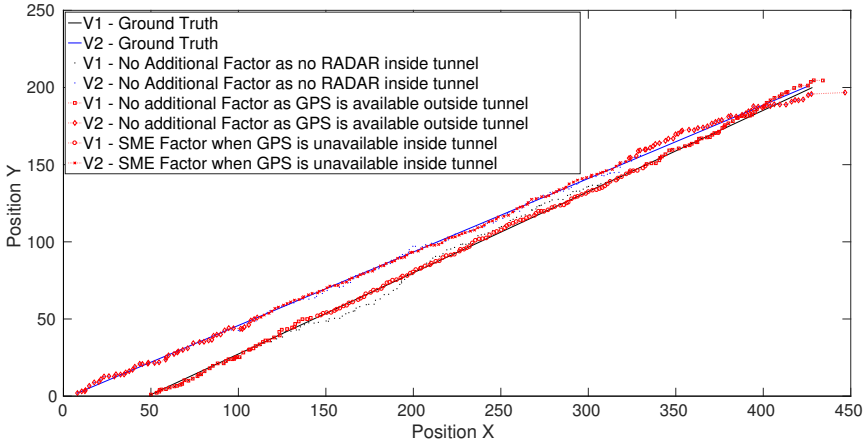
However, in introduction, the ineffectiveness of the topology factor for scenarios like tunnels and underground parking garage was highlighted. In such scenarios, GPS is available during part of the trajectory. For example, before a vehicle enters a tunnel, GPS measurements are available, but when it enters the tunnel, the direct view of the satellites is lost and GPS measurements are no longer available. When the vehicle exits the tunnel, the satellites are again visible and the GPS measurements are available again.

Therefore, in real scenarios the sensors measurements are not always available. In the tunnel scenario explained previously, if an infrastructure radar is installed in the tunnel, then the radar measurements are only available inside the tunnel. These measurements are not available before the vehicle enters the tunnel and after the vehicle exits the tunnel.

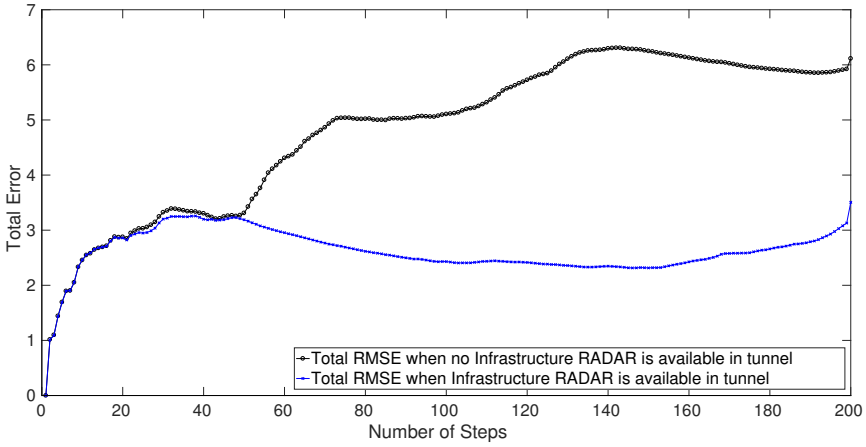
The fusion system should be able to adapt to such scenarios and fuse the data, as it is available. Such property of the system can be called as "plug and play". The "plug" phase adds the new, previously unseen sensor measurements when they are available. In the above scenario, when the vehicle enters the tunnel and incorporates the measurements from a radar installed in the tunnel in its factor graph formulation reflects the "plug" phase. The "play" phase fuses the newly added measurements to arrive at new fused state estimates. For the tunnel scenario when the system solves the graphs after adding the radar measurements to have improved state estimates reflects the "play" phase. When the measurements are no longer available, the system should automatically uses only the other available sensor measurements. For example, when the vehicle exits the tunnel, it does not receive any radar measurements but the GPS is restored. Therefore, instead of the SME factors, the graph has the GPS factors.

In such a real scenario, system should calculate the new state estimate soon as the new sensor measurements appear and not wait till the end, when all the measurements are available to formulate the complete graph and then run the offline batch optimizer. Therefore, an online incremental smoothing algorithm like iSAM2 [93] (supported in GTSAM) is required. Further the GTSAM framework supports the above described plug and play feature [98] thereby providing an efficient platform for appearing or disappearing sensor measurements.

In this section, a tunnel scenario with two vehicles on a ground plane is simulated. The tunnel is



(a)



(b)

Figure 6.13: Simulation of two vehicles through the tunnel, demonstrating the use of SME factor using online iSAM2 algorithm. (a) Ground truth and fused trajectories for 2 vehicles. (b) Total RMSE for system of 2 vehicles.

equipped with a radar sensor whose location information is known. Therefore, the radar system can perform the transformation of measurements from its local coordinates to global coordinate system. The simulation is run for 200 steps. At the $51st$ step vehicles enter the tunnel and lose the GPS connection. At the $151st$ step they exit the tunnel and the GPS is restored without any delay. Inside the tunnel, the infrastructure radar system is able to measure and communicate the positions of the two vehicles. The scenario can be understood from Figure 6.12.

Figure 6.13(a) shows the trajectories of the two vehicles. For the case when the tunnel does not have a radar sensor (hence no SME factor), it can be seen that the fused trajectories of the vehicles drift farther from the ground truth after they enter the tunnel. However, when the tunnel is equipped with the infrastructure radar and the SME factor is added as the constraint between the trajectories when the vehicle is inside the tunnel, the fused trajectories remain closer to the ground truth. The trajectory resulting from the SME factor is shown in the red color. For the scenarios with and without the radar, first and last 50 steps have odometer and GPS as factors. Inside the tunnel, without the radar, only

odometer factors are added to the graph. However, with radar, both odometer and SME factors are added to the graph. This additional information in the form of SME factor helps to overcome the drift often experienced by odometer like sensors and forces the solution towards the true position.

The result can also be analyzed from the RMSE of the total system shown in Figure 6.13(b). Through to the $50th$ time step, both the systems have almost the same total system RMSE values. Starting $51st$ time step the RMSE error increases for the system which does not have the infrastructure radar (hence no SME factors). However, for the system with the infrastructure radar (hence SME factors) the total system RMSE falls as radar has a lower error covariance.

6.8 Summary

The SME factor presented in this chapter, provides another novel mechanism to fuse the external radar measurements with internal sensors for multi-target scenario without performing any measurement-to-target association. Unlike the topology factor, the measurement in the local radar coordinates cannot be used to formulate the SME factor. However, with a minimal overhead of conversion of radar local coordinate measurements into to a global coordinates, the formulated SME factor is able to lower the state estimation error when GPS measurements are not available. This is beneficial for the scenarios like tunnels and underground parking garages. The coordinate transformation is possible as the radar can be configured with its precise location in the Global Coordinate System.

The non-linear least square algorithm used for state estimation is moderately able to handle an increase in the number of vehicles. For higher number of participants like 30, the scalability of solution suffers. Further we only need to transmit the transformed measurements and not the covariances, and the original covariance of the radar is only required to be transmitted once for the fusion. This keeps the bandwidth requirement to the minimum.

For SME factor the feasibility is demonstrated using both, the offline algorithm of Levenberg-Marquardt and the online iSAM2 [93] for final state estimation. The iSAM2 algorithm also helps to demonstrate the "plug and play" property, which is an important requirement for the futuristic cooperative fusion systems for highly assisted or autonomous vehicles, where the environment has always some sensors available for improved state estimation.

Hence, we are able to generally meet 5 of the 6 challenges highlighted in Section 1.3. Therefore, the SME factor addresses the challenges of cooperative localization in a GPS devoid and clutter free environment.

Chapter 7

DSRC Factor

7.1 Motivation

The previous two Chapters 5 and 6 introduced two novel additions to factor graphs for autonomous vehicles, the topology factor and the SME factor. They were used for improving state estimation using external infrastructure radar sensor, internal odometer and/or internal GPS using V2I communication mechanisms. However, modern day highly automated and autonomous vehicles are already equipped with Vehicle-to-Vehicle (V2V) communication mechanisms. In this chapter, we present a new factor, which considers this Vehicle-2-Vehicle communication mechanisms for further improved state estimation.

Chapter 3 defined the constraints for cooperative localization. Particularly point 3[1] highlights that communication methods are not used to identify the individual vehicles. For the previously proposed mechanisms using V2I communication, this constraint was strictly adhered. However, modern day vehicles at least capable of Assisted Driving come installed with Dedicated Short Range Communication (DSRC) mechanisms, which can identify the individual vehicles surrounding the ego vehicle in real time. Therefore, the assumption highlighted in Chapter 3 Point 3 changes. The new assumption can be stated as follows:

- V2V bi-directional communication using DSRC receiver and transponder is available which already supports the identification of neighbouring vehicles.

Additionally, the previous assumptions remain same, which are summarized as follows:

- It is assumed odometer and GPS are available.
- An externally deployed radar system can monitor the vehicles.
- V2I bi-directional communication is available and cannot identify the individual vehicles based on the communication mechanisms.
- Lastly, a clutter free environment is assumed.

[1] To perform cooperative localization, the thesis assumes an error free V2X bi-directional communication capability is available. Further, various communication methods are not used to identify the individual vehicles. There are various radio-ranging methodologies like received signal strength, time of arrival, time difference of arrival, Doppler Shift and angle of arrival [78] to identify individual targets. It is assumed that any external sensor, like infrastructure radar does not undertake any such action to identify the individual targets

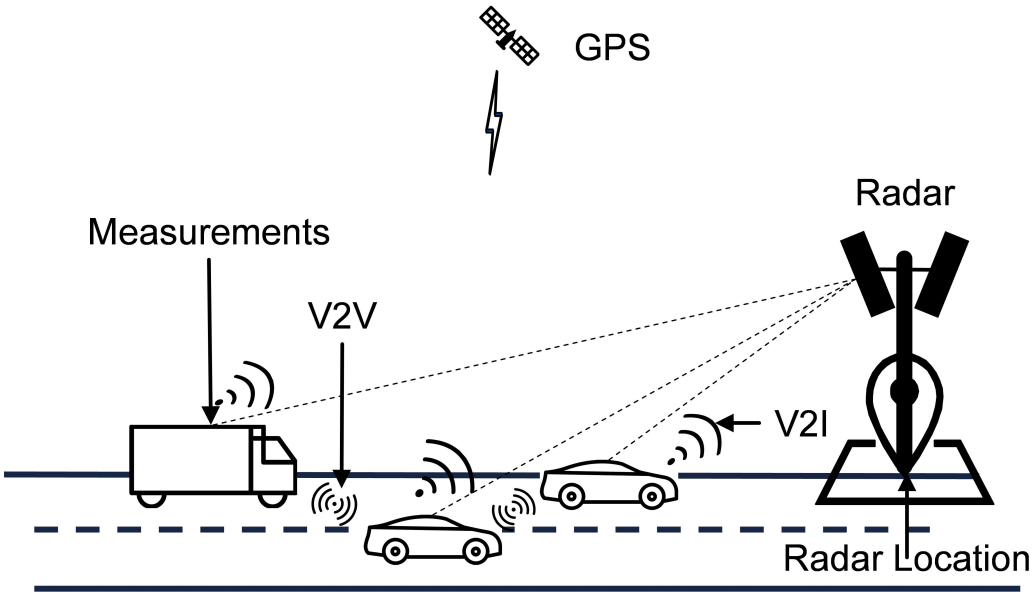


Figure 7.1: The figure highlights the constraints and set up for DSRC factor. It is assumed the location of radar sensor is known and the radar measurements are incorporated as SME factor. Each vehicle is equipped with DSRC receiver and transponder and is able to uniquely identify the vehicles up to a certain distance.

The above scenario can be understood from Figure 7.1. The results of the researched factor were published in the conference paper:

1. Gulati et. al.: Graph Based Cooperative Localization Using Symmetric Measurement Equations and Dedicated Short Range Communication. 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2017.

7.2 SLAM Formulation

For the improved state estimation, as mentioned in the motivation we have access to measurements from an odometer, GPS, infrastructure radar and DSRC sensor. We use the inter-vehicle distances from the DSRC (explained in the next Section 7.3) . Therefore, the estimated state X^*, as defined in Equation 4.23 can be further extended with the DSRC measurements as:

$$\begin{aligned}\mathbb{X}^* = \arg\min_{X} \Bigg(& \underbrace{\sum_{1}^{n} (\|((\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i)\|^2_{\Sigma_i^o})}_{\text{Odometer}} + \underbrace{\sum_{1}^{m} (\|((\mathrm{g}(x_{i_k})) - z_k^g)\|^2_{\Sigma_k^g})}_{\text{GPS}} \\ & + \underbrace{\sum_{1}^{l} (\|((\mathrm{r}(x_{i_k})) - z_k^r)\|^2_{\Sigma_k^r})}_{\text{Radar}} + \underbrace{\sum_{1}^{p} (\|((\mathrm{s}(x_{i_k})) - z_k^s)\|^2_{\Sigma_k^s})}_{\text{DSRC}} \Bigg)\end{aligned} \tag{7.1}$$

where the mathematical terms describing odometer, GPS and radar are same as previously defined. $\mathrm{s}(\,\cdot\,)$ is the DSRC measurement function which represents the state of the vehicle which has to be estimated,

i.e. its location. z_k^s is the measurement generated from the mapping of actual distance using DSRC (discussed later) to the state of the vehicle. Σ_k^s is the covariance matrix for the distance measured using the DSRC sensor.

To incorporate the radar measurements, any of the topology factor (Chapter 5) or the SME factor (Chapter 6), can be used. This chapter assumes the location of the radar is known, therefore uses the SME factor to formulate the radar measurements.

7.3 Dedicated Short Range Communication

Ad-hoc communication in vehicular network among vehicles has emerged as a challenge for connected vehicles. For real time wireless communication, a dedicated Wireless Local Area Network (WLAN) protocol IEEE802.11p has been implemented. It has a two-way short- to- medium-range wireless communications capability that permits very high data transmission which is critical in communications-based active safety applications. In Report and Order FCC-03-324, the Federal Communications Commission (FCC) allocated 75 MHz of spectrum in the 5.9 GHz band for use by Intelligent Transportations Systems (ITS) vehicle safety and mobility applications [6]. FCC had allocated this bandwidth as early as in 1999 for the automotive use [99]. Similarly European Telecommunication Standards Institute (ETSI) allocated 30 MHz in 5.9 GHz band in 2008 [100]. Various other authorities around the world have also reserved dedicated bandwidth. This bandwidth is called Dedicated Short Range Communication (DSRC). Most of the modern day cars capable of semi or fully automated driving are equipped with the DSRC transceivers and transponders.

Bai et. al. [101] analysed various physical characteristics of DSRC. We mention here a few of them. For detailed specifications, please refer to the DSRC standard (FCC [99]). DSRC is for data-only systems and has a maximum theoretical range of about $1000m$ for Line-Of-Sight conditions. However, this range is lower in practice and highly depends on the obstacles and the objects in the environment. Two important properties of DSRC are: (a) Using DSRC, vehicles can be uniquely identified. (b) Second is that distance of neighbouring uniquely identified vehicles from the ego vehicle can be estimated.

7.3.1 Distance measurement using DSRC

The distance measurement is achieved using various radio-ranging methodologies like Received Signal Strength (RSS), Time of Arrival (TOA), Time Difference of Arrival (TDOA), Doppler Shift and Angle of Arrival (AOA). Each of the methods have their benefits and problems. There exists a lot of literature for achieving the same like [102], [78], [37]. We just give a quick overview of some of the methodologies. The interested reader can refer to the above-mentioned literature for further details.

Like the name suggests, RSS uses the ratio of received strength and the power of the signal to estimate the range. The path loss model governs the attenuation of signal strength and this is used for range estimation. To measure the distance, DSRC packets are broadcasted and based on the signal strength of the reply received the distance from the answering transmitter can be estimated. The replier transmitter can be a mobile or a static target.

TOA uses signal propagation time to estimate the distance and requires the clocks of sender and receiver to be synchronized. To measure the distance, DSRC packets are time stamped and broadcasted

and based on the time stamp of the reply received, the distance from the answering transmitter can be estimated. Again, the replier transmitter can be a mobile or a static target.

As TOA requires time synchronization between the receiver and the transmitter, it makes TOA more complex and expensive than other approaches. To avoid this complexity and the cost, other time-based approaches have been proposed. TDOA is one such technology, which partially avoids the problem of time synchronization. In TDOA, two signals are received from a pair of transmitters, in one receiver. The time difference between these two signals is used for estimation. For such a system, the receiver synchronization is not required but the transmitters still need to be synchronized.

Researchers have also come up with more strategies that are novel, for example a Two-Way TOA. In this more than one communication between a pair of transmitter and receiver is considered for distance estimation [103]. After the delay estimation of the signal processing has been calculated for both transmitter and receiver, the final message and corresponding acknowledgement between them is exchanged. Using the total time consisting of two-way signal flight, the distance between the nodes is calculated.

Doppler Shift is the difference between the emitted frequency and the observed frequency. This is commonly referred to as 'Frequency Difference of Arrival'. Using the combination of vehicular speeds from the data over DSRC and Doppler shift, the range can be calculated.

AOA is used to get the bearing of received signals. It measures the Angle of Arrivals of a signal from a transmitter using antenna arrays. Like other methods, it is most efficient with direct Line-of-Sight. It also suffers from issues arising from reflection and scattering and hence not very efficient.

All the methods suffer from various degrees of measurement noise, multi-path fading, interference, non-line-of-sight errors. Alam et. al. [78] provides the performance comparison for some of the above mentioned methods.

The distance of neighbouring vehicles from ego-vehicle has already been used for improved state estimation [52, 53, 54, 55, 56, 104]. In this chapter we propose another factor which is similar to the the previous work of using V2V radio technologies to perform the cooperative localization. However, the previous methods uses fusion methods like Kalman Filter, Extended Kalman Filter etc. but we propose the use of factor graphs. Section 2.3 gives a detailed overview of the papers which use DSRC for cooperative localization. The distance information for neighbouring vehicles obtained using DSRC range can be formulated as *DSRC range factor* to improve the state estimation of the ego-vehicle.

7.4 DSRC Range Factor Formulation

The DSRC range factor incorporates the distance of vehicles, which are present around the ego vehicle. The distance measurement can be calculated from any of the methods mentioned in the previous section and can be represented as:

$$z_k = \mathrm{s}(x_i) + n^d \tag{7.2}$$

where z_k is the distance of the k^{th} vehicle, s is any of the range calculating functions as described in previous section for ego vehicle at position x_i, n^d represents the Gaussian measurement noise.

As the distance can be calculated by any of the methods based on DSRC, the error in measurement depends on the used method and is adjusted while formulating the corresponding factors.

SME Factor Formulation: The external radar measurement is formulated as a SME factor as explained in Chapter 6.

7.5 Jacobians for the Non-Linear Least Square Optimization

Similar to the topology factor, the proposed DSRC factor also requires the first derivative, i.e. the Jacobians, of the functions to numerically solve the non-linear equations. This section derives the Jacobians from the respective equations, required for the optimization process.

Odometer Factor Jacobian: The odometer Jacobian remains same as Equation 5.12, that is:

$$Jacobian_{odom} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{7.3}$$

GPS Factor Jacobian: GPS factors use the same Jacobians as Equation 5.15.

$$Jacobian_{gps} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{7.4}$$

SME Factor Jacobian: The Jacobian for SME factor for symmetric functions $\{\mathrm{r_l}(\,\cdot\,) \mid \forall\, l = 1 \cdots N\}$ for N targets in a two dimension space can be written as:

$$\begin{aligned} Jacobian^{l}_{sme} &= \frac{\partial(\mathrm{r_l}((x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)))}{\partial x_n \partial y_n} \\ &= \begin{bmatrix} l \cdot x_n^{l-1} & 0 \\ 0 & l \cdot y_n^{l-1} \end{bmatrix} \end{aligned} \tag{7.5}$$

For details of the calculation of Jacobian for the formulated SME factor the reader is advised to see Chapter 6 Section 6.5.1.

7.5.1 DSRC Factor Jacobian

Equation 7.2 is the DSRC range factor function. The Jacobian for a node k is obtained by getting the partial derivative with x_i.

$$Jacobian^{k}_{dsrc} = \frac{\partial(\mathrm{s}(x_i))}{\partial x_i} = 1 \tag{7.6}$$

7.6 Measurement Uncertainties / Covariances

As explained in Section 5.6, to calculate the optimal state estimate covariances of the participating sensor have to be used. As seen, Equation 7.1 requires the covariance matrices Σ^o_k, Σ^g_k, Σ^r_k and Σ^d_k, for the odometer, GPS, radar and DSRC factors. This section derives these respective covariances.

Odometer Sensor Covariance: The sensor manufacturer provides the covariance, Σ^o_k for the odometer sensor.

GPS Sensor Covariance: For GPS factor formulation, like odometer sensor, we use the covariance provided by the sensor manufacturer.

SME Covariance: For a position (x, y) in two-dimensional plane, where σ_x^2 and σ_y^2 are the variances for the radar for x and y, assuming x and y are independent, the covariance of the SME factor for function r_n can be written as:

$$\begin{aligned}\Sigma^{r_n} = n^2 \cdot [x_1^{2 \cdot n-2} + x_2^{2 \cdot n-2} + \cdots + x_N^{2 \cdot n-2}] \cdot \sigma_x^2 + \\ n^2 \cdot [y_1^{2 \cdot n-2} + y_2^{2 \cdot n-2} + \cdots + y_N^{2 \cdot n-2}] \cdot \sigma_y^2\end{aligned} \tag{7.7}$$

For details of the calculation of above covariance for the formulated SME factor, the reader is advised to see Chapter 6 Section 6.6.1.

7.6.1 DSRC Covariance

The DRSC range calculation is dependent upon the physical method used to establish the value (e.g. AOA, TOA, TDOA, FDOA etc.). It has to be noted that each of these methods have their own set of errors e.g. 'angle' error or a 'hyperbolic error' for the TDOA system. What we are interested in is the property that various techniques introduce an error of 'x metres' in the final calculation of distance from the ego-vehicle, independent of the method.

