Very-Large-Scale Distributed Map Making Using Multiple Vehicles

Ashkan Amirsadri

A thesis submitted for the degree of
Doctor of Philosophy
The Australian National University

September 2014

Except where otherwise indicated, this thesis is my own original work.

Ashkan Amirsadri
12 September 2014

to my precious family

Acknowledgments

TO BE WRITTEN.

Abstract

The last two decades have witnessed a growing global demand for high quality, up-to-date mapping datasets from urban environments. However, the majority of current map making solutions are labour intensive, cost ineffective and error prone. This has prompted digital map publishers to seek automated solutions for creating accurate, reliable maps. This work is inspired by a real-world project called AutoMap where objects of interest and assets visible from the road scene are automatically extracted from video data captured by land vehicles and geo-located to form a map.

This thesis provides a flexible solution to the problem of building and maintaining a very-large-scale map using multiple vehicles. In particular, we consider producing a map of landmarks on the scale of thousands of kilometres in an outdoor environment. A setup as described in this work enables a continuously updated database of road scene information at a fraction of the cost compared to the manual alternative. According to the proposed framework, geographically located information from the road scene is gathered continuously (over long periods of time) on a very large scale by a fleet of distributed vehicles. Each vehicle is equipped with a range of low-cost sensors including three cameras, a Global Positioning System (GPS), an Inertial Measurement Unit (IMU), a processor and a 3G modem. In the developed solution, the mapping algorithm is distributed across different vehicles each given the task of producing and updating a local map. The vehicles selectively (and asynchronously) communicate maps to and from a central station in a bandwidth-constraint environment. This central station combines the potentially overlapping local maps to compile a global map. The developed multi-vehicle data fusion framework complies with the essential practical requirements of the AutoMap project. In particular, our solution is efficient in terms of both computational complexity and communication bandwidth. Moreover, the proposed communication architecture is scalable and is capable of dealing with time-varying overlapping map sizes.

This thesis also addresses a particular, yet prominent aspect of mapping systems, namely quality assessment. A new concept of map quality for specialised road mapping applications such as AutoMap is established. We derive a particular type of error measure, known as the Directional Map Error (DIMER) metric, which is capable of reflecting the accuracy of landmark maps in a more meaningful way. The new metric can be tuned to fit a wide spectrum of mapping scenarios and can be deployed by both scientific and business communities as a tool for comparing the performance of different mapping techniques. We also devise a systematic approach known as the Covariance Trajectory Perturbation (CTP) algorithm which is capable of enhancing the DIMER-based quality of obtained maps when incorporated into an EKF-SLAM structure. The effectiveness of this approach in general single-vehicle and multi-vehicle settings are examined.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The last two decades have witnessed a growing global demand for high quality, up-to-date mapping datasets from urban environments. Different application domains such as location-based services, asset management and route finding navigation can benefit from accurate geo-location of road scene objects. However, the majority of current map making solutions are labour intensive, cost ineffective and error prone. This has prompted digital map publishers to seek automated, cost-effective solutions for creating accurate, reliable maps of objects and assets visible from the road.

The inspiration behind this work stems from a real-world project called *AutoMap*. This project addresses the above problems by developing cost-effective technologies for the automatic creation of digital maps to support the growing data requirement of the personal navigation market. The AutoMap project uses advanced computer vision techniques to automatically extract and geolocate road signs and other objects of interest[1] from video footage captured by survey land vehicles.

This thesis considers the problem of building and maintaining a very-large-scale map using multiple vehicles. In the proposed solution, geographically located information from the road scene is gathered continuously on a very large scale by a fleet of distributed vehicles and sent back to a central server where a global database is compiled.

The current work explores several different aspects of this general problem area. Accurate and reliable geo-localisation, efficient communication, computational feasibility and quality assessment are ~~among~~ the main topics addressed in this thesis. The methods used in this project belong to a broader research area involving information fusion and simultaneous localisation and mapping. While the current chapter addresses some of these disciplines from a high level point of view, these topics are further elaborated in this thesis from a more technical, detailed perspective.

The ability to collect, transmit and process information has been studied extensively in the recent decades as part of the universal phenomenon commonly known as the information age. Sensor enriched infrastructures can be found in many application domains such as transportation, defence, surveillance and engineering in today's world. The abundance of information in the world makes it difficult to translate this information into useful and sensible data. This makes the search to find

[1] This thesis will preliminary focus on traffic signs as landmarks of interest.

intelligent solutions to integrate and interpret different sources of information desirable.

Information fusion aims at a synergistic exploitation of multiple information sources to obtain more information than is present in any individual source [Yan et al., 2007; Braun et al., 2009]. In other words, by utilising the complementary properties of the different information sources, the information fusion process attempts to reduce ambiguities and uncertainties in the measured information. In typical data fusion and estimation problems, multiple noise-corrupted random variables need to be fused together to obtain an improved estimate about the underlying state of a system together with a measure of accuracy.

Simultaneous Localisation and Mapping (SLAM) is a popular technique in the literature [Durrant-Whyte and Bailey, 2006; Kim and Sukkarieh, 2003; Dissanayake et al., 2000b; Gutmann and Konolige, 1999] to address complex mapping problems under conditions of process and sensor noise and possible modeling errors. This algorithm first appeared in a seminal paper by Smith, Self and Cheeseman [Smith and Cheeseman, 1986] and it has received a considerable amount of attention by the robotics community [Lázaro and Castellanos, 2010; Menegatti et al., 2009; Castellanos et al., 2000; Feder et al., 1999]. Broadly speaking, SLAM is the process of concurrently building a map of the environment while using the map to estimate the location of the robot in that unknown environment.

1.1 Multi-vehicle Simultaneous Localisation and Mapping

Although most of the initial interest in SLAM considered the problem of mapping and localisation with a single vehicle, the first decade of the twenty-first century saw a substantial interest in multi-vehicle localisation and mapping. The advantages of using multiple, cooperative, vehicles in exploration and mapping applications, compared to the single vehicle case are well known and intuitive, e.g. redundancy, improved accuracy in mapping etc. [Fox et al., 2000; Burgard et al., 2000; Cao et al., 1997].

At a high-level, information fusion is the fundamental tool required for multi-vehicle SLAM as, on an abstract level, the problem is about combining numerous sources of information (that may be correlated) about a common parameter in order to increase one's knowledge about that parameter.

In multi-vehicle SLAM the problem of where this fusion occurs and how information is shared is a practical problem and is one that motivates much of the work in the first three chapters of this thesis (along with similar work discussed subsequently). Different multi-vehicle data fusion architectures have been suggested and implemented for tasks such as autonomous navigation [Bryson and Sukkarieh, 2005; Kim, 2004; Williams, 2001], exploration and mapping [Nerurkar et al., 2009; Stipes et al., 2006; Thrun, 2001] and target tracking [Nettleton, 2003; Ong et al., 2003; Liggins et al., 1997].

The most obvious and traditional data fusion architecture is a fully-centralised one where all the raw sensor data from multiple sources is transmitted to a central station for fusion (e.g. using a large Kalman filter). Works such as [Fenwick et al., 2002] and [Thrun and Liu, 2005] provide fully-centralised approaches to the multi-vehicle SLAM problem. The main disadvantage of a centralised solution is the communication and networking complexity required. The central station also offers a single point of failure and thus centralised solutions in general are less redundant and robust. In addition, the sophistication and heavy computational load at the central station might lead to an undesirable computational bottleneck. However, centralised solutions are convenient in numerous practical applications where it is undesirable for the vehicles to communicate between themselves.

In contrast to the centralised systems, fully decentralised architectures often have no central processing station. In such systems, individual stations (e.g. individual vehicles) can perform data fusion in a fully autonomous manner, while receiving information from and transmitting information to other particular stations. In other words, fusion occurs locally at each station on the basis of local observations and the information received from neighbouring stations. Examples of decentralised SLAM can be found in [Nettleton et al., 2006; Sharon et al., 2003; Ong et al., 2003]. Of course, in general, decentralised solutions are more robust to failure of a given station.

Both fully-centralised and conventional decentralised architectures have proven to be effective in numerous mapping applications. However, without additional local processing it turns out that both methods fail to provide a practical and flexible solution to large-scale (millions of mapping points) mapping where limited bandwidth and processing power is a real concern. This is particularly true when the constraint of a centralised architecture is dictated by the problem. This will be discussed in more details in the current and next chapters as part of the real-world motivation behind this work.

Despite some fundamental work (e.g. [Nettleton et al., 2006; Nettleton, 2003]), the problem of selective communication has not been extensively addressed in the study of multi-vehicle information fusion (e.g. [Bryson and Sukkarieh, 2007]). In large-scale, low-bandwidth mapping applications, sending all the local information to the central station is not feasible due to the limited system and communication resources present in practice. Information tailoring is necessary to avoid high communication costs and other bandwidth constraints in a distributed data collection system. Consequently, only the most valuable information should be selected and transmitted. This is the avenue that we follow in Chapter 3.

In addition, the majority of the existing multi-vehicle SLAM techniques suffer from the growing size of the local maps within individual nodes. Due to the large number of features and the rapidly increasing map size, the SLAM algorithm fails to fulfil the requirements of large-scale applications. The ramification is an immense memory and computational load on the vehicles. Consequently, appropriate strategies must be applied to limit the size of the SLAM filters in very large-scale environments. We discuss a particular pruning strategy in Chapter 3.

Figure 1.1: Road sign detection and localisation in the AutoMap project.

1.2 Thesis Motivation

Digital maps quickly become obsolete and need to be updated regularly[2]. Traditional map making solutions are labor intensive, error prone, slow and cost ineffective. A large portion of the map development cost using these methods is due to manual methods, since they rely heavily on human labour. The ramification is the low speed and inaccuracies in the mapping process. In addition, the data collection devices usually utilised by map making companies[3] employ high-grade sensors, thus are not affordable for large-scale mapping applications. AutoMap addresses the above problems by developing cost-effective technologies for the automatic creation of digital maps to support the growing data requirement of the personal navigation market. At its core, the project uses advanced computer vision techniques [Overett et al., 2009] to automatically extract and geolocate road scene objects (e.g. traffic signs) from recorded video footage captured by survey land vehicles (see Figure 1.1). Such objects are of interest to third party companies like mapping companies and road asset managers. Figure 1.2 shows an example map output from this project.

One of the solutions offered by the AutoMap project is a passive data collection scheme using a set of low-cost in-vehicle sensor platforms. In this solution, geographically located information from the road scene is gathered continuously (over long periods of time) on a very large scale by a fleet of distributed vehicles such as taxis, garbage trucks, delivery vans etc. and selectively sent back to a central server where a global database is compiled (see Figure 1.3). Figure 1.4 gives an indication of the size of the problem that we are addressing. The video captured by the vehi-

[2] Anecdotally, 5-15 % of the road sign inventory changes every year.

[3] The terms map mapping company, mapping company and road mapping companies are used interchangeably throughout this thesis.

Figure 1.2: Example map output from the AutoMap project. The road signs of interest are extracted and geo-located. Location: Canberra, Australia (Source: Google Earth).

cles is automatically analysed to extract road signs of interest. Such information is currently collected in a manual fashion and updated only every few years which is a very cumbersome and error prone process. A setup as described in this work enables a continuously updated database of road scene information at a fraction of the cost compared to the manual alternative. Each sensor platform, as installed in each fleet vehicle, consists of three cameras, a Global Positioning System (GPS), a 3-axis accelerometer, a 3-axis gyroscope, a 3-axis magnetometer, a processing unit and a 3G modem[4]. Data from the sensors are continuously stored on a local hard drive, and later analysed by the local processing unit in order of importance to maximise a cost function representing the value of extracted information.

Analysing the vast amount of information gathered from the sensors and transmitting it back to the central server is a challenging task as the platform installed in each vehicle suffers from a number of constraints. These constraints can be categorised as 1) Communication bandwidth 2) Processing power, and 3) Memory and storage. One of the key constraints this thesis sets out to address is the limited communication bandwidth provided by the 3G modem. The limited communication bandwidth not only makes it impossible to transmit all raw sensory data to a central server and analyse it there, but even the amount of extracted, symbolic information poses a challenge. Clearly, a communication architecture that allows selective communication is needed to handle this case.

In general, there are different parties that can benefit from certain accurate information regarding road signs (e.g. type and geo-location). These groups include, but not limited to:

1. Navigation device owners
2. Asset owners (road authorities and city councils)
3. Mapping companies, content providers and surveying/engineering firms

[4]This type of sensor bundle is now also becoming commonplace in mobile phones and tablets.

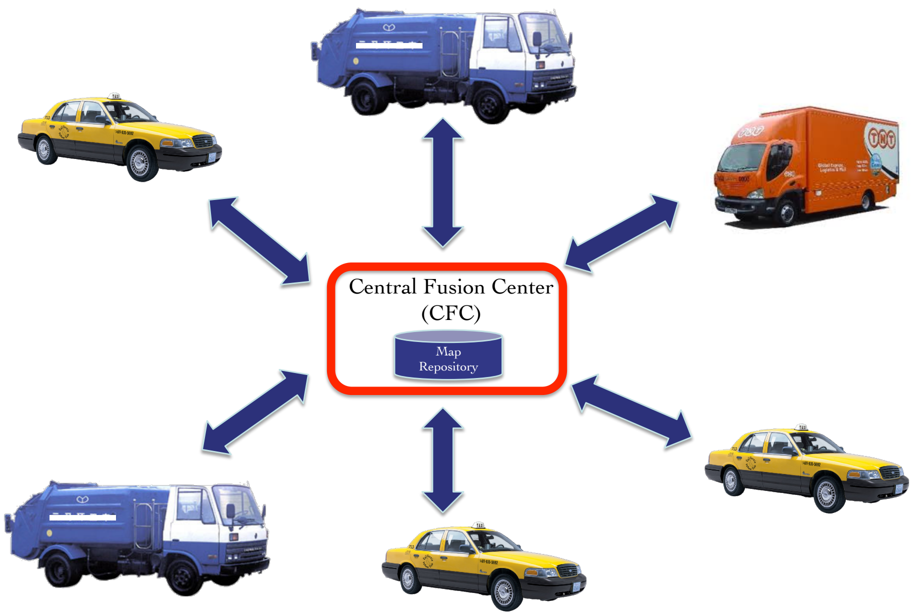


Figure 1.3: The distributed information fusion model keeps the central map repository up-to-date.

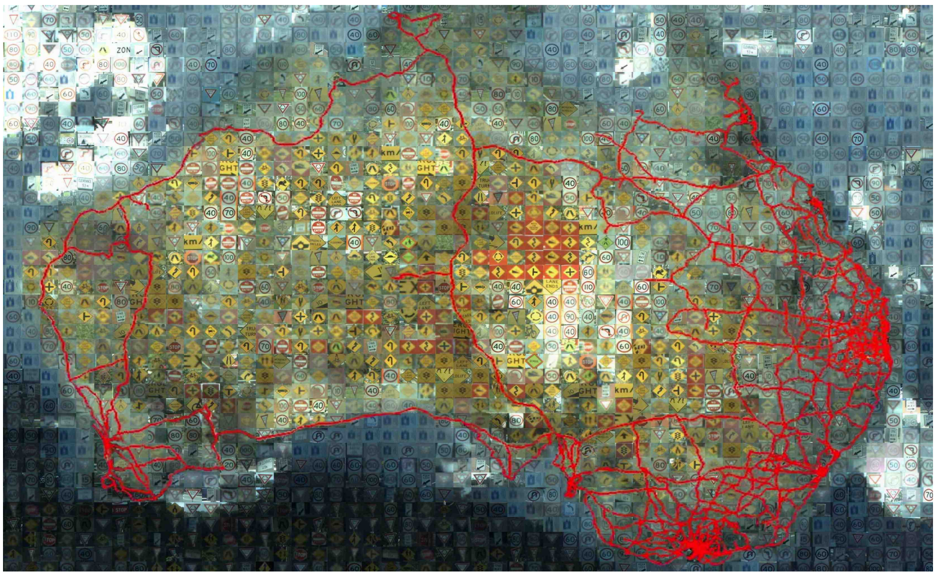


Figure 1.4: The red lines on the Australia map indicate the roads from which video has been captured and automatically analysed for road signs (as of 2011). The challenge addressed in this thesis is how to further automate this task by having a fleet of ad-hoc surveying vehicles efficiently communicating their observations to a central server. Since 2011, many more roads have been added to this map. Gary Overett is thanked for constructing the image.

The first group comprises the consumers of satellite/navigation devices. The AutoMap project can provide this group with a better user experience by offering richer, fresher, more reliable maps. Asset owners (such as road authorities and city councils) constitute the second group of map users. These parties are interested in accurate positioning of road signs in order to construct an up-to-date inventory for different roads. In addition, accurate monitoring of potential changes to the location of road signs (also known as change detection) is an important aspect of road asset management. The third group includes companies who control the personal navigation market by providing map contents for the interested parties (such as the first two groups). Sensis and TomTom are two examples of this group. These companies seek to acquire diverse, high quality maps consisting of the geo-location of road signs of interest in the environment. The AutoMap project can offer competitive edge to this group by providing maps at a reduced cost and a higher update rate (due to the automated map making process). This information is mainly used to provide navigation and routing advice for the first group and create digital map inventories for the second group. The block diagram presented in Figure 1.5 visualises the relationship between the AutoMap project and the above beneficiary parties.

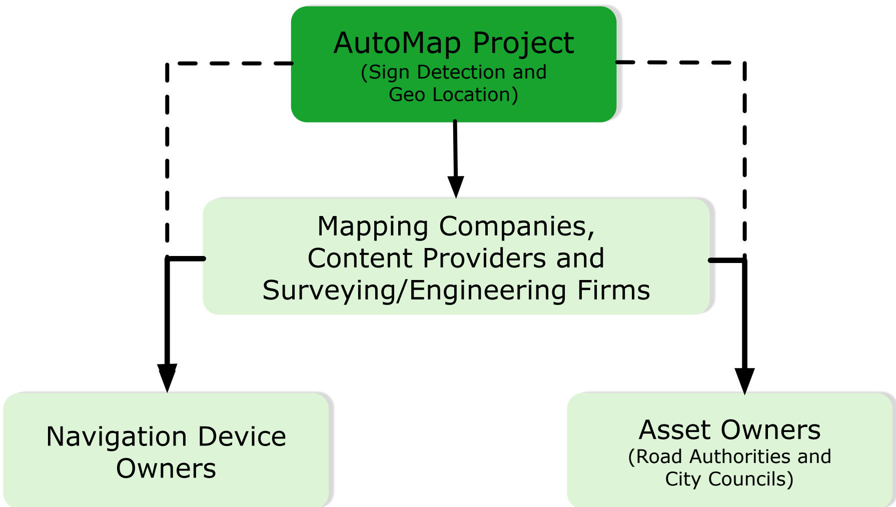


Figure 1.5: The relationship between the AutoMap project and different beneficiary parties.

1.3 Quality Assessment in Mapping Applications

A particular, yet prominent facet of mapping applications is the way in which their performance is evaluated. Accessing meaningful tools for systematic comparison between the results of various mapping techniques is desirable to the scientific as well as the business community. To this end, the majority of the works in the literature have attempted to utilise strategies to demonstrate the effectiveness of their methods and the precision of their results based on a set of criteria. Notwithstanding, it is widely believed that (e.g. [Jaulmes et al., 2009; Kümmerle et al., 2009; Mourikis and Roumeliotis, 2006]) the robotic mapping community lacks a generally accepted, standard methodology for quality assessment and comparison between the results of different algorithms. Few researchers have recognised the importance of a generic quality metric which can be applied to calculate a quantitative measure of precision that thoroughly manifests the characteristics of systems in real-world applications. Most existing solutions fail to provide a comprehensive evaluation in specialised mapping scenarios, since they do not fully reflect the quality and accurateness of mapping processes. Moreover, a large majority of the existing solutions are subjective, influenced by individual perceptions and hence debatable. This work sets out to address this issue and clarify the concept of 'quality assessment' for practical robotic mapping applications. Chapter 4 introduces a new map error measure known as *DIMER metric* which is able to capture the quality of mapping frameworks in a more meaningful way compared to commonly used methods.

Having access to a generic quality measure for robotic mapping applications (such as the DIMER metric) then poses new practical questions. The main question would be 'how can a given mapping system be revised so as to generate more accurate maps when judged using the new map quality metric?'. This specific problem falls into the more general area of criteria-based estimation in which certain criteria are optimised in order to achieve performance improvements. This is the main subject we aim to address in Chapter 5 of this thesis. Specifically, the incorporation and application of the newly designed DIMER metric in different components of the previously mentioned distributed multi-vehicle mapping system is studied.

1.4 Thesis Contribution

The first contribution of this work is the development of a multi-vehicle data fusion framework for a real-world inspired road mapping application. We introduce a hierarchical data fusion architecture and a communication scheme that allows the communication of sub-maps of arbitrary size. In 3.3, a practical pruning algorithm based on a measure of information gain is introduced to overcome the problem of progressively growing map sizes within individual vehicles. Moreover, the required communication bandwidth is reduced significantly by selectively transmitting sub-maps with the largest information contribution to a central server, where a global map repository is maintained. The proposed communication architecture is flexible in the sense that it is capable of dealing with dynamically changing map sizes in the entire system. In addition, the fusion algorithm offered in this work ensures that map estimates are integrated in a consistent and robust fashion. The general implementation complies with the essential practical requirements of the AutoMap project, as the real-world inspiration behind this work.

The second contribution of this thesis is the development of a new concept of map quality for specialised road mapping applications (such as AutoMap). In Section 4.5, we introduce and formulate a new map error measure known as DIMER metric which is able to capture the accurateness of mapping systems in a more meaningful way compared to traditional methods. The DIMER metric is developed for two fundamentally distinct cases, depending on the accessibility of ground-truth information.

In addition, this work investigates the incorporation of the newly designed DIMER metric into some of the most common estimation and mapping frameworks with the aspiration of producing high-quality map estimates. A new methodology called the Covariance Trajectory Perturbation (CTP) is developed in Section 5.4 which is capable of enhancing the quality of obtained maps when integrated into the standard EKF-SLAM structure. The applicability and performance of the CTP method is analysed in detail in Section 6.3. This solution is then also systematically expanded to the previously mentioned multi-vehicle SLAM system with efficient communication. Finally, this thesis examines the impact of incorporating a filtering structure known as the converted measurement Kalman filter with debiasing compensation (D-CMKF) into the non-linear SLAM problem in order to reduce the unwanted bias effect in the mapping process. The performance of the resulting mapping system and the behaviour of the DIMER metric is analysed for the multi-vehicle SLAM setting. It will be shown that this integration can effectively reduce the estimated map error with respect to the true map, in both local and global maps.

1.5 Thesis Structure

This thesis consists of seven chapters. With the exception of the current chapter, each chapter begins with an introduction and concludes by a summary. The literature review regarding each topic is provided in its respective chapter.

Chapter 2 presents the theoretical background and preliminaries required to address the large-scale multi-vehicle mapping problem outlined in this work. The objective is to evaluate and provide the information in line with the concepts used in this thesis. An overview of some of the most popular statistical estimation and filtering techniques is conducted. The simultaneous localisation and mapping (SLAM) is presented as one of the key solutions to the problem of map making in the presence of process and observation noise. In addition, the chapter formulates the problem at hand and discusses the existing conceptual and technical challenges in designing a scalable mapping framework. A summary of the principal literature describing different methodologies to the multi-vehicle SLAM problem is also presented and the shortcomings of the existing methods are described.

Chapter 3 provides a flexible solution to the problem of multi-vehicle SLAM for very-large-scale mapping applications. An overview of different components of the mapping system (such as the local SLAM filter, channel filter and the central fusion centre) is presented. A data fusion framework based on the popular Covariance Intersection (CI) algorithm is devised to tackle the problem of redundant information propagation. To avoid excessive communication costs, a communication scheme is proposed which operates by selectively transmitting the most informative information from individual vehicles. Moreover, a practical pruning algorithm is applied to restrict the size of local maps inside the vehicles.

Chapter 4 is concerned with quality assessment in practical mapping systems with a focus on the real-world application in this work. A review of some of the most common techniques used to evaluate the quality of localisation and mapping systems is provided. The shortcomings of the existing methods are discussed and the need for a more meaningful quality measure for practical mapping applications is justified. A new map quality measure known as DIMER metric is devised which incorporates type and spatial orientation of the existing map elements.

Chapter 5 investigates different techniques for criteria-based estimation and mapping with a direct focus on incorporating the DIMER metric into different facets of mapping systems. The practical problem of fusing two or more maps with unknown correlation in order to achieve a single, more accurate map (when judged using the DIMER criteria) is explored. For this purpose, the covariance intersection algorithm is revisited in this chapter, this time with the DIMER metric as its minimisation criterion. Furthermore, a systematic method known as criteria-based covariance trajectory perturbation (CTP) is proposed to enhance the quality of maps obtained using the EKF-SLAM algorithm. To tackle an unwanted bias problem in the map estimates (amplified due to the utilisation of the CTP method), a debiasing technique coupled with a converted measurement Kalman filtering structure (D-CMKF) is devised and integrated into the EKF-SLAM structure.

Chapter 6 presents the simulation results of the methods developed throughout this thesis. At first, the effectiveness of the proposed multi-vehicle mapping solution for large-scale environments is demonstrated. The results pertaining to the new DIMER metric and criteria-based mapping is presented next. In particular, an extensive analysis on the applicability and performance of the covariance perturbation method is provided. The incorporation of the proposed algorithms into the distributed mapping system is investigated. Furthermore, the effect of applying the D-CMKF structure to the local EKF-SLAM filter of individual vehicles is studied. The performance of the system is assessed and compared against other discussed solutions.

Finally, **Chapter 7** provides the conclusion of this work along with suggestions on areas of future work.

Chapter 2

Background and Problem Formulation

2.1 Introduction

The first objective of this chapter is to provide a theoretical background and preliminaries required to address the large-scale multi-vehicle mapping problem in this thesis. The second objective is to present and formulate the problem at hand and to discuss the existing conceptual challenges and difficulties that must be tackled in designing a scalable multi-vehicle mapping framework. The limitations and constraints imposed on our particular system are elaborated and the subtle practical differences with the existing methods are outlined.

The structure of this chapter is as follows. Section 2.2 provides an overview on some of the most popular state-of-the-art statistical estimation and filtering techniques. The Kalman Filter (KF), a practical versatile procedure which combines noisy sensor measurements to estimate the state of a system with uncertain dynamics, will be presented first. Its nonlinear counterpart, the Extended Kalman Filter (EKF), will be formulated and discussed subsequently as the de-facto approach for nonlinear estimation and sensor fusion in state space. These two algorithms form the cornerstones of most of the stochastic filtering in this work. This section also includes a brief synopsis on estimation in information space and a discussion on the information form of the Kalman filter. The information space offers some interesting characteristics which will be exploited in the next chapter for communication and fusion of data in the proposed multi-vehicle mapping system.

Simultaneous Localisation and Mapping (SLAM) is presented in Section 2.3 as one of the most widely used tools in the literature to address complex mapping problems under the influence of process and sensor noise and possible modelling errors. Broadly speaking, SLAM is the process of concurrently building a map of the environment while using the map to estimate the location of the robot in that unknown environment.

Section 2.4 discusses the problem of localisation and mapping using multiple vehicles. The strategical advantages of using multiple, cooperative, vehicles in exploration and mapping applications, compared to the single vehicle case will is explored. Furthermore, the current literature is surveyed and various algorithms and techniques applied in different multi-vehicle scenarios is introduced and their respective properties are discussed.

Typically, one of the key decisions that has to be made prior to the design of a multi-agent multi-sensor system is the way the system is set up from an architectural point of view. A brief review of some of the most popular multi-vehicle data fusion architectures is conducted in Section 2.5. Three main architectures namely centralised, decentralised and hierarchical are analysed and the advantages and disadvantages of each architecture are discussed.