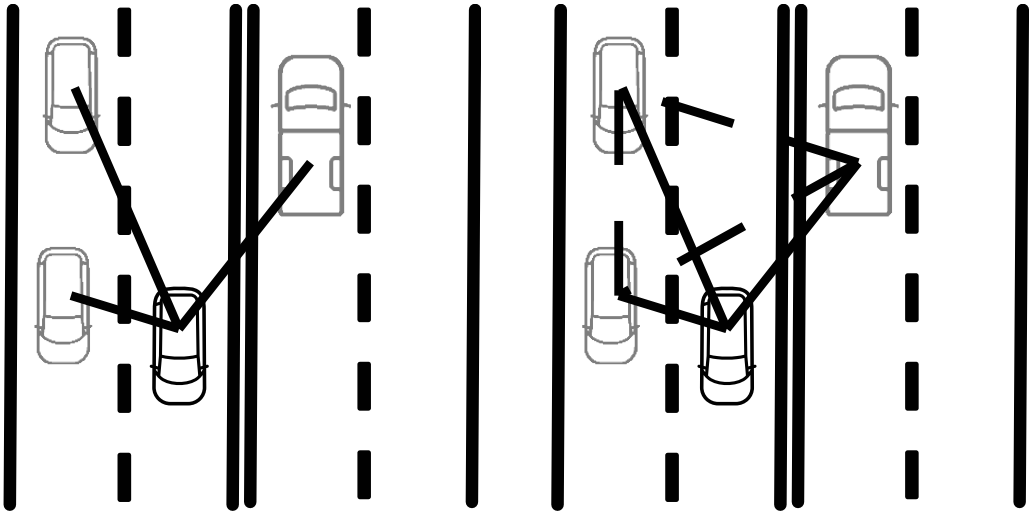
As shown by Alam et. al. [78] different method have different standard deviations. In fact, statistic shown by Alam et. al. show that the standard deviation for the same methodology changes with the distance between the target and the receiver. Therefore, to have a comprehensive analysis we can not use any one fixed standard deviation.

7.7 Evaluation

7.7.1 Simulation Setup

Cooperative localization using DRSC can be primarily achieved using star architecture or cluster architecture. Star architecture means only direct connection between ego vehicle and the neighbouring vehicles is considered. This results in an architecture with ego vehicle in the center and other vehicles around the ego vehicle connected to it, resulting in a star formation. Figure 7.2(a) highlights the star architecture. The black is the ego-vehicle. The black lines consist of the connection between the ego-vehicle and neighbouring vehicles in the range.

For a cluster architecture, not only the direct connections between the ego vehicle and neighbouring vehicles are considered but also the connections between other vehicles, which are connected to the ego vehicle, are considered. This implies all possible connections between all the vehicles, which are directly connected to the ego vehicle. Therefore, in terms of measurements cluster architecture is the super set of the star architecture. This can be better understood from Figure 7.2(b). Again, the black is the ego-vehicle with direct connection to the neighbouring vehicles also shown with black lines. However, the cluster architecture also considers the communication between other vehicles plotted in dotted black for calculation of state estimation of ego-vehicle.



(a) (b)

Figure 7.2: Ego vehicle is in black. (a) Star Architecture: Only direct connection between ego vehicle and the neighbouring vehicles (b) Cluster Architecture: All possible connections between all the vehicles, which are directly connected to the ego vehicle.

In this chapter, direct distance measurement using DSRC is considered, hence only the scenario of star architecture is simulated. The cluster architecture needs more information to be sent from the participating vehicles over the data connection.

A highway scenario with **16 vehicles** with an infrastructure radar mounted across the lanes of the highway is simulated as shown in Figure 7.3. The infrastructure radar is assumed to have equal field of view in both the directions. The simulation is run for 200 time steps.

Simulated vehicles provide their own odometer measurements and GPS location in global coordinates. As an SME factor is used to correlate the vehicles, the position information of radar is assumed to be known. Hence, the infrastructure radar provides measurements after the transformation to the global coordinates for vehicles in its field of view without performing any data association. All the simulated sensors are assumed to have zero mean Gaussian Noises. The covarainces assumed are as $diag\,[1.0, 1.0]$, $diag\,[10.0, 10.0]$ and $diag\,[0.5, 0.5]$ for the odometer, the GPS and the radar respectively.

Alan et. al. [78] point the error for various ranging methodologies using DSRC from as good as $1\,m$ and worse even up to $10\,m$. So instead of using one error value, the simulations is run for error in range measurement between $1\,m - 6\,m$. Also, as already mentioned in the previous section, the DSRC signals can have the range up to $1000\,m$ and is highly dependent on the Line-of-Sight conditions and obstacles. Therefore, $1000\,m$ is difficult to achieve. For the simulation, DSRC range of the ego vehicle (depicted in black in Figure 7.3 is assumed from $100\,m$ to $600\,m$.

Results from the simulation are compared three ways, between:

- The fused trajectory only using odometer and GPS.
- The fused trajectory for odometer, GPS and SME factor; and
- The fused trajectory for odometer, GPS, SME factor and DSRC range factor.

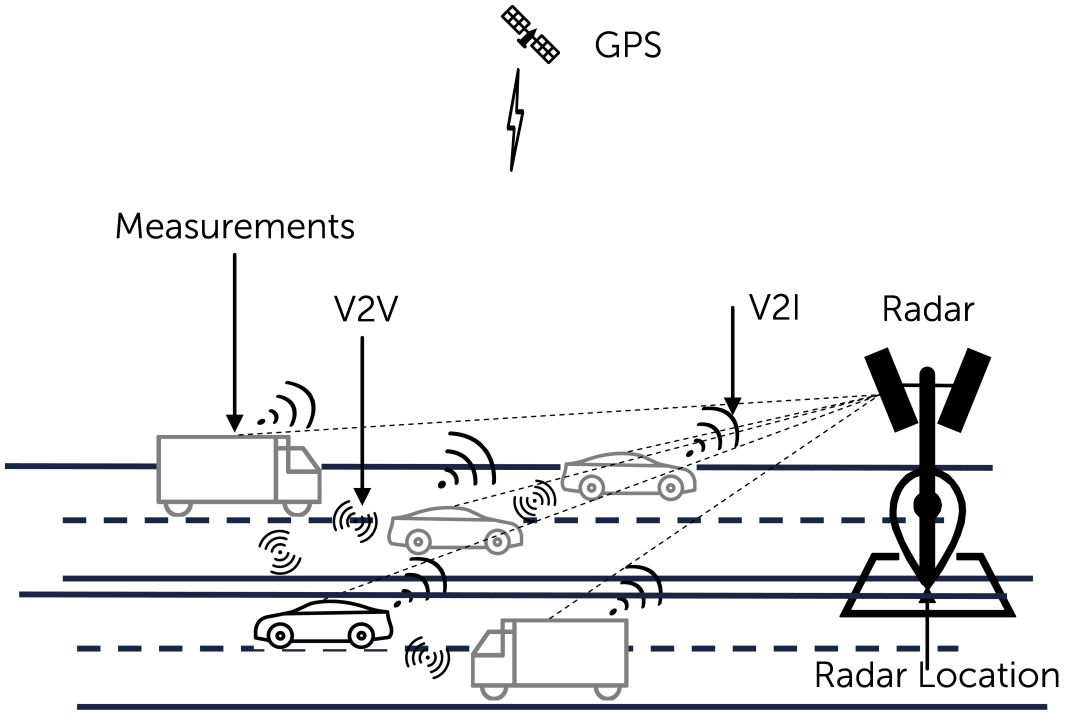


Figure 7.3: Simulation scenario of a highway with vehicles. Simulated radar is mounted above the ground on a beam across the highway. The radar is assumed to have a field of view in both the directions. The vehicles represent the approximate state of their trajectory at some time t.

Then the above experiments are repeated with different error in measurements and different DSRC ranges.

To clearly analyse the benefits of adding DSRC factors, a Monte-Carlo simulation of 1000 iterations is used. To avoid the influence of any other sensor measurement, the simulated measurements from the odometer, GPS and radar are kept same in all the 1000 iterations.

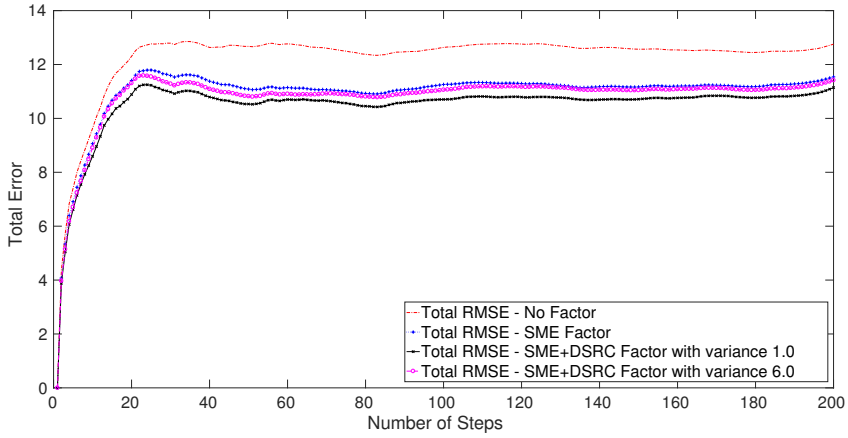
Root Mean Square Error: Like previously proposed factors, we compare the performance using Root Mean Square Error. That is,

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{GroundTruth}})^2} \tag{7.8}$$

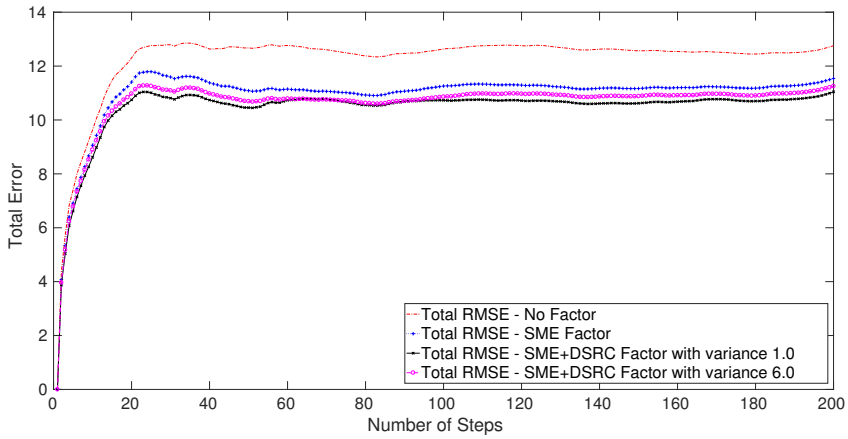
For details of RMSE and the above equation refer to Section 5.7.2.

7.7.2 Results

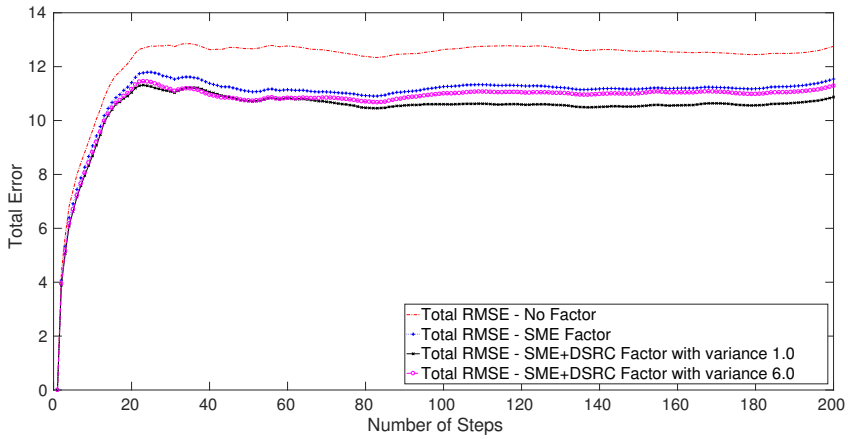
Figures 7.4 and 7.5 show the RMSE Error for one simulation run of the 1000 Monte Carlo simulations. Figures 7.4(a) to 7.4(c) show the total system RMSE results for the simulation of the DSRC range of $100\,m$, $200\,m$, and $300\,m$. Figures 7.5(a) to 7.5(c) show the total system RMSE results for the simulation of the DSRC range of $400\,m$, $500\,m$, and $600\,m$. For each DSRC range the RMSE error of $1\,m$ and $6\,m$ against the case of no factor and SME factor is plotted. From each of the figures it is can be seen that the combination of SME and DSRC factors give a superior performance than the other two. Although the



(a)

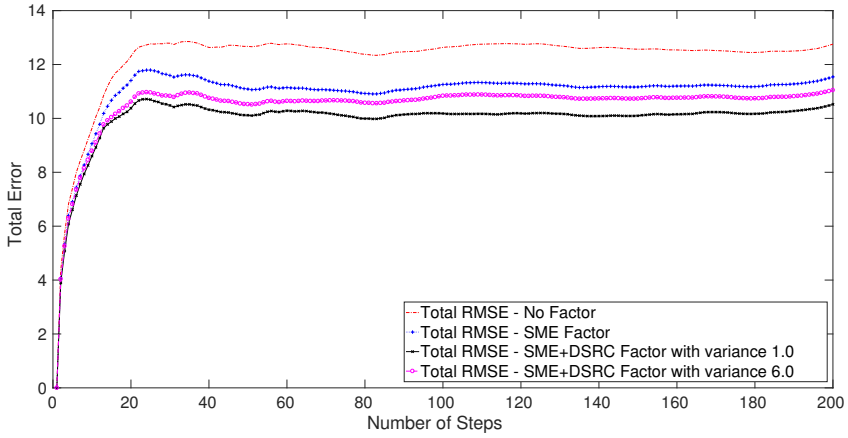


(b)

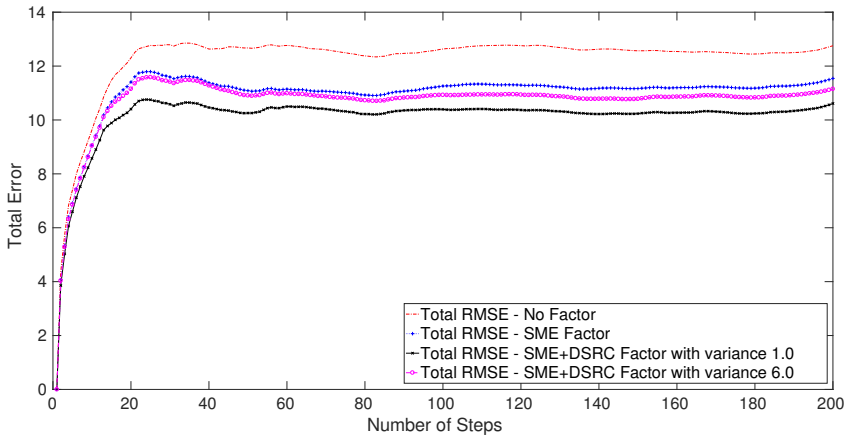


(c)

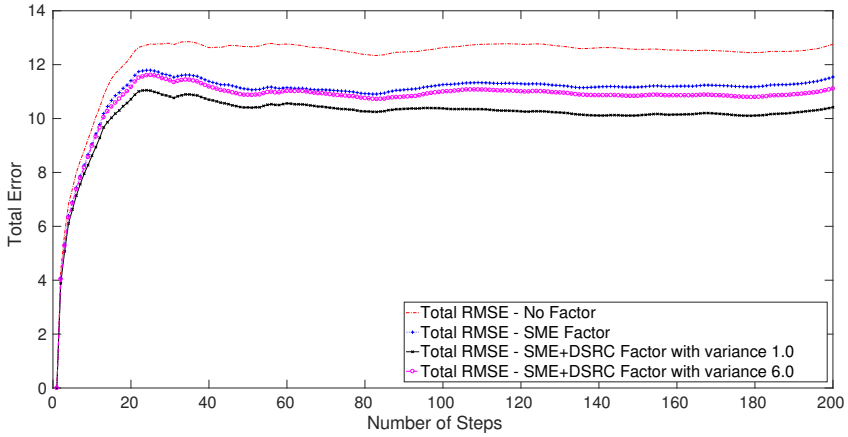
Figure 7.4: (a) Total system RMSE with DSRC range as 100 m. (b) Total system RMSE with DSRC range as 200 m. (c) Total system RMSE with DSRC range as 300 m.



(a)



(b)



(c)

Figure 7.5: (a) Total system RMSE with DSRC range as 400 m. (b) Total system RMSE with DSRC range as 500 m. (c) Total system RMSE with DSRC range as 600 m.

Table 7.1: Average total system RMSE values after 200 steps for different DSRC ranges and different errors for 1000 iterations. This is for **16 vehicles** and per vehicle RMSE value is lower.

No Additional Factor	No DSRC Factor	DSRC Range (m)	Standard Deviation (m)						Average Number of DSRC Range Factors
			1	2	3	4	5	6	
12.7551	11.5419	100	11.1620	11.1944	11.2368	11.2810	11.3283	11.3571	515
12.7551	11.5419	200	11.0086	11.0841	11.1594	11.2210	11.2818	11.3270	633
12.7551	11.5419	300	10.8503	10.9263	11.0065	11.0840	11.1600	11.2199	762
12.7551	11.5419	400	10.7180	10.8004	10.8915	10.9846	11.0678	11.1436	913
12.7551	11.5419	500	10.5989	10.7162	10.8202	10.9219	11.0147	11.0954	1052
12.7551	11.5419	600	10.4531	10.5961	10.7208	10.8378	10.9368	11.0285	1166

Table 7.2: Average number of DSRC Range Factors used for 1000 iterations for **16 vehicles**.

DSRC Range (m)	Standard Deviation (m)						Average for this range
	1	2	3	4	5	6	
100	515.495	515.362	515.190	515.214	515.135	515.342	515
200	633.387	633.824	633.939	633.900	633.800	633.724	633
300	762.224	762.539	762.578	762.630	762.626	762.610	761
400	912.082	912.577	912.909	913.055	913.055	912.964	912
500	1051.732	1051.600	1051.600	1051.761	1051.761	1051.998	1052
600	1166.163	1166.180	1166.391	1166.615	1166.535	1166.498	1166

SME incorporates full information but adding DSRC range factor adds further information of distance between the vehicles to the given factor graph and hence achieves better performance.

The total system RMSE for the $1\,m$ and $6\,m$ for all the DSRC ranges is analysed because the graphs for various standard deviations are very close and is difficult to observe a statistically significant gain. This can be clearly seen in Table 7.1. The table shows the average final RMSE values obtained from Monte Carlo simulation of 1000 iterations. The simulation was run for DSRC range for $100\,m - 600\,m$, for the DSRC error ranging between $1\,m - 6\,m$. The first column shows the total system RMSE error without any additional factor. The second column shows the total system RMSE error with only SME factor and no DSRC range factor. Both the first and the second columns show the same values because as already mentioned, the simulations are run keeping the odometer, the GPS and the radar measurements constant and only DSRC ranges are simulated. Therefore, in the absence of the DSRC range factor there is no change in the total system RMSE errors for the first and the second columns. Also mentioned in the last column are the average number of DSRC range factors added for the various DSRC ranges.

It can be seen from Table 7.1 that:

- For each of the DSRC range ($100\,m - 600\,m$), with an increase in the DSRC range error ($1\,m - 6\,m$), generally the RMSE value increases. This is expected and implies that the lower is the error in the measurement, the more accurate information it adds for the state estimation and hence the final calculated state estimate is closer to the ground truth.

- For each DSRC range error ($1\,m - 6\,m$) in the measurement, with an increase in the range ($100\,m - 600\,m$), generally the RMSE value decreases. This is also expected as with an increase in the DSRC range for the same DSRC range error in the measurement implies more DSRC range factors can be added, thereby contributing more information to the factor graph, and enabling an improved state estimate. This is confirmed by the last column of the table, which shows the average number of DSRC range factors added to the factor graph for the given range.

Table 7.1 also presents some more interesting results:

- It can be seen that the final total system RMSE value for the DSRC range of $100\,m$ with an error of $1\,m$ is 11.1620. Moreover, for the DSRC range of $600\,m$ with an error of $6\,m$ is 11.0285. The error reduces by almost 1.2%. This general trend of the reduction in error can be observed in the full table. Hence, it can be deduced that for a little increase in DSRC range error for a bigger DSRC range can perform better than the smaller DSRC range error with the smaller DSRC range. This happens because as the number of factors added to the system increase and hence also increases the information added to the graph. This helps in reducing the overall error in the system. However, this observation is true for a linear increase in the error and cannot be generalized because if errors increase non-linearly there may not be a significant gain.

- For a lower DSRC range with a high DSRC range error implies lowering the number of detected vehicles around the ego vehicle and increasing the total system error. This is seen for the range $100\,m$ with the DSRC range error of $6\,m$. This error is still lower than when there is no DSRC factor added to the system. In fact, for any case of DSRC range and DSRC range error, the average system error is always lower than with only SME factor and no DSRC factor. Therefore, total error

is still upper bounded by the error of SME factor, which would be the case that no vehicle is present in the DSRC range of the ego vehicle.

The last coloumn of Table (7.1) shows the average number of DSRC range factors added to that particular DSRC range. The complete average DSRC range factors can be seen in Table 7.2.

7.8 Summary

The DSRC factor presented in this chapter provides another way of improved state estimation using the DSRC range estimation. In addition to the DSRC, radar measurements from the infrastructure are used as SME factors. The idea presented in this chapter goes in the direction of improving the state estimation using a complete V2X communication mechanisms. With the presented technique, an increase in the precision of the localization of the ego and the surrounding vehicles is achieved. This increase happens because of the increase in the information content which effectively reduces the uncertainty and improves the state estimation. Precise localization increases the effectiveness of various technologies (like collision avoidance and path planning) used in semi and fully autonomous vehicles. By using SME, one of the critical problems of data association in cooperative localization is also addressed. Other challenges of bandwidth issue and scalability are also addressed. In the simulation, the trade off of higher range vs higher error rate in the range measurement using DSRC is also demonstrated.

The DSRC data exchange capability can be exploited to have a true V2V communication, in order to get even better state estimates of the ego vehicle using factor graphs. That would help in the cluster architecture as shown in Figure 7.2(b).