Major challenges in building and maintaining maps in large-scale environments are outlined in Section 2.6. Fundamental issues that need to be considered and rectified in the implementation of large-scale mapping systems are outlined and the shortcomings of the existing methods are discussed.

A description of the real-world practical project under study and the main resource constraints imposed on the project are described in Section 2.7. The devised solution must be able to cope with these limitations across a range of practical situations. Therefore, addressing the existing issues are of absolute necessity in the design and formulation of the large-scale distributed mapping system.

Finally, Section 2.8 devises a hierarchical architecture with a central base station to combine the local maps obtained from individual vehicles into a global map. In this chapter, the proposed architecture is addressed from a high-level point of view. Different components of the distributed mapping system, the task associated with each component and the way these system entities interact with each other are presented in Chapter 3.

2.2 Statistical Filtering Techniques

There is an abundance of literature on the area of statical estimation and filtering. The aim of this section is to establish the mathematical framework for different estimation algorithms deployed throughout this thesis. These algorithms are either directly implemented in this work, or their analysis and properties have been considered essential in decision making regarding the design of the large-scale multi-vehicle mapping system. The current background material on estimation algorithms is presented for completeness; thus, advanced readers may skip this section.

2.2.1 Kalman Filter (KF)

The Kalman filter (KF) is a recursive estimation technique that estimates the underlying states of a linear, dynamical system through a process of prediction and update. The algorithm minimises the mean squared error (MSE) between the estimated and the real value of a state $\mathbf{x}$.

The Kalman filter operates under two main assumptions. Firstly, the system under study is a linear time-varying system and secondly, all error terms and measurements have a Gaussian (Normal) probability density distribution. If any of the aforementioned assumptions are violated, the algorithm will no longer guarantee an efficient, optimal state estimation. Therefore, these two conditions are imperative for proper functionality and optimality of the Kalman filter in any application.

The operation of the filter relies on the proper definition of a dynamic process model, an observation model and a stochastic model [Brown et al., 1992] which are presented next.

Linear Process Model[1]

A linear, time-varying discrete-time system can be expressed mathematically using the state space representation via the following difference equation:

$$\mathbf{x}(k) = \mathbf{F}\mathbf{x}(k-1) + \mathbf{B}\mathbf{u}(k) + \mathbf{G}\mathbf{w}(k), \tag{2.1}$$

where $\mathbf{x}(k)$ is the state vector at time k, $\mathbf{F}$ is a linear state transition matrix relating the current state to the previous state at time $k-1$, $\mathbf{u}(k)$ is the input control vector while $\mathbf{B}$ is the model that links the control vector to the current state, and $\mathbf{w}(k)$ represent the process noise which relates to the state vector through matrix $\mathbf{G}$.

The process noise $\mathbf{w}(k)$ is a Gaussian white noise that accounts for the inherent uncertainties in the state transition matrix and control input and is described according to

$$\begin{aligned} \mathrm{E}[\mathbf{w}(k)] &= 0 \qquad \forall k, \\ \mathrm{E}[\mathbf{w}(k)\mathbf{w}^T(j)] &= \begin{cases} \mathbf{Q}(k) & k = j \\ 0 & k \neq j. \end{cases} \end{aligned}$$

In other words, $\mathbf{w}(k)$ is assumed to be a zero mean, uncorrelated random sequence with covariance $\mathbf{Q}(k)$.

Linear Observation Model[2]

At time k an observation $\mathbf{z}(k)$ of the state $\mathbf{x}(k)$ is taken according to

$$\mathbf{z}(k) = \mathbf{H}\mathbf{x}(k) + \mathbf{v}(k), \tag{2.2}$$

where $\mathbf{H}$ is the linear observation model which links the current state to the observation vector. In addition, $\mathbf{v}(k)$ is the observation noise that models the uncertainties associated with the made observation. The observation noise is also assumed to be a zero mean, uncorrelated random sequence with covariance $\mathbf{R}(k)$,

$$\mathrm{E}[\mathbf{v}(k)] = 0 \qquad \forall k,$$

[1] The terms 'Process model' and 'state transition model' are used synonymously throughout this thesis.

[2] The terms 'observation model' and 'measurement model' are used synonymously throughout this thesis.

$$
\mathrm{E}[\mathbf{v}(k)\mathbf{v}^T(j)] = \begin{cases} \mathbf{R}(k) & k = j \\ 0 & k \neq j. \end{cases}
$$

Moreover, the process and observation noises are assumed to be mutually exclusive (uncorrelated) and therefore do not affect each other,

$$
\mathrm{E}[\mathbf{w}(k)\mathbf{v}^T(j)] = 0 \qquad \forall k, j
$$

Given the above observation model and the process model presented earlier, the Kalman filter computes a recursive estimate of the system's states at time k, given all observations up to time k. This estimate is denoted by $\mathbf{x}(k|k)$ and is referred to as *a priori* state estimate throughout this thesis. Likewise, $\mathbf{x}(k|k-1)$ is called the *a posteriori* state estimate at time k, given all observations up to and including time $k-1$.

In essence, the Kalman filter estimates the unknown states of a process using a form of feedback control in a recursive way. The filter estimates the process state at a specific time and then obtains the feedback in form of noisy observations. Based on this, two groups of equations can be defined for the filter, time update equations and measurement update equations. The time update equations propagate in time to get the current state and error covariance estimates. This phase of the algorithm is also referred to as the prediction step. Then, the filter uses the measurement update equations as the feedback to obtain an improved estimate of the states. This is referred to as the update (or estimation) step in the filtering literature. Therefore, the KF recursively conditions the current estimate on all of the past observations [Welch and Bishop, 1995]. This will provide an efficient computational mean to estimate the state of a stochastic process. The prediction and update steps are described below.

KF Prediction Step

The aim of the prediction step of the Kalman filter is to predict the states based upon the past estimations. Hence, the predicted state is derived by taking the expectation of Equation (2.1) with zero process noise, given all the previous observations denoted by $\mathbf{Z}^{k-1}$, according to:

$$
\begin{aligned}
\mathbf{x}(k|k-1) &\triangleq \mathrm{E}[\mathbf{x}(k|\mathbf{Z}^{k-1})] \\
&= \mathbf{F}\mathbf{x}(k-1|k-1) + \mathbf{B}\mathbf{u}(k). \qquad (2.3)
\end{aligned}
$$

The uncertainty in the predicted state at time k is represented by $\mathbf{P}(k|k-1)$ and is determined by taking the expected value of the variance of the error in the state at time k conditioned upon all observations up to time $k-1$ as shown below.

$$
\begin{aligned}
\mathbf{P}(k|k-1) &\triangleq \mathrm{E}[(\mathbf{x}(k) - \mathbf{x}(k|k-1)(\mathbf{x}(k) - \mathbf{x}(k|k-1)^T|\mathbf{Z}^{k-1}] \\
&= \mathbf{F}\mathbf{P}(k-1|k-1)\mathbf{F}^T + \mathbf{G}\mathbf{Q}(k)\mathbf{G}^T. \qquad (2.4)
\end{aligned}
$$

where $\mathbf{P}(k-1|k-1)$ corresponds to the error covariance matrix from the filter's previous step.

KF Update Step[3]

After the occurrence of an observation $\mathbf{z}(k)$ at time k, the updated state is derived by taking the expectation of Equation (2.1) with zero process noise, given all the observations up to and including time k.

$$
\begin{aligned}
\mathbf{x}(k|k) &\triangleq \mathrm{E}[\mathbf{x}(k|\mathbf{Z}^k)] \\
&= \mathbf{x}(k|k-1) + \mathbf{K}(k)\nu(k). \qquad (2.5)
\end{aligned}
$$

where $\mathbf{K}(k)$ and $\nu(k)$ and are known as the *Kalman gain* and the *innovation vector* respectively. The Innovation vector (or *residual*) is determined by subtracting the predicted observation with zero observation noise from the measured one,

$$\nu(k) = \mathbf{z}(k) - \mathbf{H}\mathbf{x}(k|k-1). \qquad (2.6)$$

Equation (2.5) updates the state vector by adding a weighting on the innovation to the latest prediction. The Kalman gain $\mathbf{K}(k)$ which acts as the weighting factor in this equation is chosen so as to minimise the trace of the state covariance matrix. Under the aforementioned Gaussian assumptions on process and observation noise, this is equivalent to the minimisation of the mean squared error of the state estimate. The conventional (minimal trace) Kalman gain is derived as[4]

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{H}\mathbf{S}^{-1}(k), \qquad (2.7)$$

where $\mathbf{S}(k)$ represents the innovation covariance and is determined according to

$$\mathbf{S}(k) = \mathbf{H}\mathbf{P}(k|k-1)\mathbf{H}^T + \mathbf{R}(k) \qquad (2.8)$$

The covariance matrix, which represents the uncertainty in the estimated state, is derived by taking the expected value of the variance of the error in the states at time k given all observations up to time k,

[3] Please note that the update phase of the Kalman filter may be referred to as 'estimation phase'.
[4] The Kalman gain and its derivation are discussed further in Chapter 5.

$$\mathbf{P}(k|k) \triangleq \mathrm{E}[(\mathbf{x}(k) - \mathbf{x}(k|k-1)(\mathbf{x}(k) - \mathbf{x}(k|k-1)^T|\mathbf{Z}^k]$$
$$= [\mathbf{I} - \mathbf{K}(k)\mathbf{H}]\mathbf{P}(k|k-1)[\mathbf{I} - \mathbf{K}(k)\mathbf{H}]^T + \mathbf{K}(k)\mathbf{R}(k)\mathbf{K}^T(k) \quad (2.9)$$

The above formula is also known as the *Joseph form* of the covariance update equation and is valid for any value of the gain $\mathbf{K}(k)$. This equation can be simplified further if the optimal Kalman gain form Equation (2.7) is used. This results in the below covariance update equation:

$$\mathbf{P}(k|k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{H})\mathbf{P}(k|k-1) \quad (2.10)$$

As mentioned before, the above covariance matrix is the minimal trace solution for the Kalman filter.

2.2.2 Extended Kalman Filter (EKF)

In many real-world practical applications, the underlying process and observation models are nonlinear. This precludes the use of the above-mentioned Kalman filter algorithm. However, a different version of the Kalman filter, known as the Extended Kalman Filter (EKF) can be utilised for non-linear systems. The EKF operates by continuously linearising the system model before applying the linear estimation techniques [Maybeck, 1979]. Although there are few theoretical results to indicate the validity of this algorithm, it has been proven that the EKF can operate successfully when the model nonlinearity is relatively benign. The foundation for the construction of the EKF is the Kalman Filter which was outlined in Subsection 2.2.1. EKF equations and the manner in which the filter handles the nonlinearity problem in the system is presented here.

Non-linear Process Model

A non-linear time-varying system in discrete-time can be expressed according to

$$\mathbf{x}(k) = \mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k), \mathbf{w}(k)), \quad (2.11)$$

where $\mathbf{f}(.,.,.)$ is a non-linear state transition function which relates the current state to the previous state $\mathbf{x}(k-1)$, current control input $\mathbf{u}(k)$ and process noise $\mathbf{w}(k)$. The zero-mean Gaussian noise assumptions stated earlier for the Kalman filter also hold for the non-linear system equations.

Non-linear Observation Model

In the general case, the non-linear observation model is expressed by

$$\mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k), \mathbf{v}(k)) \quad (2.12)$$

where $\mathbf{h}(.,.)$ is the observation function at time k which links the observation to the current state and observation noise vector.

EKF Prediction Step

In a similar fashion to what was described in Subsection 2.2.1 for the Kalman filter, the predicted state for the EKF is evaluated with zero process noise, according to

$$\mathbf{x}(k|k-1) = \mathbf{f}(\mathbf{x}(k-1|k-1), \mathbf{u}(k), 0) \tag{2.13}$$

The resulting predicted covariance matrix is given by

$$\mathbf{P}(k|k-1) = \nabla\mathbf{f}_x(k)\,\mathbf{P}(k-1|k-1)\nabla\mathbf{f}_x^T(k) + \nabla\mathbf{f}_w(k)\mathbf{Q}(k)\nabla\mathbf{f}_w^T(k) \tag{2.14}$$

where $\mathbf{P}(k-1|k-1)$ corresponds to the previous error covariance matrix from the estimation step. The term $\nabla\mathbf{f}_x(k)$ in the above equation is the Jacobian[5] of the non-linear state transition function $\mathbf{f}$, with respect to the previous state estimate $\mathbf{x}(k|k-1)$ and the term $\nabla\mathbf{f}_w(k)$ is computed as the Jacobian of the same matrix with respect to the process noise vector $\mathbf{w}(k)$. Both $\nabla\mathbf{f}_x(k)$ and $\nabla\mathbf{f}_w(k)$ are Jacobian matrices which have to be calculated at each prediction step by taking the partial derivatives of $\mathbf{f}$ with respect to their corresponding variables. As a result, the best linear approximation of the state transition function around the current state estimate is obtained. However, the presence of Jacobian terms in the EKF adds a higher level of complexity to the system.

EKF Update Step

After the occurrence of an observation, the state vector is updated by

$$\mathbf{x}(k|k) \;=\; \mathbf{x}(k|k-1) \;+\; \mathbf{K}(k)\nu(k), \tag{2.15}$$

where $\nu(k)$ or the innovation vector is calculated by subtracting the predicted observation with zero observation noise from the measured one,

$$\nu(k) \;=\; \mathbf{z}(k) \;-\; \mathbf{h}(\mathbf{x}(k|k-1), 0). \tag{2.16}$$

As before, the optimal Kalman gain is chosen so as to minimise the trace of the updated covariance matrix. Therefore,

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\nabla\mathbf{h}_x^T(k)\mathbf{S}^{-1}(k), \tag{2.17}$$

where $\mathbf{S}(k)$ is called the innovation covariance which is calculated according to,

$$\mathbf{S}(k) = \nabla\mathbf{h}_x(k)\mathbf{P}(k|k-1)\nabla\mathbf{h}_x^T(k) + \nabla\mathbf{h}_v(k)\mathbf{R}(k)\nabla\mathbf{h}_v^T(k). \tag{2.18}$$

[5]The Jacobian matrix is defined as the matrix of all first-order partial derivatives of a function [Tay et al., 1998].

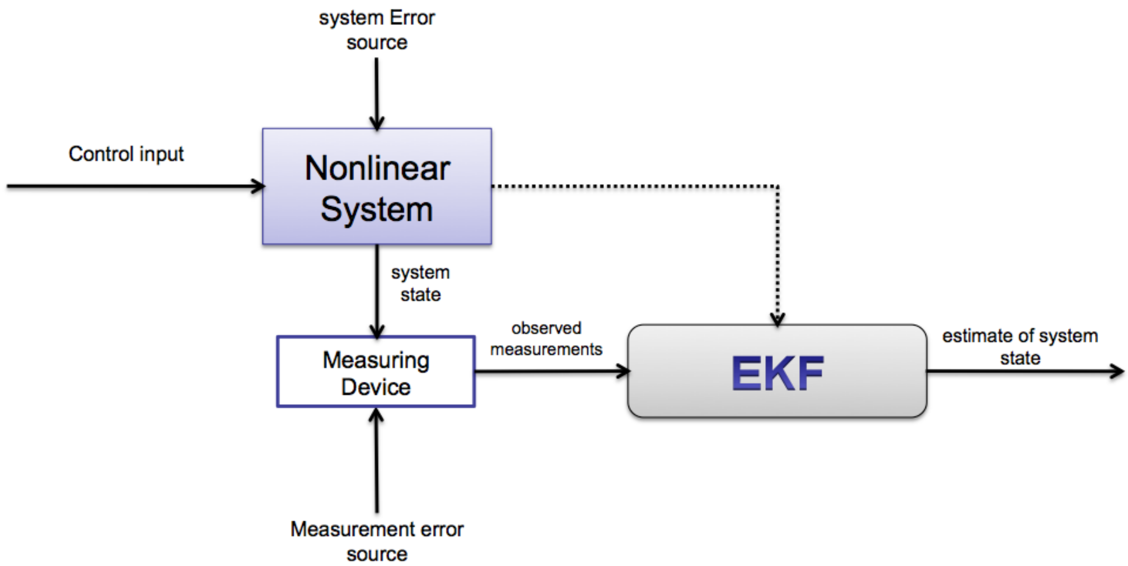


Figure 2.1: A diagram illustrating the operation of the extended Kalman filter (EKF)

In the above equation, the term $\nabla \mathbf{h}_x(k)$ is the Jacobian of the current observation model with respect to the previous state estimate $\mathbf{x}(k-1|k-1)$. Similarly, $\nabla \mathbf{h}_v(k)$ is the Jacobian of the same function with respect to the observation noise $\mathbf{v}(k)$.

Finally, the updated covariance matrix is formed by using

$$\begin{aligned} \mathbf{P}(k|k) &= [\mathbf{I} - \mathbf{K}(k)\nabla \mathbf{h}_x(k)]\mathbf{P}(k|k-1)[\mathbf{I} - \mathbf{K}(k)\nabla \mathbf{h}_x^T(k)]^T \\ &+ \mathbf{K}(k)\nabla \mathbf{h}_v(k)\mathbf{R}(k)\nabla \mathbf{h}_v^T(k)\mathbf{K}^{\mathbf{T}}(k). \end{aligned} \tag{2.19}$$

A block diagram for the extended Kalman filter is illustrated in Figure 2.1.

2.2.3 Information Filter (IF)

The information filter (IF), also known as the *inverse covariance filter*, is an estimation technique which is numerically equivalent to the Kalman filter. However, rather than using the conventional state space representation, this filter is expressed in terms of measures of information. In the information filter framework, the estimated state $\mathbf{x}(k|k)$ and estimated covariance matrix $\mathbf{P}(k|k)$ are replaced respectively by the so-called information state vector $\mathbf{y}(k|k)$ and information matrix $\mathbf{Y}(k|k)$[6] according to the following definitions:

$$\mathbf{y}(k|k) \triangleq \mathbf{P}^{-1}(k|k)\mathbf{x}(k|k) \tag{2.20}$$

$$\mathbf{Y}(k|k) \triangleq \mathbf{P}^{-1}(k|k) \tag{2.21}$$

[6] $\mathbf{Y}(k|k)$ is also known as Fisher information.

It is well known (c.f. [Durrant-Whyte and Henderson, 2008; Bar-Shalom et al., 2004]) that a duality exists between the information filter and the conventional Kalman filter in a way that the prediction step of the information filter is related to the update step of the Kalman filter, and the update of the information filter is comparable to the prediction step of the Kalman filter [Anderson and Moore, 2012]. The following prediction and update equations for the information filter are obtained for the linear process and observation models described by Equations (2.1) and (2.2). We simply provide the final equations here. A full derivation can be found in the early book by Maybeck [Maybeck, 1979].

IF Prediction Step

The predicted information state vector is derived according to

$$\mathbf{y}(k|k-1) = \mathbf{L}(k|k-1)\mathbf{y}(k-1|k-1) + \mathbf{Y}(k|k-1)\mathbf{B}(k)\mathbf{u}(k) \tag{2.22}$$

where $\mathbf{L}(k|k-1)$ is called the similarity transform matrix as is defined by

$$\mathbf{L}(k|k-1) \triangleq \mathbf{Y}(k|k-1)\mathbf{F}(k)\mathbf{Y}(k-1|k-1)^{-1} \tag{2.23}$$

The predicted information matrix is given by

$$\mathbf{Y}(k|k-1) = \Big(\mathbf{F}(k)\mathbf{Y}^{-1}(k-1|k-1)\mathbf{F}^T(k) + \mathbf{G}(k)\mathbf{Q}(k)\mathbf{G}^T(k)\Big)^{-1} \tag{2.24}$$

IF Update Step

Within the information filter framework, when an observation $\mathbf{z}(k)$ happens at time k, the information state vector contribution and its corresponding information matrix associated with that observation are defined as

$$\mathbf{i}(k) \triangleq \mathbf{H}^T(k)\mathbf{R}^{-1}(k)\mathbf{z}(k) \tag{2.25}$$
$$\mathbf{I}(k) \triangleq \mathbf{H}^T(k)\mathbf{R}^{-1}(k)\mathbf{H}(k) \tag{2.26}$$

The update equations of the information filter are simply obtained by adding these respective information contributions to the information vector and the information matrix according to

$$\mathbf{y}(k|k) = \mathbf{y}(k|k-1) + \mathbf{i}(k) \tag{2.27}$$
$$\mathbf{Y}(k|k) = \mathbf{Y}(k|k-1) + \mathbf{I}(k) \tag{2.28}$$

For the general scenario of incorporating n synchronous observations at time k, the sum of information contributions from each observation $\mathbf{z}_j(k)$ is added to the information vector and information matrix as

$$\mathbf{y}(k|k) = \mathbf{y}(k|k-1) + \sum_{j=1}^{n} \mathbf{i}_j(k) \tag{2.29}$$

$$\mathbf{Y}(k|k) = \mathbf{Y}(k|k-1) + \sum_{j=1}^{n} \mathbf{I}_j(k) \tag{2.30}$$

Therefore, with regards to information integration, all update operations are additive [Nettleton et al., 2006; Ong et al.; Thrun et al., 2003]. As a result, the update step is quite simple compared to that of the Kalman filter[7]. This is an appealing characteristic and one of the main advantages of the information filter (and more generally, data fusion in information space) that will be exploited in Chapter 3.

2.2.4 Extended Information Filter (EIF)

The extended information filter is the nonlinear version of the information filter and is numerically equivalent to the extended Kalman filter presented earlier in this section. Although the EIF is not used directly in this thesis, its equations are presented here for completeness. The following prediction and update equations are obtained for the nonlinear process and observation models described by Equations (2.11) and (2.12). A complete derivation of the non-linear information filter equations along with some of its applications are shown in [Thrun et al., 2002].

EIF Prediction Step

$$\mathbf{y}(k|k-1) = \mathbf{Y}(k|k-1)\mathbf{f}(\mathbf{x}(k|k-1), \mathbf{u}(k), 0) \tag{2.31}$$

$$\mathbf{Y}(k|k-1) = \left[\nabla\mathbf{f}_x(k)\mathbf{Y}^{-1}(k-1|k-1)\nabla\mathbf{f}_x^T(k) + \nabla\mathbf{f}_w(k)\mathbf{Q}(k)\nabla\mathbf{f}_w^T(k)\right]^{-1} \tag{2.32}$$

EIF Update Step

The information vector contribution and its associated information matrix are

$$\mathbf{i}(k) = \nabla\mathbf{h}_x^T(k)\left[\nabla\mathbf{h}_v(k)\mathbf{R}(k)\nabla\mathbf{h}_v^T(k)\right]^{-1}[\nu(k) + \nabla\mathbf{h}_x(k)\mathbf{x}(k|k-1)] \tag{2.33}$$

$$\mathbf{I}(k) = \nabla\mathbf{h}_x^T(k)\left[\nabla\mathbf{h}_v(k)\mathbf{R}(k)\nabla\mathbf{h}_v^T(k)\right]^{-1}\nabla\mathbf{h}_x(k) \tag{2.34}$$

where the innovation vector $\nu(k)$ is given by Equation (2.16).

Similar to the linear information filter, the update equations of the extended information filter are simply obtained using

[7] Nevertheless, the prediction step is comparatively more complex as opposed to the Kalman filter.

$$
\begin{aligned}
\mathbf{y}(k|k) &= \mathbf{y}(k|k-1) + \mathbf{i}(k) && (2.35)\\
\mathbf{Y}(k|k) &= \mathbf{Y}(k|k-1) + \mathbf{I}(k) && (2.36)
\end{aligned}
$$

2.3 Simultaneous Localisation and Mapping (SLAM)

Simultaneous Localisation and Mapping (SLAM) is a popular technique in the literature [Durrant-Whyte and Bailey, 2006; Kim and Sukkarieh, 2003; Dissanayake et al., 2000b; Gutmann and Konolige, 1999] to address complex mapping problems under conditions of process and sensor noise and possible modeling errors. This algorithm first appeared in a seminal paper by Smith, Self and Cheeseman [Smith and Cheeseman, 1986] and it has received a considerable amount of attention by the robotics community [Castellanos et al., 2000; Feder et al., 1999; Menegatti et al., 2009; Lázaro and Castellanos, 2010]. Broadly speaking, SLAM is the process of concurrently building a map of the environment while using the map to estimate the location of the robot in that unknown environment. In essence, SLAM tries to estimate both the robot and map states with successive observations. By tracking the relative position between the robot and the detected *landmarks*[8] in the environment, both the position of the robot and the position of the landmarks can be estimated simultaneously.

A handful of techniques have been proposed for solving the SLAM problem. In [Thrun, 2002], Sebastian Thrun provides a comprehensive survey on existing robotic mapping strategies such as Maximum likelihood estimation, expectation maximisation, extended Kalman filter (EKF), extended information filter (EIF) and particle filtering. Notwithstanding the effectiveness of all the aforementioned techniques in various applications, algorithms based on Kalman filtering are widely used to probabilistically estimate the robot and map objects in the SLAM context and hence form the basis of most of the work in this thesis. In practice, the extended Kalman filter (addressed in Section 2.2) is often used as an ad-hoc approximation for non-linear systems and for fusing the information collected by the robot in a recursive fashion [Leonard and Durrant-Whyte, 1992]. In other words, EKF calculates Gaussian posterior estimates based on the location of the detected landmarks and the mobile robot.

We now provide the mathematical foundation for the single-vehicle SLAM algorithm employed in this work. This framework along with the applied notation has been mostly adopted from the influential paper by Dissanayake et al. [Dissanayake et al., 2001] and closely resembles that of the classical paper by Smith et al. [Smith et al., 1990]. In the context of our mapping application, the robot is a moving land vehicle and the landmarks are the stationary road signs with their position estimates to be determined. Please note that, as will be seen later in this thesis, beside position, some types of landmarks (e.g. road signs) have a secondary attribute namely orientation (or surface normal) that may be of interest in some mapping applications.

[8]Please note that the terms landmark and feature will be used interchangeably in this thesis.

However, without loss of generality, this chapter, as well as Chapter 3, only considers the unknown position of landmarks. The more general case of landmark's state consisting of position and orientation will be discussed in detail in Chapter 4.

2.3.1 State and Covariance Representation in SLAM

Consider a mobile vehicle traversing an environment containing a population of navigable landmarks with unknown time-invariant locations. The vehicle is equipped with a sensor (e.g. laser scanner, camera) or a bundle of sensors which are able to detect the environmental landmarks of interest and provide relative observations (measurements) of those landmarks with respect to the vehicle itself.

The state of the SLAM system $\mathbf{x}(k)$ comprises the state of the vehicle along with the state of all the detected landmarks. The state of the vehicle at time k is denoted $\mathbf{x}_v(k)$ and usually consists of the position and orientation of the moving platform. The state of the i^{th} landmark, normally consisting of an estimate of its position, is denoted $\mathbf{x}_{mi}(k)$. The joint subscript m indicates that the variable is akin to some landmark in order to differentiate between vehicle and landmark states[9]. The position vector of all the registered landmarks is denoted $\mathbf{x}_m(k)$, therefore

$$\mathbf{x}_m(k) = [\mathbf{x}_{m1}^T(k), \mathbf{x}_{m2}^T(k), \ldots, \mathbf{x}_{mN}^T(k)]^T \tag{2.37}$$

where N denotes the current number of landmarks stored in the SLAM filter.

Following the Bayesian notation used earlier in Section 2.2, the estimated state vector and its associated covariance matrix (in block form) for the SLAM system may be written as

$$\mathbf{x}(k|k) = \begin{bmatrix} \mathbf{x}_v(k|k) \\ \mathbf{x}_m(k|k) \end{bmatrix} \tag{2.38}$$

$$\mathbf{P}(k|k) = \begin{bmatrix} \mathbf{P}_{vv}(k|k) & \mathbf{P}_{vm}(k|k) \\ \mathbf{P}_{mv}(k|k) & \mathbf{P}_{mm}(k|k) \end{bmatrix} \tag{2.39}$$

with $\mathbf{P}_{vv}(k|k)$ and $\mathbf{P}_{mm}(k|k)$ representing the covariance of the vehicle states and registered landmark states respectively, while the terms $\mathbf{P}_{mv}(k|k)$ and $\mathbf{P}_{vm}(k|k)$ denoting the cross-covariance between the vehicle and landmark states.

In this section, we restrict our discussion to the general process and observation models for SLAM, without considering a particular type of system. A more detailed operation of the SLAM filter will be discussed in Chapter 3 when addressing the local SLAM filter in the multi-vehicle mapping system.

[9] The superscript has been reserved for future chapters to indicate the vehicle ID in multi-vehicle mapping systems.

2.3.2 General Vehicle and Landmark State Transition Models for SLAM

The general form of the state transition model for SLAM is mathematically expressed by the following difference equation

$$\mathbf{x}(k) = \mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k), \mathbf{w}(k)) \tag{2.40}$$

where similar to the notation used in 2.2.2, $\mathbf{f}$ is the nonlinear state transition function, $\mathbf{u}$ is the control input vector and $\mathbf{w}$ is the vector of process noise. Explicitly, Equation (2.40) can be partitioned into the vehicle and the map state dynamic model.

The motion model of the vehicle is represented using a non-linear discrete-time state transition equation as follows:

$$\mathbf{x}_v(k) = \mathbf{f}_v(\mathbf{x}_v(k-1), \mathbf{u}_v(k), \mathbf{w}_v(k)) \tag{2.41}$$

where $\mathbf{f}_v$ is the vehicle's state transition function, $\mathbf{u}_v$ is the control input vector and $\mathbf{w}_v$ is the vector of process noise errors pertaining to the vehicle.

Similarly, the general state transition equation for the registered landmarks is given by

$$\mathbf{x}_m(k) = \mathbf{f}_m(\mathbf{x}_m(k-1), \mathbf{w}_m(k)) \tag{2.42}$$

The dynamic model for the i^{th} landmark is trivial and is given by

$$\mathbf{x}_{mi}(k) = \mathbf{x}_{mi}(k-1) \tag{2.43}$$

simply because the landmark locations are assumed stationary in SLAM.

As a result, the general state transition model of Equation (2.40) can be written by combining the process model equations described in (2.41) and (2.42) as

$$\begin{bmatrix} \mathbf{x}_v(k) \\ \mathbf{x}_m(k) \end{bmatrix} = \begin{bmatrix} \mathbf{f}_v(\mathbf{x}_v(k-1), \mathbf{u}_v(k), \mathbf{w}_v(k)) \\ \mathbf{f}_m(\mathbf{x}_m(k-1), \mathbf{w}_m(k)) \end{bmatrix} \tag{2.44}$$

2.3.3 General Observation Model for SLAM

As mentioned before, it is assumed that the moving vehicle is equipped with a sensor (or set of sensors) that provide relative observations of the landmark with respect to the vehicle in terms of sensor measurements. The observation model provides the relationship between these measurements, the vehicle and landmark states. The non-linear observation model in discrete-time is given by

$$\mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k), \mathbf{v}(k)) \tag{2.45}$$

where $\mathbf{h}(.,.)$ is a general observation function, while $\mathbf{v}(k)$ is the vector associated with observation noise. Equation (2.45) is used in order to predict the sensor measurements for different landmarks. This equation can be simplified to reflect the fact that the predicted measurement vector for an arbitrary landmark i, is only a function of the vehicle state $\mathbf{x}_v(k)$ and the landmark state itself $\mathbf{x}_{mi}(k)$.

$$\mathbf{z}_i(k) = \mathbf{h}(\mathbf{x}_v(k), \mathbf{x}_{mi}(k)) + \mathbf{v}(k) \tag{2.46}$$

Notice that the general premise on observation noise $\mathbf{v}(k)$ has now been relaxed compared to the generic model of Equation (2.45), by assuming additive observation noise.

The next three subsections will address the operation of the SLAM algorithm. As stated before, the extended Kalman filter is used in this work as a state estimator for localisation and mapping using SLAM.

2.3.4 SLAM Prediction Step

Similar to the EKF equations of Section 2.2.2, the state vector and its covariance matrix are propagated using the nonlinear process model $\mathbf{f}$ and its corresponding Jacobian matrix according to

$$\begin{aligned} \mathbf{x}(k|k-1) &= \mathbf{f}(\mathbf{x}(k-1|k-1), \mathbf{u}(k), 0) && (2.47) \\ \mathbf{P}(k|k-1) &= \nabla\mathbf{f}_{\mathbf{x}}(k)\, \mathbf{P}(k-1|k-1)\, \nabla\mathbf{f}_{\mathbf{x}}^T(k) + \nabla\mathbf{f}_{\mathbf{w}}(k)\mathbf{Q}(k)\nabla\mathbf{f}_{\mathbf{w}}^T(k) && (2.48) \end{aligned}$$

2.3.5 SLAM Update Step

In the event of occurrence of an observation, if the observed landmark is already stored in the map and is successfully associated with one of the registered landmarks, the current state vector and covariance matrix are updated according to

$$\begin{aligned} \mathbf{x}(k|k) &= \mathbf{x}(k|k-1) + \mathbf{K}(k)\nu(k), && (2.49) \\ \mathbf{P}(k|k) &= [\mathbf{I} - \mathbf{K}(k)\nabla\mathbf{h}_{\mathbf{x}}(k)]\mathbf{P}(k|k-1)[\mathbf{I} - \mathbf{K}(k)\nabla\mathbf{h}_{\mathbf{x}}^T(k)]^T \\ &+ \mathbf{K}(k)\mathbf{R}(k)\mathbf{K}^{\mathbf{T}}(k). && (2.50) \end{aligned}$$

where the associated innovation vector, Kalman gain and innovation covariance are respectively determined using

$$\begin{aligned} \nu(k) &= \mathbf{z}(k) - \mathbf{h}(\mathbf{x}(k|k-1), 0) && (2.51) \\ \mathbf{K}(k) &= \mathbf{P}(k|k-1)\nabla\mathbf{h}_{\mathbf{x}}^T(k)\mathbf{S}^{-1}(k) && (2.52) \\ \mathbf{S}(k) &= \nabla\mathbf{h}_{\mathbf{x}}(k)\mathbf{P}(k|k-1)\nabla\mathbf{h}_{\mathbf{x}}^T(k) + \mathbf{R}(k) && (2.53) \end{aligned}$$

Please note that topics such as data association in SLAM are beyond the scope of this work and hence will not be mentioned in this chapter.

2.3.6 State and Covariance Augmentation in SLAM

In the event of occurrence of an observation, if the observed landmark is not matched with any of the already registered landmarks in the map, it must be added to the existing map through extension of the state vector and covariance matrix described by Equation (2.38) and (2.39).

Let $\mathbf{g}_i(.,.)$ be a nonlinear initialisation function that maps the current vehicle state estimate and the new observation to the new landmark estimate at time k according to

$$\mathbf{x}_{mi}(k) = \mathbf{g}_i(\mathbf{x}_v(k), \mathbf{z}(k)) \tag{2.54}$$

Then, the augmented state vector can be written as

$$\mathbf{x}_{\text{aug}}(k) = \begin{bmatrix} \mathbf{x}_v(k) \\ \mathbf{x}_m(k) \\ \mathbf{g}_i(\mathbf{x}_v(k), \mathbf{z}(k)) \end{bmatrix} \tag{2.55}$$

The augmented covariance matrix is also formed according to[10]

$$\mathbf{P}_{\text{aug}}(k) = \begin{bmatrix} \mathbf{P}_{vv}(k) & \mathbf{P}_{vm}(k) & (\nabla_v \mathbf{g}_i(k)\mathbf{P}_{vv}(k))^T \\ \mathbf{P}_{mv}(k) & \mathbf{P}_{mm}(k) & (\nabla_v \mathbf{g}_i(k)\mathbf{P}_{vm}(k))^T \\ \nabla_v \mathbf{g}_i(k)\mathbf{P}_{vv}(k) & \nabla_v \mathbf{g}_i(k)\mathbf{P}_{vm}(k) & \gamma(k) \end{bmatrix} \tag{2.56}$$

where the block covariance matrix $\gamma(k)$ is given by

$$\gamma(k) = \nabla_v \mathbf{g}_i(k)\mathbf{P}_{vv}\nabla_v \mathbf{g}_i(k)^T + \nabla_z \mathbf{g}_i(k)\mathbf{R}(k)\nabla_z \mathbf{g}_i(k)^T \tag{2.57}$$

2.4 Multi-vehicle Simultaneous Localisation and Mapping

Although most of the initial interest in SLAM considered the problem of mapping and localisation with a single vehicle, the first decade of the twenty-first century saw a substantial interest in multi-vehicle localisation and mapping. There are numerous practical robotic applications in which a fleet of distributed sensor platforms are utilised in order to gather information about an environment and fuse this information into a consistent map. The advantages of using multiple, cooperative, vehicles in exploration and mapping applications, compared to the single vehicle case have been proven and are well known in the literature [Cao et al., 1997; Fox et al., 2000; Burgard et al., 2000] and are briefly mentioned here.

Deployment of a single sensing platform is often not sufficient for data collection and map building in large-scale environments. Performing complex tasks is also proven to be difficult using only one robotic agent. Also, in general, multiple agents

[10] See [Kim, 2004; Williams, 2001] for more details on derivation of the augmented covariance matrix.

have the potential to carry out a mission quicker that a single agent[11]. In addition, the utilisation of one very expensive capture platform reduces the fault-tolerance of the system due to the reliance on one individual entity, whereas the redundancy created by more than one vehicle mollifies the system failure problem. Most importantly, in many exploration and mapping applications, using several vehicles provides improved accuracy and other performance benefits due to the observation of map objects by more than one vehicle. This alleviates the sensors′ uncertainty and localisation errors, particularly in situations where the robots have different sensing and localisation capabilities [Fox et al., 2000]. All these aspects are of paramount conceptual significance and will be seen in the mapping application addressed in this work.

2.4.1 Existing Work

In the area of navigation and mapping by multiple vehicles, the work of Sebastian Thrun [Thrun, 2001; Fox et al., 2000; Burgard et al., 2000], Hugh Durrant-Whyte [Williams et al., 2002; Durrant-Whyte and Henderson, 2006] and their respective research groups are specifically notable. Different methods and algorithms were proposed and analysed by the researchers in these groups for effective and accurate mapping and navigation using multiple robots. For example, Makarenko & Durrant-whyte [Makarenko and Durrant-Whyte, 2004] provide an algorithm for Bayesian data fusion for multiple vehicles. Rosencrantz et.al. [Rosencrantz et al., 2003] present a scalable Bayesian technique for decentralised state estimation for multiple platforms in dynamic environments. Fox et al. [Fox et al., 2000] use a sample-based version of Markov localisation [Cassandra et al., 1996] which is capable of localising mobile robots at any time.

Several papers such as [Simmons et al., 2000; Singh and Fujimura, 1993] investigate the techniques for coordinating multiple robots in their task of exploring and mapping large environments. Burgard et al. [Burgard et al., 2000] present a method for efficiently coordinating a team of robots for achieving their tasks in exploring the environment. The key idea in such works is that the cost of reaching an unexplored location and its utility is simultaneously taken into account while planning the paths for different robots. Singh and Fujimura [Singh and Fujimura, 1993] designed an exploration strategy to guide the multiple mobile robots with different motion and sensing capabilities to explore an unknown bounded region and obtain a global map while avoiding the obstacles. Within such applications the robots are usually provided with sufficient computational power and memory to process all the collected measurements and store a map of the complete region. Each mobile robot can also communicate with all other robots with negligible delay. Although there are fundamental differences between these applications and the distributed mapping system in this work, certain concepts from coordinated exploration and path planning robotic applications have been borrowed by the work presented in this thesis. The main dif-

[11] However, this may not always be the case due to the interference between robots [Schneider-Fontan and Mataric, 1998].

ference between the pair is that the mobile vehicles that are tasked with mapping the environment in the present work cannot be controlled and coordinated to explore unknown locations or locations where map items are of poor quality. We are simply interested in building a consistent and accurate picture of the environment using collaborative survey vehicles whose future paths and behaviour are uncontrollable and undetermined to a great extent.

Different works [Williams et al., 2002; Fenwick et al., 2002; Fenwick, 2001] have attempted to extend the single-vehicle simultaneous localisation and mapping (SLAM) to multiple vehicles. The primary sophistication in directly extending the single-vehicle mapping systems to multiple vehicles arises from the need to incorporate the cross-correlation between different landmark estimates stored on each vehicle [Julier and Uhlmann, 2001b]. For example, Fenwick et al. [Fenwick et al., 2002] combine all of the collaborating vehicle state estimates into a single state vector in a rigorous way. In a similar way, all the position estimates of the observed landmarks from all the vehicles are combined together. The authors then define a single estimate that incorporates all of the vehicle and landmark estimates. The general collaborative covariance matrix is also constructed by combining all the covariances and cross-covariances. Once collaborating vehicles are added into the defined state and covariance matrices, the multi-vehicle prediction and update equations take on the same general form as the single vehicle estimation algorithms. Despite the straightforward nature of such solutions, they are highly unsuitable for deployment in extremely large environments with a large number of landmarks, as will be discussed subsequently in this chapter.

A variety of stochastic estimation techniques have been reported in different works to estimate and maintain the bounded location of robots and the map landmarks. The nonlinear SLAM algorithm has been extensively implemented in the past using the extended Kalman filter (addressed in Section 2.2) [Kim and Sukkarieh, 2003; Williams et al., 2000; Leonard and Durrant-Whyte, 1991]). EKF-based SLAM approaches calculate a fully correlated posterior estimate about robot pose and landmark maps. However, these algorithms rely on the strong assumption of Gaussian distribution for both robot motion and sensor noise. While the EKF solution to the SLAM problem has received considerable interest, alternative approaches also appear in the literature. Lu & Milios [Lu and Milios, 1997], Thrun et al. [Thrun et al., 1998] and Gutmann & Konolige [Gutmann and Konolige, 1999] use batch estimation techniques to tackle the simultaneous map building and localisation problem for mobile robots. In these works, the data gathered by the robot is stored and processed in a batch manner to build the maps of the environment in which the robot has operated[12]. A relatively more recent batch processing method called GraphSLAM was introduced by Thrun and Montemerlo in [Thrun and Montemerlo, 2006]. GraphSLAM operates by using graphical networks and applying optimisation techniques to the offline SLAM problem. It is claimed that the algorithm is capable of generating

[12]Such problems are sometimes referred to as *offline* SLAM problems. These problems require memorising all data and postponing the mapping process until after the robot's operation is complete [Thrun and Montemerlo, 2006]

maps with around 10^8 features in approximately 30 seconds. Despite its spectacular performance, the GraphSLAM algorithm is not feasible for many practical application due to its offline nature; i.e. it requires the accumulation of all data during mapping, and consolidating this data into a map after the robot's exploration process is complete.

Other approaches to the SLAM problem have tried to eliminate the need for rigorous mathematical models of the vehicle and sensing properties and have relied instead on more qualitative knowledge of the nature of the environment [Brooks, 1986; Levitt and Lawton, 1990]. For instance, Kuipers & Byun [Kuipers and Byun, 1991] have proposed a robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. Sequential Monte Carlo methods, widely known as Particle filters, have also been used for both single-robot and multi-robot global localisation [Grisetti et al., 2007; Doucet et al., 2001]. Several works [Ong et al.; Nettleton et al., 2006; Thrun et al., 2002] employ the information form of the Kalman filter as an effective approach to the SLAM problem. Lie & Thrun [Eustice et al., 2005; Thrun et al., 2004; Liu and Thrun, 2003] present some results for outdoor SLAM using sparse extended information filters. These techniques and other alternative approaches to the SLAM problem have their own particular strengths and weaknesses. As mentioned previously, the work in this thesis will mainly rely on the EKF as the primary tool for simultaneous mapping and localisation problem. However, the interesting properties of the information form of the EKF will also be exploited in the multi-vehicle mapping paradigm in Chapter 3.

At a high-level, information fusion is the fundamental tool required for multi-vehicle SLAM as, on an abstract level, the problem is about combining numerous sources of information (that may be correlated) about a common parameter in order to increase one's knowledge about the parameter.

In multi-vehicle SLAM the problem of where this fusion occurs and how information is shared is a practical problem and is one that motivates much of the work in Chapter 3 (along with similar work discussed subsequently). Different multi-vehicle data fusion architectures have been suggested and implemented for tasks such as autonomous navigation [Bryson and Sukkarieh, 2005; Kim, 2004; Williams, 2001], exploration and mapping [Nerurkar et al., 2009; Stipes et al., 2006; Thrun, 2001] and target tracking [Liggins et al., 1997; Nettleton, 2003; Ong et al., 2003]. Section 2.5 will address three main architectures used by different researchers for multi-vehicle/multi-sensor data fusion.

2.5 Multi-vehicle Data Fusion Architectures

The aim of this section is to discuss different multi-vehicle data fusion architectures and to investigate their respective properties in data fusion applications.

Within the multi-sensor data fusion literature, three main architectures can be seen for combining information from distributed sources. These architectures are known as centralised, decentralised and hierarchical.

Please note that non-identical definitions and categorisations have been provided in the literature by the information fusion and robotics community for various data fusion architectures. These definitions are sometimes conflicting and not consistent with each other. Most notably, the terms hierarchical and distributed are subject to debate in the literature. For instance [Cao et al., 1997] presents the hierarchical architecture as a sub-category of decentralised systems, while [Durrant-Whyte and Henderson, 2006] considers a central processor in the hierarchical architecture and categorises it in the centralised group. To avoid any confusion and in order to be consistent, this thesis will present the clear definition of each architecture, as used throughout this text. We provide a brief description of each architecture along with their respective advantages and disadvantages as a useful background for making future design decisions in this work.

2.5.1 Fully Centralised Architecture

The most obvious and traditional data fusion architecture is a fully-centralised one where all the measurements made by sensors from multiple sources are directly transmitted to a central station in a raw format where they are processed by a single algorithm (e.g. using a large Kalman filter), almost in the same way as single sensor systems. Works such as [Fenwick et al., 2002], [Mourikis and Roumeliotis, 2004], [Walter and Leonard, 2004] and [Thrun and Liu, 2005] provide fully-centralised approaches to the multi-vehicle SLAM problem.

In this architecture, little or no local processing of information is performed and the central server has complete centralised control over the interpretation and integration of information. The primary disadvantage of a centralised solution is the communication and networking complexity required. As more sensors and information sources are incorporated, the functional requirements and the substantial complexity of the data fusion system grow. Due to the nature of this architecture, a severe computational burden is usually imposed on the fusion centre. The resulting sophistication and heavy computational load at the central station might lead to an undesirable computational bottleneck. Since the central station offers a single point of failure, centralised solutions in general are less redundant and robust. Moreover, due to the communication of all the sensor measurements back to the central processor, this setup is not efficient in terms of data transmission and communication bandwidth. However, centralised solutions are convenient in numerous practical applications where it is undesirable for the vehicles to communicate between themselves. A block diagram for a fully centralised system is depicted in Figure 2.2.

2.5.2 Decentralised Architecture

In contrast to the centralised systems, fully decentralised architectures often have no central processing station and no common communication system. In such systems, individual stations (e.g. individual vehicles) can perform data fusion in a fully autonomous manner, while receiving information from and transmitting information

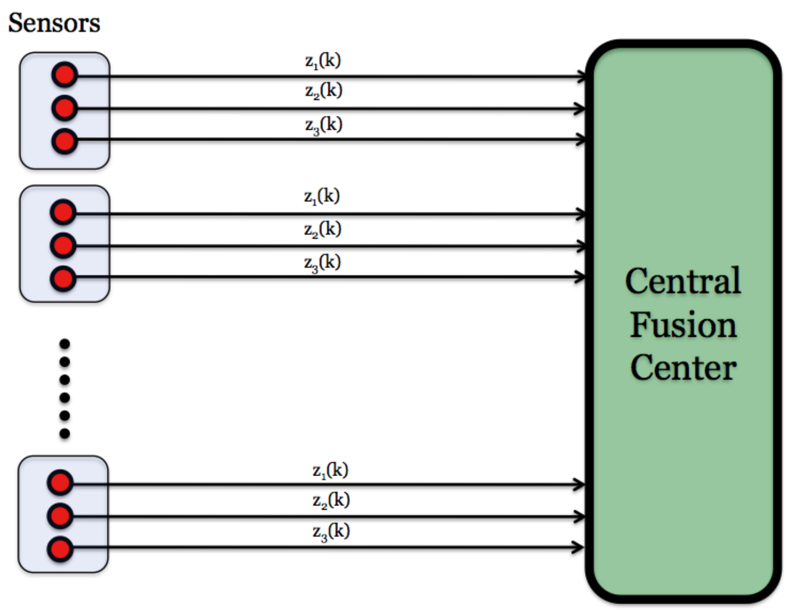


Figure 2.2: A fully centralised data fusion architecture

to other particular stations. In other words, fusion occurs locally at each station on the basis of local observations and the information received from neighbouring stations. Note that if the networking topology resembles a complete graph then such decentralised systems offer no advantage in terms of communication requirements. Of course, in general, decentralised solutions are more robust to failure of a given station. A block diagram for the decentralised architecture is illustrated is Figure 2.3.

[Durrant-Whyte and Henderson, 2006] characterises a decentralised fusion centre by three important constraints:

1. There is no single central fusion centre; no one node should be central to the successful operation of the network.

2. There is no common communication facility; nodes cannot broadcast results and communication must be kept on a strictly node-to-node basis.

3. Sensor nodes do not have any global knowledge of sensor network topology; nodes should only know about connections in their own neighbourhood

These constraints provide a number of important characteristics such as scalability, survivability and modularity which give decentralised systems a major advantage over conventional sensing architectures, particularly in defence and military applications. Examples of decentralised SLAM can be found in [Nettleton et al., 2006; Sharon et al., 2003; Ong et al., 2003].

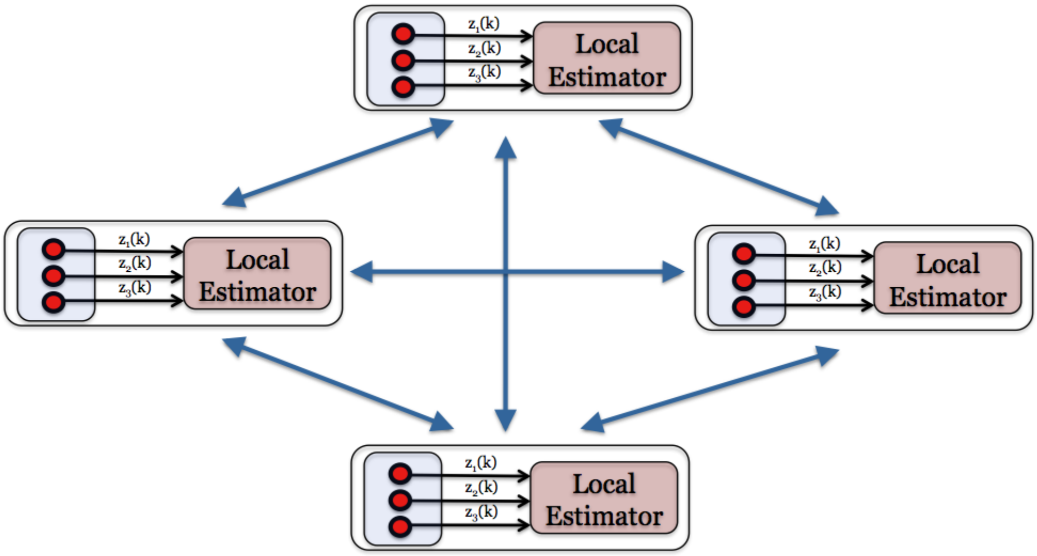


Figure 2.3: A fully decentralised data fusion architecture

2.5.3 Hierarchical Architecture

In essence, the hierarchical architecture aims at increasing the 'intelligence' of local nodes by allowing different levels of information processing. Compared to fully centralised architectures, this means moving away some of the sophisticated processing tasks at the central server at the cost of losing complete control over the low-level sensor information. Since more processing occurs locally, the overwhelming computational and communication burden can be removed from the fusion centre. Also, because the sensors are granted a virtual intelligence in such systems, they can be constructed in a modular manner. The degree to which local processing takes place at a sensor node varies from simple validation and data compression up to the full construction of tracks, running complex estimation algorithms and full interpretation of information locally. The disadvantage of this architecture is placing a specific and often rigid structure on the fusion system [Durrant-Whyte and Henderson, 2006]. Examples of hierarchical systems can be found in [Hashemipour et al., 1988; Blackman and Popoli, 1999; Dai and Du, 2009]. A single-level hierarchical system is depicted in Figure 2.4.

2.6 Major Challenges in Building Scalable Maps Using SLAM

Building and maintaining maps in extremely large environments has always been one of many practical challenges in the SLAM community. One can think of a handful of real-world applications where a number of mobile robots are tasked with gathering information about an extensive environment filled with a large population of landmarks. This information needs to be interpreted and presented in terms of a

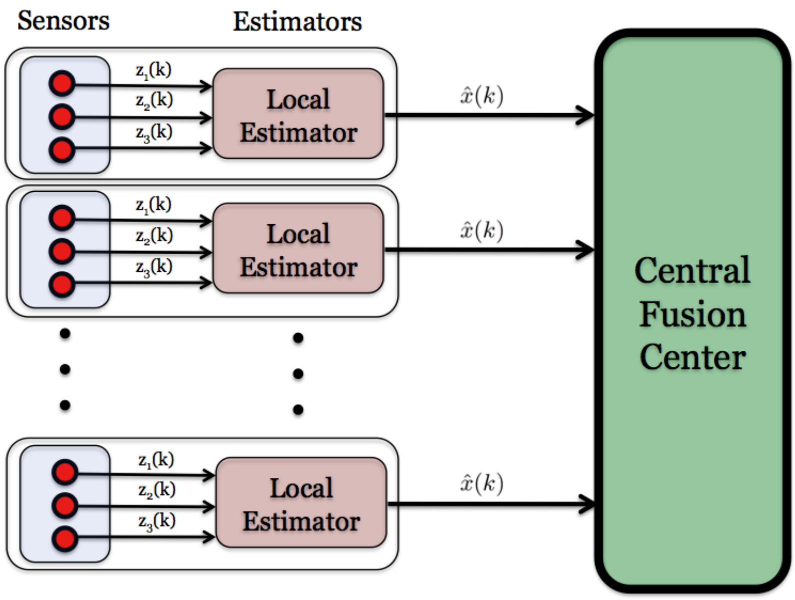


Figure 2.4: A single-level hierarchical data fusion architecture

consistent global picture of the environment. The mapping problem addressed in this thesis is of such practical nature.

Practical challenges associated with mapping large-scale environments is multi-faceted and generally arise due to the fact that the complexity of the SLAM system increases as new landmarks are explored. Typically, the main ramifications are in terms of storage, computational complexity and memory requirements, all caused by the large number of map elements and the well-known cross-correlation problem described below.