Chapter 8

Clutter

8.1 Motivation

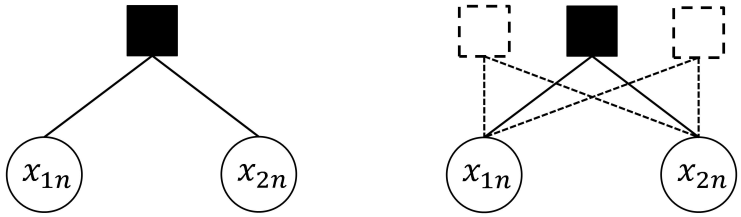
Previous chapters introduced the novel concept of the topology factor, the SME factor and the DSRC range factor. All of the proposed factors form the part of the proposed solution of cooperative localization using V2X communication. The topology and the SME factor are formulated from radar measurements which is present as an infrastructure sensor. While using the radar, an assumption of clutter free environment has been mentioned in Chapter 3 and point 4[1]. However, in reality this is not the case.

Clutter in the domain of radar is a known phenomenon [79]. The clutter constitutes of measurements from a radar, which do not correspond to any target, i.e. they are the false positives. These can be understood as *Ghost Objects* as they do not map to any real object. The clutter measurements using the topology or the SME factors effects adversely on the state estimation. In this chapter, the problem of clutter for the topology factor is analysed and addressed.

Consider an example for two vehicles in the field of view of an infrastructure radar and the radar returns 3 measurements. That is, an extra clutter measurement, which does not belong to any real target. There is no way to find, without performing a measurement-to-target association or the data association, which measurement does not belong to any target. This implies 3 topology factors (${}^3C_2 = 3$) will be formulated from all the radar measurements and added between the two vehicle states. That is, 2 of the 3 topology factors are composed of a clutter measurement and contribute false information for the final state estimation. The problem can be understood from Figure 8.1.

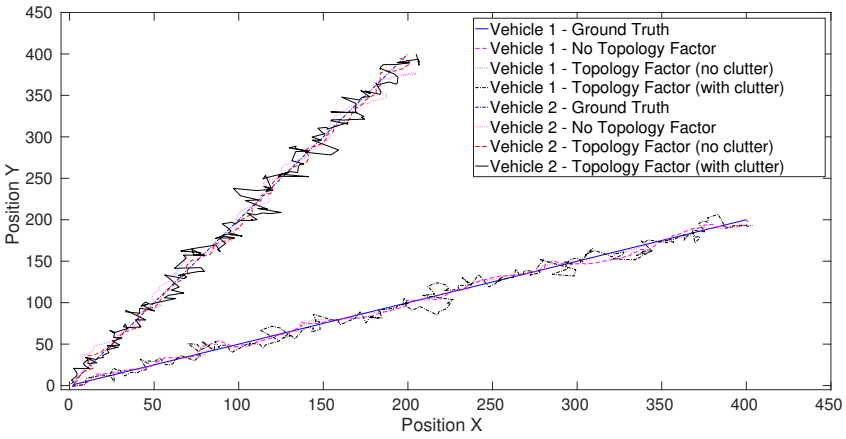
The problem becomes worse for higher number of targets or higher number of clutter measurements. For example, for 1 clutter measurement for 4 targets results in 5 topology factors (${}^5C_4 = 5$), 4 of which (80%) are based on the clutter measurement. For 2 clutter measurements for 2 targets, gives 6 topology factors (${}^4C_2 = 6$), of which 5 are false.

[1] Clutter poses a significant challenge as any false measurements in the system negatively effect the state estimation. Skolnik [79] explains various clutter characteristics and its impact on the radar performance. One of the common problems experienced by such dynamic target tracking radars is that of 'Ghost Objects', i.e. measurements which do not correspond to any true object but are presented by radar as valid measurements. In the research work presented in this thesis, first a clutter free environment is assumed. Nevertheless, in Chapter 8, the impact of clutter on the state estimation is considered. It goes further introducing methodology to address the challenges of clutter, when combined with some of the proposed methodologies earlier presented in this Thesis.

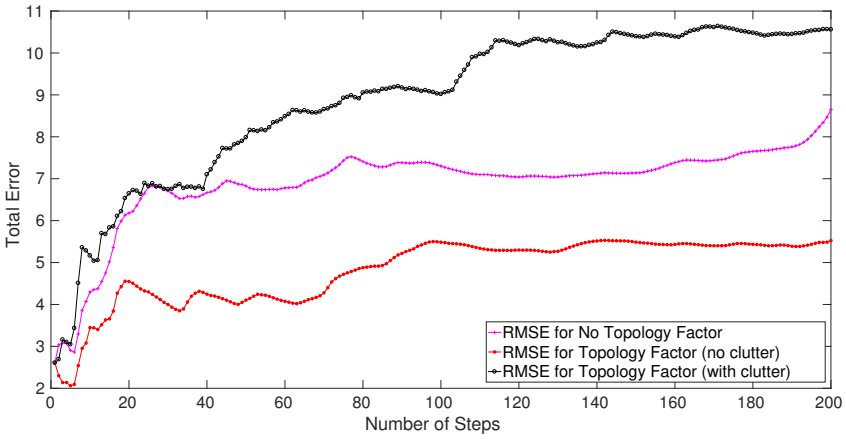


(a) (b)

Figure 8.1: State of two vehicles and the topology factor. Squares represents the topology factor constructed with radar measurements. The circle with X_{yz} represents the z^{th} state estimation of the y^{th} target. (a) The ideal case of no clutter, there exists only one topology factor for the two targets. (b) The radar gives three measurements with one of them being a clutter; hence, there are three possible topology factors. The black filled square represents the topology factor from the true target measurements and the empty square with dotted lines represent the ones with the clutter measurements.



(a)



(b)

Figure 8.2: (a) Trajectories of two vehicles. (b) Corresponding complete system RMSE values.

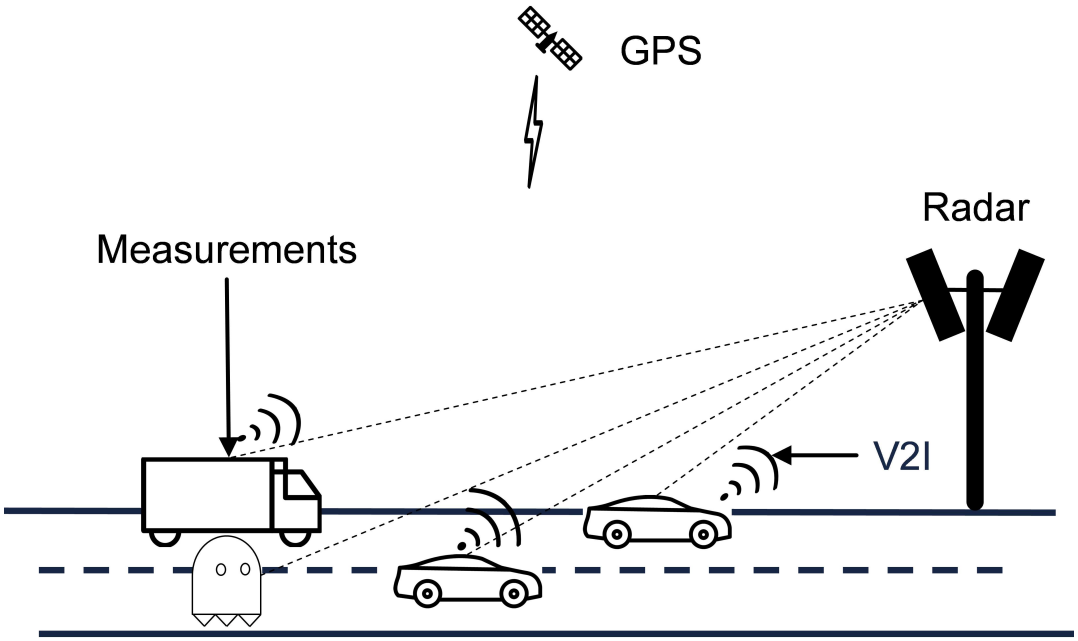


Figure 8.3: Radar results in four measurements, three from vehicles and one from "Ghost Object"

The result from the topology factor in presence of clutter measurements for two vehicles can be seen in Figure 8.2. In presence of clutter additional topology factors get added which increase the total RMSE of the system significantly (seen in Figure 8.2(b)). The total error of the trajectory without the use of topology factor is better than the one using all the topology factors including the ones formulated from the clutter.

This chapter goes one-step further in the direction of realistic scenario and relaxes the constraint of clutter free environment. The scenario can be understood from Figure 8.3. It can be seen that a radar results in four measurements, three from the vehicles and one from the Ghost Object. The research presented in this chapter is in the direction of formulating robust topology factors and has been published in following conferences:

1. Gulati et. al.: Robust cooperative localization in a dynamic environment using factor graphs and Probability Data Association filter. 20th International Conference on Information Fusion (FUSION), 2017.

2. Gulati et. al.: Robust vehicle Infrastructure Cooperative Localization in Presence of Clutter. 21st International Conference on Information Fusion (FUSION), 2018 (Accepted).

8.1.1 Solutions

The problem of clutter for a radar is mostly handled at the hardware. However, despite efficient hardware filters some clutter still is presented as the valid measurements from the radar sensor. Various fusion mechanisms handle the clutter differently.

Random Finite State (RFS) based tracking algorithms have frequency threshold parameter which keeps any newly initialized track in temporary phase. Only after the measurement is seen for a minimum

threshold, is the track finalised otherwise it is deleted.

Multi-Hypothesis Tracking maintains simultaneous trees of all possible tracks and prunes them regularly to remove the falsely initialized paths.

Probability Data Association Filter discards any measurements, which lie outside the configured threshold called Gate, and assigns probabilities to the remaining measurements. The final measurement is then the weighted sum of the measurements inside the Gate. Therefore, it can partially filter the clutter measurements.

SLAM based solutions rely heavily on the front end to provide the correct measurements. Sünderhauf et. al. [105] show the effect when outliers like data association errors and the false loop closures are formulated in the graph. They used the Manhattan Data Set [106] compromising of 3500 poses and 2099 loop closures, and corrupted the dataset by introducing 100 additional wrong loop closures. The basic SLAM set up could not converge to the correct solution. They proposed the solution of "switchable constraints" which adds additional variable which controls the contribution of each loop closure, and "switches off" the loop closures which increase the total error in the system.

8.2 SLAM Formulation

SLAM formulation to handle the clutter remains same as represented in Equation (5.1), that is:

$$\mathbb{X}^* = \arg\min_X \Bigg(\underbrace{\sum_1^n \|(\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i\|^2_{\Sigma_i^o}}_{\text{Odometer}} + \underbrace{\sum_1^m \|(\mathrm{g}(x_{i_k})) - z_k^g\|^2_{\Sigma_k^g}}_{\text{GPS}} + \underbrace{\sum_1^l \|(\mathrm{r}(x_{i_k})) - z_k^r\|^2_{\Sigma_k^r}}_{\text{Radar}} \Bigg) \tag{8.1}$$

where the mathematical terms describing odometer and GPS are same as previously defined. The $\mathrm{r}(\cdot)$ is the topology factor function (Equation (5.9)) which represent the state of the vehicle which has to be estimated, i.e. the location. z_k^r is the measurement generated from the mapping of actual radar measurements to the state of the vehicle. Σ_k^r is the covariance matrix for the topology factor.

8.3 Robust Topology Factor Using Probability Data Association Filter

8.3.1 Robust Factors

Sünderhauf et. al. [105] proposed the concept of weights which can be attached to the least square formulation which guides the optimizer to either consider the formulation or completely remove it during

the optimization process. Attaching similar weights to topology factor in Equation (8.1) would result in:

$$\mathbb{X}^* = \arg\min_X \left(\sum_1^n \|(\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i\|^2_{\Sigma_i^o} + \sum_1^m \|(\mathrm{g}(x_{i_k})) - z_k^g\|^2_{\Sigma_k^g} + \sum_1^l \boldsymbol{\beta}^{\mathbf{k}}_{\mathbf{i_k}} \cdot \|(\mathrm{r}(x_{i_k})) - z_k^r\|^2_{\Sigma_k^r} \right) \tag{8.2}$$

where $\beta^k_{i_k}$ is the weight for topology factor at each position i_k and all other terms have the same meaning as in Equation 8.1. The weight can range between 0 and 1, thereby increasing or decreasing the total error contributed by that factor. The proposed solution by Sünderhauf et. al. in [105] is to let the optimizer control the value of $\beta^k_{i_k}$. The first solution proposed in this thesis is to obtain it using Probability Data Association (PDA) Filter. The next section gives a quick overview of the concepts of PDA Filter.

8.3.2 Probability Data Association Filter

Probability Data Association Filter (PDA Filter) is a well-known Bayesian target-tracking filter. Instead of using one measurement to update the state, it utilizes maximum possible measurements of the target. As this is not the topic of this thesis, here only a brief overview of the process is mentioned. For detailed explanation the reader can refer to [107] and [108].

Assume only one target having a linear motion is present and the track for the target has been initialized. A measurement validation region, called a Gate, is constructed around the predicted region. All the actual measurements from the sensor outside this Gate are dropped. Remaining measurements inside this Gate are considered validated measurements. For each of the validated measurements an associative probability is calculated which is used as the weight for the calculation of the final measurement.

If z_t be the set of validated multiple observations at a time t and m_t be the number of observations at time t. Then we get:

$$\mathbb{Z}_t = \{z_t^1, \cdots, z_t^{m_t}\} = \{z_t^i\}_{i=1}^{m_t} \tag{8.3}$$

The association probabilities for the set z_t are then calculated as

$$\beta_t^i = \frac{e_i}{\sum_{i=0}^{m_t} e_i}, \tag{8.4}$$

where,

$$e_i = \exp\left(-\frac{1}{2} \cdot (\tilde{z}_t^i)^T \cdot S_t^{-1} \cdot \tilde{z}_t^i \right), i = 1, \ldots, m_t \tag{8.5}$$

$$e_0 = \left(\frac{2\pi}{\gamma} \right)^{n_z/2} \cdot m_t \cdot c_{n_z} \cdot \frac{(1 - P_D P_G)}{P_D} \tag{8.6}$$

where S_t, P_G and P_D denote the innovation covariance, the generation probability (of the target originated measurement falling within the validation gate) and the detection probability (the correct measurement is detected) respectively. $\tilde{z}_t^i$ denotes the innovation. c_{n_z} is the volume of the n_z dimensional unit hypersphere ($c_1 = 2, c_2 = \pi, c_3 = 4\pi/3, etc.$). γ is the threshold for the validated measurements by defining the following validation region:

$$\{z^i : (\tilde{z}_t^i)^T S_t^{-1} \tilde{z}_t^i \leq \gamma\} \tag{8.7}$$

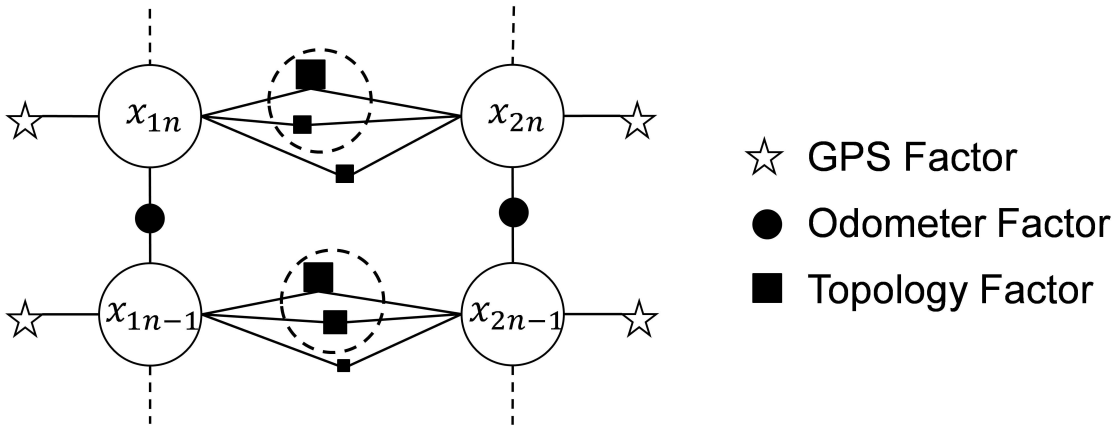


Figure 8.4: (a) Topology factor implemented with PDA Filter. The dotted circle represents the Gate. The size of the square inside the Gate represents the weight of that topology factor allocated by the PDA Filter.

8.3.3 Proposed Solution

Topology measurements are single dimensional values. The proposed solution uses PDA Filter to track these topology measurements and filter the outlier values. The remaining validated values are assigned an association probability, which is used as the weight, $\beta_{i_k}^k$, for each validated topology measurement in Equation 8.2.

The solution can be understood from Figure 8.4. The figure shows the state of two vehicles at $(n-1)^{st}$ and n^{th} time step. The radar results in more than two measurements (three in this case); therefore, it has three topology factors, which are plotted as squares. The dotted circle represents the Gate for the topology measurements, i.e. the topology factors, which satisfies the Gate, are considered and rest are removed. For each of the validated measurement probability is calculated using PDA Filter. In the figure, the size of the squares inside the Gate represents the probability of the measurement. Hence, only validated topology factors contribute to the final solution based on the weights.

8.3.4 Jacobians for the Non-Linear Least Square Optimization

Since it a topology factor, the Jacobian derived in Chapter 5 Section 5.5 are used here. The Jacobians for odometer, GPS and topology factor are represented in Equations 5.12, 5.15 and 5.18 respectively.

8.3.5 Measurement Uncertainties / Covariances

Like Jacobians, covariances derived in Chapter 5 Section 5.6 are also used here. Odometer and GPS covariances are obtained from the sensor manufacturers. Topology covariance from Equation 5.27 is used.

8.3.6 Evaluation

8.3.6.1 System Setup

The above approach has been only evaluated on a simulated data set. The simulation is implemented with two vehicles for 200 steps.

Since the proposed methodology uses the topology factor formulation, the simulated sensors are odometer, GPS and an infrastructure radar. All the simulated sensors have zero mean Gaussian Noises. The covariance assumed are $diag\,[1.0, 1.0]$, $diag\,[15.0, 15.0]$ for odometer and GPS respectively. The radar has one valid reading with covariance $diag\,[0.5, 0.5]$ and the remaining is clutter having uniform distribution. For the tests $1 - 3$ measurements are randomly generated against each vehicle. The time step interval is taken as 1.

For tracking the topology measurements, PDA Filter has the detection probability and the generation probability of 0.99. It is assumed that the corresponding track detection and initialization has been already successful. Then using PDA Filter the probabilities to the resulting topologies for a given state are assigned.

The results from the simulation are compared four ways, between:

1. The fused trajectory only using odometer and GPS measurements.
2. The fused trajectory for odometer, GPS measurements and topology factor with probability 1 for each state (assuming no clutter).
3. The fused trajectory for odometer, GPS measurements and multiple topology factors for each state (assuming clutter) with probability 1 assigned to each of them. This implies we incorporate all the topology factors resulting from radar.
4. The fused trajectory for odometer, GPS measurements and multiple topology factors for each state (assuming clutter) with probability assigned to each of them using PDA Filter (the proposed solution).

Two such simulations are performed, one set with linear trajectories and the second with random trajectories.

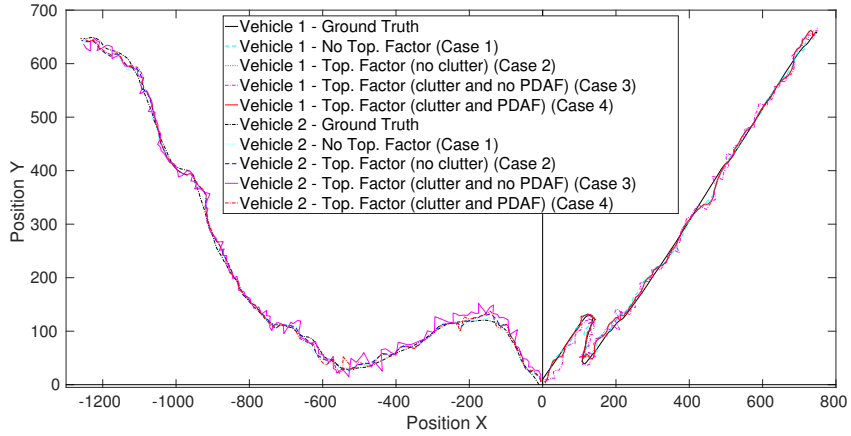
Root Mean Square Error: Like previously proposed factors, we compare the performance using Root Mean Square Error. That is,

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{GroundTruth}})^2} \tag{8.8}$$

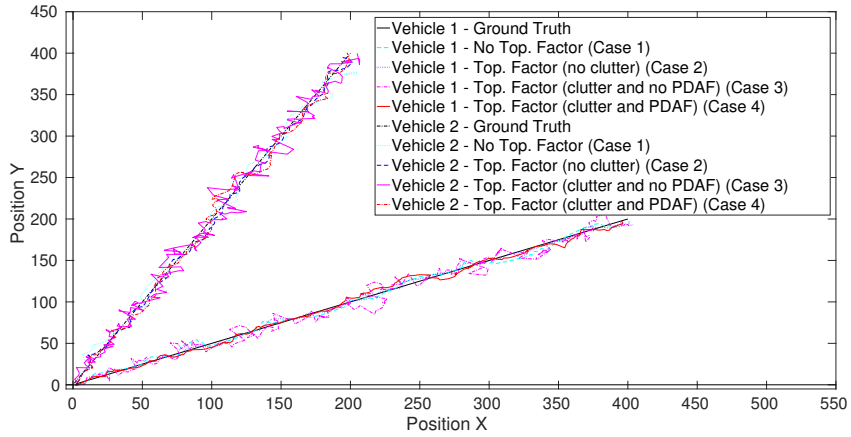
For details of RMSE and the above equation refer to Section 5.7.2.