In general, a fundamental part of the SLAM algorithm is the knowledge of correlations between the map and vehicle estimates. In fact, it is shown that (e.g. in [Dissanayake et al., 2001]) the cross-correlations in the covariance matrix which maintain the relationships between the vehicle's state estimates and landmark estimates in SLAM is the reason for the long-term convergence of the filter. Therefore, the propagation of the covariance matrix of the full map is vitally important to the solution of the SLAM problem. The full-covariance SLAM (FC-SLAM) operates by maintaining all the existing cross-correlations presented in the covariance matrix of the estimates. Recalling Section 2.3, the covariance matrix for a total number of n landmarks can be written by expanding the block components of Equation (2.39) according to

$$\mathbf{P}(k|k) = \begin{bmatrix} \mathbf{P}_{vv}(k|k) & \mathbf{P}_{v1}(k|k) & \mathbf{P}_{v2}(k|k) & \dots & \mathbf{P}_{vn}(k|k) \\ \mathbf{P}_{1v}(k|k) & \mathbf{P}_{11}(k|k) & \mathbf{P}_{12}(k|k) & \dots & \mathbf{P}_{1n}(k|k) \\ \mathbf{P}_{2v}(k|k) & \mathbf{P}_{21}(k|k) & \mathbf{P}_{22}(k|k) & \dots & \mathbf{P}_{2n}(k|k) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{nv}(k|k) & \mathbf{P}_{n1}(k|k) & \mathbf{P}_{n2}(k|k) & \dots & \mathbf{P}_{nn}(k|k) \end{bmatrix} \tag{2.58}$$

where similar to the notation used before, the term $\mathbf{P}_{vv}(k|k)$ is the block covariance pertaining to the vehicle's state estimate, $\mathbf{P}_{ii}(k|k)$ is the covariance of the i^{th} landmark's state estimate and $\mathbf{P}_{ij}(k|k)$ is the cross-correlation between the estimates of two arbitrary landmarks i and j.

For a map consisting of $\mathcal{O}(n)$ landmarks, the storage requirements is proportional to $\mathcal{O}(n^2)$ and the computational cost is proportional to $\mathcal{O}(n^3)$. This is because the entire map of n landmarks needs to be updated at each step by the filter. This means the map is updated not only after the observation of a single landmark, but also following the incorporation of any input measurement from navigation sensors. These sensors (e.g. accelerometers and gyroscopes) usually operate with a high update rate, sometimes reaching a few hundred samples per second. Therefore, maintaining all the cross-correlation terms in FC-SLAM makes the algorithm computationally intractable in environments with a large number of landmarks. As a result, the propagation of the covariance matrix stated by (2.58) is only practically feasible for small maps when n is of the order of a few hundred landmarks. The limited computational power and communication bandwidth in most applications does not allow the tracking and recording of all the correlations in the multi-vehicle system.

To tackle the above-mentioned problem, several papers [Guivant and Nebot, 2003, 2002; Williams, 2001; Dissanayake et al., 2000a] have proposed different theoretical and practical solutions to ameliorate the computational inefficiency in SLAM. Basically, these papers aim at reducing the computational cost of the filter caused by the full update of the state and covariance matrices by considering sub-optimal solutions or using other map management techniques. These approaches only address the single-vehicle SLAM problem. The naive extension of these solutions to large-scale multi-vehicle SLAM scenarios where remote vehicles share landmark information (either directly or through a central station) is ineffective in most cases due to the significance of overlapping landmarks and their correlations in such applications. Despite some effort, no fully versatile solution for these practical, real-world problems has been offered to the best of our knowledge.

Another related issue in large-scale mapping applications is the problem of mathematical consistency that needs to be addressed. This problem is particularly important in data fusion applications where overlapping maps are shared between several distributed vehicles. Within most filtering frameworks, it is not possible to integrate information from multiple sources unless they are independent or have known cross-covariance [Chong et al., 1990]. Although, the cross-correlations between the map objects play a crucial role in the multi-vehicle SLAM, they may cause inconsistency problems in data fusion networks. The effect of redundant information and information double-counting is a serious issue that must be analysed and handled with care. Common sources of information double-counting and system inconsistencies will be discussed in more details in the next chapter. Moreover, Appendix A provides a general analysis on consistency when fusing two generic estimates.

So far we have outlined some of the most arduous challenges associated with mapping extensive environments. As described, traditional map making solutions are usually not deployable in their classic forms in these situations where the robots are required to operate robustly over long periods of time. These solutions are typically prone to failure if implemented in practice. As a result, a versatile, scalable solution is sought which is able to cope with the large-scale virtue of these problems and face the multitude of challenges imposed on the system. The utilised system should be able to accommodate algorithms capable of handling extremely large environments. The solution must be scalable with respect to criteria such as the number of landmarks, the number of robots exploring the environment and the physical size of the map[13].

2.7 Project Description

As mentioned in the Introduction Chapter, the work in this thesis is inspired by a real-world project called AutoMap where assets visible from the road scene are automatically extracted from recorded video and geo-located to form a map. One of the solutions offered by the AutoMap project is a passive data collection scheme using a set of low-cost in-vehicle sensor platforms. In this solution, geographically located information from the road scene is gathered continuously (over long periods of time) on a very large scale by a fleet of distributed vehicles such as taxis, garbage trucks, delivery vans etc. (see Figure 2.5) and sent back to a central server where a global database is compiled. Advanced computer vision algorithms [Overett et al., 2009] are deployed to automatically extract and geolocate objects such as road signs from recorded video that are of interest to third party companies like mapping companies and road asset managers [Petersson, 2014]. Such information is currently collected in a manual fashion and updated only every few years which is a very costly and error prone process. A setup as described in this work enables a continuously updated database of road scene information at a fraction of the cost compared to the manual alternative. Each fleet vehicle in this setup is equipped with a low-cost sensor platform consisting of three cameras, a Global Positioning System (GPS), a 3-axis accelerometer, a 3-axis gyroscope, a 3-axis magnetometer, a processing unit and a 3G modem (See Figure 2.6). The low-grade accelerometer and gyroscope are embedded in a six-degree of freedom MEMS-based[14] inertial measurement unit (IMU). The utilisation of low-cost sensors enables the development of data collection sensor platforms at a reasonable cost. Data from the sensors are continuously stored on a local hard drive inside the vehicle, and later analysed by the local processing unit in order of importance to maximise a cost function representing the value of extracted information. The in-vehicle sensor platforms are able to send and receive information to and from a central server using a 3G connection. In addition to tasks such as fusing partial information from the vehicles, the central server also acts as the main map repository which accommodates the global map. The global database maintains information such as type and geolocation of all landmarks detected via the exploring vehicles.

[13] See [Julier and Uhlmann, 2001b] for a discussion on these criteria
[14] MEMS stands for Micro-Electrical-Mechanical-Systems

Figure 2.5: Passive data collection using third party vehicles.

The first objective of this work is to develop a distributed data collection model, which is able to effectively incorporate the collected measurements from different vehicles to gradually build a map of the environment. We are interested in producing a map of landmarks on the scale of thousands or even millions of kilometres of road network. This will be the main focus of the current and next chapter. As its second main objective, this thesis addresses an important aspect of practical mapping applications, namely quality assessment. In particular, we seek a well-defined concept for map quality which is able to reflect the accuracy of mapping systems in a meaningful way. This is mainly motivated by the lack of a widely accepted scientific methodology for comparing the results of existing mapping techniques. In addition, the recent technological advancements and the emergence of specialised mapping systems (such as AutoMap) call for application-driven performance metrics that are able to capture the particulars of such systems and can be applied in different practical scenarios. Chapters 4 and 5 are concerned with quality assessment and criteria-based estimation in mapping applications.

Figure 2.6: Low-cost sensors used inside the in-vehicle data collection platforms.

We now turn our focus to the design of the previously discussed distributed mapping system using multiple vehicles. Like many other real-world applications, this project suffers from variety of limitations and is constrained by the available resources. Section 2.7.1 presents the main resource constraints imposed on the project which are needed to be carefully considered in the design and implementation of the large-scale distributed mapping system.

2.7.1 Resource Constraints in the AutoMap Project

One of the key restrictions is the low-cost nature of the sensors utilised inside the in-vehicle data collection units. It is well known (c.f. [Aggarwal et al., 2006; Godha, 2006]) that low-cost sensors are usually associated with large measurement noise and other inaccuracies due to their intrinsic natures. Most notably, the notorious error characteristics of low-grade inertial sensors are troublesome in navigation and mapping applications where the continuous positioning information of a moving platform is required. For example, temperature dependency and highly nonlinear characteristics of low-cost inertial sensors can potentially cause drift and misalignment errors during navigation. As a result, if not dealt with properly, the solution suffers from unbounded error growth with time, leading to degraded navigation performance in the long term. Consequently, determining the associated errors (such as noises, biases, drifts and scale factor instabilities) becomes indispensable in the utilisation of these sensors in real-world navigation applications.

To address this problem, Appendix B provides the theoretical and experimental development of a calibration scheme to overcome the intrinsic limitations of a low-cost inertial measurement unit. The two-stage calibration algorithm was developed and tested successfully on the prototype MEMS IMU (similar to the ones deployed inside the in-vehicle sensor platforms) to determine the deterministic and stochastic errors of the sensor. This work makes use of artificial observations known as pseudo-velocity measurements resulting from a specific scheme of rotation to calibrate the IMU in the laboratory environment. The proposed structure is then modified and utilised as a basis for the IMU's error estimation in outdoor navigation applications.

Besides the low-cost nature of the deployed sensors, there are other major limitations associated with the system. Analysing the vast amount of information gathered from the sensors and transmitting it back to the central server is a challenging task as the platform installed in each vehicle suffers from a number of constraints. These constraints can be categorised as

1. Communication bandwidth and associated cost

2. Processing power

3. Memory and storage

One of the key constraints this thesis sets out to address is the limited communication bandwidth provided by the 3G modem. The limited communication bandwidth not only makes it impossible to transmit all raw sensory data to a central server and analyse it there, but even the amount of extracted, symbolic information poses a challenge (see Example 1 below). Clearly, a communication architecture that allows selective communication is needed to handle this case.

In addition, the available processing power, memory and storage inside the vehicles are limited mainly due to the resources allocated for computationally expensive computer vision algorithms. This is particularly important to consider given the very large scale virtue of this mapping problem. Hence, main challenges explained in Section 2.6 are mostly applied to the current project. Consequently, traditional multi-vehicle map making solutions are not applicable given the rapidly increasing map size and the limited available resources.

Example 1. *Consider a scenario with n vehicles collecting measurements and tasked at mapping a given environment. Each vehicle traverses d kilometres per day and each kilometre contains m map objects (road signs) on average. The size of each vehicle's map is given by*

$$N = m \cdot d \tag{2.59}$$

and it is assumed this map size is initialised at the start of the day. A map represented by a covariance matrix is then assumed to require $b \cdot N^2$ bytes to transmit and the communication cost for each byte is given by c. Without any loss of generality, we presume that each of these **n** *vehicles are tasked with improving a previously existing map in a central communication node, hence no new landmark is being observed/transmitted in this scenario. The communication protocol requires k transmissions of $b \cdot N^2$ bytes per kilometre of road data. A simple calculation shows that the total communication cost using this method is*

$$C_{total} = d^3 m^2 nkbc \tag{2.60}$$

per day. The communication cost (and bandwidth) in this example is proportional to the cube of the distance driven by each vehicle over a fixed period of time. Consequently, the above solution is not feasible for very-large-scale applications like AutoMap which exhibit limited system and communication resources. For example, let $n = 10$, $d = 200$, $m = 10$, $k = 0.1$, $b = 8$, $c = \$3 \times 10^{-8}$ (\$30 for 1GB of 3G data[15]) and $N = md = 10 \times 200$, then $C_{total} = \$192$ per day. In this case, the cost of complete communication is prohibitive and a more efficient solution is required. Similar analysis can be done for the processing power and memory requirements.

This example will be revisited in Chapter 6.

2.8 The Proposed Distributed Data Fusion Model

The nature of the project described above, demands a versatile solution which is capable of coping with extremely large-scale environments and overcome challenges and limitations addressed in Sections 2.6 and 2.7.

Although fully-centralised and conventional decentralised architectures have been proven to be effective in numerous mapping applications, without additional local processing it turns out that both methods fail to provide a practical and flexible solution to large-scale (millions of mapping points) mapping where limited bandwidth

[15] This is the average cost in Australia as of October 2013.

and processing power is a real concern. This is particularly true when the constraint of a centralised architecture is dictated by the problem.

This section introduces a single-level hierarchical architecture with a central base station, called the central fusion centre (CFC), to combine the local maps obtained from individual vehicles into a global map. The strategical advantages of having this central station as a communication hub can be justified in the context of the mapping application and based on the system constraints elaborated in 2.7.1. Since the primary objective is to retain an accurate map of an unknown environment, the central server is essential in maintaining the very large-scale map repository that can be easily accessed at any time by any vehicle. In addition, in terms of robustness, the safety of information at the server can be guaranteed, whereas, the components of a survey vehicle on a mission are generally prone to failure and information loss. As argued in Section 2.7, by virtue of the very-large-scale nature of the problem, it is practically unrealistic to process and maintain all the map data locally at individual vehicles.

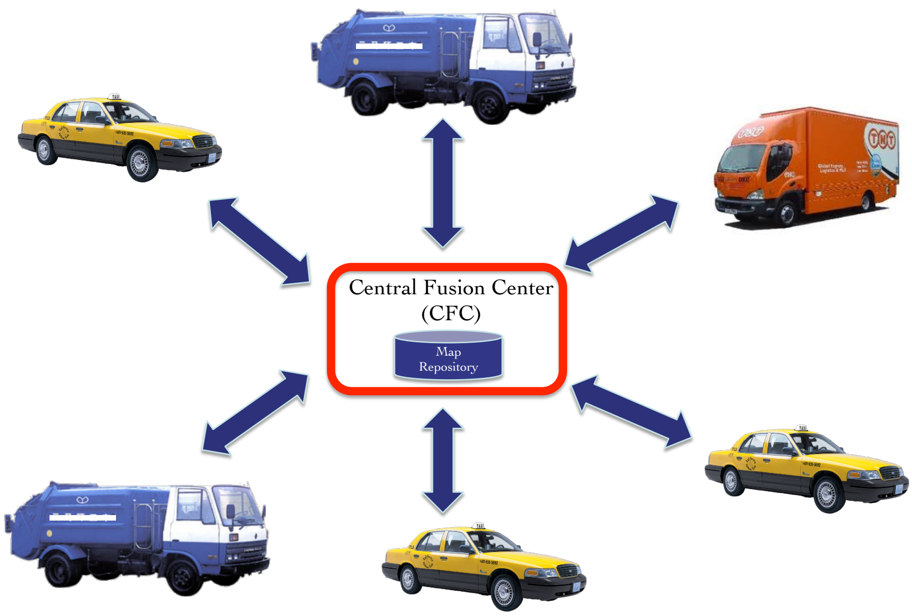


Figure 2.7: The distributed information fusion model keeps the central map repository up-to-date.

The hierarchical architecture aims to increase the processing done locally by the individual vehicles. A global map is maintained at the CFC and the individual vehicles construct local maps via SLAM (along with the fusion of local sub-maps from the CFC). These local maps are transmitted between the vehicles and the CFC over a cellular network; see Figure 2.7. A local SLAM algorithm is implemented in each vehicle in order to retain a local map of the detected landmarks and concurrently estimate the location of the vehicle as it explores the environment. Each vehicle shares a selection of its local information with the CFC (via a cellular network). The

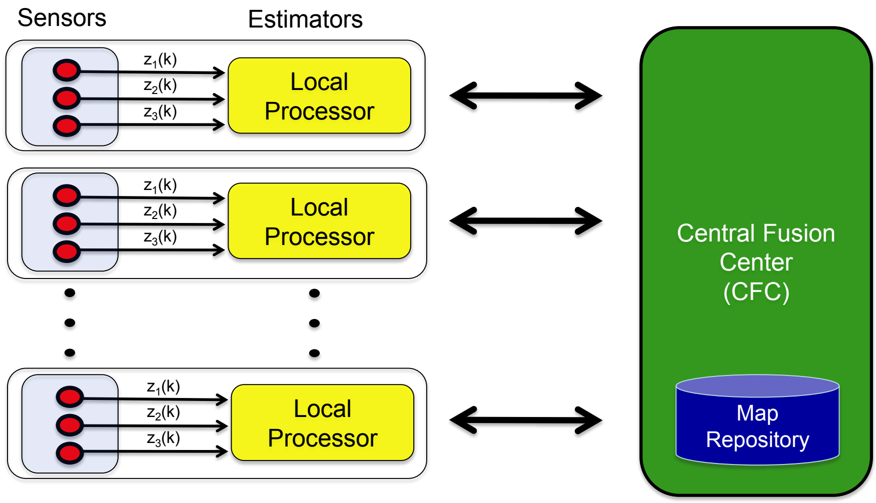


Figure 2.8: A single-level hierarchical architecture

CFC is responsible for maintaining a global map and for integrating the information collected by the vehicles in a consistent fashion (see Figure 2.8). A feedback configuration in the system provides a route for the communication of sub-maps of the global map back to the local filters in individual vehicles. As such, individual vehicles indirectly have access to the information obtained by other vehicles in the system. In addition, the feedback can potentially improve the data collection process with expected advantages of earlier detection, enhanced tracking, and more reliable identification [Xiong and Svensson, 2002].

Despite some fundamental work (e.g. [Nettleton et al., 2006; Nettleton, 2003]), the problem of selective communication has been widely neglected in the study of multi-vehicle information fusion (e.g. [Bryson and Sukkarieh, 2007]). In large-scale, low-bandwidth mapping applications, sending all the local information to the central station is not feasible due to the limited system and communication resources present in practice. Information tailoring is necessary to avoid high communication costs and other bandwidth constraints in a distributed data collection system. Consequently, only the most valuable information should be selected and transmitted. This is the avenue that we follow in Chapter 3.

In addition, the majority of the existing multi-vehicle SLAM techniques suffer from the growing size of the local maps within individual nodes. Due to the large number of features and the rapidly increasing map size, the SLAM algorithm fails to fulfil the requirements of large-scale applications. The ramification is an immense memory and computational load on the vehicles. Consequently, appropriate strategies must be applied to limit the size of the SLAM filters in very large-scale environments. We discuss a particular pruning strategy in Chapter 3.

2.9 Summary

This chapter conducted a review of some of the most widely used stochastic estimation techniques. The simultaneous localisation and mapping (SLAM) algorithm was presented subsequently as one of the key solutions to the problem of map making under conditions of process and sensor noise and other modelling uncertainties. A summary of the principal literature describing different methodologies to the multi-vehicle SLAM problem was presented, to provide sufficient information to set the context of the proposed research. Three main multi-vehicle data fusion architectures were reviewed and their most important properties were noted. Practical challenges associated with building and maintaining maps in extensive environments were discussed and the shortcomings of the existing strategies in mapping such environments were addressed.

A description of the real-world practical project under study and the main constraints and limitations associated with it was provided. Low-cost sensors and other resource constraints such as limited communication bandwidth, processing power, memory and storage were explained in this chapter. There is an absolute necessity to carefully address these practical restrictions in the design and formulation of the large-scale distributed mapping system. Based on the problem formulation provided in this chapter, a single-level hierarchical architecture with a central base station was devised in Section 2.8. The proposed framework was only addressed from a high-level point of view in this chapter. Details pertaining to the operation of the distributed map building framework such as sensor fusion, efficient communication, consistent map data fusion and local map management will be addressed in Chapter 3.

Chapter 3

Efficient Map Building in Very-Large-Scale Environments

3.1 Introduction

The goal of this chapter is to provide a flexible, intelligent solution to the problem of building and maintaining a very-large-scale map using multiple vehicles. In particular, we aim at producing a map of landmarks on the scale of thousands of kilometres in an outdoor environment. We consider the distributed data collection model with hierarchical architecture proposed in the previous chapter for the real-world inspired road mapping application described earlier.

The devised algorithm is distributed across multiple vehicles each given the task of producing and updating a local map. The vehicles are equipped with a range of sensors and selectively communicate maps to and from a central station in a bandwidth-constraint environment. The potentially overlapping local maps are asynchronously transmitted back to a central fusion centre where a global map repository is maintained. As its main contribution, this chapter addresses two of the most common issues of mapping in large-scale environments, namely, computational complexity and limited communication bandwidth.

The content of this chapter can be split into two main parts. The first part (Section 3.2) presents an overview of different components of the multi-vehicle mapping system. The local SLAM filter which is the local implementation of the single-vehicle SLAM algorithm is outlined first. The state-space equations based on the extended Kalman filter are provided for this filter. The information-based representation for states and covariances of a given estimate is addressed next, as it forms the basis for communication and fusion of a group of landmarks in this chapter. The channel filter, a structure responsible for maintaining the common information between the distributed vehicles and the server, is introduced along with a discussion on its operation. Moreover, a short overview of the system's medial component known as the central fusion centre is presented. The second part of this chapter (Section 3.3), considers the communication sequence between a single vehicle and the central fusion centre. Six different sequence of steps containing transmission, fusion and update of map information in different system components is discussed. We propose a communication architecture which is scalable and is capable of dealing with time-varying overlapping map sizes. A general data fusion framework based on the popular covariance intersection algorithm is devised to tackle the problem of redundant information propagation that is caused by communicating sub-maps of arbitrary size in the network. The solution is efficient in terms of computational complexity, memory requirements and communication bandwidth.

3.2 Distributed System Overview

This section provides a detailed description of different system components utilised in the multi-vehicle data fusion architecture outlined in Chapter 2. The local SLAM filter (LSF), the channel filter (CHF) and the central fusion center (CFC) are addressed in this section.

3.2.1 The Local SLAM Filter (LSF)

The local SLAM filter is a local implementation of the single-vehicle SLAM algorithm. In this work the LSF is executed in the standard state space. In summary, the LSF estimates a state vector and a covariance matrix based on the observed sensor measurements and the information received from the central server (e.g. as an initial prior). The state and covariance at the LSF in vehicle i is given by

$$\hat{\mathbf{x}}^i(k|k) = \begin{bmatrix} \hat{\mathbf{x}}^i_v(k|k) \\ \hat{\mathbf{x}}^i_m(k|k) \end{bmatrix} \tag{3.1}$$

$$\mathbf{P}^i(k|k) = \begin{bmatrix} \mathbf{P}^i_{vv}(k|k) & \mathbf{P}^i_{vm}(k|k) \\ {\mathbf{P}^i_{vm}}^T(k|k) & \mathbf{P}^i_{mm}(k|k) \end{bmatrix} \tag{3.2}$$

where vehicle and map components are denoted by the subscripts v and m respectively. The dimension of $\hat{\mathbf{x}}^i(k|k)$ is different for each i as the local environment (e.g. the number of landmarks observed etc.) is different for each vehicle. The LSF is merely considered as a component which takes the sensor measurements and the external contributing information as inputs and estimates the vehicle and map states through a recursive process of prediction and update.

It is important to note that an appropriate pruning algorithm must be employed by the LSF in order to prevent an undesirable growth of the estimates' state vector and covariance matrix in a large-scale environment. The significance of the pruning is to minimise the memory requirements and the computational complexity of the local vehicle. As a consequence, different vehicles in the network will have different and non-excessive map sizes. Landmark pruning will be discussed in more details in Section 3.3.

A detailed description of the local SLAM filter deployed in the multi-vehicle mapping system in this thesis is provided here.

Vehicle and Map State Vectors and Covariance Matrices in LSF

We now provide the state vector and covariance matrix for the vehicle and map components of the local SLAM filter inside a nominal vehicle. We drop the vehicle's superscript i for this discussion, as only a single vehicle is considered.

In this chapter as well as all the related simulations in Chapter 6, the state of the vehicle at time k is represented by its pose (2-D position and orientation) relative to a base Cartesian coordinate system with the following mean and covariance

$$\hat{\mathbf{x}}_v(k|k) = [\hat{x}_v(k|k), \hat{y}_v(k|k), \hat{\phi}_v(k|k)]^T \tag{3.3}$$

$$\mathbf{P}_{vv}(k|k) = \begin{bmatrix} P_{x_v x_v}(k|k) & P_{x_v y_v}(k|k) & P_{x_v \phi_v}(k|k) \\ P_{y_v x_v}(k|k) & P_{y_v y_v}(k|k) & P_{y_v \phi_v}(k|k) \\ P_{\phi_v x_v}(k|k) & P_{y_v x_v}(k|k) & P_{\phi_v \phi_v}(k|k) \end{bmatrix} \tag{3.4}$$

Also, in this chapter, the state of the observed landmarks at time k is represented by the combined vector of their 2-D position estimates with respect to the same coordinate system[1]. The map covariance matrix $\mathbf{P}_{mm}(k|k)$ includes the correlation information (the diagonal terms) and the cross-correlation information between the landmarks (off-diagonal terms). The cross-correlation terms encapsulate the dependencies amongst different landmarks in the map. Theoretically speaking, as the number of observations increases, the dependency between the map elements also increases as the map becomes fully correlated in the limit.

$$\hat{\mathbf{x}}_m(k|k) = [\hat{x}_{m1}(k|k) \quad \hat{y}_{m1}(k|k) \quad \ldots \quad \hat{x}_{mn}(k|k) \quad \hat{y}_{mn}(k|k)]^T \tag{3.5}$$

$$\mathbf{P}_{mm}(k|k) = \begin{bmatrix} P_{x_{m1}x_{m1}}(k|k) & P_{x_{m1}y_{m1}}(k|k) & \ldots & P_{x_{m1}x_{mn}}(k|k) & P_{x_{m1}y_{mn}}(k|k) \\ P_{x_{m2}x_{m1}}(k|k) & P_{x_{m2}y_{m1}}(k|k) & \ldots & P_{x_{m2}x_{mn}}(k|k) & P_{x_{m2}y_{mn}}(k|k) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{x_{mn}x_{m1}}(k|k) & P_{x_{mn}y_{m1}}(k|k) & \ldots & P_{x_{mn}x_{mn}}(k|k) & P_{y_{mn}x_{mn}}(k|k) \\ P_{y_{mn}x_{m1}}(k|k) & P_{y_{mn}y_{m1}}(k|k) & \ldots & P_{y_{mn}x_{mn}}(k|k) & P_{y_{mn}y_{mn}}(k|k) \end{bmatrix} \tag{3.6}$$

LSF State Transition Model Equations

The vehicle's motion model in this chapter is considered to be two-dimensional, i.e., the vehicle operates on a flat surface where its state at each point is described by a 2-D position vector and a heading. All landmarks are assumed stationary with no process noise (see section 2.3.4 for the general state transition models in SLAM). The states of the vehicle are propagated using the following state space equations

[1]The multi-vehicle simulations pertaining to this chapter are all performed under this two dimensional assumption for the model of the landmarks. Nevertheless, this setup may differ in other chapters depending on the context. For example, Chapters 4 and 5 assume a pose map containing the position and orientation of the observed landmarks.

$$\begin{bmatrix} \hat{x}_v(k) \\ \hat{y}_v(k) \\ \hat{\phi}_v(k) \end{bmatrix} = \begin{bmatrix} \hat{x}_v(k-1) + \Delta t V(k) \cos\left(\hat{\phi}_v(k-1) + \gamma(k)\right) \\ \hat{y}_v(k-1) + \Delta t V(k) \sin\left(\hat{\phi}_v(k-1) + \gamma(k)\right) \\ \hat{\phi}_v(k-1) + \frac{1}{B}\Delta t V(k) \sin\left(\gamma(k)\right) \end{bmatrix} + \begin{bmatrix} w_x(k) \\ w_y(k) \\ w_\phi(k) \end{bmatrix} \tag{3.7}$$

where $V(k)$ and $\gamma(k)$ are the system's control inputs which denote the velocity and steer angle respectively, while B is the vehicle's wheelbase (the distance between the front and rear axles). Δt is the time difference between two consecutive steps k and $k-1$ in the state space.

The EKF-based propagation of the covariance matrix is performed using Equation (2.14).