8.3.6.2 Simulation Results

Figures 8.5(a) and 8.5(b) show the ground truth, and the trajectories from the four methods for two vehicles. It can be seen that the trajectory for the proposed approach (Case 4) is closer to the ground truth and performs better than no topology (Case 1). However, it performs little worse when compared to



(a)

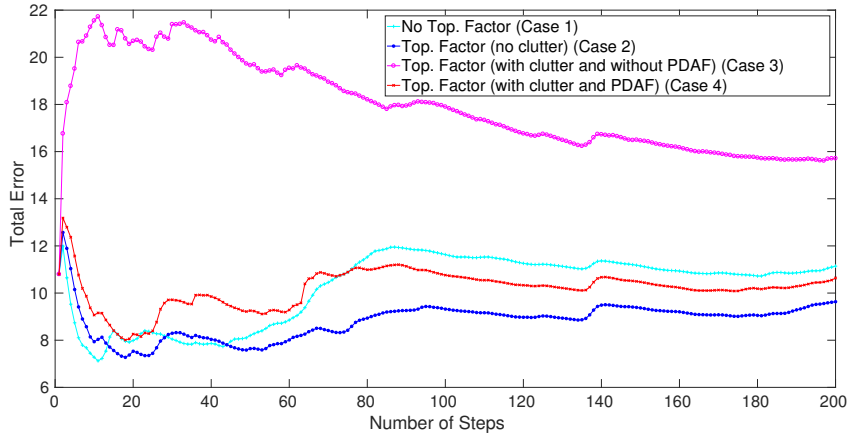


(b)

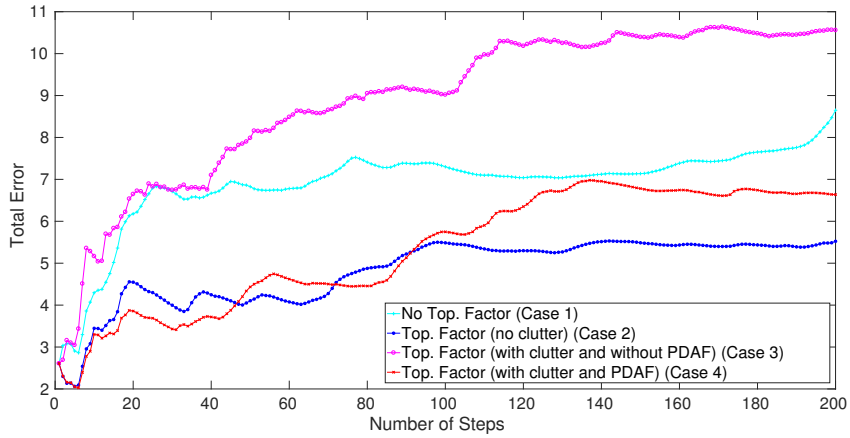
Figure 8.5: Ground truth and fused trajectories for two-vehicle simulation. Case 2 without clutter (unrealistic scenario) and with topology, results in the trajectory near to the ground truth. Case 3 with clutter (realistic scenario) includes all clutter measurements from radar and with topology, results in the worst trajectory. Case 4 with clutter (realistic scenario) assigned weights using a PDA Filter and with topology, although is worse than Case 2, results in a better trajectory than the Case 3 which is a realistic scenario and Case 1 (which does not use any topology).

the Case 2 where it uses only one measurement with the value $\beta_{i_k}^k$ (or the probability) as 1. Nevertheless, the Case 2 also assumes no clutter, which is untrue in real environments. In addition, if we use all the measurements (including clutter) from the radar (Case 3), the trajectory is the worst.

This can be further verified by the total RMSE values of the system as plotted in graphs in Figures 8.6(a) and 8.6(b) for Figures 8.5(a) and 8.5(b) respectively. The RMSE for Case 3 (with clutter and without PDA Filter) is even higher than the Case 1 of no topology factor. This is because clutter significantly increases the error by adding false information to the graph. Although the Case 2 (without clutter) performs the best (as it does not add any false information to the formulated graph) but does not reflect the real environment. Hence, the RMSE of using PDA Filter with cluttered measurements (Case 4) suggests that this is an effective solution for real environments where clutter and additional noise are



(a)



(b)

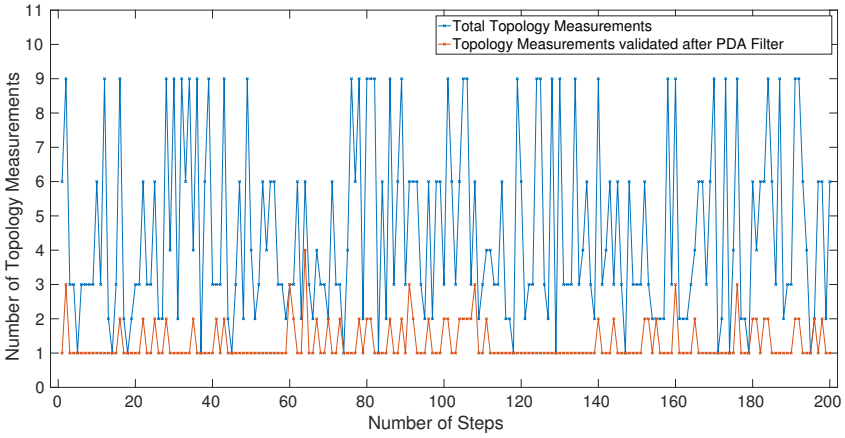
Figure 8.6: RMSE for two-vehicle simulation. Case 2 without clutter (unrealistic scenario) and with topology, has the least error. Case 3 with clutter (realistic scenario) includes all clutter measurements from radar and with topology, has the maximum error. Case 4 with clutter (realistic scenario) assigned weights using a PDA Filter and with topology, although has more error than Case 2, but performs better than Case 3 which a realistic scenario and Case 1 (which does not use any topology).

present.

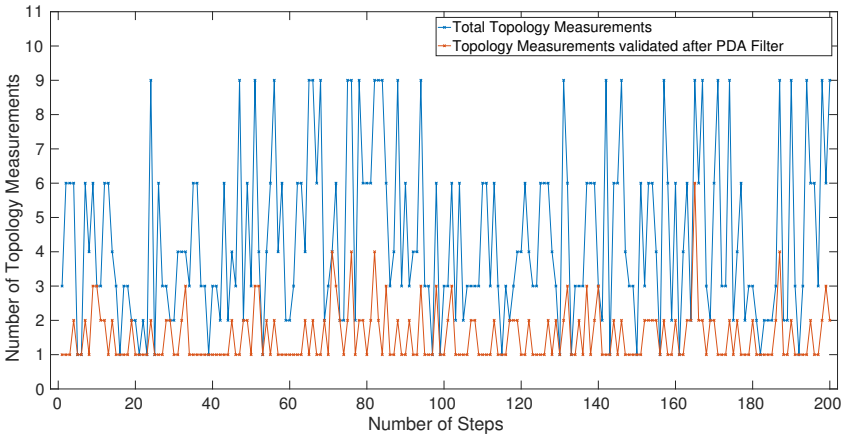
Figures 8.7(a) and 8.7(b) compares the total topology measurements for all the steps against the topology measurements selected by PDA Filter, which are effectively used by the optimizer for estimating the final states for Figure 8.5.

8.3.7 Remarks

The results presented here assume two-vehicle system, thereby resulting in one topology measurement, which can be tracked using a PDA Filter. The clutter is assumed uniform and results in topology measurements, which are assumed to have a linear motion. Therefore, they can be tracked using PDA Filter which uses a linear Kalman Filter. To extend the system to more than two vehicles further investigation needs to be carried out in order to correctly determine the "motion model" of the topology



(a)



(b)

Figure 8.7: Comparison of validated measurements by PDA Filter and the total measurements for Fig. 8.5(a) and Fig. 8.5(b).

Measurements which can be then efficiently tracked using PDA Filter.

8.4 Robust Topology Factors

8.4.1 Motivation

Last Section 8.3 introduced a way to address the problem of clutter using PDA Filter. The simulation was run only on two-vehicle scenario and some of the problems were highlighted in subsection 8.3.7. The following are the challenges:

1. In order to extend the concept of the PDA Filter to address the clutter for any number of vehicles, a generic motion model of the topology measurements has to be developed which can be tracked using PDA Filter. This even if possible, mathematically is difficult to arrive at.

2. Sünderhauf et. al. [105] presented a solution which filtered the outliers without any additional external method.

Therefore, a solution, which is self contained and independent of any kind of external method, would be optimal. Further analysis of radar clutter handling reveals [109, 110, 111, 79] that there exist different clutter models for different radar types. Therefore, extending the previously proposed solution to a generic multi-target scenario may not be feasible.

8.4.2 Clutter from Infrastructure Radar

This section gives a quick overview of the functioning of an infrastructure radar. This helps in better understanding the kind of clutter, which can be observed on a scenario like that of a highway. For detailed functioning of a radar, reader is advised to consult references like the Radar Handbook by Skolnik [79] and online references like [112].

A typical radar deployed in a traffic scenario to oversee $4-8$ lanes is a Ground Moving Target Indicator (GMTI) Radar. As the name suggests, the aim for an GMTI Radar is to detect a moving target and reject the measurements from fixed or slow-moving unwanted targets like buildings, trees or rain. These radar types fall under the Frequency-Modulated Continuous-Wave Radar (FMCW Radar). These are special kind of sensors, which radiate continuous transmission power with change in the frequency. That is, the transmission signal is modulated in frequency. This increase or decrease of the frequency of the sent signal is measured against the change of frequency in the received signal. This helps in measuring the distance of the moving object and the relative velocity simultaneously.

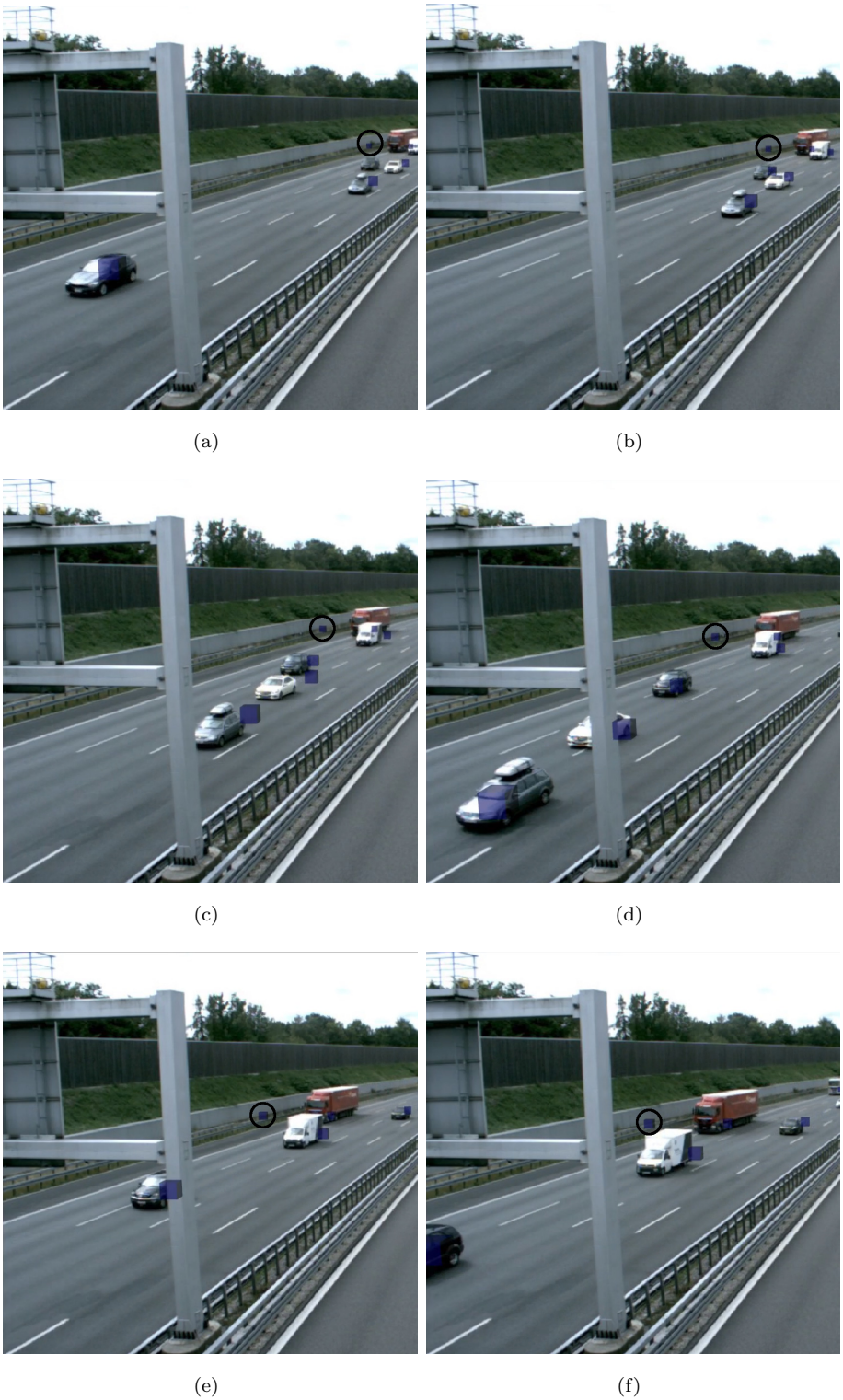
The state-of-the-art MTI Radars use real time sophisticated circuits and physics principles to operate. Therefore, they also have hardware based filters to remove a lot of clutter from the scenarios. Details of such filters can be found in [79]. This is also a topic of active research in radar domain. Despite all the advancements, sometimes high winds, rain and reflections result in detections which could not be filtered at hardware and appear as clutter measurements which do not belong to any real target. This can induce errors in the state estimation. All these clutter measurements which escaped the signal-level sensor filtration process have to be addressed in the high-level fusion process. Therefore, we consider a specific situation (infrastructure radar on a highway) which allows us to make use of higher-level information to get rid of the clutter that usual methods cannot deal with.

To understand the real clutter phenomenon the actual radar data was also analysed, which was collected as part of the Providentia Project [113] on German A9 Highway. Figure 8.8 highlights one set of such data measurements of the radar plotted as cubes on the Camera data. In Figures 8.8(a)$-$8.8(f) the clutter point is highlighted with a black circle.

The contribution of this section is to propose a solution for above kind of realistic clutter scenarios for topology factor for improved state estimation.

8.4.3 Proposed Solution

The topology factor requires the coordinates of the targets detected by the radar. The factor performs optimal when the radar detects the same number of targets as the number of pair of odometer/GPS measurements received from the vehicles. However, in presence of clutter as seen in Figure 8.8, radar will result in more measurements than the targets, but the fusion system will still receive the same number of odometer/GPS pair of measurements. This results in more topology factors between the states, which influence the final state estimation.

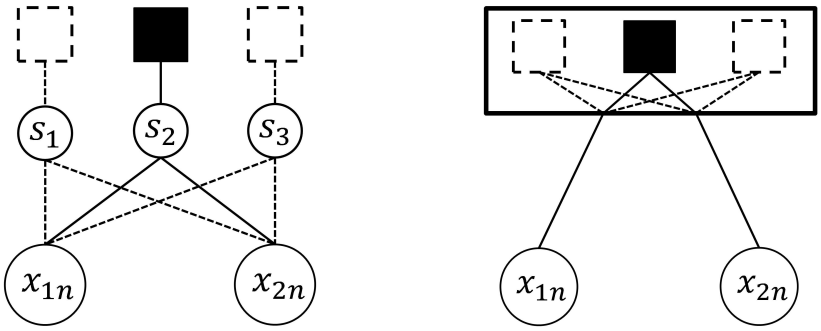


(a) (b)

(c) (d)

(e) (f)

Figure 8.8: Measurements detected by the radar (plotted as cubes) when mapped on the Camera plane highlight the clutter points. The radar could not filter all the possible clutter measurements. In Fig (a) the clutter point is marked with a circle. Its progression can be seen through all the subsequent pictures (b-f) where it is also encircled. Source: [113]



(a) (b)

Figure 8.9: State of two vehicles and the topology factor. The squares represents the topology factor constructed using radar measurements. The circle with X_{yz} represents the z^{th} state estimation of the y^{th} target. The circle with S_i represent the Switch Variable for the i^{th} topology factor. (a) A possible solution using Switch Variables for topology factor for the scenario depiction in 8.1(b) (b) The minimization topology factor which has three *sub-factors* and uses the one which results in least error for the scenario depiction in 8.1(b)

8.4.3.1 Switchable Topology Factor

Sünderhauf et al. [105] proposed the solution of switchable constraints. This added a latent variable which *switches off* the factors that contribute higher errors to the final state estimation. A possible solution would be to apply the switchable constraints for the topology factors, i.e. introduce one switch variable per topology factor. The topology factors contributing higher error to the final state estimates would be switched off, i.e. the ones from the clutter measurements. Applying switchable constraints from [105] on topology factor from Equation (8.2) results in :

$$\mathbb{X}^* = \arg\min_X \Bigg(\underbrace{\sum_1^n \|(\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i\|^2_{\Sigma_i^o}}_{\text{Odometer}} + \underbrace{\sum_1^m \|(\mathrm{g}(x_{i_k})) - z_k^g\|^2_{\Sigma_k^g}}_{\text{GPS}} + \underbrace{\sum_1^l \boldsymbol{\Psi(s_{i_k}^k)} \cdot \|(\mathrm{r}(x_{i_k})) - z_k^r\|^2_{\Sigma_k^r}}_{\text{Radar}} \Bigg) \tag{8.9}$$

where Ψ is the switch function, $s_{i_k}^k$ is the switch variable for the k^{th} factor and all other terms have the same meaning as in Equation 8.1.

Equation 8.9 is the *switchable topology factor*. For each of the formulated topology factor, switch variable is assigned which is controlled by the optimizer. During optimization, the variable can be enabled or disabled.

The solution can be understood from Figure 8.9(a). As seen in Figure 8.1(b), for two vehicles with one clutter measurement from radar, three topology factors are formulated. For a switchable topology factor a switch variable S_i is added for each of the constructed topology factor.

8.4.3.2 Minimization Topology Factor

For scenarios with clutter, there are more measurements from radar than the actual vehicles. If there were m_v actual vehicles and M_v radar measurements with $M_v \geqslant m_v$, then there are ${}^{M_v}C_{m_v}$ possible topology factors. Therefore, topology measurement z_k^r will actually be a member of all possible subsets of ${}^{M_v}C_{m_v}$ elements. Representing the set of topology measurements as $\mathcal{Z}_k$, the radar formulation from Equation 8.2 for one step can be written as:

$$\|(\mathrm{r}(x_{i_k})) - \mathcal{Z}_k^r\|_{\Sigma_k^r}^2 \tag{8.10}$$

where all other terms have the same meaning as in Equation 8.1

Therefore the task of the optimizer is to find the topology measurement subset in $\mathcal{Z}_k^r$ which results in the least optimization error.

Also among all the possible topology measurements, it is known that only one of them is based on the true radar measurement. Therefore, only that should result in the least error.

Hence, instead of the topology factor controlled by a switch variable or by the optimizer, a novel minimization topology factor is proposed as follows:

$$\left\| \min \|(r(x_{i_k})) - \mathcal{Z}_k^r\| \right\|_{\Sigma_k^r}^2 \tag{8.11}$$

where "min" is the minimization factor which returns the minimum error from all the possible *sub-factors* from the set of measurements in $\mathcal{Z}_k$.

Substituting it back in Equation 8.2, the new SLAM formulation becomes:

$$\begin{aligned} \mathbb{X}^* = \arg\min_X \Bigg(& \underbrace{\sum_1^n \|(\mathrm{f}(x_{i-1}, u_{i-1}))_k - x_i\|_{\Sigma_i^o}^2}_{\text{Odometer}} + \underbrace{\sum_1^m \|(\mathrm{g}(x_{i_k})) - z_k^g\|_{\Sigma_k^g}^2}_{\text{GPS}} \\ & + \underbrace{\sum_1^l \left\| \min \|(\mathrm{r}(x_{i_k})) - \mathcal{Z}_k^r\| \right\|_{\Sigma_k^r}^2}_{\text{Radar}} \Bigg) \end{aligned} \tag{8.12}$$

where all the terms have the same meaning as in Equations 8.1 and 8.11.

The solution can be understood from Figure 8.9(b). As seen in Figure 8.1(b), for two vehicles with one clutter measurement from a radar, three topology factors are formulated. Figure 8.9(b) shows the minimization topology factor that contains *sub-factors* for all possible combinations (here 3). Only the one, which results in the least error, is chosen. This way the minimization topology factor partially controls the optimizers behaviour.

8.4.4 Jacobians for the Non-Linear Least Square Optimization

Since a minimization topology factor is also a topology factor, the Jacobian derived in Chapter 5 Section 5.5 are also used here. The Jacobians for odometer, GPS and topology factor are represented in Equations 5.12, 5.15 and 5.18 respectively.

8.4.5 Measurement Uncertainties / Covariances

Like Chapter 5 Section 5.6, odometer and GPS covariances are obtained from the sensor manufacturers.

8.4.5.1 Minimization Topology Factor Covariance

Topology covariance from Equation 5.27 is dependent on the actual measurements. Therefore, each of the composed multiple sub-factors in minimization topology factor will have different error standard deviations or the covariances. However, the minimization topology factor can only have one covariance. This covariance should be large enough that it suffices all the sub-factors.

The biggest covariance from all possible covariances can cover all the possible covariances and therefore, we use that as the covariance for the minimization factor. If $\mathcal{C}$ represents the set of all possible covariances defined for sub-factors using Equation 5.27, then covariance for the minimization topology factor can be written as:

$$\Sigma_t^{\max} = \max(\mathcal{C}) \tag{8.13}$$

8.4.6 Evaluation

8.4.6.1 System Setup

The proposed solution is evaluated on a simulated data set, which is based on real radar observations on German A9 Highway. The simulation is implemented with up to four vehicles on a highway for 200 steps. Since the proposed methodology uses topology factor, the simulated sensors are odometer, GPS and an infrastructure radar. All the simulated sensors have zero mean Gaussian Noises. The covariance assumed are $diag\,[1.0, 1.0]$, $diag\,[10.0, 10.0]$ for odometer and GPS respectively. The radar has one valid reading with covariance $diag\,[0.1, 0.1]$.

As already mentioned in Section 8.4.2, MTI Radars are used in infrastructure and are capable of filtering the static clutter using hardware filters. The big risk of clutter comes from "Ghost Objects" which result from motion originating from phenomena like high winds, rain or reflections. From our observations of the recorded data, we notice the clutter is most likely seen moving parallel to the vehicle. Therefore, clutter measurements are simulated which move parallel to the vehicle movement.

Various characteristics for the simulated clutter are defined rigorously. The clutter can appear along any randomly chosen vehicle (any of the 4 simulated vehicles). The length of the clutter trajectory is also random with 5 to 10 time steps. The clutter can occur on any side of the simulated vehicle. The clutter measurements are divided into sets of 1 to 5. Therefore, there are at least 5 clutter measurements in one simulation. The maximum number of simultaneous clutter measurements at any given time step is 3. So theoretically, the maximum number of clutter points in the system is $3 \times 5 \times 10 = 150$.

The simulation also makes sure that clutter measurements should not jump around or across the simulated vehicles at every time step. In addition, after one set of clutter is generated, the system waits for at least 30 time steps before a new set may be randomly generated. We randomize various properties of the clutter measurements in order to get a realistic simulation of the system.

A random sample of ground truth and the corresponding clutter measurements of up to 1 simultaneous clutter measurements for 2 vehicles on a highway, generated using the above methodology, is shown in

Figure 8.10(a). Figure 8.10(b) shows up to 2 simultaneous clutter measurements for 3 vehicles on a highway. In addition, Figure 8.10(c) displays up to 3 simultaneous clutter measurements for 4 vehicles on a highway. By "up to 3 simultaneous clutter measurement", we mean that the simulator randomly generates the clutter measurements, therefore it can be only 1 or 2 or 3 measurements.

The results from the simulation are compared and contrasted in five ways. Various cases for the comparison are:

1. The fused trajectory only using odometer and GPS measurements.
2. The fused trajectory for odometer, GPS measurements and topology factor for each state (assuming no clutter).
3. The fused trajectory for odometer, GPS measurements and multiple topology factors for each state (assuming clutter). This implies all the topology factors resulting from radar are added to the graph
4. The fused trajectory for odometer, GPS measurements and multiple topology factors for each state (assuming clutter) with switchable constraints. That the switchable topology factor is used. The switch variables control the contribution of each of the topology factor. The linear version of the switch function is used.
5. The fused trajectory for odometer, GPS measurements and minimization topology factor with all possible topology factors as sub-factors for each state (assuming clutter).

A Monte Carlo simulation of the above-mentioned cases is performed 1000 times and results are analysed.

Root Mean Square Error: Like previously proposed factors, we compare the performance using Root Mean Square Error. That is,

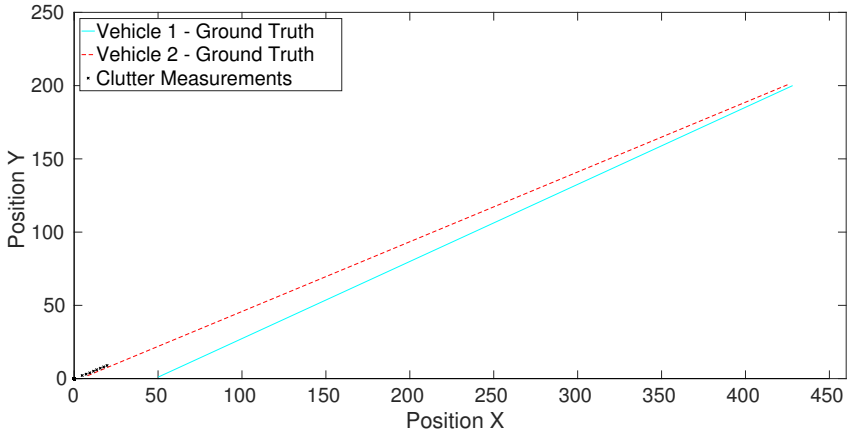
$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{GroundTruth}})^2} \tag{8.14}$$

For details of RMSE and the above equation refer to Section 5.7.2.

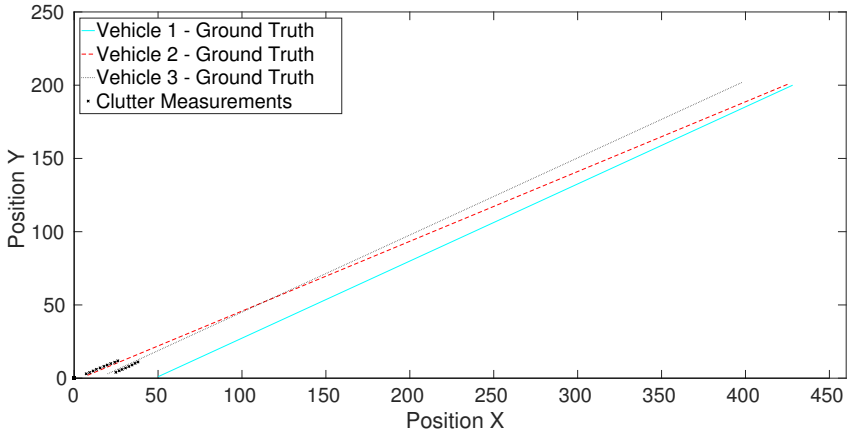
8.4.6.2 Simulation Results

Figures 8.11(a), 8.11(b) and 8.11(c) show the trajectories of two vehicles on the highway with up to 1, 2 and 3 clutter measurements. It is not very clear but on close inspection, it can be seen that trajectories resulting from use of minimization topology factor (Case 5) are closer to the ground truth than the one resulting with the switchable topology factor (Case 4) and when all possible topology factors are used (Case 3).

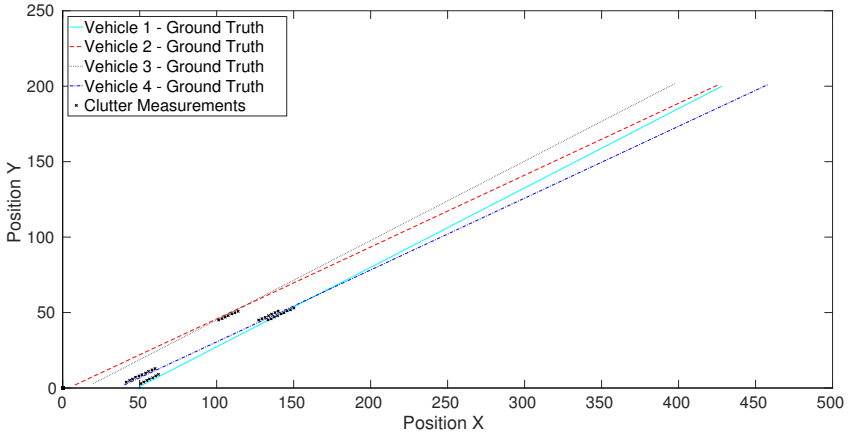
For further inspection the total system RMSE for the above are plotted in Figures 8.12(a), 8.12(b) and 8.12(c) respectively. It is very clear from the RMSE values that minimization topology factor performs optimally and is able to curtail the effect of the clutter in the system. Minimization topology factor is able to handle the increase in the number of clutter measurements and its performance remains optimal. This happens because the minimization topology factor partially guides the optimizer externally to arrive at an optimized state estimate and suppresses the outliers from misguiding the optimizer.



(a)

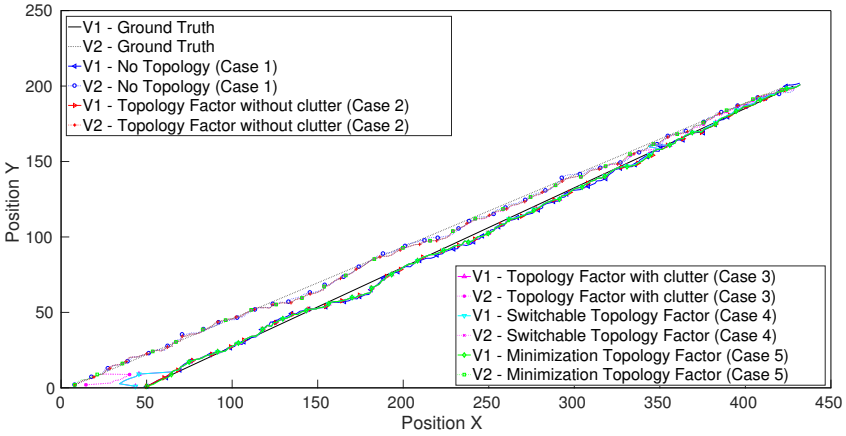


(b)

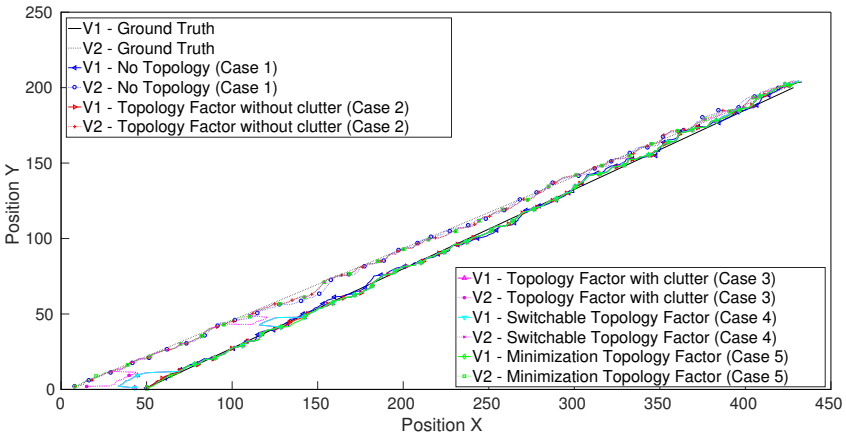


(c)

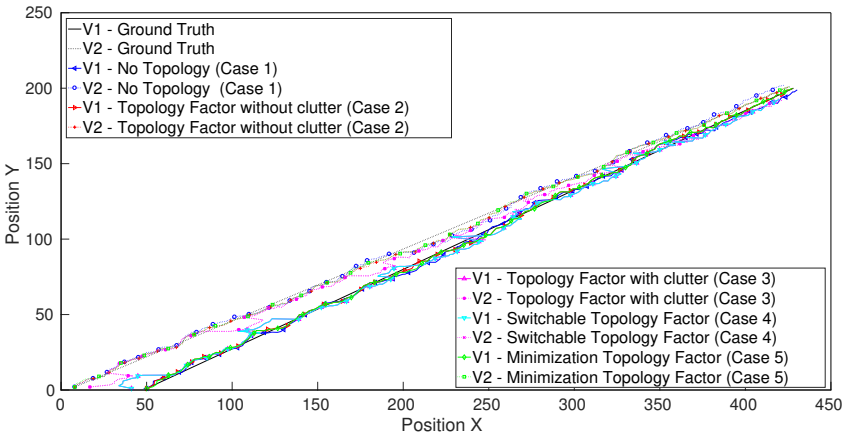
Figure 8.10: (a) 2 vehicles on a highway with up to 1 clutter measurement. (b) 3 vehicles on a highway with up to 2 clutter measurement. (c) 4 vehicles on a highway with up to 3 clutter measurement.



(a)

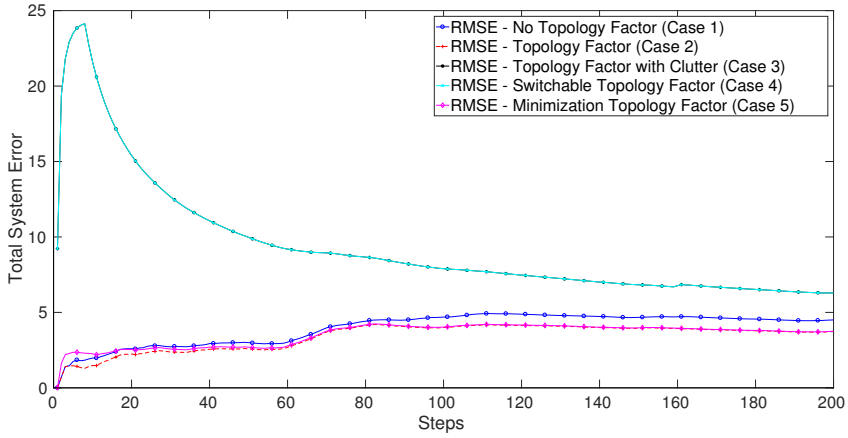


(b)

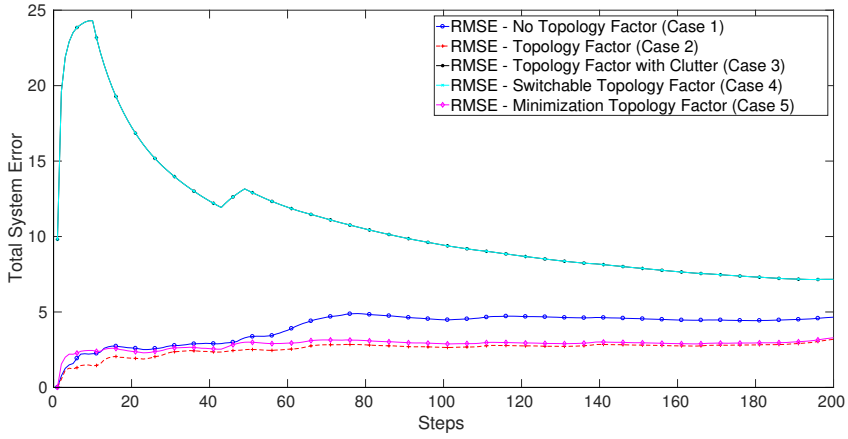


(c)

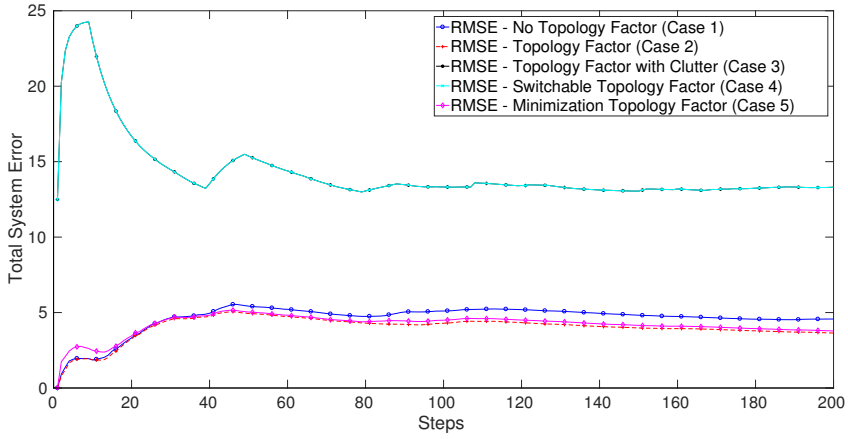
Figure 8.11: (a) Two vehicles on a highway with up to 1 clutter measurement. (b) Three vehicles on a highway with up to 2 clutter measurement. (c) Four vehicles on a highway with up to 3 clutter measurement.



(a)



(b)



(c)

Figure 8.12: (a) Two vehicles on a highway with up to 1 clutter measurement. (b) Three vehicles on a highway with up to 2 clutter measurement. (c) Four vehicles on a highway with up to 3 clutter measurement.

On the other hand, the switchable topology factor does not perform optimally and with an increase in the number of clutter measurements, the performance degrades further. Further investigation of the phenomenon lead to the point that the optimizer is not able to tune the switch variables successfully and hence all factors, including the ones formulated from the clutter, end up contributing to the final state estimate.

Since increase in the vehicles will explode the number of trajectories, for further analysis and comparison of the results refer to Table 8.1, which shows the average total system RMSE for 1000 iterations for 2, 3 and 4 vehicles. The simulation is run for all the Cases and for up to 1, 2 and 3 simultaneous clutter measurements. Each row in the table uses the same set of simulated data. This implies that for a simulation run for up to 1 clutter measurement for 2 vehicles all the 5 cases use the same set. Case 1 and 2, ignore the clutter measurements. The RMSE is rounded off to 4 decimal points. The last row mentions the average number of clutter measurements rounded off to an integer.

It can be clearly seen from Table 8.1 that among all the cases where the clutter measurements are used to construct the topology factor, the solution of minimization topology factor is superior. The performance does not match the ideal case, but this is expected as sometimes the solution may use a clutter measurement during the optimization because the minimization topology factor returned a lower error. In addition, since the radar has a higher precision therefore the lower error gets a higher weight and the solution may be swayed away from the true value. Overall the solution successfully stops the effect of the false information and lets the optimizer converge to a solution which is closer to the ground truth.

For lower number of clutter measurements the solution with switchable constraints also performs optimally. This is visible for 3 and 4 vehicles for up to 1 clutter measurement. In this scenario only an average of 22 clutter points were present for 3 and 4 vehicles. However, for a 2 vehicle simulation, when the number of clutter points per vehicle increases, the case 3 of adding all the topology factors and case 4 for using switchable topology factor, both result in bigger total system RMSE errors. On the other hand, the proposed solution of case 5 performs optimally. This trend can be seen again in the case of up to 3 simultaneous clutter measurements for 2, 3 and 4 vehicles. For up to 3 clutter simultaneous measurements, the simulation generated an average of 45 clutter measurements. Therefore, clutter measurement per vehicle for 2 is quiet high and for per vehicle for 4 vehicles is low. For 4 vehicles, the Case 3 with switchable topology factor is able to curtail the influence of Clutter, but minimization topology factor is superior and the RMSE is not far from the ideal case of no clutter. Moreover, for higher clutter measurements per vehicle i.e. for 2 vehicle scenario, the error is very high for both cases 3 and 4. On the other hand Case 5 i.e. minimization topology factor still performs optimally. Hence the minimization topology factor is able to handle the increase in the measurements originating from clutter efficiently.

The RMSE performance for case 3 and 4, for 2 vehicle scenario is same. This can happen when the switch variables are set to 1 for all the factors thereby all the factors equally contribute to the final optimization. Another interesting observation is, when Case 3 (that is adding all the topology factors) performs better than the case 4 (that is the switchable topology factor). This is possible when the optimizer ends up tuning the wrong topology factor with higher contribution.

Minimization topology factor avoids both the pitfalls as the minimization topology factor partially controls the optimizers behaviour.

Table 8.1: Average total system RMSE for 1000 iterations for 2, 3 and 4 vehicles. (TF = Topology Factor, Add All = Add all possible Topology Factors, SC = Topology Factor with Switchable Constraints, Min = Minimization Topology Factor)

			TF/Up-to 1 Clutter			**TF/Up-to 2 Clutter**			**TF/Up-to 3 Clutter**		
NoV	**No TF**	**TF/NC**	**Add All**	**SC**	**Min**	**Add All**	**SC**	**Min**	**Add All**	**SC**	**Min**
	Case 1	**Case 2**	**Case 3**	**Case 4**	**Case 5**	**Case 3**	**Case 4**	**Case 5**	**Case 3**	**Case 4**	**Case 5**
2	4.2194	3.1798	7.1168	7.1168	**3.2173**						
	4.2114	3.1869				15.3884	15.3884	**3.2545**			
	4.2023	3.1837							22.2722	22.2722	**3.2871**
3	5.1788	4.1613	5.6935	5.4637	**5.2990**						
	5.1888	4.1854				6.2963	6.2185	**5.5548**			
	5.1753	4.1564							7.7915	6.8170	**5.7304**
4	5.9432	4.7399	4.8610	5.0605	**4.8079**						
	5.9562	4.7497				5.1731	6.1106	**4.8836**			
	5.9629	4.7642							5.4044	7.0797	**4.9908**
Avg Clutter				22			33			45	

Table 8.2: Clutter in the system for 1000 iterations. NoV = Number of Vehicles, Min = Minimum Number, Max = Maximum Number, Avg = Average Number

	Up-to 1 Clutter		**Up-to 2 Clutter**		**Up-to 3 Clutter**	
NoV	**Min/Max**	**Avg**	**Min/Max**	**Avg**	**Min/Max**	**Avg**
2	5/46	22.3	5/83	33.4	5/116	44.5
3	5/49	22.1	5/81	33.6	5/113	45.9
4	5/47	22.4	5/86	33.4	5/123	44.7

The last row of Table 8.1 mentions only the average clutter measurements in the system. The actual minimum, maximum and the average number of clutter values for 1000 iterations in each case is presented in Table 8.2. As it can be seen, during our simulation there was always some clutter present in the system.

8.4.6.3 Remarks

Current implementation of the minimization topology factor uses the linear search for the minimum error. The calculation of error is just a mathematical subtraction operation, but with an increase in the number of clutter with respect to the true targets, all possible combinations need to be iterated. All these processes can be executed in parallel using multi-threaded approach, thereby avoiding any performance loss.

8.5 Summary

Among all the possible solutions presented in this chapter, the minimization topology factor provides a novel way of improved state estimation in a dynamic environment in presence of clutter. Additionally, the minimization topology factor retains all the properties of the original topology factor; i.e. bandwidth limitations; data association uncertainties; unknown coordinate transformations; and scalability.

Although the robustness of the minimization topology factor has been demonstrated using offline algorithm of Levenberg-Marquardt for final state estimation, the iSAM2 [93] algorithm of GTSAM can also be used to perform an online state estimation of the variables.

The minimization topology factor covers all of the 6 challenges highlighted in Section 1.3. Therefore, it has the full potential to perform the cooperative localization in a realistic environment, where there exist clutter measurements from an infrastructure radar.

Chapter 9

SME Factor vs GLMB Filter

9.1 Motivation

Associating the measurements from an external precise sensor with an internal off-the-shelf odometer/GPS for multi target scenario is a challenging task. With an increasing number of targets, the combinatorial possibilities increase as well. Therefore, data or measurement-to-track association has to be an integral and expensive part of any solution performing multi-target multi-sensor cooperative localization for improved state estimation.

In Chapters 5 and 6 novel ways of cooperative localization using factor graph were introduced. The critical challenge of data association was avoided completely to improve state estimates using cooperative localization. However, there exist other state-of-the-art methods which perform multi-target multi-sensor data fusion and solve the challenge of data association implicitly.

In this chapter, a comparative study against one of such methods is performed in order to evaluate the advantages and disadvantages of the proposed methods. The method of factor graph using SME factor (Chapter 6) is compared against Random Finite Set (RFS) based Generalized Labeled Multi-Bernoulli (GLMB) Filter [114]. The results of the comparison were published in the conference paper:

1. Gulati, Dhiraj; Sharif, Uzair et. al.: Data association - solution or avoidance: Evaluation of a filter based on RFS framework and factor graphs with SME. 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2017.