LSF Obervation Model Equations

This chapter assumes that the landmarks are observed using a range-bearing sensor[2]. The following observation model is then used to express the relationship between the current vehicle and map state and the range/bearing observations.

$$\mathbf{z}(k) = \begin{bmatrix} z_R(k) \\ z_\theta(k) \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_v(k) - \hat{x}_i(k))^2 + (\hat{y}_v(k) - \hat{y}_i(k))^2} \\ \arctan\left(\frac{\hat{y}_v(k) - \hat{y}_i(k)}{\hat{x}_v(k) - \hat{x}_i(k)} - \hat{\phi}_v(k)\right) \end{bmatrix} + \begin{bmatrix} v_R(k) \\ v_\theta(k) \end{bmatrix} \tag{3.8}$$

where $\hat{x}_i(k)$ and $\hat{y}_i(k)$ are the location estimates of the observed landmark. Subscript m has been safely dropped here since there is little confusion about the landmark and vehicle variables.

The EKF-based update stage for the local SLAM filter is performed using Equations (2.49-5.46).

Landmark Initialisation in LSF

The initial position estimate of a newly observed landmark can be computed using an initialisation function $\mathbf{g}_i(\hat{\mathbf{x}}_v(k|k-1), \mathbf{z}(k))$ as described in section 2.3.6. This function can be derived from Equation (3.8) and is shown here.

$$\begin{bmatrix} \hat{x}_i(k) \\ \hat{y}_i(k) \end{bmatrix} = \begin{bmatrix} \hat{x}_v(k|k-1) + z_R(k) \cos\left(\hat{\phi}_v(k|k-1) + z_\theta(k)\right) \\ \hat{y}_v(k|k-1) + z_R(k) \sin\left(\hat{\phi}_v(k|k-1) + z_\theta(k)\right) \end{bmatrix} \tag{3.9}$$

Given the above initialisation function, Equations (2.55) and (2.56) are used to augment the state vector and covariance matrix of the local SLAM filter.

[2]In actual fact, observations from a camera can also be used to compute range and bearing estimates for a detected feature. Therefore, this assumption is in-line with the practical setup in the AutoMap project. See Section 2.7 for a list of sensors used in this project.

3.2.1.1 Map Information

As described in Section 2.2.3, given a state estimate $\hat{\mathbf{x}}(k|k)$ with covariance $\mathbf{P}(k|k)$, the information vector and information matrix are defined by a bijective mapping

$$\hat{\mathbf{y}}(k|k) = \mathbf{P}^{-1}(k|k)\hat{\mathbf{x}}(k|k) \tag{3.10}$$
$$\mathbf{Y}(k|k) = \mathbf{P}^{-1}(k|k) \tag{3.11}$$

The reason for considering the information-based representation for states and covariances is that the interpretation, communication and the fusion of a group of estimates is more convenient in this form.

From Equation (3.2) we then define the total map information at each vehicle i as

$$\hat{\mathbf{y}}^i_{mm}(k|k) = {\mathbf{P}^i_{mm}}^{-1}(k|k)\hat{\mathbf{x}}^i_m(k|k) \tag{3.12}$$
$$\mathbf{Y}^i_{mm}(k|k) = {\mathbf{P}^i_{mm}}^{-1}(k|k) \tag{3.13}$$

Eric Nettleton's thesis [Nettleton, 2003] provides two important results concerning the cross-correlation between vehicle state estimates $\hat{\mathbf{x}}^i(k|k)$ under some pretty common assumptions. Suppose that the size of $\hat{\mathbf{x}}^i_m(k|k)$ is the same for all i; i.e., we can think of $\hat{\mathbf{x}}^i_m(k|k)$ as a local estimate of the complete global map. Also suppose that the association (i.e., ordering) amongst the elements of $\hat{\mathbf{x}}^i_m(k|k)$ is consistent between vehicles and that

$$\mathrm{E}[(\hat{\mathbf{x}}^i_m(k|k) - \mathbf{x}_m(k))(\hat{\mathbf{x}}^j_m(k|k) - \mathbf{x}_m(k))^\top] = \mathbf{0} \tag{3.14}$$

where $\mathbf{x}_m(k)$ is the actual map of the environment. Then the information vector and the information matrix of the best, linear unbiased, estimate of the global map are simply obtained by

$$\hat{\mathbf{y}}_{mm}(k|k) = \sum_i \hat{\mathbf{y}}^i_{mm}(k|k) \tag{3.15}$$
$$\mathbf{Y}_{mm}(k|k) = \sum_i \mathbf{Y}^i_{mm}(k|k) \tag{3.16}$$

Moreover, under these assumptions

$$\mathrm{E}[(\hat{\mathbf{x}}^i_v(k|k) - \mathbf{x}^i_v(k))(\hat{\mathbf{x}}^j_v(k|k) - \mathbf{x}^j_v(k))^\top] = \mathbf{0} \tag{3.17}$$

where $\mathbf{x}^i_v(k)$ is the actual i^{th} vehicle location.

However, in practice the assumption that

$$\mathrm{E}[(\hat{\mathbf{x}}^i_m(k|k) - \mathbf{x}_m(k))(\hat{\mathbf{x}}^j_m(k|k) - \mathbf{x}_m(k))^\top] = \mathbf{0} \tag{3.18}$$

is typically not justified and individual vehicle maps $\hat{\mathbf{x}}^i_m(k|k)$ may only partially overlap and be of different sizes. Therefore, the results of Nettleton above are not always applicable (as noted in much of Nettleton's own work, e.g. [Nettleton et al., 2006]).

3.2.1.2 Selective Communication

Given the practical scenario envisioned for this work, it follows that limited communication bandwidth constrains the transmission of information to and from the central server. Consequently, the accuracy of the central map should be optimised in some manner as a function of the information sent by the individual vehicles under the limited bandwidth constraints. More generally, the desired quality of the map, the available communication bandwidth and the available processing power at the server side determine the type and the rate of information that needed to be collected and transmitted. These factors as well as the available local processing facility at the vehicles control the place where the integration and assimilation of information should be performed. As discussed later in Subsection 3.3.4, we use the information gain (between the local sub-maps known at the central server and the improved maps resulting from the local SLAM algorithm) as a measure to select the most informative sub-map within the local SLAM algorithm for communication.

3.2.2 Channel Filter (CHF)

A channel filter is a popular structure in decentralised data fusion architectures and is used to maintain an estimate of the common information between particular nodes. In a general decentralised network, a channel filter on node i connected to node j maintains the common information vector $\hat{\mathbf{y}}_{ij}(k|k)$ and the common information matrix $\mathbf{Y}_{ij}(k|k)$. Furthermore, the channel filter is responsible for synchronisation of the incoming and outgoing information from the local filter employed by a node. A variation of the channel filter concept is used in the work described here to keep track of the common information between each node and the central server (this will be discussed in Section 3.3). Under the independence assumptions discussed previously by Nettleton, and where complete and overlapping maps are shared between two nodes i and j then

$$\begin{aligned}\hat{\mathbf{y}}^j_{CH}(k|k) &= \hat{\mathbf{y}}^j_{CH}(k|k-1) + [\hat{\mathbf{y}}^i_{mm}(k|k) - \hat{\mathbf{y}}^j_{CH}(k|k-1)] \\ &= \hat{\mathbf{y}}^i_{mm}(k|k) \qquad (3.19)\\ \mathbf{Y}^j_{CH}(k|k) &= \mathbf{Y}^j_{CH}(k|k-1) + [\mathbf{Y}^i_{mm}(k|k) - \mathbf{Y}^j_{CH}(k|k-1)] \\ &= \mathbf{Y}^i_{mm}(k|k) \qquad (3.20)\end{aligned}$$

where $\hat{\mathbf{y}}^j_{CH}(k|k)$ and $\mathbf{Y}^j_{CH}(k|k)$ denote the j^{th} channel's information vector and information matrix at time k given the updated information at time k from the i^{th} data source. However, if the channel map information and the transmitted map have different sizes and/or there is some cross-correlation between the shared information and the existing data in the channel then this approach may lead to inconsistent estimates of the common information between nodes. Broadly speaking, if the statistics of the correlations can be tracked down and identified, the full joint probability function [Papoulis and Probability, 1991] can be used to obtain the minimum mean squared error (MMSE) estimates of the common information. Otherwise, one of the many suboptimal approaches should be applied.

Different methods have been developed to address the data fusion problem when exact knowledge of the correlation between information sources is not available. Methods based on Kalman filtering (KF) simply ignore the unmodeled correlations by assuming independence between the prior estimation error and the new information error. This presumption has been sufficient for a wide range of practical situations and has been successfully implemented in applications such as navigation [Kim et al., 2006], sensor fusion [Amirsadri et al., 2012b], map building Dissanayake et al. [2001] and target tracking [Blackman and Popoli, 1999]. Nevertheless, since the independence assumption is only an approximation to reality, it can potentially lead to serious problems. An example of such a case is the famous 'double counting' problem in distributed sensor networks which leads to information redundancy and over-confident estimates resulting from discarding the common information between two nodes; see [Chen et al., 2002]. In practice, a typical solution to avoid over-confident estimation relies on artificially inflating the covariance of the combined estimate. This method is ad-hoc and unreliable as the level of inflation cannot be precisely quantified and is largely application dependent.

The inconsistency issue caused by ignoring the correlation can be tackled by applying conservative fusion algorithms. Perhaps one of the most popular methods for this purpose is the Covariance Intersection (CI) algorithm. CI was first introduced in a seminal paper by Simon Julier and Jeffrey Uhlmann [Julier and Uhlmann, 1997] and has since been used in a wide spectrum of applications, particularly in the field of decentralised and distributed fusion [Wang and Li, 2010; Uhlmann et al., 1999].

The CI method is also employed in this work to overcome the aforementioned inconsistency issue in the channel filter created by combining dependant, correlated information. Appendix A provides a formal justification for the use of CI in the subsequent discussions. It establishes a result concerning estimation consistency, CI and fusion while neglecting cross-correlations. We are not aware of a similar formal argument along the lines given in appendix A for justifying CI (and we show that for some cross-correlations simply neglecting the cross-correlation will outperform CI and remain consistent).

A synopsis of the CI algorithm is provided here.

3.2.2.1 Covariance Intersection (CI) Algorithm

The covariance intersection (CI) algorithm is a conservative method to consistently combine two or more estimates (e.g. running state estimates and sensor measurement) when the correlation among them is unknown. In the context of multi-vehicle mapping, it is used to manage the double counting of information when combining the information from different vehicles.

Consider the problem of combining two estimates A and B with unknown degree of correlation into an estimate C at an arbitrary time k. Given the mean and the covariance of the two estimates[3] as:

$$
\begin{aligned}
A \quad &: \quad \hat{\mathbf{x}}_A(k|k), \qquad \mathbf{P}_A(k|k) && (3.21)\\
B \quad &: \quad \hat{\mathbf{x}}_B(k|k), \qquad \mathbf{P}_B(k|k) && (3.22)
\end{aligned}
$$

The mean and the covariance of estimate C are calculated according to:

$$
\begin{aligned}
\hat{\mathbf{x}}_C(k|k) &= \mathbf{P}_C(k|k)\left(\omega\mathbf{P}_A^{-1}(k|k)\hat{\mathbf{x}}_A(k|k) + (1-\omega)\mathbf{P}_B^{-1}(k|k)\hat{\mathbf{x}}_B(k|k)\right) && (3.23)\\
\mathbf{P}_C(k|k) &= \left(\omega\mathbf{P}_A^{-1}(k|k) + (1-\omega)\mathbf{P}_B^{-1}(k|k)\right)^{-1} && (3.24)
\end{aligned}
$$

where $\omega\epsilon[0,1]$ is usually selected based on some heuristic to minimise some criteria of uncertainty. One of the most commonly used methods is to select the coefficient ω so as to minimises the determinate of the resulting covariance matrix $\mathbf{P}_C(k|k)$. The CI update can be written more naturally in the information form using Equations (3.10) and (3.11) as:

$$
\begin{aligned}
\hat{\mathbf{y}}_C(k|k) &= \omega\hat{\mathbf{y}}_A(k|k) + (1-\omega)\hat{\mathbf{y}}_B(k|k) && (3.25)\\
\mathbf{Y}_C(k|k) &= \omega\mathbf{Y}_A(k|k) + (1-\omega)\mathbf{Y}_B(k|k) && (3.26)
\end{aligned}
$$

where $\hat{\mathbf{y}}_C(k|k)$ and $\mathbf{Y}_C(k|k)$ are the information vector and the information matrix of the fused estimate C respectively.

The main benefit of using CI in data fusion applications is the ability of this algorithm to generate consistent estimates, regardless of the degree of correlation between the information sources. However, CI often results in highly conservative estimates, i.e., the estimated covariance can be much larger than the actual covariance. Therefore, alternative methods which provide less conservative estimates would be preferred. This is simply because CI always provides an upper-bound of the true covariance.

Notwithstanding its conservative nature, the CI method will be used in Section 3.3 to calculate the common information between each node and the data from the central server and to overcome the inconsistency issue described earlier in this section.

[3] The covariance intersection technique is based upon the assumption that measurements or states can be described with Gaussian probability density functions.

3.2.3 Central Fusion Center (CFC)

The medial component in the single-level hierarchical architecture introduced in Section 2.8 is a base station called the central fusion centre (CFC). The CFC is responsible for assimilating the local sub-maps transmitted from individual vehicles into a consistent global map. The resulting global map is maintained in a central map repository at the server and is accessible at any time. More specifically, the global database accommodates information such as type, geolocation (along with its associated uncertainty) of all landmarks detected by the survey vehicles. Moreover, as mentioned in the previous chapter, a feedback configuration in the system provides a route for the transmission of sub-maps of the global map back to the local SLAM filters (via channel filter) in remote vehicles. As such, individual vehicles indirectly have access to the information obtained by other vehicles in the system.

The strategical importance of having this central base station as a communication hub can be justified in the context of the mapping application and based on the resource limitations outlined in Section 2.7.1. As explained in Chapter 2, due to the size of the mapping problem under study, it is impractical to process and store all the map data locally at individual vehicles. Furthermore, in terms of system robustness, the safety of information at the CFC can be guaranteed in the setup, whereas, the components of a survey vehicle on a mission are generally prone to failure and information loss.

3.3 Efficient Sub-map Communication and Fusion

In this section we consider the communication sequence for a single vehicle, e.g. one of the components shown in Figure 2.8, and discuss the process of information fusion when the shared information between the CFC and an individual vehicle is correlated and of differing sizes. The proposed communication block diagram is shown in Figure 3.1. As shown in the figure, a channel filter (CHF) has been added to the vehicle (see 3.2.2). Note that such a channel filter is also identically replicated at the CFC for each vehicle. The CHF maintains an information vector $\hat{\mathbf{y}}_{CH}(k|k)$ and matrix $\mathbf{Y}_{CH}(k|k)$ representing the newly acquired and shared information.

We consider the following sequence of steps:

1. Communicating the CFC information to the vehicle
2. Updating the channel filter using the map information from the CFC
3. Updating the local SLAM filter
4. Selecting the local vehicle sub-map to communicate to the CFC
5. Updating the channel filter using the selected sub-map from the LSF
6. Updating the global map using the communicated information from the local vehicles

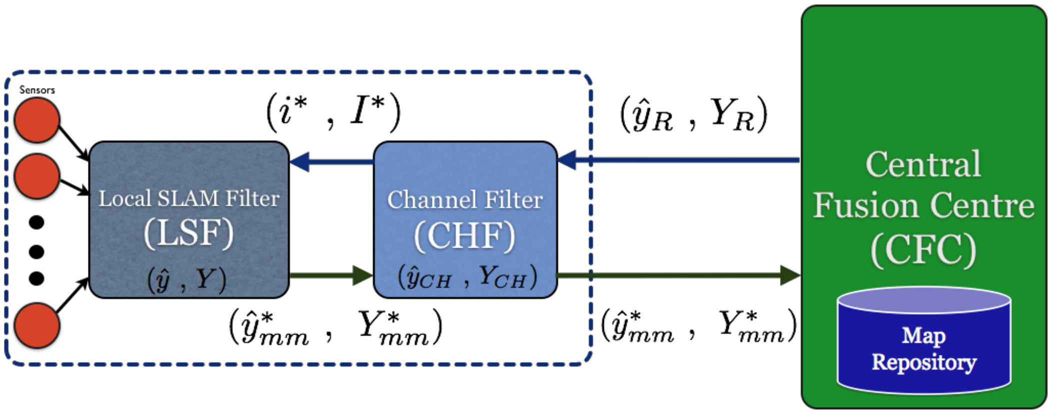


Figure 3.1: Single-vehicle Information Communication Block Diagram

In this thesis, the first three steps and the last three steps are referred to as downlink and uplink respectively.

3.3.1 Communicating the CFC Information to the Vehicle

All the landmarks[4] within the global map held by the CFC that are in a pre-defined radius (r_{regional}) around the vehicle are transmitted to the vehicle[5].

The reason behind sending 'all' the regional landmarks to the local vehicle can be justified with two arguments. Firstly, those landmarks in the CFC which satisfy the accuracy requirements[6] will be sent back to the vehicle in order to provide the vehicle's local SLAM filter with a reasonably accurate set-point to help the localisation of the vehicle (due to the existing correlation between the vehicle and the landmarks). Secondly, those groups of CFC landmarks which do not fulfil the accuracy requirements will be communicated to the vehicle to serve as prior information. This information can be potentially improved using the fresh measurements obtained by the in-vehicle sensor platform. In addition, previously recorded landmarks can potentially help the data association task in the local vehicle. If the vehicle discovers new landmarks whilst it explores the environment, it most certainly means that they have not been previously reported back to the CFC by any of the distributed vehicles (at least until the previous communication). Therefore all of those newly discovered landmarks should be sent back to the server. Also, when it comes to the previously registered landmarks, a decision has to be made on how to select the most informative sub-map to communicate. This will be clarified further in Section 3.3.4.

[4] When we say information is transmitted it is typically meant that the corresponding state vector (or information space representation) and the corresponding marginalised covariance (or information space equivalent) is transmitted.

[5] We assume that the CFC can access the global coordinates of the sensor platforms on demand.

[6] The accuracy requirement is usually defined with respect to the mapping application. This will be elaborated further in Chapter 4.

This so-called regional map that is sent from the CFC to the i^{th} vehicle is denoted by $\mathbf{M}_R^i(\hat{\mathbf{y}}_R^i, \mathbf{Y}_R^i)$. This information will be received at the communication channel filter (CHF) of the local vehicle.

3.3.2 Updating the Channel Filter Using the Map Information from the CFC

After receiving the regional map from the CFC, the channel filter needs to be updated. This update is performed by combining the newly arrived regional map with the previous channel information; i.e., $\hat{\mathbf{y}}_{CH}^i(k|k-1)$ and $\mathbf{Y}_{CH}^i(k|k-1)$. Please note that the channel filter will never maintain any states other than map states, since vehicle information is never communicated.

Let's assume that the communicated regional information map and the existing information map in the channel filter are given by $\mathbf{M}_R^i(\hat{\mathbf{y}}_R^i, \mathbf{Y}_R^i)$ and $\mathbf{M}_{CH}^i(\hat{\mathbf{y}}_{CH}^i, \mathbf{Y}_{Ch}^i)$ respectively.

In general, the channel map information and the transmitted map have different sizes and/or there is some cross-correlation between the shared information and the existing data in the channel. Therefore, using a scheme similar to Equations (3.19) and (3.20) for updating the channel may lead to inconsistent estimates of the common information between the vehicle and the CFC. To overcome this inconsistency we employ the covariance intersection (CI) algorithm (discussed in 3.2.2.1) to calculate the common information between two nodes.

We now drop the superscript i where there is no danger of confusion (and in this section we consider only the communication between the CFC and a single vehicle i). The CI algorithm requires both information matrices to be of the same size. Thus, define the map domain $\mathcal{M}_F$ as the union of the landmarks in the channel $\mathbf{M}_C(\hat{\mathbf{y}}_{CH}, \mathbf{Y}_{CH})$ and the regional map $\mathbf{M}_R(\hat{\mathbf{y}}_R, \mathbf{Y}_R)$ as shown in Figure 3.2.

Two projection matrices are defined $\mathbf{G}_{R2F}$ and $\mathbf{G}_{C2F}$ and consist of 0 and 1 elements. These matrices inflate $\hat{\mathbf{y}}_R$ and $\hat{\mathbf{y}}_{CH}$ to match the cardinality of the union $\mathcal{M}_F$ by padding those components in each respective vector by zero when the landmark indexed by that component is present only in the other vector. The CI algorithm is then given by

$$\begin{aligned}
\hat{\mathbf{y}}_{CH}(k|k) &= \omega[\mathbf{G}_{C2F}(k|k)\hat{\mathbf{y}}_{CH}(k|k-1)] \\
&+ (1-\omega)[\mathbf{G}_{R2F}(k|k)\hat{\mathbf{y}}_R(k|k)] \qquad (3.27)\\
\mathbf{Y}_{CH}(k|k) &= \omega[\mathbf{G}_{C2F}(k|k)\mathbf{Y}_{CH}(k|k-1)\mathbf{G}_{C2F}^T(k|k)] \\
&+ (1-\omega)[\mathbf{G}_{R2F}(k|k)\mathbf{Y}_R(k|k)\mathbf{G}_{R2F}^T(k|k)] \qquad (3.28)
\end{aligned}$$

where $\mathbf{y}_{CH}(k|k)$ and $\mathbf{Y}_{CH}(k|k)$ denote the i^{th} channel's information vector and information matrix at time k given the updated information at time k from the regional sub-map.

The new information received from the CFC is given by

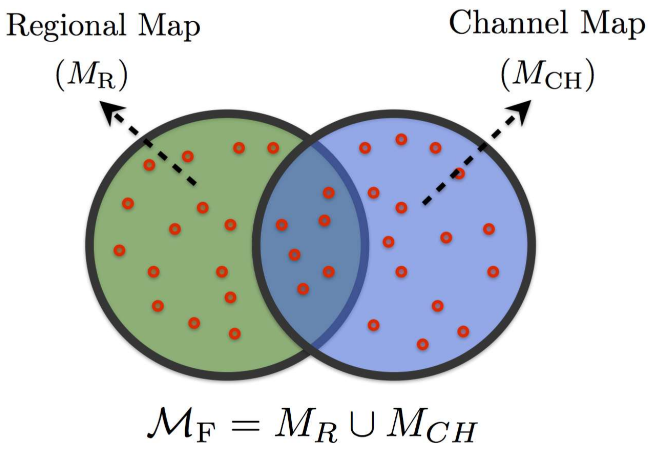


Figure 3.2: The map domain $\mathcal{M}_{\mathrm{F}}$ is the union map obtained by combining the regional map domain sent from the CFC and the existing map domain in the channel filter.

$$\mathbf{i}^*(k|k) = \hat{\mathbf{y}}_{CH}(k|k) - \mathbf{G}_{C2F}\hat{\mathbf{y}}_{CH}(k|k-1) \tag{3.29}$$

$$\mathbf{I}^*(k|k) = \mathbf{Y}_{CH}(k|k) - \mathbf{G}_{C2F}\mathbf{Y}_{CH}(k|k-1)\mathbf{G}_{C2F}^T \tag{3.30}$$

The information increment is sent to the LSF, e.g. see Figure 3.1 to be combined with the locally running SLAM filter. Computing the increment prevents double counting of information in the LSF as discussed next.

3.3.3 Updating the Local SLAM Filter

When the local SLAM filter receives the information increment from the channel filter it uses this information to update its estimates. For this purpose, proper projection matrices $\mathbf{G}_{N2H}$ and $\mathbf{G}_{L2H}$ are defined as previously discussed in order to inflate the information increment vector $\mathbf{i}(k|k)$ and the local information vector $\hat{\mathbf{y}}(k|k)$ to the size of the union domain $\mathcal{M}_H$. In constructing the former projection matrices, in addition to padding the respective vectors with zeros at those elements corresponding to the non-overlapping landmarks, we must also pad components into $\mathbf{i}^*(k|k)$ with zero to correspond with the vehicle components in $\hat{\mathbf{y}}(k|k)$. Recall no vehicle state is communicated. The update is done according to:

$$\hat{\mathbf{y}}(k|k) = \mathbf{G}_{L2H}\hat{\mathbf{y}}(k|k-1) + \mathbf{G}_{N2H}\mathbf{i}^*(k|k) \tag{3.31}$$

$$\mathbf{Y}(k|k) = \mathbf{G}_{L2H}\mathbf{Y}(k|k-1)\mathbf{G}_{L2H}^T + \mathbf{G}_{N2H}\mathbf{I}^*(k|k)\mathbf{G}_{N2H}^T \tag{3.32}$$

and as the LSF is typically executed in the standard state space it follows that $\mathbf{x}(k|k) = \mathbf{Y}^{-1}(k|k)\hat{\mathbf{y}}(k|k)$ and $\mathbf{P}(k|k) = \mathbf{Y}^{-1}(k|k)$ as before.

3.3.4 Selecting the Local Vehicle Sub-map to Communicate to the CFC

This algorithm is motivated by the AutoMap practical application where the primary objective is to construct and maintain a high-quality global map at a centralised station using information collected (and pre-processed to some degree) at local vehicles. Since the communication bandwidth is limited as previously noted, the 'most informative' sub-map needs to be selected and transmitted back to the CFC. The term 'most-informative' sub-map is necessarily ambiguous. Intuitively one would like to consider the available communication resources and subject to this constraint then select those landmarks in the local vehicle's map that will reduce the uncertainty in any resulting global map constructed at the CFC.

There are numerous measures of informativeness. The simplest method involves selecting a sub-map based on the measured information gain. In this application, the information gain is computed by taking the information matrix of the available local landmarks (in the LSF) and comparing this with the existing channel information (all the information transmitted from the LSF previously). We define the information gain of the local map according to:

$$\mathbf{I}(k|k) = \mathbf{Y}_{mm}(k|k) - \mathbf{G}_{C2M}(k|k)\mathbf{Y}_{CH}(k|k)\mathbf{G}_{C2M}^{T}(k|k) \tag{3.33}$$

where $\mathbf{Y}_{mm}(k|k) = \mathbf{P}_{mm}{}^{-1}(k|k)$ and an appropriate (as in previous arguments) inflation matrix $\mathbf{G}_{C2M}$ has been used.

Assume that $\mathbf{I}(k|k)$ encodes the information gain regarding a total number of p landmarks. Due to the existing communication constraints, the information of q landmark ($q < p$) will be transmitted where q is determined by the available bandwidth or an allocated communication budget for time k. The q landmarks with the highest information gain will be selected for transmission. The simple method which is used here is done by picking up the q landmarks with the largest diagonal elements in the $\mathbf{I}(k|k)$ matrix. The selected information sub-map for communication to the CFC will be denoted by $\hat{\mathbf{y}}^{*}_{mm}(k|k)$ and $\mathbf{Y}^{*}_{mm}(k|k)$. This information sub-map is sent to the channel filter prior to transmission to the CFC (see Figure 3.1).

3.3.4.1 Pruning the Local SLAM Filter

As mentioned before in this thesis, in large-scale mapping applications, it is imperative to prevent the size of the local map within the individual vehicles from growing unboundedly. To achieve this, pruning algorithm is implemented at each communication time to limit the size of the SLAM filter. Landmarks with the lowest information gain are eliminated from both the LSF and CHF of each vehicle to reduce the size of the local map to a pre-defined constant n_{pr}, without comprising the integrity of the system.

The salient point here is that, this pruning method is distinct from the standard computationally efficient solutions to the SLAM problem, in a sense that the information (and cross-information) of the discarded landmarks is not lost, due to the previous communication of information to the CFC. This information can be restored locally at any time by downloading the map information from the server.

Note: Different methods and tools can be utilised in our specific application to select the sub-map to communicate. For example, to compare the amount of contribution provided by different sub-maps, the sign types can also be accounted for and prioritised based on the significance of the sign localisation accuracy for each sign type.

3.3.5 Updating the Channel Filter Using the Selected Sub-map from the LSF

When the information sub-map presented by $\hat{\mathbf{y}}^*_{mm}(k|k)$ and $\mathbf{Y}^*_{mm}(k|k)$ arrives at the channel filter, the channel update is performed using the covariance intersection method similar to what was shown in Section 3.3.2. As mentioned earlier in this chapter, CI will yield conservative results when combining sub-maps of arbitrary size with unknown correlations. This update is done according to:

$$
\begin{aligned}
\hat{\mathbf{y}}_{CH}(k|k) &= \omega[\mathbf{G}_{C2Q}(k|k)\hat{\mathbf{y}}_{CH}(k|k-1)] \\
&+ (1-\omega)[\mathbf{G}_{S2Q}(k|k)\mathbf{y}^*_{mm}(k|k)] \qquad (3.34) \\
\mathbf{Y}_{CH}(k|k) &= \omega[\mathbf{G}_{C2Q}(k|k)\mathbf{Y}_{CH}(k|k-1)\mathbf{G}^T_{C2Q}(k|k)] \\
&+ (1-\omega)[\mathbf{G}_{S2Q}(k|k)\mathbf{Y}^*_{mm}(k|k)\mathbf{G}^T_{S2Q}(k|k)] \qquad (3.35)
\end{aligned}
$$

where $\mathcal{M}_Q$ is the union of the landmarks presented in the communicated sub-map and the channel filter as shown in Figure 3.3. Projection matrices $\mathbf{G}_{C2Q}$ and $\mathbf{G}_{S2Q}$ are defined appropriately to inflate the sub-maps to the size of the union $\mathcal{M}_Q$.

After the complete update of the channel filter using Equations (3.34) and (3.35), the selected information sub-map $\hat{\mathbf{y}}^*_{mm}(k|k)$ and $\mathbf{Y}^*_{mm}(k|k)$ will be transmitted to the central fusion centre to be fused with the existing global map. The salient point is that the sub-map presented by $\hat{\mathbf{y}}^*_{mm}(k|k)$ and $\mathbf{Y}^*_{mm}(k|k)$ contains the entire history of the landmarks in it and not simply an increment of information. This has certain desirable properties such as robustness to system failure and the ability to retrieve the information in case of communication failure.

3.3.6 Updating the Global Map Using the Communicated Information from the Local Vehicles

As described earlier, a channel filter structure is used in each node to keep track of the common information between that node and the central fusion centre (Figure 3.1).

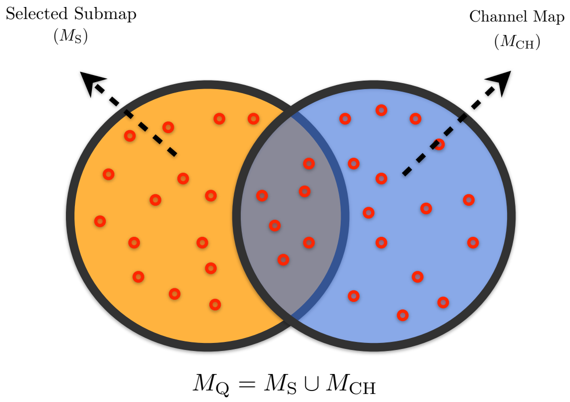


Figure 3.3: $\mathcal{M}_Q$ is the union information map obtained by combining the landmarks in the selected sub-map $M_{\mathrm{S}}(\hat{\mathbf{y}}^*_{mm}, \mathbf{Y}^*_{mm})$ and the existing landmarks in the channel filter $M_{\mathrm{CH}}(\hat{y}_{CH}, Y_{CH})$. The small (red) circles represent the landmarks.

The exact same structure is required inside the CFC for tracking the common information between the CFC and each vehicle. Therefore, for every vehicle in the system, there exists a channel filter at the server (Figure 3.4). This channel is a replica of the CHF inside the corresponding vehicle. As such, when the CFC receives a sub-map

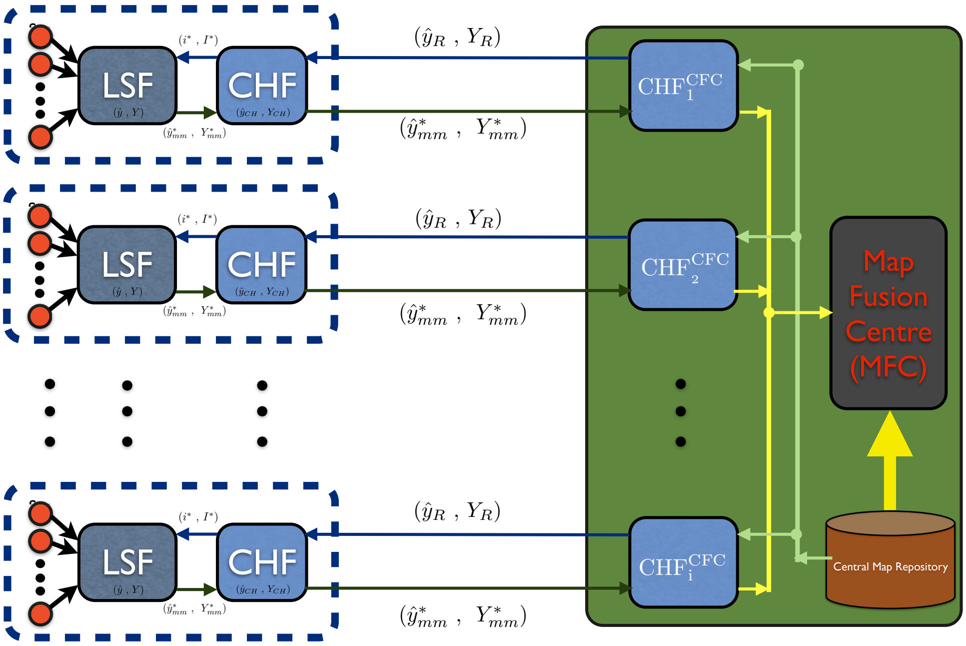


Figure 3.4: The internal structure of the central fusion centre. For every vehicle, there is a corresponding channel filter at the CFC. The map fusion centre (MFC) integrates the information increment of the received map from the vehicles into a global map and sends it to the central map repository.

information from the i^{th} node, it updates the corresponding channel filter ($\mathrm{CHF}_\mathrm{i}^\mathrm{CFC}$) using similar update steps described in Section 3.3.2. Once updated, it calculates the increment of new information it has just received from node i that has not previously been fused in the global map in a similar way to Equations (3.29) and (3.30). A structure called the map fusion centre (MFC), then adds the information increments received from all the exploring vehicles to the existing information map. In this way, the central map becomes complete over time and the distributed vehicles continue to keep it up-to-date. Furthermore, when a regional map is communicated from the CFC to the i^{th} node, the central channel filter corresponding to that node ($\mathrm{CHF}_\mathrm{i}^\mathrm{CFC}$) needs to be updated using CI, in a manner similar to Equations (3.34) and (3.35).

3.4 Summary

This chapter presented an efficient data fusion framework for the problem of multi-vehicle SLAM for very-large-scale road mapping applications. The solution is efficient in terms of both computational complexity and communication bandwidth. A communication algorithm was proposed which operated by intelligently transmitting the most informative sub-map (highest information gain) within the local SLAM filter to the server. A practical pruning algorithm based on information gain was applied to overcome the problem of growing map sizes at the local nodes. The applicability of covariance intersection algorithm was discussed and its used was justified. The proposed communication architecture is capable of dealing with dynamically changing map-sizes in the system and is able to consistently fuse this map information in order to build a global map. The mapping solution is potentially scalable to environments with thousands of vehicles and many millions of landmarks.

Chapter 4

Quality Assessment in Map Making Applications

4.1 Introduction

In the previous two chapters of this thesis an efficient data fusion framework was devised for the problem of multi-vehicle SLAM in very-large-scale environments. The work was inspired by a real-world road mapping application called AutoMap with the objective of producing an accurate map of road signs using a number of dedicated surveying vehicles. A very prominent aspect of such specialised mapping systems (and more generally, every mapping system) is the way in which their performance is evaluated. Addressing this issue is the main focus of the current chapter.

In general, the problem of determining the quality of a given map estimate is subject to debate for robotic mapping applications because no standard method is available. Although a myriad of different techniques have been presented in the literature to tackle the problem of localisation and mapping, the field researchers have found it difficult to reach a consensus on a generic approach to assess the performance of a variety of mapping systems. This precludes the existence of a widely accepted scientific methodology for comparing a multitude of existing mapping techniques. In addition, the recent technological advancements and the emergence of specialised mapping systems call for application-driven performance metrics that are able to capture the particulars of such systems and can be applied in different practical scenarios.

This chapter explicitly pursues a comprehensive discussion on measures of map quality with a focus on road mapping frameworks. A well-defined concept for map quality is sought which can reflect the accuracy of mapping systems in a meaningful way. A new, generic directional map quality metric is introduced which can be tuned to fit a wide spectrum of practical applications depending on priorities in localisation direction and specific types of landmarks. This measure is designed in a way that is conceptually compelling for specialised road mapping applications (such as the work carried out in this thesis) and can be potentially employed by both scientific and business community to serve as a tool for comparing the performance of different mapping algorithms.

This chapter is organised as follows: Section 4.2 presents an overview on some of the most popular techniques used in the literature to assess the quality and performance of localisation and mapping applications and highlights the motivation behind the present work. Fundamental questions such as "what is map quality?" and "how to compare the results of different mapping algorithms" are elaborated. Section 4.3 describes a taxonomy of map quality metrics based on various practical factors. This systematic categorisation sets up the foundation for designing a new quality metric. Some of the practical factors that are needed to be considered in the design and development of a map quality metric for road applications are discussed in Section 4.4. These considerations are discussed around topics such as requirements of different groups of map users, geometry of road structure and the accessibility of information. Finally, Section 4.5 formulates a new map error metric which incorporates the classification of map elements and the priorities in localisation directions. Two types of directional map error (DIMER) metrics are proposed depending on the availability of ground-truth information. We also establish the link between the new and existing quality metrics.

4.2 Motivation and Background

As outlined before in this thesis, one of the principal goals of mobile robotics research is the creation of a map from noisy sensor data collected by a robot as it explores an unknown environment. A myriad of different solutions have been presented in the literature to tackle the problem of robot localisation and mapping[1]. One of the fundamental questions that needs to be answered in any mapping application is "how should the performance of the mapping system be evaluated?". The significance of this question lies in the fact that it is generally desirable to study the impact of different algorithms, utilised system components and environmental/experimental conditions on the performance of mapping systems. In addition, finding an appropriate answer to the above question enables the systematic comparison between the results of various mapping techniques. This has prompted different works in the literature to employ strategies to demonstrate the effectiveness of their methods and the precision of their results. Notwithstanding, it has been argued that (e.g. [Jaulmes et al., 2009; Kümmerle et al., 2009; Mourikis and Roumeliotis, 2006]) the robotic mapping community lacks a generally accepted, standard methodology for quality assessment and comparison between the results of different algorithms. This stems from the fact that it is generally difficult to use a single measure of performance for a wide range of application areas and practical scenarios. Most existing solutions fail to provide a comprehensive evaluation in specialised mapping scenarios, since they do not fully reflect the quality and accurateness of mapping processes. Moreover, a large majority of the existing solutions are subjective, influenced by individual perceptions and hence debatable. A practical quality measure that thoroughly manifests the performance of a mapping systems needs to be driven by the specific character-

[1]Chapter 2 summarises some of the most popular localisation and mapping methods.

istics of that particular application. This is the avenue we follow in this chapter.

This work strives to address these issues and to clarify the concept of "quality assessment" for practical robotic mapping applications, before formulating a new quality metric for specialised road mapping applications. In order to provide the required background, this section reviews some of the state-of-the-art quality assessment techniques used in the literature and outlines their key strengths and weaknesses. Although this discussion can be around most of the existing mapping algorithms, the probabilistic mapping methods (such as SLAM) are the main focus of this chapter.

In general, assessing the performance of mapping algorithms is multifaceted. Different aspects of the mapping systems might be taken into account and examined for this purpose. Ideally, all factors included in the mapping system as well as the final obtained map and the application of the end map should be accounted for in the performance evaluation process. However, quantifying and incorporating all of these aspects in a single measure is infeasible for most practical systems due to the introduced complexity. Therefore, more simplistic methods are usually employed depending on the specifications and requirements of mapping applications. For instance, a naive approach may see the average required communication bandwidth or the computational complexity as performance criteria for the multi-vehicle mapping system described in Chapter 3; i.e. the effectiveness of the system is judged based on communication or computational efficiency. However, using such statistics puts very little emphasis on the overall performance of the mapping system.

A relatively straightforward way of comparing certain robotic systems is judging them by their performance in a competition setting. RoboCup [Rob, 2013] and DARPA Grande Challenge [DAR, 2013] are two famous examples of such competitions. Although individual subsystems are not directly judged, such competition scenarios allow the level of system integration and the merit of certain engineering skills to be ranked for different participating groups [Wulf et al., 2008]. Search and rescue robotic operations [Kleiner et al., 2006; Piniés et al., 2006], robots in simulated Mars environments [ESA, 2008] and cleaning robots [EPF, 2002] are some of the robotic challenges with a defined mission objective as the main criterion to evaluate the performance of different algorithms. However, it can be argued that due to the diversity of hardware/software components used by different participating parties, the competition outcome does not reflect a fair comparison between the merit of utilised mapping algorithms.

Another popular approach used by the scientific community to assess the quality of localisation and mapping techniques is the use of benchmarking methods. These techniques require some sort of ground-truth data about the environment in which the robot operates. Benchmarking the outcome can be performed with respect to the location of robot or map features. The published works on classical SLAM, especially works carried out around autonomous navigation (e.g. [Guivant et al., 2000; Durrant-Whyte et al., 1996]), focus on the algorithm's performance based on the robot localisation error, while little emphasis is put on the error contained in the obtained map. A more recent example is [Kümmerle et al., 2009] in which the estimated position of the robot is compared with a reference position. Therefore, instead of the output map, the poses of the robot during data acquisition are considered. The main advantage of such techniques is that they enable the comparison of different algorithms regardless of the fashion used to represent their generated maps. However, as mentioned before, such measures do not provide any information on the quality of the output map from the algorithm. A more common approach for benchmarking robotic mapping systems is to assess the output map with respect to a reference map. The type of the generated map (e.g. occupation grid maps, position of features/beacons maps, superposition of scans) plays an important role in the manner in which the performance of the system is examined. For example, numerous applications in the area of feature-based estimation use the distance (Euclidean or Mahalanobis distance) between the estimated and true feature locations. In the area of grid-based estimation techniques, visual inspection methods (such as image similarity) are used to benchmark and compare the produced maps. It is also possible to quantitatively measure the correlation between the estimated map and the ground-truth.

Simulation environment is commonly used to generate the ground-truth data and state variables for conducting and repeating experiments in defined conditions (example works include [Bryson and Sukkarieh, 2007; Nettleton et al., 2006; Kim, 2004]). However, the downside is, since it is almost impossible to inclusively model all aspects of different applications, simulation environment inevitably differs from real-world in various respects.

In addition to artificially generated environments, publicly available datasets are extensively used in benchmarking applications to provide a platform to replicate experiments for evaluation and comparison of different methodologies. For example, the Victoria Park benchmark dataset [J. Guivant and Nebot., 2007] has become a popular benchmarking tool within the SLAM research community. Nevertheless, for most real-world applications, ground-truth data is not readily available mainly due to the difficulties involved in the process of gathering this information. More specifically, getting ground-truth information for large scale outdoor environments (like AutoMap) is cumbersome. Hence, alternative tools for a standard, meaningful comparison between different methods are still desirable for these cases.

In the absence of absolute ground-truth information, different criteria are used by the scientific community to evaluate the performance of different mapping algorithms. In Kalman filter based estimation algorithms (e.g. EKF-SLAM), one of the most common approaches is to determine the accurateness of the estimator by making use of the error covariance matrix pertaining to the estimates. The covariance matrix is essentially a measure of uncertainty in the obtained estimate which is generated by the estimator[2]. Error ellipses are popularly used to denote the uncertainty of features' position estimates.

Different measures of the error covariance matrix have been used in the literature. Trace and determinant are the two of the mostly employed types of measures (e.g. [Julier and Uhlmann, 2007; Alriksson and Rantzer, 2006; Sim and Roy, 2005;

[2]This was one of the main methods used in Chapter 3.

Andrade-Cetto et al., 2005; Vidal-Calleja et al., 2004; Amirsadri et al., 2012a]). Geometrically, the determinant is closely related (although not equal) to the volume encapsulated by the error ellipse associated with the estimate's covariance matrix, while the trace has a more subtle interpretation which is based on the lengths of the axes of the ellipse[3]. Metrics dependant on both vehicle and landmark uncertainties are also utilised for a different agenda. For example, in [Feder et al., 1999], an influential paper on adaptive navigation and mapping, the authors introduce a metric for adaptive sensing represented by the sum of the areas of the error ellipses of the vehicle and feature estimates in the map.

In summary, this section described some of the most common approaches for evaluating the performance of localisation and mapping systems. As mentioned earlier, in an ideal world, all factors included in the mapping system should be taken into consideration for a comprehensive quality assessment in robotic applications. However, this is clearly infeasible in a practical sense. As a result, only certain aspects of a mapping system can be considered in a realistic scenario depending on the characteristics and priorities of the particular application. In applications like AutoMap where the final output map is considered as the primary outcome, a sensible decision regarding the performance metric is to emphasise on the quality of that map, regardless of the method with which it has been acquired. This is the avenue we follow in the remainder of this chapter. We seek a well-defined map quality metric to quantitatively assess the performance of localisation and mapping methods such as SLAM. Another significant aspect of such systems is assessing the quality of its output in relation to particular mapping profiles for specialised mapping applications. In order to have a generic way of evaluating the map quality, we devise a metric that can be tuned to fit a wide spectrum of applications. As a preliminary step, Section 4.3 describes a general taxonomy of different map quality metrics.

4.3 Taxonomy of Different Map Quality Metrics

This section describes a taxonomy of map quality metrics with the purpose of setting up the context for designing a new map quality measure for specialised road mapping applications. These metrics can be categorised with respect to different criteria. Different schemes of classification are driven by various factors and points of view such as the properties of the available map, available information and other geometrical or statistical elements. The review of different map metric classifications provides a systematic tool to compare the existing map quality assessment techniques. This section is complementary to the discussion provided in Section 4.2. Here, for simplicity, we only concentrate on two-dimensional feature maps, as it is also closely related to the map making process studied in this thesis.

A relatively simple, yet fundamental categorisation is based on the availability of ground-truth information in the calculation of the map quality metric. Two distinct

[3]See Section 4.5 for more details.

types of measures can be considered for this case

1. Metrics relying on ground-truth information
2. Metrics independent of ground-truth information

Metrics belonging to the first group are frequently used in computer simulations to test the performance of different algorithms. The Euclidean or Mahalanobis distance which are calculated based on the distance between the estimated and true feature positions are examples of such metrics. However, as mentioned in the previous section, these metrics are subject to the availability of ground-truth feature information. Also, such simple measures do not capture the particulars of certain applications. Consequently, their use in real-world practical applications is restricted. In such cases, the second group of metrics can be used in lieu of the first group. These metrics are typically correlated with a quantity representing the map uncertainty. For instance, metrics based on the covariance matrix (e.g. trace or determinant of the map covariance matrix) are commonly used as an uncertainty criteria for feature maps in scenarios where a ground-truth map is inaccessible. Albeit, these metrics are indirectly connected to the metrics of the first group due to the fact that they are often calculated to approximate the distance with respect to the true map. In fact, the covariance can be calculated using the following expected value:

$$\mathbf{P} \triangleq \mathrm{E}\left[(\mathbf{X}-\mathrm{E}(\mathbf{X}))(\mathbf{X}-\mathrm{E}(\mathbf{X}))^T\right] \tag{4.1}$$

where the state vector $\mathbf{X}$ is a general random variable with $\mathrm{E}(\mathbf{X})$ representing its mean value.

The taxonomy of map quality metrics can also be in relation to different map properties. For example, the type of the given map plays a pivotal role in the metric chosen to measure the map's quality. In addition to the position, many applications also consider another attribute for stationary feature maps namely *landmark orientation*. Broadly speaking, orientation can be defined for landmarks with a third dimension (e.g. surface, heading, pointer) where the mean orientation estimation is also of importance in the map making process[4]. Therefore, map quality metrics can be divided into the following three categories, based on the availability of position and orientation of map elements.

1. Metrics based only on landmarks' position
2. Metrics based only on landmarks' orientation
3. Metrics based on both position and orientation of landmarks

In principle, probabilistic estimation techniques such as SLAM estimate a mean along with a measure of uncertainty (e.g. covariance matrix) for the position of map

[4]Orientation will be explicitly defined for road signs in Section 4.4.

elements. For example, for a 2-D position map acquired through the use of the EKF-SLAM algorithm (see Chapter 3), the estimated position mean for a landmark is in the form of a 2×1 matrix and the associated covariance is a 2×2 matrix. This leads to another categorisation for the map quality metric based on the available statistics provided by the mapping algorithm:

1. Metrics based only on estimated mean
2. Metrics based only on estimated variance
3. Metrics based on both mean and variance of the estimate

The last categorisation considered for the taxonomy of map quality metrics in this section is based on isotropy, i.e. whether the properties of the metric vary depending on the factors such as direction or orientation. As will be seen in Section 4.4 this classification is vital for understanding the underlying metric sought for road mapping applications. In general, all map quality metrics can be divided into the following two groups:

1. Isotropic metrics
2. Non-isotropic metrics

The majority of the map quality metrics used by the robotics community (see Section 4.2) are isotropic, meaning that the geometry of the quality metric is the same, regardless of direction. Measures that are solely based on the trace or determinant of the covariance matrix are examples of isotropic metrics.

This section highlighted some of the most important classifications of map quality metrics in robotics applications. It goes without saying that the taxonomy of quality metrics is not limited to the discussion provided here. In what follows in Section 4.4, we set up the requirements for the formulation of a new map quality metric for road mapping frameworks.

4.4 Practical Considerations in Measuring the Map Quality in Road Applications

This section is concerned with some of the practical considerations in design and development of map quality metrics for road applications. Thus far in this chapter, we have addressed quality assessment in general mapping frameworks from a high-level point of view. A handful of different techniques were outlined in Section 4.2 and a general taxonomy was conducted in Section 4.3. In this section, we narrow down our discussion to quality assessment for road mapping applications. More specifically, we focus on the AutoMap project explained previously in this thesis[5]. We

[5]Please note the terms road sign, landmark and map element are used interchangeably throughout this section.

start by providing arguments and examples of reasons why most of the de-facto map quality metrics struggle to provide a comprehensive, generic approach to assess the quality of maps consisting of road signs. This will serve as a justification for the need for a more advanced quality metric for similar road applications. We then provide some of the practical issues that are needed to be considered prior to designing a new metric. These considerations are discussed around factors such as requirements of different user groups, geometry of road structure and information availability. The current section and the discussion provided previously in this chapter set up the basis for Section 4.5 where a new map quality metric is developed and mathematically formulated.

4.4.1 Requirements of Different User Groups

A very important facet that needs to be considered in designing a specialised map quality metric is the targeted user-group. Before making design decisions, one needs to distinguish between different groups of map users and the subtle differences between their requirements. For this purpose, this section considers four main groups that can benefit from a meaningful map quality metric.

1. Digital Mapping companies
2. Road authorities and asset managers
3. Map developers
4. Users of satellite navigation devices (end users)

The first group is comprised of map making companies (e.g. Sensis, TomTom, Nokia and others) who control the personal navigation market. These companies seek to acquire accurate and reliable map estimates consisting of the geo-location of road signs of interest in the environment. This information is mainly used for *routing* purposes. Through the integration of road signs (along with other extracted data from the roads), these companies aim at creating quality maps with an enhanced user experience. For instance, Sensis interprets the road sign data in order to provide efficient and accurate turn-by-turn navigation advice and other location-based features for their satellite navigation devices. For this purpose, the maps must satisfy a certain degree of accuracy where both position and orientation accuracies are important. For a particular road sign, the main concern of this group is to be able to identify the specific road that the sign is applied to.

Road authorities and asset managers constitute the second group studied in this section. These parties are interested in accurate positioning of road signs in order to construct a valid and up-to-date database of the installed traffic signs in different roads. In addition, accurate monitoring of potential changes to the location of road signs (also known as change detection) is an important aspect of road asset management. In order to do this, these groups must have an accurate idea of the sign location and its initial state in the first place. Moreover, sign inventories are used to keep track of statistics of road signs in different areas. This can also facilitate maintenance procedures deemed essential for certain road signs. This group is interested in whether or not different road signs comply with road safety standards (e.g. how close is a given sign to the road) in order to prevent hazardous situations. Therefore, road asset managers care about the asset itself for various maintenance and compliance purposes, whereas mapping companies care about certain statistics regarding the sign essentially for routing purposes, to be able to transfer this information to the end users.

The third group is the party responsible for extracting the information of road signs and developing high-quality digital maps for mapping companies and asset managers. The AutoMap project is a good example of this group[6]. Access to a reliable tool for map quality assessment is of vital practical importance to this group, since the algorithm development in such applications is usually performed in order to optimise a certain quality criteria. Therefore, it is evidently advantageous to use a customisable measure which can be tuned with respect to the requirements of a specific mapping application. In this way, one can improve the map making process by building better quality maps for the other user groups discussed here. This is the main avenue we follow in Chapter 5.

Finally, the fourth group of users who can potentially benefit from a comprehensive map quality metric are the end users. This group includes the regular people who are the users of personal satellite navigation devices. Although the end users may not directly use all the available statistics regarding different signs on a daily basis, accurate geo-localisation of the signs with respect to roads is important to them. Particularly, the emergence of a variety of location-based services for the users of in-vehicle navigation devices has created the necessity for accurate information regarding the traffic signs applied to different roads. The requirements of this group regarding map quality are tightly related to those of the first group (the mapping companies), as the latter should essentially reflect the needs of the former. In other words, the needs of mapping companies are derived from the needs of the end users. For example, the end users want to know which speed limit apply to a certain road, can they drive in a specific direction, etc. These needs are transferred into the needs of mapping companies such as Sensis.

Prior to designing a new quality metric, one must make a distinction between the way in which the above groups are going to evaluate the quality of their maps. Consequently, a map quality metric that accommodates the basic requirements and priorities of different groups of map users is preferred. The above-mentioned descriptive needs must first be transformed into equivalent technical needs. For instance, as explained before, mapping companies are required to determine the specific road any given sign is applied to. Furthermore, road authorities are interested in specifying the proximity of traffic signs to different roads. To be able to do this, parallel and orthogonal accuracies with respect to the road must be distinguished and separated in a quality assessment. This concept will be elaborated in Subsection 4.4.2.

[6]Note that this group can be a part of the organisational pipeline of a mapping company itself. However, for the sake of the discussion in this thesis, they are presumed as a separate entity.