9.2 Generalized Labeled Multi-Bernoulli (GLMB) Filter

The result obtained for the comparison of RFS based cooperative localization against the factor graph based SME factor were achieved in collaboration with Mr. Uzair Sharif.

Random Finite Sets (RFS): We first summarize the basic formulation of RFS methods. For detailed explanations, the reader is advised to refer to [115].

Mathematically, the RFS framework lumps together all the objects' states and observations into finite-sets referred to as multi-object state X_k and multi-object observation Z_k at any time k. These sets are

random in the number of elements. If $\mathbb{X} \subset \mathbb{R}$ and $\mathbb{Z} \subset \mathbb{R}$ are the complete state spaces, then subsets can be defined as:

$$\begin{aligned} X_k &= \{x_{k,1}, \cdots, x_{k,n(k)}\} \in \mathcal{F}(\mathbb{X}) \\ Z_k &= \{z_{k,1}, \cdots, z_{k,n(k)}\} \in \mathcal{F}(\mathbb{Z}) \end{aligned} \tag{9.1}$$

where $\mathcal{F}(cdot)$ denotes the collection of finite sets.

This unifying notion of entire system's state and observation enables the application of Bayesian-Inferential Statistics in the derivation of optimal multi-object Bayes *prediction-update* recursion, which is given by [115]:

$$\pi_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X)\ \pi_{k-1}(X|Z_{1:k-1})\ \delta X, \tag{9.2}$$

$$\pi_k(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)\ \pi_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X)\ \pi_{k|k-1}(X|Z_{1:k-1})\ \delta X}, \tag{9.3}$$

where $Z_{1:k} = (Z_1, \cdots, Z_k)$ denotes the measurement history up to time k, $\pi_{k|k-1}(\cdot\,|Z_{1:k-1})$ and $f_{k|k-1}(\cdot\,|\,\cdot)$ denote multi-object prediction and transition functions from time $k-1$ to k, $\pi_k(\cdot\,|Z_{1:k})$ and $g_k(\cdot\,|\,\cdot)$ denote the multi-object posterior and likelihood function at time k. The integrals are the Finite Set Statistics (FISST) integrals defined for a function $f : \mathcal{F}(\mathbb{X}) \to \mathbb{R}$ as:

$$\int f(X)\delta X = \sum_{i=0}^{\infty} \frac{1}{i!} \int f(\{x_1, \ldots, x_i\}) dx_1 \cdots dx_i. \tag{9.4}$$

As seen above there are multiple integrals on $\mathcal{F}(\mathbb{X})$, therefore the optimal Bayes filter is only computationally tractable for very small number of objects within the system.

Labelled Random Finite Sets: As already mentioned, the sets are random and the targets can not be uniquely identified between two sets of state estimates of consecutive time steps. Therefore, to facilitate the target identification a concept of Labelled random finite sets were first introduced in [116]. In this each element in the state set is augmented with a unique label which gets propagated to the next time step. This enables us to estimate the identity or trajectory of a single target in a multi-target scenario.

Mathematically, for any $\tilde{X} \subset \mathbb{X}\times\mathbb{L}$, let $\mathcal{L}(\tilde{X})$ denotes the set of labels of $\tilde{X}$ i.e. $\mathcal{L}(\tilde{X}) \triangleq \{l : (x, l) \in X\}$. Then a labelled RFS with a state space $\mathbb{X}$ and discrete label space $\mathbb{L}$ is an RFS on $\mathbb{X} \times \mathbb{L}$ such that for each realization $\tilde{X}$ we have $|\mathcal{L}(\tilde{X})| = |(\tilde{X})|$. [116]

Multi-Bernoulli RFS: An RFS $\boldsymbol{X}$ is said to be Bernoulli when the cardinality distribution of the sets is a Bernoulli distribution. The probability is r of being singleton whose only element is distributed according to the probability density p. A multi-Bernoulli RFS $\boldsymbol{X}$ is then the union of the M independent Bernoulli RFSs with probability densities $p^i\ \forall\ i = 1, \cdots, M$.

Generalized Labeled Multi-Bernoulli (GLMB) Filter: Under the assumptions of standard multi-object transition and observation models, Vo et al. in [116] have derived a special case of labelled RFS on $\mathbb{X} \times \mathbb{R}$ called the "GLMB RFS" with probability density of the form:

$$\tilde{\pi}(\tilde{X}) = \delta_{|\bar{X}|}(|\mathcal{L}(\tilde{X})|) \sum_{c\in\mathbb{C}} w^{(c)}\left(\mathcal{L}(\tilde{X})\right) [\tilde{p}^{(c)}]^{\bar{X}} \tag{9.5}$$

where $\mathbb{C}$ is a discrete index set; $\delta_{|\bar{X}|}(|\mathcal{L}(\tilde{X})|)$ is the distinct label indicator of $\bar{X}$; w is the weight; $w^{(c)}$, and $\tilde{p}^{(c)}$ satisfy:

$$\begin{aligned} \int \tilde{p}^{(c)}(x,\ell)dx &= 1 \\ \sum_{c\in\mathbb{C}} \sum_{L\subseteq\mathbb{L}} w^{(c)}(L) &= 1 \end{aligned} \tag{9.6}$$

As shown in [117], given a GLMB initial density, subsequent multi-object densities are GLMBs and can be computed exactly by a tractable recursion and also have an exact solution to Equations (9.2) and (9.3).

For our comparison to compute the prediction-update recursion we make use of the Gaussian-Mixture implementation of the GLMB filter as provided in [118]. This implementation assumes linear-Gaussian constraints on target dynamics and sensor observations. Under these assumptions, the individual target states take on the form of Gaussian Mixtures on $\mathbb{X}$.

9.3 Evaluation

9.3.1 System Setup

To compare the two approaches, a simulation of 2 vehicles on a ground plane is set up. This is later extended for 3 and 4 vehicles. The tests are coded in C++ and run on an Ubuntu 16.04 LTS 64-bit machine with 16 GB RAM and Intel(R) Core(TM) i7-4710MQ CPU @ 2.50GHz processor. It is important to note that the computational time will be influenced by different operations already being undertaken on the system and any performance analysis can only provide an indication.

Simulated vehicles are assumed to have odometer and GPS sensors. The GPS sensor provides the measurement in global coordinates. The simulated infrastructure radar is mounted such that it is able to observe the vehicles for the complete trajectory. Since SME factor is compared, it is assumed that the radar location is known. Therefore, the measurements are provided in global coordinates without performing any data association. The measurement noises are assumed to be Gaussian. The covariance are assumed as $diag\,[1.0, 1.0]$, $diag\,[9.0, 9.0]$ and $diag\,[0.1, 0.1]$ for the odometer, the GPS and the radar respectively.

It is assumed that there is no miss-detection or clutter, therefore the detection probability for GLMB $P_d = 1.0$. The infrastructure sensor is able to observe the entire trajectory, therefore the survival probability $P_s = 1.0$. The multi-sensor GLMB filter is implemented via the Gibbs sampling technique, the filter is configured with maximum association hypothesis $H^{max} = 100$. Further the GLMB [114] is a Gaussian Mixture implementation, which assumes that the targets move with a constant velocity model.

Results from the simulation are compared two ways, between:

- The fused trajectory for GLMB using GPS and radar.
- The fused trajectory for odometer, GPS, and SME factor (formulated using radar).

Current implementation of GLMB Filter assumes the measurements from the sensors as the target positions in the global coordinate framework. On the other hand, the odometer provides only the difference between the two consecutive states of the vehicles. Using the cumulative odometer measurements

to calculate the position increases the covariance for each subsequent step. This results in the failure of the GLMB tracker. Hence, the odometer sensor is not used for our final comparison for GLMB Filter. Further comparison is performed with two other covariances of the radar as $diag\,[3,3]$ and $diag\,[5,5]$.

9.3.2 Performance Comparison

In this chapter we are using two different fusion methods to do the final state estimation. Each of these methods have their own set of comparison metrics and at times are not directly portable to the other methods. Therefore, we explore other possible methods for the comparison.

Optimal Sub-pattern Assignment (OSPA) is one of the commonly used performance metric for RFS based fusion methods. This was proposed first in [119] and calculates the miss-distance[1] and the corresponding error between the calculated and the estimated individual target states. If X is the set of estimated states with a cardinality m and Y is the set of true target states with a cardinality n, then the OSPA measure is defined as:

$$d_p^c(X,Y) = \Big(\frac{1}{n}\Big(\min \sum_{i=1}^{m} d^c(x_i, y_{\pi(i)})^p + c^p(n-m)\Big)\Big)^{1/p} \tag{9.7}$$

if $m \leq n$ and $d_p^c(X,Y) = d_p^c(Y,X)$ otherwise. Here, $d^c(x,y) \triangleq min(c, d(x,y))$ is the distance between x and y, cut-off at c. For the comparison c is assigned 100. We also use OSPA metric to measure and compare the performance of both the approaches.

The performance is also measured by calculating Root Mean Square Error (RMSE) value for the complete system. That is,

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{i_{est}} - x_{i_{GroundTruth}})^2} \tag{9.8}$$

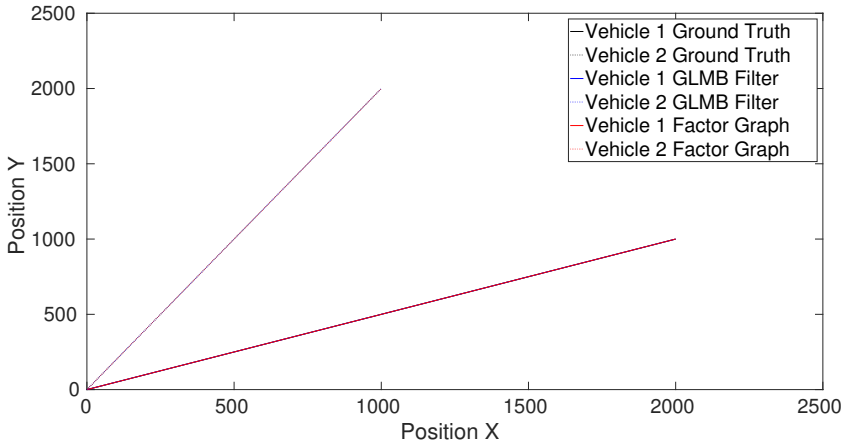
For details of RMSE and the above equation refer to Section 5.7.2.

9.3.3 Results

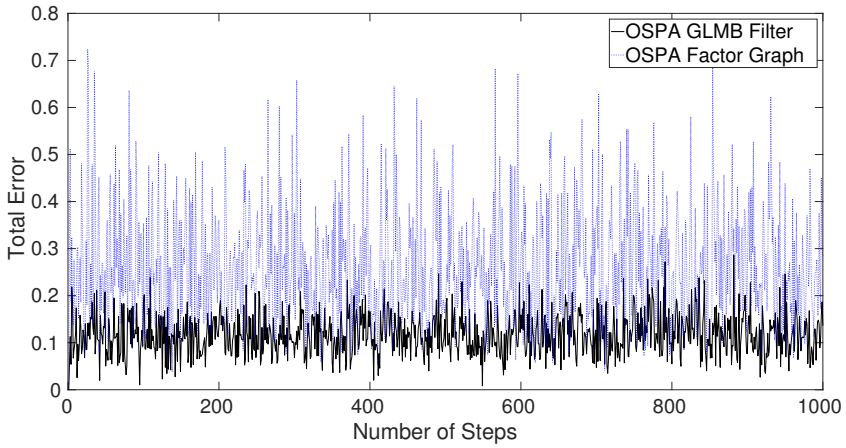
Figure 9.1(a) shows the ground truth and the fused trajectories for 2 vehicles using GLMB Filter and factor graph based SME factor. Since the fused trajectories are very close to the ground truth, it is difficult to analyse the performance. Therefore, Figure 9.1(b) shows the OSPA for the 1000 steps. Again, it becomes difficult to judge the performance. Hence, we plot the total system RMSE for the same as shown in Figure 9.1(c). It can be clearly seen GLMB Filter performs better than the SME factor approach. For the first few steps, the RMSE is very high, but it falls and quickly stabilizes. This is intentionally kept high because the GLMB Filter is unable to detect all the targets at the beginning.

GLMB Filter does not always detect the targets and results in state estimates of the targets in a random order, hence calculating the total system RMSE becomes challenging. For the above case of two vehicles, the data was manually inspected to calculate the total system RMSE. This is not feasible for all tests. Therefore we apply the principle of cumulation, similar to RMSE, on OSPA and calculate the

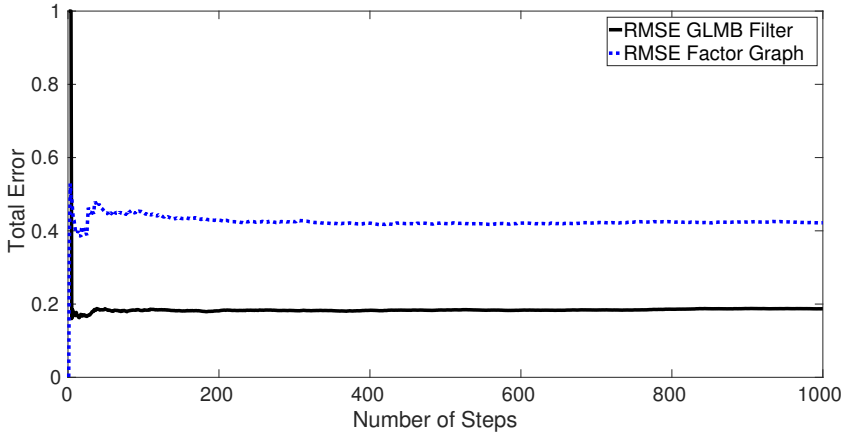
[1] FISST provides natural generalizations of the concept of miss distance to multi-target situations. Let $X = \{x_1, \cdots, x_n\}$ and $Y = \{y_1, \cdots, y_m\}$. Then the simplest definition of the multi-target miss-distance between X and Y is the Hausdorff distance. This is defined by $d_H(X,Y) = \max\{d_0(X,Y),\ d_0(Y,X)\}$, $d_0(X,Y) = \max_x \min_y \|x-y\|$. [120] p. 402



(a)



(b)



(c)

Figure 9.1: Radar covariance as $[.1, .1]$ (a) Ground truth; Trajectories calculated from GLMB Filter and factor graph. (b) OSPA for fused trajectories from GLMB Filter and factor graph. (c) Total system RMSE for fused trajectories from GLMB Filter and factor graph.

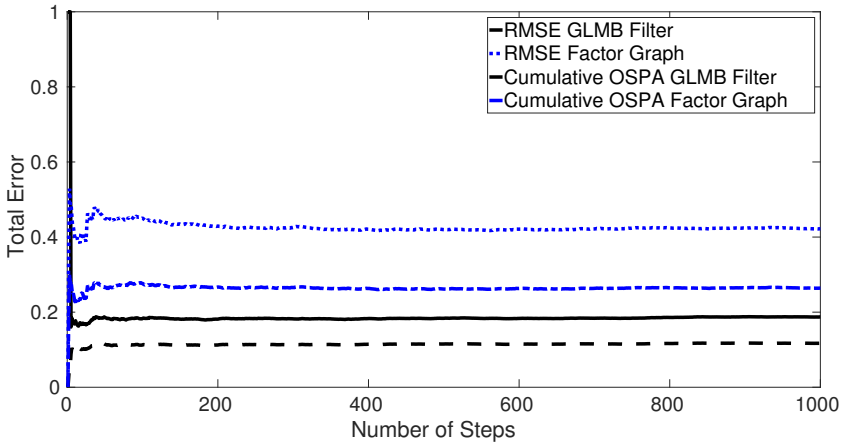
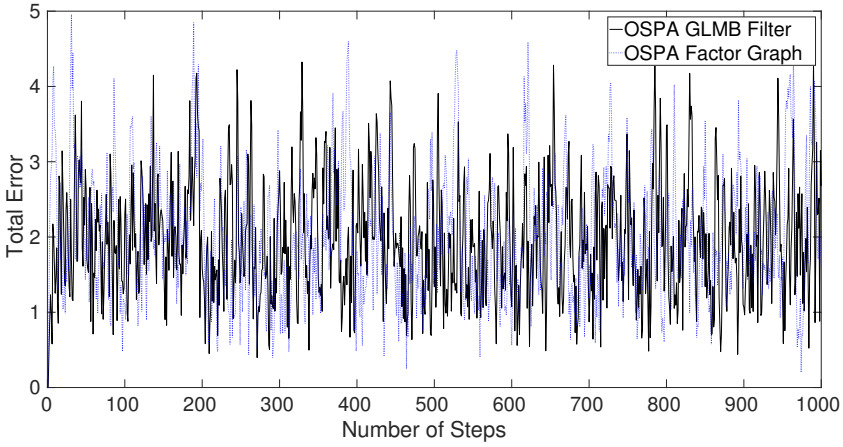
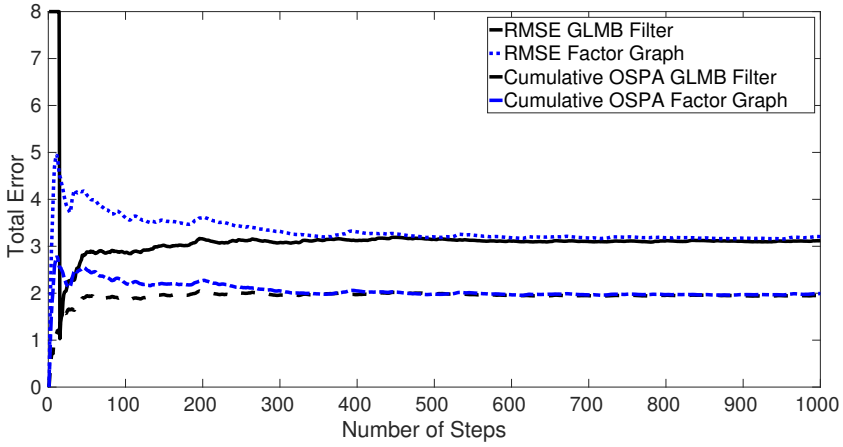


Figure 9.2: Radar covariance as $[.1, .1]$. Total system RMSE and Cumulative OSPA for fused trajectories from GLMB Filter and factor graph.



(a)



(b)

Figure 9.3: Radar covariance as $[3, 3]$ (a) OSPA for fused trajectories from GLMB Filter and factor graph (b) Total system RMSE and Cumulative OSPA for fused trajectories from GLMB Filter and factor graph.

"cumulative OSPA" as:

$$C - OSPA = \frac{\sum_{i=1}^{N} d_p^c}{N}$$

where d_p^c is defined in Eq. 9.7, and N is the number of steps. The plot for the cumulative OSPA and RMSE for the above example is shown in Figure 9.2.

From Figure 9.2, it can be seen that cumulative OSPA not only conveys the same information as total system RMSE but also does not suffer the problem of undetected targets and the correct order of the fused results.

GLMB Filter performs quite optimally because it gives high weight to the radar measurements, which have a very low covariance. On the other hand, factor graph avoids the data association using SME factor, which is a derived pseudo-measurement. The corresponding covariance is also a derived one and is not able to match the native radar covariance.

For further evaluation of the differences, the covariance of the radar is changed to $diag\,[3, 3]$. Figure 9.3(a) shows the OSPA and Fig. 9.3(b) shows the total system RMSE and cumulative OSPA for the same. It can be seen both the methods perform almost similarly. Figure 9.4(a) shows the OSPA and Fig. 9.4(b) shows the RMSE and cumulative OSPA for the radar covariance of $diag\,[5, 5]$. With an increase in the radar covariance, the GLMB Filter degrades faster than the SME factor based on factor graphs. This is because factor graph also uses the odometer and hence this extra information keeps the degradation in check.

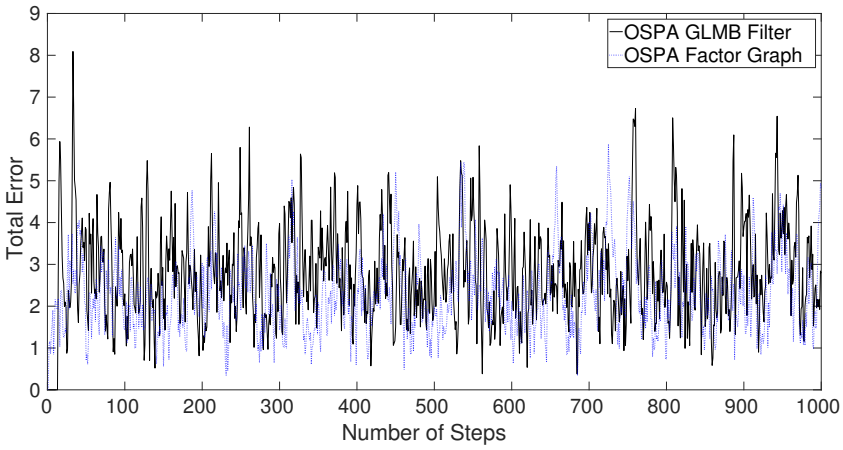
To further evaluate the stability of both the solutions, a Monte-Carlo Simulation of 1000 iterations for $2, 3$ and 4 targets is performed. The comparison is done using the above-proposed Cumulative OSPA, as already mentioned above that calculating RMSE for GLMB Filter is a challenging task. Table 9.1 shows the corresponding results rounded up to three decimal places.