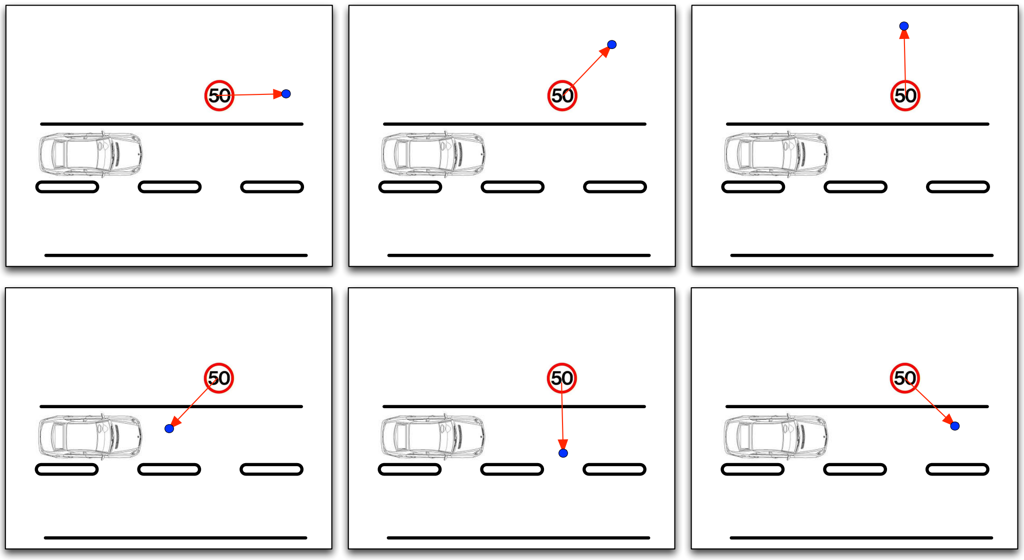


Figure 4.1: An example of six different position estimates for a sample speed sign. The magnitude of the Euclidean distance between the estimated (blue dot) and true position (the 50 km/h speed sign) is identical in all cases.

To allow more degrees of freedom in the system, different sign types will be treated differently in the map quality assessment process. In general, road signs can be divided into two distinct groups: 1- regulatory signs 2- advisory signs. Broadly speaking, regulatory signs are the signs intended to instruct the drivers and other road users in different circumstances. These signs are used to reinforce various traffic laws and regulations. Disregarding regulatory signs will lead to law violations and carry a legal penalty due to their sensitive nature. Stop signs, speed limits and no entry signs are a few examples of the many types in this group. On the other hand, as implied by their names, advisory signs describe a range of signs that are used to recommend certain instructions to the road users. Although disregarding advisory signs may not necessarily be deemed as a traffic offence, obeying them will increase road safety and diminish the risk of road incidents. Caution signs and advisory speed limits are examples of such signs. Typically, advisory signs are considered less critical than regulatory signs. This is one reason why one wants to treat different sign types differently in map quality assessment. This will be the focus of Subsection 4.4.3.

4.4.2 Directional Priorities

As mentioned before, in some mapping scenarios (particularly in road applications) the localisation accuracy for certain types of landmarks is more significant along one particular direction compared to the others. The following discussion is used to exemplify this claim for a simple scenario consisting of a single landmark.

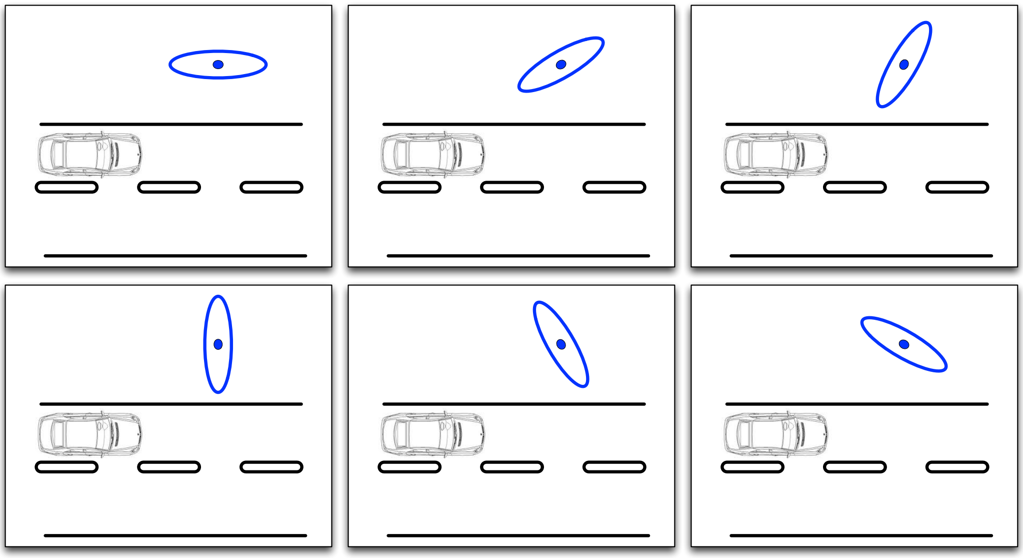


Figure 4.2: An example of six different uncertainty ellipses pertaining to the position estimates of a sample road sign. The trace and determinant of these ellipses are identical, despite their different relative orientation with respect to the road.

Consider a hypothetical scenario where a number of distinct position estimates have been obtained for a particular road sign through the employment of a number of non-identical mapping algorithms. For the first example, we assume that the true position of this sample road sign is completely known. Figure 4.1 shows the estimated position mean and the true sign position for each of the given estimates. The stretch of road corresponding to the sample road sign can also be seen in the figure. Also, for the sake of the argument, suppose that the localisation accuracy for this particular road sign is m times more important along the road's perpendicular axis compared to its parallel axis. It can be argued that the generality of this presumption can be deemed valid for specific road signs (e.g. speed signs[7]) in AutoMap. We refer to this concept as *directional priorities* in this thesis and we further elaborate that in the future[8].

The Euclidean distance between the position estimates and the actual road sign position is one of the most common existing measures for calculating the map error in this scenario. This distance has been shown using a vector for each of the estimates in figure 4.1. Although the Euclidean distance is identical for all six cases, their respective qualities must be different due to the previously mentioned assumption on directional priorities. For example, it is logical that the estimate shown in Box No.1 has a higher quality compared to estimate of Box No.5, since it has got a much smaller perpendicular position error with respect to the road.

[7]Since speed signs are placed to specify the speed limit on a road stretch, the positioning inaccuracies in horizontal directions (with respect to the road) are more tolerable compared to vertical errors.

[8]These priorities will be defined as a function of sign type later in this chapter.

Now for the second example, we consider a similar, yet more realistic scenario in which the ground-truth positions are not available. Suppose that six different position estimates (including mean and covariance) are given (See Figure 4.2). Without loss of generality, we assume that these six given estimates have the same mean, but different covariances. Since the actual sign locations are not accessible in this case, the covariance of these position estimates are used to calculate and compare the error associated with each map. The uncertainty ellipse associated with each estimate's covariance is shown in the figure.

Recalling Section 4.2, trace and determinant of the covariance matrix are two of the most common metrics for map quality assessment in similar scenarios. Employment of either of these isotropic metrics will yield the same value for all the ellipses in Figure 4.2. However, assuming the same directional priorities as before, (most probably) these maps must have different qualities in terms of directional position accuracy, since their parallel and perpendicular uncertainties with respect to the road frame are evidently not alike.

Consequently, it can be concluded from the above two hypothetical examples (Figs. 4.1 and 4.2) that there is a necessity to apply a more versatile metric for quality assessment in specialised road mapping applications such as AutoMap. In each example, identical error values for each case resulting from applying conventional methods (simple Euclidean distance and trace/determinant respectively) suggests that these isotropic metrics are not sufficiently effective in determining the map quality under similar circumstances. As a result, strategies to integrate directional priorities in the calculation of map quality metrics need to be further examined. This issue will be addressed in detail later in this chapter.

4.4.3 Type Priorities

In addition to directional priorities, there are other complementary factors that can be capitalised in determining the quality for a given map of road signs. One main factor which is considered in this thesis is the classification of the sign type (i.e. speed, speed, caution, no entry, etc.). Practically speaking, different sign types are generally regarded (or weighted) differently in terms of localisation accuracy[9]. Consider a map populated by a large number of road signs in which different sign types are present. The overall error associated with the map would typically be the sum of error for individual landmarks. We argue that, for applications (like AutoMap) where different kinds of landmarks are perceived differently, this may not be the most sensible approach to quantify the error. We propose a weighting concept for each type of landmark depending on their contribution to the total map error. In this text, we use the term *sign type priorities* to refer to the above concept and we further elaborate its application in Section 4.5.

By virtue of the above practical factors, the need for a more sophisticated, meaningful metric can be justified. In Section 4.5, we introduce a well-defined metric that can be utilised by the scientific and business communities to inclusively evaluate the

[9]Refer to Subsection 4.4.1 for the fundamental differences between regulatory and advisory signs.

quality of a given map in specialised road mapping applications. The intention is to later incorporate the new quality metric in multi-vehicle SLAM scenarios addressed in Chapter 3. The remaining of this section is concerned with addressing some of the other practical considerations in the design and theoretical development of this new metric.

4.4.4 Information Accessibility

As discussed earlier, there are various preliminary factors that are needed to be considered when designing a metric for map quality assessment in real-world applications. This subsection will address the information accessibility concept for constructing a metric for measuring the quality of maps comprised of road signs.

Generally speaking, the proposed metric will have to be a precisely defined mathematical concept which can be used with respect to the requirements of particular mapping applications. Therefore, it is imperative to define a metric which is computable based on the available information. This is the direct follow up from the discussion provided on taxonomy of quality metrics in Section 4.3. Intuitively, one should develop a metric which is based on the specifications of the available map. For example, a measure based on the position of the vehicle would clearly not make sense for a map which only consists of position estimates of map elements.

In this thesis, two main spatial attributes are considered for road signs. These attributes are position and orientation of the sign with respect to the Cartesian frame of reference (in other words, we consider the *pose* attribute for a given road sign). To clarify the orientation concept, we provide the following definition.

Definition 1. *Under the fairly reasonable assumption that road signs have a flat surface, the sign orientation in defined as the counter-clock-wise angle between the normal vector to the sign's surface and the x-y plane in the global Cartesian frame. Therefore, orientation of a sign is the same as the orientation of the vector perpendicular to the sign's surface and is defined in the* $(-\pi, \pi]$ *interval.*

Note that the sign orientation definition is based on the assumption that the majority of signs have a flat surface and are installed in a way that they are orthogonal to the ground's surface. This, in turn, makes the sign's normal vector parallel to the road's surface. As a result of these assumptions, a unique, well-defined orientation can be defined for any given sign (see Figure 4.3).

In addition to the main spatial parameters (position and orientation), sign type is also considered to characterise a given road sign. The implemented computer vision algorithms used to detect road signs of interest from video footages are capable of identifying the type of the extracted signs. Broadly speaking, these three parameters are sufficient to describe a road sign for the purpose of the application addressed here.

Throughout the remaining of this chapter we concentrate on maps of road signs comprising of 2-D position and orientation estimates. Each estimate is represented using a mean and a corresponding covariance. We further assume that the true sign

Figure 4.3: Sign Orientation is defined with respect to the normal vector perpendicular to its surface.

type is also available for the existing signs in the map. Consequently, based on the availability information and the nature of the mapping application, the following different options can be nominated for use in the calculation of the new map quality metric (sign type information has been excluded in this categorisation):

1. Measures that depend only on position
2. Measures that depend only on pose
3. Measures that depend only on position and its variance
4. Measures that depend only on pose and its variance
5. Measures that depend only on position variance
6. Measures that depend only on pose variance

Presuming the availability of the above statistics, the decision on the choice of the quality metric depends on which type of map measure is more appropriate for our specific road application. For instance, on one level, a measure which is characterised by position-only estimates might be the kind of map quality factor that one wants to define, and on another level, a quality metric based on pose, rather than just position, might be more appealing and suitable for a specific purpose.

As argued before in this section, measures based on position-only estimates are not deemed sufficient to correctly reflect the quality of a map encompassing road signs due to the existence of the inherited directional priorities. Therefore, a logical option would be to somehow incorporate the road heading information in the calculation of the new map error metric. Nevertheless, this information is generally not readily accessible as it is not directly presented in the output of most mapping techniques. This stems from the fact that the majority of map representations are essentially used to describe different aspects of the existing map elements, rather than other environmental objects. Note that, in theoretical sense, the SLAM algorithm produces the track of the vehicle as well as the position of the detected features. However, the only deliverable piece of information for most mapping applications is merely the information of map elements. This restriction is particularly valid for map creation in large-scale environments where maintaining all the vehicle's positioning information is not feasible due to the size of the problem. Therefore, an alternative must be sought to replace the road and vehicle heading information in developing the new quality metric.

As a general rule, in the world of road signs, each sign is installed with respect to a stretch of road. We make an assumption that even though any given road has a finite width, it is defined according to the centre-line of its lanes. Therefore, a single line is associated to each road regardless of its width. For a typical road network, the heading of the road varies from point to point, thus making it difficult to mathematically define a fixed heading for a road stretch. Therefore, for our particular road mapping application, we define a point-wise heading depending on where this quantity is being measured.

Definition 2. *The heading at any particular point of a given road is defined as the angle between the road tangent at that point and the global Cartesian frame. Road heading is denoted using β and is defined in the interval $[0, \pi)$.*

More specifically, we purposely define the 'road heading for a specific road sign' as the heading of the road at the intersection point of the road and the sign's surface. A numerical example of the point-wise road heading concept has been shown in Figure 4.4.

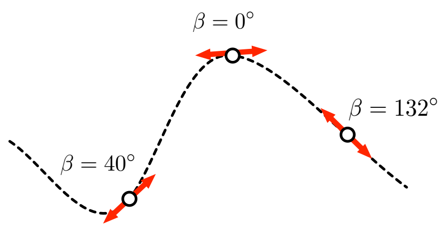


Figure 4.4: Point-wise road heading ($0 \leq \beta < \pi$).

As can be tentatively deducted from Definitions 1 & 2 , sign orientation and road heading are closely related concepts. This is mainly due to the fact that, for the majority of signs, the sign's flat surface is approximately orthogonal to the stretch of the road they apply to[10]. Thus, for a given sign[11], the sign normal vector is roughly

[10]Since the imaginary horizontal continuation of most signs' surfaces (form the sides) intersects the road at a 90° degree.

[11]Note that some sign types (e.g. one way, parking, etc.) are parallel to the road. Without loss

parallel to the stretch of road and opposite to the direction in which the vehicles traverse that road (Figure 4.5 clarifies this concept). Consequently, we propose the use of sign orientation (sign normal) in lieu of the road heading (as the closest accessible measure) in developing the new map error metric. This will effectively lead to the creation of a non-isotropic metric, as it will be dependant on the spatial orientation of the landmarks (see Section 4.3).

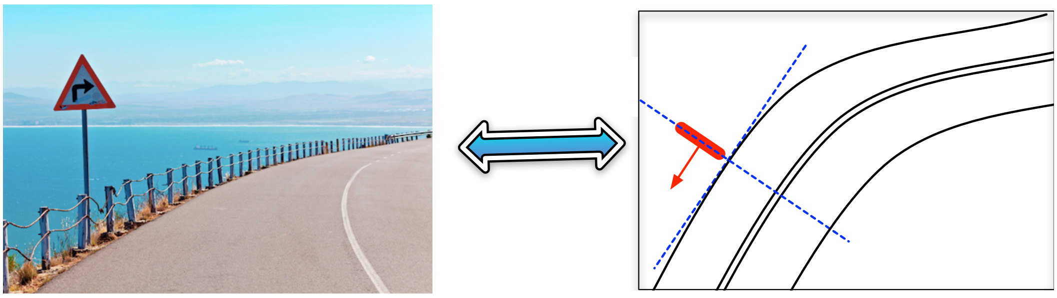


Figure 4.5: The surface of a sample road sign is usually orthogonal to the portion of the road it applies to, which in turn makes the sign's surface normal (sign orientation) parallel to the road's tangent.

The advantages of estimating the landmark orientation for road mapping applications is two-fold. Firstly, as discussed recently, the orientation integration can potentially facilitate the design of a compelling, more extensive map quality metric which is dependant on the previously mentioned directional priorities. Secondly, delivering maps consisting of pose estimates can offer certain practical benefits to mapping companies. For instance, Sensis, AutoMap's key Australian customer, can use and interpret the sign orientation information in a non-trivial procedure in their organisational pipeline (called the integration process) to obtain some partial knowledge about the geometry of roads and the surrounding environment. For example, based on the pose estimate and the type of a particular road sign, the structure of the roads pertaining to that sign can be inferred. It is important to note that finding the road corresponding to a road sign is not a trivial problem in general, specially for speed, give way and round-about signs.

This section outlined some of the most pivotal practical points needed to be considered for the design of a versatile metric for quality assessment in road mapping applications. In the next section we will introduce a new measure called *DIMER metric* based on the type of the existing map elements and the directional priorities with respect to the road geometry. The term 'DIMER' is created as an acronym of the phrase "DIrectional Map ERror".

of generality, we exclude these signs from our discussion. However, the road heading can be easily determined using the sign orientation and type for this minority. In addition, all the past and future equations can be re-derived for these signs.

4.5 Directional Map Error (DIMER) Metric

In this section we design a directional map quality metric which integrates the landmark's type information and directional priorities into the calculation of the map error metric. Two distinct cases based on the availability of ground-truth information are considered. Firstly, a preliminary quality measure based on mean-only estimates is formulated in Section 4.5.1 for the case where the true landmark locations are available. Secondly, a map quality metric based on variance-only estimates is defined in Section 4.5.2 for scenarios where the true mean values are not available. Please note that assessing the map quality is performed by calculating a form of map error in this chapter. Therefore, the concepts of 'map quality metric' and 'map error metric' are closely related. Needless to say, the lower the map error, the higher the quality and accurateness of that map.

Similar to the previous section, we choose to focus on road signs as the main landmarks of interest for this section. Despite this, the discussion presented here can be generalised and extended to other robotic applications where a more meaningful, advanced quality metric is required.

4.5.1 Ground-truth-based DIMER Metric

Consider a two-dimensional map consisting of N landmarks (e.g. N road signs). Let $\mathbf{X}_j^*$ be the vector of true position and $\hat{\mathbf{X}}_j$ the vector of position estimates for the j^{th} landmark in the map. If all the N landmarks presented in the map are identical[12] (homogenous) and the localisation importance for every landmark is the same along any arbitrary axis (there are no directional preferences), then the computation of the total map error measure ($\mathcal{M}$) for this given map is straightforward and can be calculated using[13]

$$\mathcal{M}_G = \sum_{j=1}^{N} \|\hat{\mathbf{X}}_j - \mathbf{X}_j^*\|^2 \tag{4.2}$$

where subscript G refers to the fact that the calculated map error is based on ground-truth data. As can be seen, $\mathcal{M}_G$ is simply the sum of square[14] of distance between the estimate and the ground-truth for individual landmarks (See Fig. 4.6 for a sample landmark in the map). This combined Euclidean distance is obviously the most viable metric for evaluating the quality of the map in this situation.

Now assume a realistic scenario where the landmarks in the map are of different types (non-homogenous map) and the localisation accuracy for each type is of different practical importance. As discussed in Section 4.4, in such cases, individual landmark's errors are weighted differently in the overall map error calculation

[12] Identical in this sense means that the landmarks in the map are of the same type and class.

[13] One can also consider the average error over N landmarks. However, we consider the total map error in our calculations.

[14] We remark that the use of squared error values makes the calculations analytically more tractable. Therefore, they are used in most of the formulations provided in this chapter.

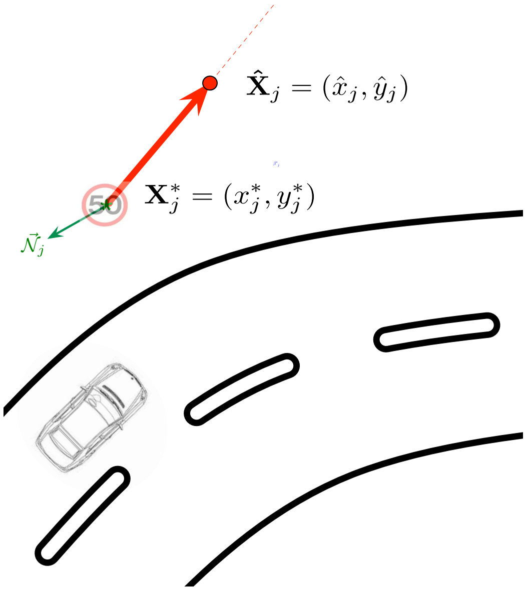


Figure 4.6: The true and estimated positions for a sample landmark j.

depending on their respective classification. We propose the introduction of a weighting factor ω for each type of landmark depending on their localisation importance. Larger weighting ω_j corresponds to higher localisation importance and therefore more contribution towards the overall map error. These weights are adjustable and can be defined in different ways for different applications based on their specific requirements[15]. A simplistic setup may choose to define $\omega \in [0, 1]$ where 0 and 1 indicate the lowest and highest priorities respectively. The weighting factors can be incorporated into the map error measure by revising Equation (4.2) according to

$$\mathcal{M}_G = \sum_{j=1}^{N} \omega(\mathfrak{T}_j) \|\hat{\mathbf{X}}_j - \mathbf{X}^*_j\|^2 \tag{4.3}$$

This equation uses the weighted sum of the squared errors pertaining to individual landmarks to compute the total map error metric. Notation $\mathfrak{T}_j$ is used to denote the type of the j^{th} landmark. As can be seen, the weighting factor ω for a landmark is a function of its type.

In order to complete the design of the new map quality metric for this case, we now need to integrate the previously discussed directional priorities. Suppose the directional importance factors for sign localisation are denoted by $\mathcal{I}_{\parallel}{}^j$ for parallel and $\mathcal{I}_{\perp}{}^j$ for perpendicular directions with respect to the road[16]. The relative importance

[15]Note that the weighting factor for a particular type of landmark is defined relatively with respect to the localisation importance of other groups of landmarks.

[16]In this chapter, the terms parallel direction, perpendicular direction and directional priorities are all defined with respect to the heading of the road. Therefore, for brevity and to avoid repetition, we do not mention this relatively with respect to the road most of the time.

factors are deliberately chosen under the following constraint:

$$\mathcal{I}_{\|}{}^{j} + \mathcal{I}_{\perp}{}^{j} = 2 \tag{4.4}$$

For example, for a hypothetical situation where the map user is only concerned with the vertical error with respect to the road, $\mathcal{I}_{\perp}$ and $\mathcal{I}_{\|}$ are set equal to 2 and 0 respectively, meaning that the parallel errors are neglected in the calculation of the directional map error metric. Note that these importance factors can be separately chosen for each landmark based on specific mapping profiles for any criteria-based mapping system. For applications with no directional preference, $\mathcal{I}_{\|}$ and $\mathcal{I}_{\perp}$ are both set equal to unity.

Let β_j^* denote the true heading of the road in which landmark j applies to ($0 < \beta_j^* < \pi$). The error vector in Fig. 4.6 can be projected into components along the road's parallel (x'') and perpendicular (y'') axes in the Cartesian Coordinate System (see Figure 4.7). Through this orthogonal decomposition, the directional priorities $\mathcal{I}_{\|}$ and $\mathcal{I}_{\perp}$ can be incorporated into Equation (4.3). These error projections are called the parallel ($\mathcal{E}_{\|}^{j}$) and perpendicular ($\mathcal{E}_{\perp}^{j}$) error metrics and are simply calculated by using the inner product properties as below:

$$\mathcal{E}_{\|}^{j} = \mathcal{I}_{\|}{}^{j}\omega(\mathfrak{T}_j)\|\hat{\mathbf{X}}_j - \mathbf{X}_j^*\|^2\cos^2(\alpha_j - \beta_j^*) \tag{4.5}$$

$$\mathcal{E}_{\perp}^{j} = \mathcal{I}_{\perp}{}^{j}\omega(\mathfrak{T}_j)\|\hat{\mathbf{X}}_j - \mathbf{X}_j^*\|^2\sin^2(\alpha_j - \beta_j^*) \tag{4.6}$$

where $-\pi < \alpha < \pi$ is the angle of the error vector with respect to the Cartesian frame (xy-plane) and is calculated according to

$$\alpha_j = \arctan\left(\frac{\hat{y}_j - y_j^*}{\hat{x}_j - x_j^*}\right) \tag{4.7}$$

Consequently, the revised total map error measure (after taking into account the directional priorities for each directional component) for a map comprising of N non-homogenous landmarks becomes

$$\begin{aligned}\mathcal{M}_G = &\left(\sum_{j=1}^{N}\mathcal{C}_{\|}^{j}\|\hat{\mathbf{X}}_j - \mathbf{X}_j^*\|^2\cos^2(\alpha_j - \beta_j^*)\right)\\ &+\left(\sum_{j=1}^{N}\mathcal{C}_{\perp}^{j}\|\hat{\mathbf{X}}_j - \mathbf{X}_j^*\|^2\sin^2(\alpha_j - \beta_j^*)\right)\end{aligned} \tag{4.8}$$

where $\mathcal{C}_{\|}^{j}$ and $\mathcal{C}_{\perp}^{j}$ are defined by

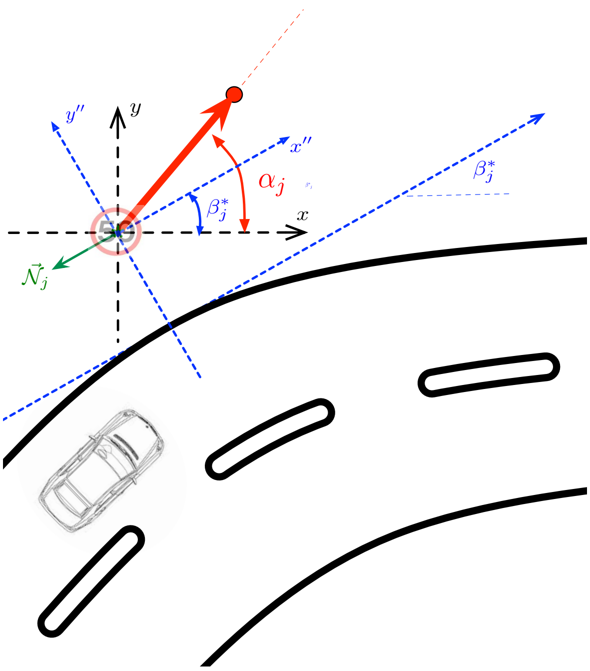


Figure 4.7: The error vector of Fig. 4.6 is decomposed into components along the road's parallel (x'') and perpendicular (y'') axes.

$$\mathcal{C}_{\parallel}^{j} \triangleq \mathcal{I}_{\parallel}{}^{j}\omega(\mathfrak{T}_j) \tag{4.9}$$

$$\mathcal{C}_{\perp}^{j} \triangleq \mathcal{I}_{\perp}{}^{j}\omega(\mathfrak{T}_j). \tag{4.10}$$

and can be adjusted for any arbitrary sign j depending on different map users and profiles. Equation (4.8) can alternatively be written as[17]

$$\mathcal{M}_G = \sum_{j=1}^{N} \left(\mathcal{E}_{\parallel}^{j} + \mathcal{E}_{\perp}^{j} \right) \tag{4.11}$$

4.5.2 Covariance-based DIMER Metric

This subsection formulates a directional map error (DIMER) metric for situations where no ground-truth information is available. As previously explained in this chapter, the covariance matrix of position estimates can be used in this case for deriving a quantitative quality metric. Once again, we examine the quality of a map consisting of N road signs. Consider a scenario where the position and orientation of the existing road signs have been estimated through the use of a stochastic filtering algorithm. For the sake of this discussion and without loss of generality, we presume

[17]Once again we remark that one might consider the average error over N landmarks.

that the position and orientation estimates of any road sign are not correlated, thus may have been obtained separately. Similar to the previous subsection, we first derive the directional error for a single landmark before formulating the DIMER metric for the full map.