Table 9.1: Average Final Cumulative OSPA values for 1000 Iterations

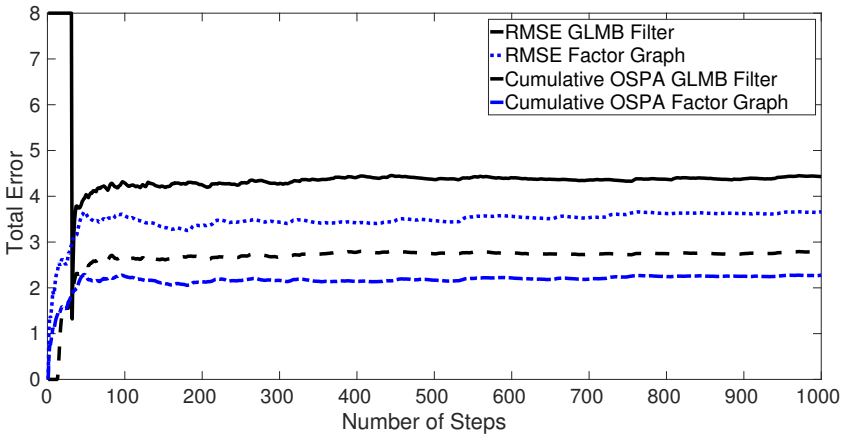
Total targets	GLMB Filter			Factor Graph		
	Covariances			Covariances		
	[.1,.1]	[3,3]	[5,5]	[.1,.1]	[3,3]	[5,5]
2	0.116	1.952	2.719	0.263	1.987	2.221
3	0.116	1.958	2.728	0.888	1.996	2.232
4	0.116	1.951	2.721	0.733	1.978	2.221

As can be seen from Table 9.1, with an increase in the number of targets, the system performance remains stable. It reflects the results discussed for a system of 2 targets.The table shows an anomaly as well. The factor graph for three vehicles seems to perform worse than two or four. One reason for the same is random data generation. We did not see such artefacts in the tests run in Chapter 6 which were simulated based on realistic radar data as seen in Figure 8.8.

Next, the execution performance of the methods is analysed by running the Monte-Carlo Simulation of 1000 iterations for $2, 3$ and 4 targets. Table 9.2 shows the corresponding results in milli-seconds rounding up to one decimal place.As can be seen from Table 9.2, GLMB Filter takes less time for all the cases of



(a)



(b)

Figure 9.4: Radar covariance as $[5, 5]$ (a) OSPA for fused trajectories from GLMB Filter and factor graph (b) Total system RMSE and Cumulative OSPA for fused trajectories from GLMB Filter and factor graph.

Table 9.2: Average Final Execution time in milli-seconds for 1000 Iterations

Total targets	GLMB Filter			Factor Graph		
	Covariances			Covariances		
	[.1,.1]	[3,3]	[5,5]	[.1,.1]	[3,3]	[5,5]
2	60.2	60.3	60.2	194.2	124.9	122.0
3	125.7	126.0	123.6	354.6	224.4	210.9
4	225.2	224.5	222.9	590.4	329.0	321.3

2, 3 and 4 targets. This is expected because factor graph based solution currently uses an offline batch optimizer, which solves the complete graph. On the other hand, GLMB being a filter only calculates the current state, thereby executing faster.

In the simulations, original measurements from the radar have been used; hence, data association is avoided in the factor graph with SME factor method with symmetrical transformations or is solved internally by the GLMB Filter. Instead of various states and covariances with each time step, only measurements are sent, keeping the bandwidth requirements to minimum for both the solutions.

9.3.4 Final Remarks

From the above simulation it can been seen the GLMB Filter performs better than the solution using the factor graph. But these numbers do not convey the complete picture. For a complete and detailed comparison it requires further testing covering the following points:

1. Although the data association is avoided for cooperative localization using external radar, the current results assume clutter free environment. In practice, this is not the case. GLMB Filter is capable to perform under the clutter scenario whereas the SME factor using factor graph would need additional solutions like [121, 122] to tackle the challenge.

2. Currently GLMB Filter is a Gaussian Mixture implementation, which assumes that the targets move with a constant velocity and linear model. Hence, it can only handle targets travelling in relative straight trajectories. For complex scenarios like constant turn or acceleration and random trajectories, additional solutions like Particle Filter have to be used. However, using Particle Filter may increase the average execution time ([123]). On the other hand, the factor graph based solution does not require any additional solution to handle complex scenarios.

3. The GTSAM framework supports the notion of plug and play [98] thereby providing an efficient platform for scenarios when not all the measurements are present from all the sensors. This was also shown in Chapter 6 Section 6.7.3. This is not possible for GLMB Filter as it should always know in advance which sensors will be used for the state estimates.

4. For online real scenarios online incremental smoothing algorithm like iSAM2 [93] (supported in GTSAM) is required.

9.4 Summary

In this chapter two state-of-the-art methods, addressing the data association challenge for multi-target multi-sensor scenarios have been considered. On one hand, the states are estimated using multi-sensor multi-object with the Generalized Labeled Multi-Bernoulli (GLMB) Filter and on the other, a factor graph solution using SME factors for cooperative localization as proposed in Chapter 6 is used. The performance is primarily assessed by cumulative OSPA with up to 4 targets.

GLMB Filter performs better when the error covariance of the radar sensor is low. However, with an increase in the error covariance the performance for GLMB Filter degrades faster than the factor graph with SME. GLMB Filter has lower execution time when compared to factor graph solution. However,

this is expected as we use the offline batch optimization for factor graph. Both the solutions address the challenges of bandwidth issue and scalability for the given test scenarios. Nevertheless, it warrants further testing with highway like scenarios where the number of vehicles can range between $20 - 40$.

Chapter 10

Conclusions and Outlook

10.1 Summary

The best autonomous driving system will not only make the job of a human driver redundant, it will also improve the fuel efficiency, decrease the traffic congestion and increase the safety and security of its passengers.

The first step towards this goal is the need of a precise state estimation of the autonomous vehicle. In this thesis I developed a framework to achieve this goal. The framework is based on sound mathematical foundation of Simultaneous Localization and Mapping (SLAM) represented using the mathematical concept of factor graphs. Factor graphs provide a convenient and easy to understand method to formulate the problem of SLAM. Various sensor measurements, direct and derived, are represented using the "factors". These factors bind various states of the vehicle. The resulting graphical structure is "solved" for the final state estimation using non-linear least squares algorithms, for example Levenberg-Marquardt [124] or iSAM2 [93].

This thesis uses an off-the-shelf C++ library developed by the Center for Robotics and Intelligent Machines (RIM) at Georgia Tech. The library is called Georgia Tech Smoothing and Mapping or GTSAM [67]. This library provides the capability of constructing the factor graphs and solving them using non-linear least squares algorithms.

The state considered in this thesis is the position of the autonomous vehicle. From the start of an autonomous vehicle to its final destination, the process of localization plays an important role. The proposed solutions estimate the location using multiple sensors, present inside and outside the vehicle. The framework utilizes both Vehicle-to-Infrastructure and Vehicle-to-Vehicle communication for the maximum benefit. Hence, the proposed solutions answers the question of "Cooperative Localization" for autonomous vehicles.

The researched solutions in this thesis have been published in peer reviewed international conferences and journals. The papers have also been presented in the conference, there by inviting discussion with the peer scientists and thus establishing their credibility and novelty.

10.2 Contributions

Traditional multi-sensor, both inside and outside the vehicle, data fusion techniques for multi-target scenarios face multiple challenges:

- Data Association: One critical challenge for sensors monitoring multi-target scenarios is to perform correct measurement-to-track association. Any incorrect association results in incorrect state estimation.
- Spatial Registration: Also known as coordinate transformation, is an important step of converting sensor measurements from heterogeneous coordinate system to a uniform coordinate system. Without this step, it is impossible to perform multi-sensor data fusion.
- Bandwidth Requirement: With increasing number of vehicles contending to perform cooperative state estimation using sensors from outside the vehicle, a minimum bandwidth utilization is an essential requirement.
- Scalability: With an increase in the participants, the task of cooperative localization for a multi target scenario should still be optimized. Therefore, the scalability of the solution is an important criterion.
- Clutter: Ghost measurements, i.e. spurious measurements of objects, which do not exists, plague the sensor systems like radar.
- Adaptability: Fusing data from a previously unseen external sensor requires the complete framework to be adaptable and support plug-and-play architecture.

The thesis investigated a generic framework addressing the above challenges to estimate the state of the vehicles using measurements from heterogeneous sensors, both inside and outside the vehicle. The researched solutions are implemented as novel factors. The sensors used from inside the vehicle are off-the-shelf automotive grade odometer and GPS. The external infrastructure sensor is a high precision radar. The mathematically sound ideas and methods presented here are further illustrated by experiments and simulations.

Chapter 5 defined an overview of a new topology factor which addresses five of the above six mentioned challenges. It assumes absence of clutter. The factor uses the sum of inter-vehicular distances added as constraints to improve the localization of the participating vehicles. The measurements from the external radar are used to formulate the proposed factor.

Chapter 6 formulated of a new SME factor which also tries to address the four of the above six mentioned challenges. Like the topology factor, it also assumes absence of clutter. It highlights the problem that the topology factor requires the presence of GPS measurements to perform optimally. Hence, scenarios like urban canyons, tunnels and underground parking garages cannot effectively use the topology factor. However, by relaxing the constraint of coordinate transformation one can use a new SME factor, resulting in superior performance.

Chapter 7 introduced another factor resulting from the distance measurements of the neighbouring vehicles using Dedicated Short Range Communication radio alongside the measurements from the external radar. This goes in the direction of proposing a complete framework utilizing both V2V and V2I communication and simultaneously addressing the above mentioned challenges.

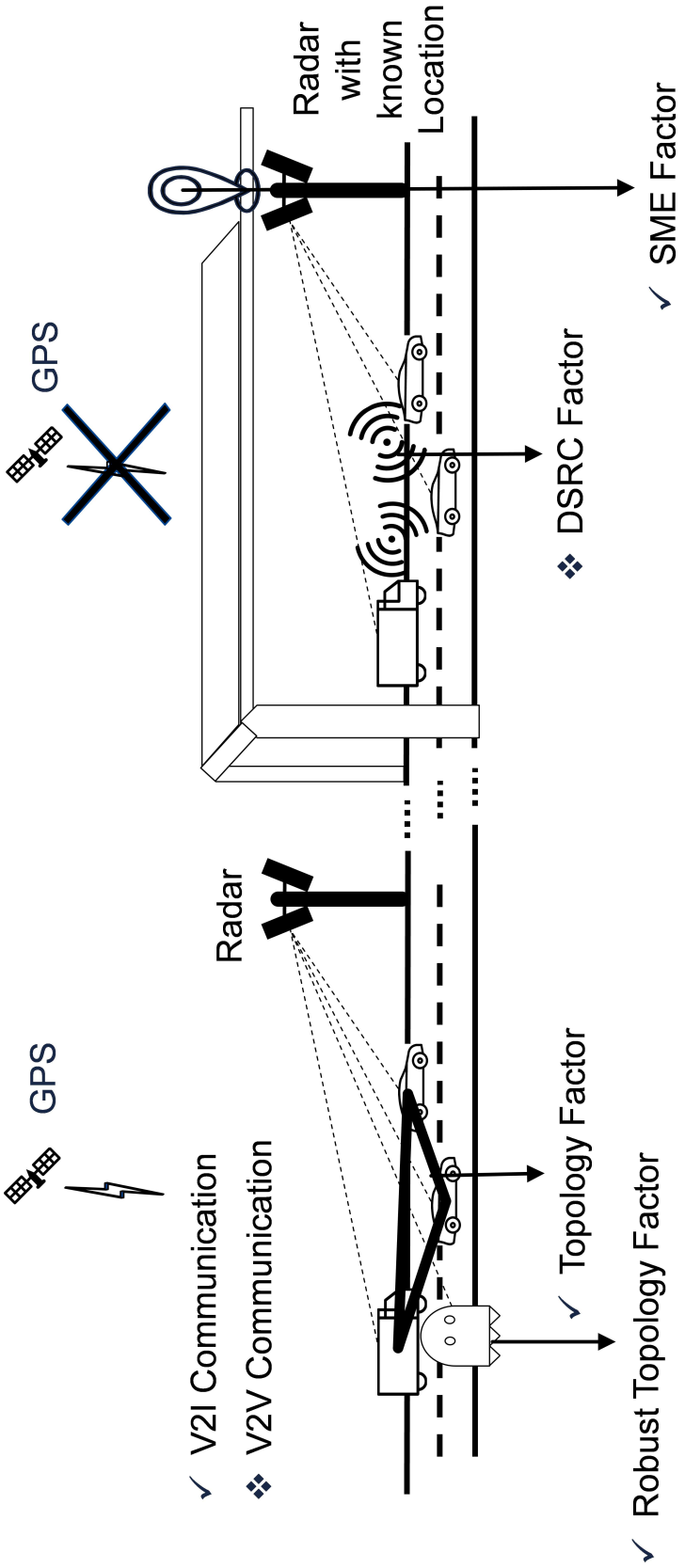


Figure 10.1: Overview of various kinds of factors researched and developed as part of the thesis

Chapter 8 demonstrated the effects of clutter in radar measurements on the topology factor and later proposed solutions to tackle it. The first proposed solution is to implement the Probabilistic Data Association (PDA) filter alongside the topology factor to remove the spurious topology factors resulting from the clutter measurements. Then it proposed an improved robust topology factor to address the clutter measurements. The simulation based on real radar data observations, highlights the power of the robust topology factor in presence of clutter measurements and successfully curtails the degradation of state estimation of the participating vehicles in cooperative localization.

Simulated comparisons are (a) either performed with presence and absence of the proposed solution or (b) are shown as improvement for state estimation against the previously proposed solution. However, Chapter 9 compared the SME factor based solution with the state-of-the-art Generalized Label Multi-Bernoulli Filter. Various pros and cons of both the solutions are highlighted.

Essentially GTSAM supports the addition of various factors at run-time, so the above proposed factors can be used independently based on the environment. Figure 10.1 shows the connection between various kinds of factors researched as part of the thesis.

The thesis offers a modular approach to data fusion, which not only addresses the above mentioned issues but also offers plug-and-play functionality for multi-sensor multi-target cooperative localization. However, this is achieved with some sacrifices to computational complexity when compared to the other Bayesian filters like Kalman and Particle Filter or Random Finite Set based solutions. Preliminary comparison has been also done in Chapter 9. Hence, it is up to the system engineers to decide what kind of approach is best suited for their needs.

10.3 Future Work

In past, factor graph based approaches have already been effectively demonstrated for a single autonomous robot. This thesis proposes a novel attempt to extend the framework of factor graphs to perform co-operative state estimation for autonomous vehicles. However, there still is a lot of scope of further advancements in theoretical and practical aspects of the concepts.

How can one address the clutter for multi-vehicle scenarios? The robust topology factor is a step in that direction. But the concept should also be extended to other factors like “SME”.

How can one utilize the data communication using DSRC? Currently I have used a star topology and used the distance information of neighbouring vehicles from the ego vehicle. However, it does not take into account distances between other vehicles, which are directly connected to the ego vehicle. How effectively can the cluster architecture be used? How much can the data communication further decrease the state estimation error?

How can one actively decide which factors to use? This requires additional rule based decision system alongside the fusion system, which can trigger the choice of factors to be used based on the quantity and quality of the sensor data received. Smart Adaptive Data Aggregation (SADA) [125, 126, 127] is one small step towards such system, which attempts to achieve the same.

How can one incorporate various other data from different new data sources like internet, Vehicle-to-People, Crowd-Sensing etc.? This requires further investigation of innovative factor formalisations from the new data sources to further decrease the error in the state estimation.

With the new communication standards like 5G, the above-proposed framework should be testable in the near future on real scenarios.

In this thesis I have used the total system RMSE to compare the performances of various proposed ideas. This is feasible for the simulation as the simulated scenarios also contain ground truth. But in real situations obtaining the ground truth is difficult. Therefore, RMSE measures are not feasible for real scenarios. One way to overcome this hurdle is to use additional expensive sensors like Differential-GPS which can provide better measurements and use these as ground truth. Other method which can be used is to consider the uncertainty associated with each state estimate which I have not considered. The lower is the uncertainty, the more certain one is of the calculated state of the vehicle.

In this thesis I have used the concept of factor graph, which easily lend themselves to formulate the problem of SLAM as a graph. These graphs are intuitive to understand, easy to formulate and represent a new way to represent and solve problems. They are one of the modern representation of the field of probabilistic programming where based on the multiple data observations with various uncertainties, a most likely observation can be calculated which best explains the observations. The scope of factor graphs need not be limited to solving the problems of autonomous vehicles, but may be extended to any autonomous system that work with uncertain data. Therefore, another direction of the future work is finding innovative problem domains which can be formulated and solved using concepts of factor graphs.

Appendix A

Publications

The following is the complete list of articles which I authored/co-authored based on the results obtained during my research work:

C1. **Dhiraj Gulati**, Vincent Aravantinos, Nikhil Somani, and Alois Knoll. Robust Vehicle Infrastructure Cooperative Localization in Presence of Clutter. In 21st International Conference on Information Fusion (FUSION), July, 2018. (Accepted).

C2. **Dhiraj Gulati**, Uzair Sharif, Feihu Zhang, Daniel Clarke, and Alois Knoll. Data association - Solution or Avoidance: Evaluation of a Filter based on RFS framework and Factor Graphs with SME. In IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Novemeber, 2017.

C3. **Dhiraj Gulati**, Feihu Zhang, Daniel Malovetz, Daniel Clarke, and Alois Knoll. Graph based Cooperative Localization using Symmetric Measurement Equations and Dedicated Short Range Communication. In IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Novemeber, 2017.

C4. **Dhiraj Gulati**, Feihu Zhang, Daniel Malovetz, Daniel Clarke, Gereon Hinz, and Alois Knoll. Graph based Vehicle Infrastructure Cooperative Localization. In 20th International Conference on Information Fusion (Fusion), July, 2017.

C5. **Dhiraj Gulati**, Feihu Zhang, Daniel Malovetz, Daniel Clarke, and Alois Knoll. Robust Cooperative Localization in a Dynamic Environment using Factor Graphs and Probability Data Association Filter. In 20th International Conference on Information Fusion (Fusion), July 2017.

C6. Feihu Zhang, Daniel Malovetz, **Dhiraj Gulati**, Daniel Clarke, and Alois Knoll. Joint Bias Estimation and Localization in Factor Graph. In IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), September, 2016.

C7. Wendelin Feiten, Susana Alcalde Bagüés, Michael Fiegert, Feihu Zhang, **Dhiraj Gulati**, and Tim Tiedemann. A new concept for a Cooperative Fusion Platform. In 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), September 2016.

C8. Susana Alcalde Bagüés, Wendelin Feiten, Tim Tiedemann, Christian Backe, **Dhiraj Gulati**, Steffen Lorenz, and Peter Conradi. Towards Dynamic and Flexible Sensor Fusion for Automotive Applications. In Tim Schulze, Beate Müller, and Gereon Meyer, editors, Advanced Microsystems for Automotive Applications 2016 Smart Systems for the Automobile of the Future, Springer International Publishing, Cham, 2016.

C9. **Dhiraj Gulati**, Feihu Zhang, Daniel Clarke, and Alois Knoll. Vehicle Infrastructure Cooperative Localization using Factor Graphs. In IEEE Intelligent Vehicles Symposium (IV), June 2016.

J1. **Dhiraj Gulati**, Feihu Zhang, Daniel Clarke, and Alois Knoll. Graph-Based Cooperative Localization Using Symmetric Measurement Equations. In Sensors 17 (6), June, 2017.

J2. Feihu Zhang, Gereon Hinz, **Dhiraj Gulati**, Daniel Clarke, and Alois Knoll. Cooperative Vehicle-Infrastructure Localization based on the Symmetric Measurement Equation Filter. In GeoInformatica, 20(2), April 2016.

W1. Christian Buckl, Michael Geisinger, **Dhiraj Gulati**, Fran J. Ruiz-Bertol, and Alois Knoll. CHROMOSOME: A Run-Time Environment for Plug & Play-Capable Embedded Real-Time Systems. In Sixth International Workshop on Adaptive and Reconfigurable Embedded Systems (APRES 2014). ACM, April 2014.

List of Figures

List of Tables

References

[1] **WHO 2015 Executive Summary (on 30 December 2017)**. http://www.who.int/violence_injury_prevention/road_safety_status/2015/Executive_summary_GSRRS2015.pdf. 2

[2] **Advanced Driver Assistant System Threats, Requirements, Security Solutions: Technical White Paper (on 31 December 2017)**. https://www.intel.com/content/dam/www/public/us/en/documents/white-papers/advanced-driver-assistant-system-paper.pdf. 2, 3, 131

[3] **Global Positioning System**. https://www.gps.gov/. 4, 10

[4] **Intelligent Transportation System**. https://www.its.dot.gov/. 4, 5

[5] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, H. Tullberg, M. A. Uusitalo, B. Timus, and M. Fallgren. **Scenarios for 5G mobile and wireless communications: the vision of the METIS project**. *IEEE Communications Magazine*, **52**(5):26–35, May 2014. 4

[6] **DSRC Fact Sheet (on 31 Jan 2018)**. https://www.its.dot.gov/factsheets/dsrc_factsheet.htm. 4, 79

[7] S. Gezici, Zhi Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu. **Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks**. *IEEE Signal Processing Magazine*, **22**(4):70–84, July 2005. 4

[8] J. J. Leonard and H. F. Durrant-Whyte. **Mobile robot localization by tracking geometric beacons**. *IEEE Transactions on Robotics and Automation*, **7**(3):376–382, Jun 1991. 9, 27

[9] I. J. Cox. **Blanche-an experiment in guidance and navigation of an autonomous robot vehicle**. *IEEE Transactions on Robotics and Automation*, **7**(2):193–204, Apr 1991. 9

[10] Charles J Cohen and Frank V Koss. **Comprehensive study of three-object triangulation**. In *Mobile Robots VII*, **1831**, pages 95–107. International Society for Optics and Photonics, 1993. 10

[11] V. Pierlot and M. Van Droogenbroeck. **A New Three Object Triangulation Algorithm for Mobile Robot Positioning**. *IEEE Transactions on Robotics*, **30**(3):566–577, June 2014. 10

[12] **Global Navigation Satellite System**. https://www.glonass-iac.ru/en/. 11

[13] **European Union Galileo positioning system**. https://www.galileognss.eu/. 11

[14] **BeiDou Navigation Satellite System**. http://www.beidou.gov.cn/. 11

[15] **Indian Regional Navigation Satellite System**. https://www.isro.gov.in/spacecraft/satellite-navigation. 11

[16] **Quasi-Zenith Satellite System**. http://www.qzss.go.jp/. 11

[17] T. Tsubouchi and S. Yuta. **Map assisted vision system of mobile robots for reckoning in a building environment**. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, **4**, pages 1978–1984, Mar 1987. 11

[18] A. Lahrech, C. Boucher, and J. C. Noyer. **Fusion of GPS and odometer measurements for map-based vehicle navigation**. In *Industrial Technology, 2004. IEEE ICIT '04. 2004 IEEE International Conference on*, **2**, pages 944–948 Vol. 2, Dec 2004. 11

[19] S. Sukkarieh, E. M. Nebot, and H. F. Durrant-Whyte. **A high integrity IMU/GPS navigation loop for autonomous land vehicle applications**. *IEEE Transactions on Robotics and Automation*, **15**(3):572–578, Jun 1999. 11

[20] H. Li, F. Nashashibi, and G. Toulminet. **Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data**. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 1657–1662, Sept 2010. 11

[21] R. Kurazume, S. Nagata, and S. Hirose. **Cooperative positioning with multiple robots**. In *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 1250–1257 vol.2, May 1994. 12, 13, 14, 15, 20, 131

[22] L. Hobert, A. Festag, I. Llatser, L. Altomare, F. Visintainer, and A. Kovacs. **Enhancements of V2X communication in support of cooperative autonomous driving**. *IEEE Communications Magazine*, **53**(12):64–70, Dec 2015. 13

[23] J. Spletzer, A. K. Das, R. Fierro, C. J. Taylor, V. Kumar, and J. P. Ostrowski. **Cooperative localization and control for multi-robot manipulation**. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, **2**, pages 631–636 vol.2, 2001. 13

[24] J. W. Fenwick, P. M. Newman, and J. J. Leonard. **Cooperative concurrent mapping and localization**. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, **2**, pages 1810–1817 vol.2, 2002. 13, 20

[25] S. I. Roumeliotis and G. A. Bekey. **Collective localization: a distributed Kalman filter approach to localization of groups of mobile robots**. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, **3**, pages 2958–2965 vol.3, 2000. 13, 14

[26] S. I. Roumeliotis and G. A. Bekey. **Distributed multirobot localization**. *IEEE Transactions on Robotics and Automation*, **18**(5):781–795, Oct 2002. 13, 16

[27] S. I. Roumeliotis and G. A. Bekey. **Synergetic localization for groups of mobile robots**. In *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*, **4**, pages 3477–3482 vol.4, 2000. 13, 20

[28] I. M. Rekleitis, G. Dudek, and E. E. Milios. **Multi-robot cooperative localization: a study of trade-offs between efficiency and accuracy**. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, **3**, pages 2690–2695 vol.3, 2002. 14

[29] Ioannis M. Rekleitis, Gregory Dudek, and Evangelos E. Milios. *Graph-Based Exploration using Multiple Robots*, pages 241–250. Springer Japan, Tokyo, 2000. 14

[30] I. Rekleitis, G. Dudek, and E. Milios. **Probabilistic cooperative localization and mapping in practice**. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, **2**, pages 1907–1912 vol.2, Sept 2003. 14, 20

[31] I. Rekleitis, G. Dudek, and E. Milios. **Experiments in free-space triangulation using cooperative localization**. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, **2**, pages 1777–1782 vol.2, Oct 2003. 14

[32] R. Madhavan, K. Fregene, and L. E. Parker. **Distributed heterogeneous outdoor multi-robot localization**. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, **1**, pages 374–381 vol.1, 2002. 14, 20

[33] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. **A probabilistic approach to collaborative multi-robot localization**. *Autonomous robots*, **8**(3):325–344, 2000. 14, 20

[34] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. **Collaborative multi-robot localization**. In *Mustererkennung 1999*, pages 15–26. Springer, 1999. 14

[35] A. Das, J. Spletzer, V. Kumar, and C. Taylor. **Ad hoc networks for localization and control**. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, **3**, pages 2978–2983 vol.3, Dec 2002. 14, 15, 20

[36] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal. **Locating the nodes: cooperative localization in wireless sensor networks**. *IEEE Signal Processing Magazine*, **22**(4):54–69, July 2005. 15, 20

[37] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, and R. J. O'Dea. **Relative location estimation in wireless sensor networks**. *IEEE Transactions on Signal Processing*, **51**(8):2137–2148, Aug 2003. 15, 20, 79

[38] T. Yoshida, A. Ohya, and S. Yuta. **Cooperative self-positioning system for multiple mobile robots**. In *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, **1**, pages 223–227 vol.1, July 2003. 15

[39] A. Howard, M. J. Matark, and G. S. Sukhatme. **Localization for mobile robot teams using maximum likelihood estimation**. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, **1**, pages 434–439 vol.1, 2002. 15, 20

[40] A. Howard, M. J. Mataric, and G. S. Sukhatme. **Putting the 'I' in 'team': an egocentric approach to cooperative localization**. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, **1**, pages 868–874 vol.1, Sept 2003. 15, 20

[41] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. **Markov localization for mobile robots in dynamic environments**. *Journal of artificial intelligence research*, **11**:391–427, 1999. 15

[42] L. Montesano, J. Gaspar, J. Santos-Victor, and L. Montano. **Cooperative localization by fusing vision-based bearing measurements and motion**. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2333–2338, Aug 2005. 15

[43] N. Trawny and T. Barfoot. **Optimized motion strategies for cooperative localization of mobile robots**. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, **1**, pages 1027–1032 Vol.1, April 2004. 15

[44] M. Di Marco, A. Garulli, A. Giannitrapani, and A. Vicino. **Simultaneous localization and map building for a team of cooperating robots: a set membership approach**. *IEEE Transactions on Robotics and Automation*, **19**(2):238–249, Apr 2003. 15, 20

[45] M Di Marco, A Garulli, S Lacroix, and A Vicino. **Set membership localization and mapping for autonomous navigation**. *International Journal of robust and nonlinear control*, **11**(7):709–734, 2001. 16

[46] A. Martinelli, F. Pont, and R. Siegwart. **Multi-Robot Localization Using Relative Observations**. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2797–2802, April 2005. 16, 18, 20

[47] C. Smaili, M. E. El Najjar, and Francois. **Multi-sensor Fusion Method Using Bayesian Network for Precise Multi-vehicle Localization**. In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pages 906–911, Oct 2008. 16

[48] J. Knuth and P. Barooah. **Distributed collaborative localization of multiple vehicles from relative pose measurements**. In *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 314–321, Sept 2009. 16

[49] V. Caglioti, A. Citterio, and A. Fossati. **Cooperative, distributed localization in multi-robot systems: a minimum-entropy approach**. In *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, pages 25–30, June 2006. 16, 20

[50] N. Karam, F. Chausse, R. Aufrere, and R. Chapuis. **Cooperative Multi-Vehicle Localization**. In *2006 IEEE Intelligent Vehicles Symposium*, pages 564–570, 2006. 16

[51] N. Karam, F. Chausse, R. Aufrere, and R. Chapuis. **Localization of a Group of Communicating Vehicles by State Exchange**. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 519–524, Oct 2006. 16

[52] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. **Vehicular ad hoc networks: A new challenge for localization-based systems**. *Computer communications*, **31**(12):2838–2849, 2008. 16, 20, 39, 80

[53] R. Parker and S. Valaee. **Vehicular Node Localization Using Received-Signal-Strength Indicator**. *IEEE Transactions on Vehicular Technology*, **56**(6):3371–3380, Nov 2007. 16, 20, 39, 80

[54] P. H. Mohammadabadi and S. Valaee. **Cooperative node positioning in vehicular networks using inter-node distance measurements**. In *2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, pages 1448–1452, Sept 2014. 16, 39, 80

[55] R. Parker and S. Valaee. **Cooperative Vehicle Position Estimation**. In *2007 IEEE International Conference on Communications*, pages 5837–5842, June 2007. 16, 39, 80

[56] M. Elazab, A. Noureldin, and H. S. Hassanein. **Integrated Cooperative Localization for Connected Vehicles in Urban Canyons**. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, Dec 2015. 16, 39, 80

[57] R. Grabowski and P. Khosla. **Localization techniques for a team of small robots**. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, **2**, pages 1067–1072 vol.2, 2001. 16

[58] R. Parker and S. Valaee. **Vehicle Localization in Vehicular Networks**. In *IEEE Vehicular Technology Conference*, pages 1–5, Sept 2006. 17, 39

[59] R. Parker and S. Valaee. **Robust min-max localization algorithm**. In *2006 IEEE Intelligent Transportation Systems Conference*, pages 1000–1005, Sept 2006. 17, 39

[60] J. Liu, B. g. Cai, and J. Wang. **Cooperative Localization of Connected Vehicles: Integrating GNSS With DSRC Using a Robust Cubature Kalman Filter**. *IEEE Transactions on Intelligent Transportation Systems*, **18**(8):2111–2125, Aug 2017. 17

[61] E. D. Nerurkar, S. I. Roumeliotis, and A. Martinelli. **Distributed maximum a posteriori estimation for multi-robot cooperative localization**. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 1402–1409, May 2009. 17, 20

[62] H. Li and F. Nashashibi. **Cooperative Multi-Vehicle Localization Using Split Covariance Intersection Filter**. *IEEE Intelligent Transportation Systems Magazine*, **5**(2):33–44, Summer 2013. 17, 20

[63] L. C. Carrillo-Arce, E. D. Nerurkar, J. L. Gordillo, and S. I. Roumeliotis. **Decentralized multi-robot cooperative localization using covariance intersection**. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1412–1417, Nov 2013. 17

[64] A. Ahmad, G. D. Tipaldi, P. Lima, and W. Burgard. **Cooperative robot localization and target tracking based on least squares minimization**. In *2013 IEEE International Conference on Robotics and Automation*, pages 5696–5701, May 2013. 17, 20

[65] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. **G2o: A general framework for graph optimization**. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3607–3613, May 2011. 17

[66] G. Huang, R. Truax, M. Kaess, and J. J. Leonard. **Unscented iSAM: A consistent incremental solution to cooperative localization and target tracking**. In *2013 European Conference on Mobile Robots*, pages 248–254, Sept 2013. 17

[67] **GTSAM, Georgia Tech Smoothing and Mapping**. `https://collab.cc.gatech.edu/borg/gtsam/`. 17, 123

[68] Vadim Indelman, Pini Gurfil, Ehud Rivlin, and Hector Rotstein. **Distributed vision-aided cooperative localization and navigation based on three-view geometry**. *Robotics and Autonomous Systems*, **60**(6):822 – 840, 2012. 17

[69] E. D. Nerurkar, K. X. Zhou, and S. I. Roumeliotis. **A hybrid estimation framework for Cooperative Localization under communication constraints**. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 502–509, Sept 2011. 18, 21

[70] E. D. Nerurkar and S. I. Roumeliotis. **A communication-bandwidth-aware hybrid estimation framework for multi-robot cooperative localization**. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1418–1425, Nov 2013. 18

[71] V. Savic and S. Zazo. **Nonparametric Belief Propagation Based on Spanning Trees for Cooperative Localization in Wireless Sensor Networks**. In *2010 IEEE 72nd Vehicular Technology Conference - Fall*, pages 1–5, Sept 2010. 18

[72] Shi Xingxi, Huang Bo, Wang Tiesheng, and Zhao Chunxia. **Cooperative multi-robot localization based on distributed UKF**. In *2010 3rd International Conference on Computer Science and Information Technology*, **6**, pages 590–593, July 2010. 18

[73] F. Zhang, H. Stähle, G. Chen, C. Buckl, and A. Knoll. **Multiple vehicle cooperative localization under random finite set framework**. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1405–1411, Nov 2013. 18, 21

[74] Rajnikant Sharma, Stephen Quebe, Randal W. Beard, and Clark N. Taylor. **Bearing-only Cooperative Localization**. *Journal of Intelligent & Robotic Systems*, **72**(3):429–440, Dec 2013. 18, 21

[75] K. Liu, H. B. Lim, E. Frazzoli, H. Ji, and V. C. S. Lee. **Improving Positioning Accuracy Using GPS Pseudorange Measurements for Cooperative Vehicular Localization**. *IEEE Transactions on Vehicular Technology*, **63**(6):2544–2556, July 2014. 18

[76] M. Rohani, D. Gingras, V. Vigneron, and D. Gruyer. **A new decentralized Bayesian approach for cooperative vehicle localization based on fusion of GPS and inter-vehicle distance measurements**. In *2013 International Conference on Connected Vehicles and Expo (ICCVE)*, pages 473–479, Dec 2013. 18

[77] Feihu Zhang, Gereon Hinz, Dhiraj Gulati, Daniel Clarke, and Alois Knoll. **Cooperative vehicle-infrastructure localization based on the symmetric measurement equation filter**. *GeoInformatica*, **20**(2):159–178, Apr 2016. 18, 21

[78] N. Alam and A. G. Dempster. **Cooperative Positioning for Vehicular Networks: Facts and Future**. *IEEE Transactions on Intelligent Transportation Systems*, **14**(4):1708–1717, Dec 2013. 24, 77, 79, 80, 82, 83

[79] Merrill Ivan Skolnik. **Radar Handbook**. 2008. 25, 91, 101

[80] R. Smith, M. Self, and P. Cheeseman. **Estimating uncertain spatial relationships in robotics**. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, **4**, pages 850–850, Mar 1987. 27

[81] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005. 28, 29

[82] X. Rong Li and V. P. Jilkov. **Survey of maneuvering target tracking. Part I. Dynamic models**. *IEEE Transactions on Aerospace and Electronic Systems*, **39**(4):1333–1364, Oct 2003. 28

[83] Sebastian Thrun. **Robotic mapping: A survey**. *Exploring artificial intelligence in the new millennium*, **1**:1–35, 2002. 29

[84] **Bayesian Network**. `https://en.wikipedia.org/wiki/Bayesian_network`. 29, 30, 131

[85] Paul Dagum, Adam Galper, and Eric Horvitz. **Dynamic Network Models for Forecasting**. In *Proceedings of the Eighth International Conference on Uncertainty in Artificial Intelligence*, UAI'92, pages 41–48, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc. 30

[86] F. R. Kschischang, B. J. Frey, and H. A. Loeliger. **Factor graphs and the sum-product algorithm**. *IEEE Transactions on Information Theory*, **47**(2):498–519, Feb 2001. 30, 31

[87] Vadim Indelman, Stephen Williams, Michael Kaess, and Frank Dellaert. **Information fusion in navigation systems via factor graph based incremental smoothing**. *Robotics and Autonomous Systems*, **61**(8):721–738, 2013. 31

[88] Hans-Andrea Loeliger. **An introduction to factor graphs**. *Signal Processing Magazine, IEEE*, **21**(1):28–41, 2004. 31

[89] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba. **A solution to the simultaneous localization and map building (SLAM) problem**. *IEEE Transactions on Robotics and Automation*, **17**(3):229–241, Jun 2001. 32

[90] M. Montemerlo and S. Thrun. **Simultaneous localization and mapping with unknown data association using FastSLAM**. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, **2**, pages 1985–1991 vol.2, Sept 2003. 32

[91] John Mullane, Ba-Ngu Vo, Martin D Adams, and Ba-Tuong Vo. **A random-finite-set approach to Bayesian SLAM**. *IEEE Transactions on Robotics*, **27**(2):268–282, 2011. 32

[92] Jorge Nocedal and Stephen J Wright. *Numerical Optimization*. Springer, 2006. 34

[93] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. **iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering**. In *2011 IEEE International Conference on Robotics and Automation*, pages 3281–3288, May 2011. 35, 51, 74, 76, 112, 121, 123

[94] Frank Dellaert and Michael Kaess. **Square Root SAM: Simultaneous Localization and Mapping via Square Root Information Smoothing**. *Int. J. Rob. Res.*, **25**(12):1181–1203, Dec 2006. 35

[95] Kai Oliver Arras. **An introduction to error propagation: Derivation, meaning and examples of Equation CY = FX CXFX T.** Technical report, ETH-Zürich, 1998. 43

[96] E. W. Kamen. **Multiple target tracking based on symmetric measurement equations**. *IEEE Transactions on Automatic Control*, **37**(3):371–374, Mar 1992. 58, 59

[97] E. W. Kamen and C. R. Sastry. **Multiple target tracking using products of position measurements**. *IEEE Transactions on Aerospace and Electronic Systems*, **29**(2):476–493, Apr 1993. 58

[98] H. P. Chiu, X. S. Zhou, L. Carlone, F. Dellaert, S. Samarasekera, and R. Kumar. **Constrained optimal selection for multi-sensor robot navigation using plug-and-play factor graphs**. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 663–670, May 2014. 74, 121

[99] **FCC DSRC (on 31 Jan 2017)**. `https://www.fcc.gov/dedicated-short-range-communications-dsrc-service`. 79

[100] **ETSI DSRC (on 31 Jan 2017)**. `http://www.etsi.org/technologies-clusters/technologies/intelligent-transport/dsrc`. 79

[101] Fan Bai, Daniel D Stancil, and Hariharan Krishnan. **Toward understanding characteristics of dedicated short range communications (DSRC) from a perspective of vehicular network engineers**. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 329–340. ACM, 2010. 79

[102] James J Caffery and Gordon L Stuber. **Overview of radiolocation in CDMA cellular systems**. *IEEE Communications Magazine*, **36**(4):38–45, 1998. 79

[103] Dennis D McCrady, Lawrence Doyle, Howard Forstrom, Timothy Dempsey, and Marc Martorana. **Mobile ranging using low-accuracy clocks**. *IEEE Transactions on Microwave Theory and Techniques*, **48**(6):951–958, 2000. 80

[104] N. Drawil and O. Basir. **Vehicular Collaborative Technique for Location Estimate Correction**. In *2008 IEEE 68th Vehicular Technology Conference*, pages 1–5, Sept 2008. 80

[105] N. Sünderhauf and P. Protzel. **Switchable constraints for robust pose graph SLAM**. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1879–1884, Oct 2012. 94, 95, 100, 103

[106] E. Olson, J. Leonard, and S. Teller. **Fast iterative alignment of pose graphs with poor initial estimates**. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 2262–2269, May 2006. 94

[107] Y. Bar-Shalom, F. Daum, and J. Huang. **The probabilistic data association filter**. *IEEE Control Systems*, **29**(6):82–100, Dec 2009. 95

[108] Yaakov Bar-Shalom and Edison Tse. **Tracking in a cluttered environment with probabilistic data association**. *Automatica*, **11**(5):451–460, 1975. 95

[109] Erik P Blasch and Mike Hensel. **Fusion of distributions for radar clutter modeling**. Technical report, AIR FORCE RESEARCH LAB WRIGHT-PATTERSON AFB OH, 2004. 101

[110] Anastassopoulos, G. A. Lampropoulos, A. Drosopoulos, and N. Rey. **High resolution radar clutter statistics**. *IEEE Transactions on Aerospace and Electronic Systems*, **35**(1):43–60, Jan 1999. 101

[111] Y. Z. Ma, C. Cui, B. S. Kim, J. M. Joo, S. H. Jeon, and S. Nam. **Road clutter spectrum of BSD FMCW automotive radar**. In *2015 European Radar Conference (EuRAD)*, pages 109–112, Sept 2015. 101

[112] **Radar Tutorials (on 21 Feb 2018)**. `http://www.radartutorial.eu`. 101

[113] **Digital Test Field (on 21 Feb 2018)**. `http://testfeld-a9.de`. 101, 102, 134

[114] B. Ngu Vo and B. Tuong Vo. **Multi-Sensor Multi-object Tracking with the Generalized Labeled Multi-Bernoulli Filter**. *ArXiv e-prints*, February 2017. 113, 115

[115] S. Reuter, B. T. Vo, B. N. Vo, and K. Dietmayer. **The Labeled Multi-Bernoulli Filter**. *IEEE Transactions on Signal Processing*, **62**(12):3246–3260, June 2014. 113, 114

[116] B. T. Vo and B. N. Vo. **Labeled Random Finite Sets and Multi-Object Conjugate Priors**. *IEEE Transactions on Signal Processing*, **61**(13):3460–3475, July 2013. 114

[117] B. N. Vo, B. T. Vo, and H. G. Hoang. **An Efficient Implementation of the Generalized Labeled Multi-Bernoulli Filter**. *IEEE Transactions on Signal Processing*, **65**(8):1975–1987, April 2017. 115

[118] B. N. Vo, B. T. Vo, and D. Phung. **Labeled Random Finite Sets and the Bayes Multi-Target Tracking Filter**. *IEEE Transactions on Signal Processing*, **62**(24):6554–6567, Dec 2014. 115

[119] D. Schuhmacher, B. T. Vo, and B. N. Vo. **On performance evaluation of multi-object filters**. In *2008 11th International Conference on Information Fusion*, pages 1–8, June 2008. 116

[120] Martin Liggins II, David Hall, and James Llinas. *Handbook of Multisensor Data Fusion: Theory and Practice*. CRC press, 2017. 116

[121] M. Baum and U. D. Hanebeck. **The Kernel-SME filter for multiple target tracking**. In *Proceedings of the 16th International Conference on Information Fusion*, pages 288–295, July 2013. 121

[122] Yong Joo Lee and Edward W. Kamen. **SME filter approach to multiple target tracking with false and missing measurements**, 1993. 121

[123] Tiancheng Li, Shudong Sun, Miodrag Bolić, and Juan M. Corchado. **Algorithm design for parallel implementation of the SMC-PHD filter**. *Signal Processing*, **119**:115 – 127, 2016. 121

[124] William H Press. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007. 123

[125] **Smart Adaptive Data Aggregation (on 1 May 2018)**. `http://www.projekt-sada.de/`. 126

[126] Susana Alcalde Bagüés, Wendelin Feiten, Tim Tiedemann, Christian Backe, Dhiraj Gulati, Steffen Lorenz, and Peter Conradi. **Towards Dynamic and Flexible Sensor Fusion for Automotive Applications**. In Tim Schulze, Beate Müller, and Gereon Meyer, editors, *Advanced Microsystems for Automotive Applications 2016: Smart Systems for the Automobile of the Future*, pages 77–89. Springer International Publishing, Cham, 2016. 126

[127] Wendelin Feiten, Susana Alcalde Bagüés, Michael Fiegert, Feihu Zhang, Dhiraj Gulati, and Tim Tiedemann. **A new concept for a cooperative fusion platform**. In *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 128–133, 2016. 126