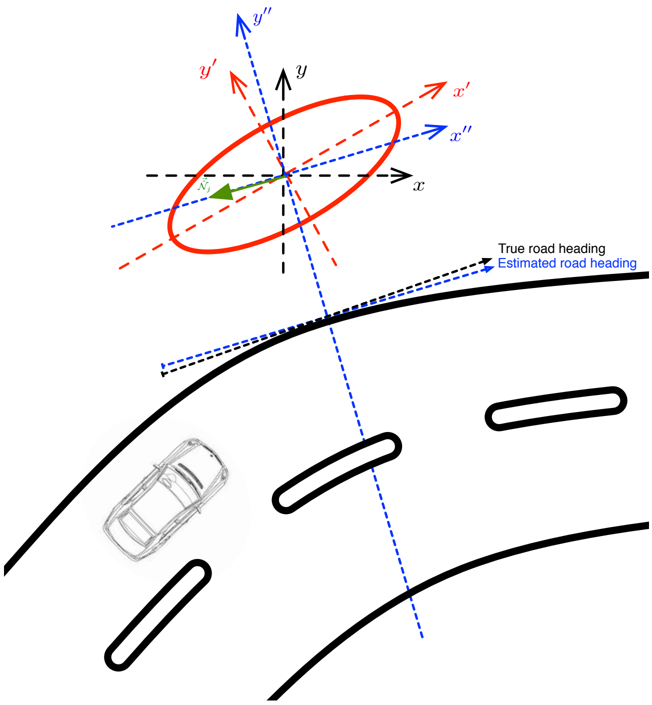


Figure 4.8: The estimated position mean and its associated covariance ellipse for a sample road sign. The road heading is estimated based on the landmark orientation estimation ($\vec{\mathcal{N}}_j$).

Consider a sample landmark j in the map as depicted in Figure 4.8. The estimated position mean and its covariance ellipse for this road sign can be seen in this figure along with the relative stretch of road. Moreover, the estimated sign orientation has been demonstrated using a normal vector ($\vec{\mathcal{N}}_j$) in the figure. As previously discussed in this chapter, an estimate of the road heading can be inferred using the sign orientation information, since the road sign's surface is perpendicular to the road at their intersection point. We denote the estimated road heading using $\hat{\beta}_j$. Now suppose $\hat{\mathbf{X}}^j$ and $\mathbf{P}^j$ be the mean and its associated covariance matrix pertaining to the position estimate of landmark j, then

$$\hat{\mathbf{X}}^j = [\hat{x}^j, \hat{y}^j]^T; \tag{4.12}$$

$$\mathbf{P}^j = \begin{bmatrix} P^j_{xx} & P^j_{xy} \\ P^j_{xy} & P^j_{yy} \end{bmatrix} = \begin{bmatrix} {\sigma^j_x}^2 & \sigma^j_{xy} \\ \sigma^j_{xy} & {\sigma^j_y}^2 \end{bmatrix} \tag{4.13}$$

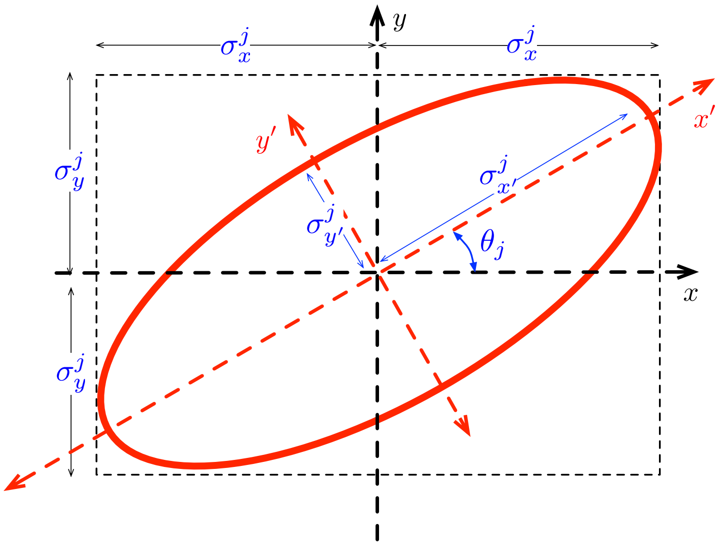


Figure 4.9: A standard error ellipse used to visualise a general 2-D covariance matrix such as the one provided in Eq. (4.13). The semi-major and semi-minor axes of the ellipse are labelled $\sigma_{x'}^j$ and $\sigma_{y'}^j$ respectively.

where σ_x and σ_y denote the standard deviation while σ_{xy} is the associated covariance of the position estimates in (4.12). The following relationship holds between these variables:

$$\sigma_{xy} = \rho_{xy} \cdot \sigma_x \cdot \sigma_y \tag{4.14}$$

with $\rho_{xy} \in [-1, 1]$ being the correlation coefficient.

In principle, covariance matrices can be geometrically represented by elliptical distributions. More specifically, in two-dimensional space, a covariance matrix like $\mathbf{P}^j$ can be visualised by a standard error ellipse such as the one shown in Fig. 4.9. Please note that in many estimation applications, the 2σ or 3σ error ellipses are used in lieu of the standard (1σ) ellipse. However, for the current discussion around map quality metrics, we concentrate on the latter case with no scaling coefficient for the standard deviation.

As a result of the principal axis theorem[18], the eigenvectors of the covariance matrix $\mathbf{P}^j$ define the principal axes of the standard error ellipse shown in Fig. 4.9. These principal axes are referred to as major and minor axes and are shown using two perpendicular axes x' and y' in the figure. The eigenvalues of the $\mathbf{P}^j$ matrix are directly related to the length of the principal axes in that they are equal to the squares of the semi-axes $\sigma_{x'}^j$ and $\sigma_{y'}^j$. Therefore,

[18]See [Mornhinweg et al., 1993] for more details on this theorem.

$$\lambda_x^j = {\sigma_{x'}^j}^2 \tag{4.15}$$
$$\lambda_y^j = {\sigma_{y'}^j}^2 \tag{4.16}$$

where $\lambda_x^j \geq \lambda_y^j \geq 0$ is implied by the obvious fact that $\sigma_{x'}^j \geq \sigma_{y'}^j$.

In a general error ellipse, the principal axes do not coincide with the Cartesian Coordinate axes (x and y), thus forming an angle with respect to it. This angle θ_j (known as the error ellipse orientation) is mathematically defined as the angle between the major axis of the ellipse (x') and the x-axis and can be calculated by

$$\theta_j = \frac{1}{2}\arctan\left(\frac{2\sigma_{xy}^j}{{\sigma_x^j}^2 - {\sigma_y^j}^2}\right) \quad , \quad \theta_j \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right] \tag{4.17}$$

Alternatively, θ_j can be written as

$$\theta_j = \frac{1}{2}\arctan\left(\frac{2P_{xy}^j}{P_{xx}^j - P_{yy}^j}\right) \quad , \quad \theta_j \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right] \tag{4.18}$$

Furthermore, the semi-major ($\sigma_{x'}^j$) and semi-minor ($\sigma_{y'}^j$) axes of the standard error ellipse are linked to the elements of the covariance matrix in (4.13) through

$${\sigma_{x'}^j}^2 = \left(\frac{P_{xx}^j + P_{yy}^j}{2} + \sqrt{\left(\frac{P_{xx}^j - P_{yy}^j}{2}\right)^2 + {P_{xy}^j}^2}\right) \tag{4.19}$$
$${\sigma_{y'}^j}^2 = \left(\frac{P_{xx}^j + P_{yy}^j}{2} - \sqrt{\left(\frac{P_{xx}^j - P_{yy}^j}{2}\right)^2 + {P_{xy}^j}^2}\right) \tag{4.20}$$

The first term in the above equations is the trace of the $\mathbf{P}^j$ matrix. It is mathematically well known that the trace of a positive semi-definite (PSD) matrix is equivalent to the sum of the matrix's eigenvalues. Therefore, Equations (4.15) and (4.16) can be rearranged as:

$${\sigma_{x'}^j}^2 = \frac{1}{2}\mathbf{Tr}(\mathbf{P}^j) + \frac{1}{2}\mathbf{Sp}(\mathbf{P}^j) \tag{4.21}$$
$${\sigma_{y'}^j}^2 = \frac{1}{2}\mathbf{Tr}(\mathbf{P}^j) - \frac{1}{2}\mathbf{Sp}(\mathbf{P}^j) \tag{4.22}$$

where $\mathbf{Sp}(\mathbf{P}^j)$ is called the spread of matrix $\mathbf{P}^j$ and is used to describe the maximum distance between the matrix's eigenvalues in the complex plane. Therefore,

$$\mathbf{Sp}(\mathbf{P}^j) = \lambda_x^j - \lambda_y^j \tag{4.23}$$

Given the above preliminaries on the nature of the covariance error ellipse, we now formulate the new covariance-based directional map error metric.

Calculating the parallel and perpendicular components of the error ellipse is slightly more subtle than the calculations in Subsection 4.5.1. However, in a similar way to Equations (4.5) and (4.6), the semi-major and semi-minor axes of the error ellipse can be decomposed into parallel and perpendicular components with respect to the heading of the road ($\hat{\beta}_j$). The enlarged view of the covariance ellipse presented in Fig. 4.8 can be observed in Figure 4.10. The principal axes of the ellipse have been marked with x' and y', while the road axes are denoted x'' and y''. The projections of the ellipse's axes onto the road's axes are calculated and rearranged to form the parallel $\mathcal{E}^j_{\parallel}$ and perpendicular $\mathcal{E}^j_{\perp}$ errors as below:

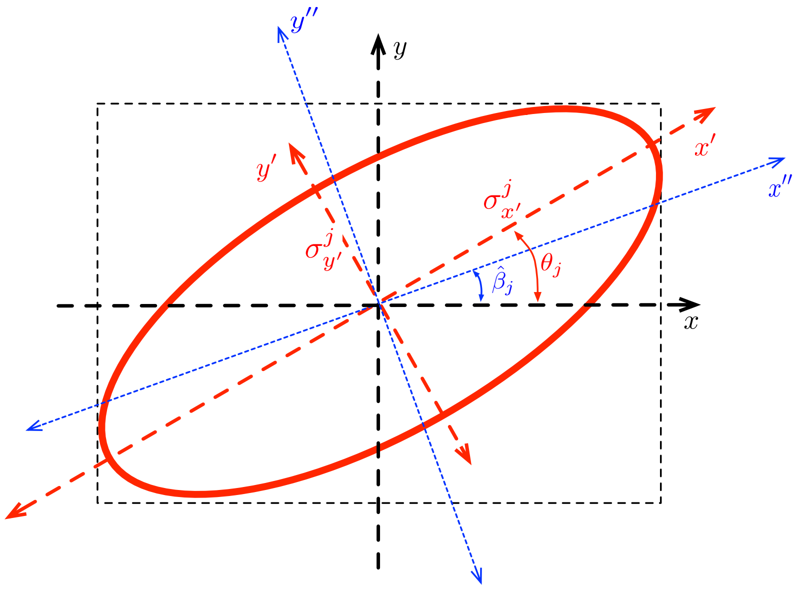


Figure 4.10: Projecting the semi-major and semi-minor axes of the ellipse to an arbitrary axis x'' with orientation β.

$$\mathcal{E}^j_{\parallel} \triangleq \mathcal{C}^j_{\parallel}\left(\sigma^2_{x'_j}\cos^2(\theta_j-\hat{\beta}_j)+\sigma^2_{y'_j}\sin^2(\theta_j-\hat{\beta}_j)\right) \tag{4.24}$$

$$\mathcal{E}^j_{\perp} \triangleq \mathcal{C}^j_{\perp}\left(\sigma^2_{x'_j}\sin^2(\theta_j-\hat{\beta}_j)+\sigma^2_{y'_j}\cos^2(\theta_j-\hat{\beta}_j)\right). \tag{4.25}$$

where, as before, sign-specific variables $\mathcal{C}^j_{\parallel}$ and $\mathcal{C}^j_{\perp}$ are defined by

$$\mathcal{C}^j_{\parallel} \triangleq \mathcal{I}_{\parallel}{}^j\omega(\mathfrak{T}_j) \tag{4.26}$$

$$\mathcal{C}^j_{\perp} \triangleq \mathcal{I}_{\perp}{}^j\omega(\mathfrak{T}_j). \tag{4.27}$$

Substituting Equations (4.21) and (4.22) in Eq. (4.24) gives

$$
\begin{aligned}
\mathcal{E}_{\parallel}^{j} &= \mathcal{C}_{\parallel}^{j}\left(\frac{1}{2}\mathbf{Tr}(\mathbf{P}^{j}) + \frac{1}{2}\mathbf{Sp}(\mathbf{P}^{j})\right)\cos^{2}(\theta_{j} - \hat{\beta}_{j}) \\
&+ \mathcal{C}_{\parallel}^{j}\left(\frac{1}{2}\mathbf{Tr}(\mathbf{P}^{j}) - \frac{1}{2}\mathbf{Sp}(\mathbf{P}^{j})\right)\sin^{2}(\theta_{j} - \hat{\beta}_{j})
\end{aligned} \tag{4.28}
$$

Rearranging the above equation and using trigonometric equalities yields

$$
\mathcal{E}_{\parallel}^{j} = \mathcal{C}_{\parallel}^{j}\left(\frac{1}{2}\mathbf{Tr}(\mathbf{P}^{j}) + \frac{1}{2}\mathbf{Sp}(\mathbf{P}^{j})\cos(2\theta_{j} - 2\hat{\beta}_{j})\right) \tag{4.29}
$$

The perpendicular error ($\mathcal{E}_{\perp}^{j}$) is also derived in a similar way as:

$$
\mathcal{E}_{\perp}^{j} = \mathcal{C}_{\perp}^{j}\left(\frac{1}{2}\mathbf{Tr}(\mathbf{P}^{j}) - \frac{1}{2}\mathbf{Sp}(\mathbf{P}^{j})\cos(2\theta_{j} - 2\hat{\beta}_{j})\right) \tag{4.30}
$$

On the other hand, reorganising the error ellipse orientation in Eq. (4.18) gives

$$
\tan(2\theta_{j}) = \frac{2P_{xy}^{j}}{P_{xx}^{j} - P_{yy}^{j}} \qquad \theta_{j} \in (-\pi, \pi] \tag{4.31}
$$

Substituting the above equation into the following trigonometry equalities

$$
\sin(2\theta_{j}) = \pm\frac{\tan(2\theta_{j})}{\sqrt{1 + \tan^{2}(2\theta_{j})}} \quad , \quad \cos(2\theta_{j}) = \pm\frac{1}{\sqrt{1 + \tan^{2}(2\theta_{j})}}
$$

yields[19]

$$
\sin(2\theta_{j}) = \frac{2P_{xy}^{j}}{\mathbf{Sp}(\mathbf{P}^{j})} \tag{4.32}
$$

$$
\cos(2\theta_{j}) = \frac{P_{xx}^{j} - P_{yy}^{j}}{\mathbf{Sp}(\mathbf{P}^{j})} \tag{4.33}
$$

Substituting Equations (4.32) and (4.33) into Equations (4.29) and (4.30) and simplifying yields:

$$
\mathcal{E}_{\parallel}^{j} = \mathcal{C}_{\parallel}^{j}\left((\frac{1 + \cos(2\hat{\beta}_{j})}{2})P_{xx}^{j} + (\frac{1 - \cos(2\hat{\beta}_{j})}{2})P_{yy}^{j} + \sin(2\hat{\beta}_{j})P_{xy}^{j}\right) \tag{4.34}
$$

$$
\mathcal{E}_{\perp}^{j} = \mathcal{C}_{\perp}^{j}\left((\frac{1 - \cos(2\hat{\beta}_{j})}{2})P_{xx}^{j} + (\frac{1 + \cos(2\hat{\beta}_{j})}{2})P_{yy}^{j} - \sin(2\beta_{j})P_{xy}^{j}\right) \tag{4.35}
$$

[19] The quadrant in which angle 2θ falls into is significant in specifying the sign of the derived sine and cosine functions in Equations (4.32) and (4.33) from the above trigonometric equations.

On the other hand, the total map error metric for landmark j is obtained by adding the parallel and perpendicular errors:

$$\mathcal{M}_C^j = \mathcal{E}_{\parallel}^j + \mathcal{E}_{\perp}^j \tag{4.36}$$

Therefore,

$$\begin{aligned}\mathcal{M}_C^j(\mathbf{P}^j) = \quad & \mathcal{C}_{\parallel}^j \quad \left((\frac{1+\cos(2\hat{\beta}_j)}{2})P_{xx}^j + (\frac{1-\cos(2\hat{\beta}_j)}{2})P_{yy}^j + \sin(2\hat{\beta}_j)P_{xy}^j \right) + \\ + \quad & \mathcal{C}_{\perp}^j \quad \left((\frac{1-\cos(2\hat{\beta}_j)}{2})P_{xx}^j + (\frac{1+\cos(2\hat{\beta}_j)}{2})P_{yy}^j - \sin(2\hat{\beta}_j)P_{xy}^j \right)\end{aligned} \tag{4.37}$$

where subscript C refers to the fact that the calculated error is based on the covariance matrix of the estimate. Alternatively, (4.37) can be written as

$$\mathcal{M}_C^j(\mathbf{P}^j) = \mathbf{Tr}(\mathbf{C}^j\mathbf{P}^j) \tag{4.38}$$

where

$$\mathbf{C}^j = \begin{bmatrix} C_{11}^j & C_{12}^j \\ C_{12}^j & C_{22}^j \end{bmatrix}$$

with

$$\begin{aligned} C_{11}^j &= \mathcal{C}_{\parallel}^j \left(\frac{1+\cos(2\hat{\beta}_j)}{2} \right) + \mathcal{C}_{\perp}^j \left(\frac{1-\cos(2\hat{\beta}_j)}{2} \right) \\ C_{22}^j &= \mathcal{C}_{\parallel}^j \left(\frac{1-\cos(2\hat{\beta}_j)}{2} \right) + \mathcal{C}_{\perp}^j \left(\frac{1+\cos(2\hat{\beta}_j)}{2} \right) \\ C_{12}^j &= \frac{1}{2}(\mathcal{C}_{\parallel}^j - \mathcal{C}_{\perp}^j)\sin(2\hat{\beta}_j) \end{aligned}$$

Consequently, the DIMER metric for a given landmark j with the estimated position covariance matrix $\mathbf{P}^j$ can be written as the trace of a weighted covariance matrix where the weighting factor depends on factors such as the estimated heading of the road, directional priorities and the sign type.

The total directional map error (DIMER) metric for a map comprising of N landmarks is then calculated by adding the error of individual landmarks in the map according to:

$$\mathcal{M}_C(\mathbf{P}^1, \ldots, \mathbf{P}^n) = \sum_{j=1}^{N} \mathcal{M}_C^j(\mathbf{P}^j) \tag{4.39}$$

Therefore,

$$\begin{aligned}\mathcal{M}_C(\mathbf{P}^1,\ldots,\mathbf{P}^n) &= \sum_{j=1}^{N}\mathcal{C}_{\|}^{j}\left((\frac{1+\cos(2\hat{\beta}_j)}{2})P_{xx}^{j}+(\frac{1-\cos(2\hat{\beta}_j)}{2})P_{yy}^{j}+\sin(2\hat{\beta}_j)P_{xy}^{j}\right)+\\ &+ \sum_{j=1}^{N}\mathcal{C}_{\perp}^{j}\left((\frac{1-\cos(2\hat{\beta}_j)}{2})P_{xx}^{j}+(\frac{1+\cos(2\hat{\beta}_j)}{2})P_{yy}^{j}-\sin(2\beta_j)P_{xy}^{j}\right)\end{aligned} \tag{4.40}$$

Or alternatively,

$$\mathcal{M}_C(\mathbf{P}^1,\ldots,\mathbf{P}^n)=\sum_{j=1}^{N}\mathbf{Tr}(\mathbf{C}^j\mathbf{P}^j) \tag{4.41}$$

One may also consider the average error over the whole number of landmark (N) according to

$$\mathcal{M}_{C,avg}(\mathbf{P}^1,\ldots,\mathbf{P}^n)=\frac{1}{N}\sum_{j=1}^{N}\mathbf{Tr}(\mathbf{C}^j\mathbf{P}^j) \tag{4.42}$$

We now provide the following important propositions regarding the directional map error metric $\mathcal{M}_C(\mathbf{P}^j)$ in Eq. (4.38).

Proposition 1. *For a given directional error measure $\mathcal{M}_C^j(\mathbf{P}^j)$ associated with landmark j and given positive coefficients $\mathcal{C}_{\|}^j$ and $\mathcal{C}_{\perp}^j$ such that $\mathcal{C}_{\|}^j=\mathcal{C}_{\perp}^j=c_j$ with $c_j>0$, all covariance matrices $\mathbf{P}^j$ that satisfy* (4.37) *have the same trace and $\mathbf{Tr}(\mathbf{P}^j)=\mathcal{M}_C^j/c_j$.*

Proof. Let $\mathcal{C}_{\|}^j=\mathcal{C}_{\perp}^j=c_j$, (4.37) becomes

$$\begin{aligned}\mathcal{M}_C^j(\mathbf{P}^j) &= c_j\left((\frac{1+\cos(2\hat{\beta}_j)}{2})P_{xx}^{j}+(\frac{1-\cos(2\hat{\beta}_j)}{2})P_{yy}^{j}+\sin(2\hat{\beta}_j)P_{xy}^{j}\right)\\ &+c_j\left((\frac{1-\cos(2\hat{\beta}_j)}{2})P_{xx}^{j}+(\frac{1+\cos(2\hat{\beta}_j)}{2})P_{yy}^{j}-\sin(2\hat{\beta}_j)P_{xy}^{j}\right)\\ &= c_j(P_{xx}^j+P_{yy}^j)\\ &= c_j\mathbf{Tr}(\mathbf{P}^j).\end{aligned}$$

Thus $\mathbf{Tr}(\mathbf{P}^j)=\mathcal{M}_C^j(\mathbf{P}^j)/c_j$. □

The above proposition establishes the link between the newly designed DIMER metric and the existing well-known measure based on trace of the covariance matrix. It demonstrates that in applications with no directional priorities for landmark positioning, the directional error metric transforms into a simple weighted trace. Moreover, it can be easily verified that for the special case where the parallel and perpendicular weights associated with the directional errors of the landmark are identical and equal to unity, the error measure will be transformed to the already-known trace of the covariance matrix $\mathbf{P}^j$.

Proposition 2. *The following statements hold for the map error measure $\mathcal{M}_C^j(\mathbf{P}^j)$ associated with landmark j and the covariance matrix $\mathbf{P}^j$ associated with the estimate of the position of j.*

1. *If $\hat{\beta}_j$ is either equal to 0 or π then*

$$\mathcal{M}_C^j(\mathbf{P}^j) = \mathcal{C}_{\|}^j P_{xx}^j + \mathcal{C}_{\perp}^j P_{yy}^j. \tag{4.43}$$

2. *If $\hat{\beta}_j = \pm\pi/2$ then*

$$\mathcal{M}_C^j(\mathbf{P}^j) = \mathcal{C}_{\|}^j P_{yy}^j + \mathcal{C}_{\perp}^j P_{xx}^j. \tag{4.44}$$

Proof. Statement (1) and (2) follow directly from evaluating (4.37) for $\hat{\beta}_j = 0$, $\hat{\beta}_j = \pi$ and $\hat{\beta}_j = \pi/2$. □

An example set of ellipses associated with covariance matrices that lead to the same $\mathcal{M}_C^j(\mathbf{P}^j)$ for a fixed $\mathcal{C}_{\|}^j$ and $\mathcal{C}_{\perp}^j$ under (4.43) where $\hat{\beta}_j = 0$ is depicted in Figure 4.11. The only dominant localisation direction in this example is the direction perpendicular to the heading of the road. Under these assumptions, counter-intuitively, all the demonstrated error ellipses have the same directional quality (see Fig. 4.12).

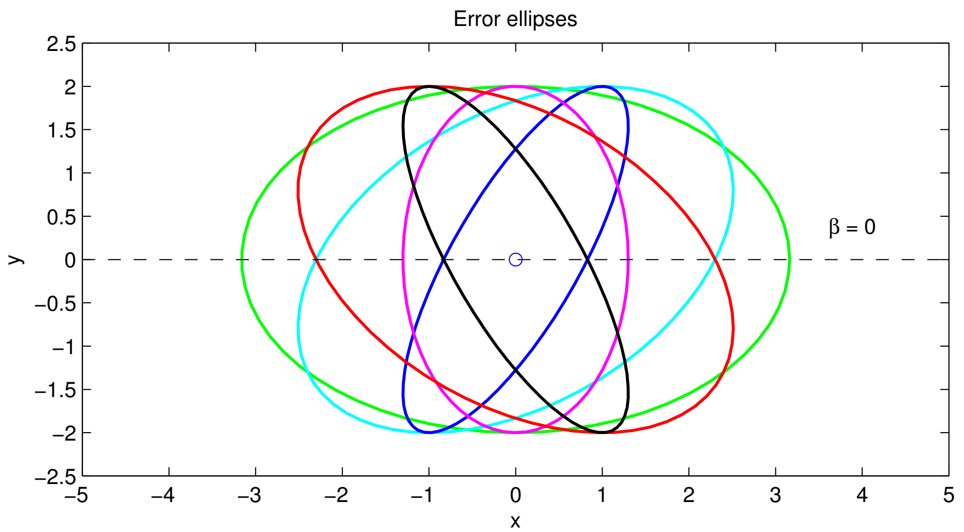
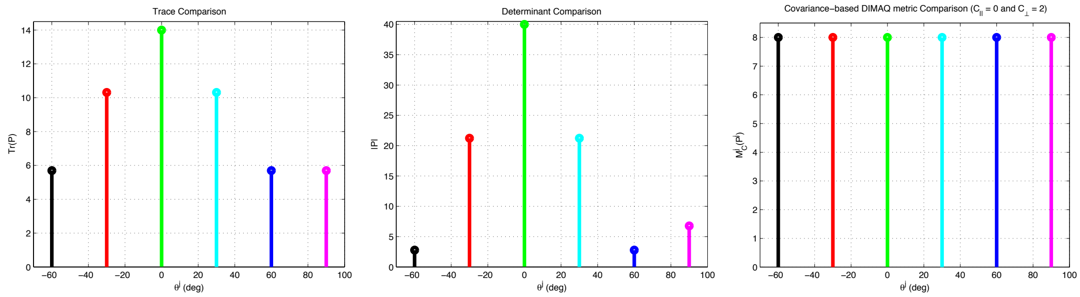


Figure 4.11: A set of covariance error ellipses that yield identical covariance-based DIMER metric, $\mathcal{M}_C^j(\mathbf{P}^j)$, for the case where the parallel component of the error is ignored, i.e. $C_{\|}^j = 0$ and $C_{\perp}^j = 2$. The heading of the road associated with the estimates is considered to be horizontal ($\beta^j = 0$).



(a) Trace comparison (b) Determinant comparison (c) DIMER comparison

Figure 4.12: Comparison of different error measures (trace, determinant and covariance-based DIMER metric) for the ellipses shown in Figure 4.11. The errors are plotted with respect to the ellipses' tilt angle θ^j.

The two propositions stated above show that the directional metric proposed in this section for quantitatively assessing the map quality is a more general case of the already-known trace metric. In other words, $\mathbf{Tr}(\mathbf{P}^j)$ is a special case of the metric described in (4.37). Consequently, the DIMER metric is a natural generalisation of the popular trace metric. A variety of examples will be provided in Chapter 6 to demonstrate the applicability and functionality of the proposed metric.

4.6 Summary

This chapter presented a review on some of the most popular techniques used to evaluate the quality of localisation and mapping applications. Some of the basic shortcomings associated with applying the existing techniques to specialised mapping problems were identified. The need for a more meaningful, versatile approach to quantitatively assess the quality of road mapping applications (such as AutoMap) was justified through demonstration of some practical examples.

A generic map error measure based on orientation and classification of map elements was devised for road signs which is deemed compelling for specialised road mapping applications. The specific requirements and expectations of different groups of map users regarding map quality metrics were studied and considered in the design process. The directional map error (DIMER) metric was developed for two fundamentally distinct cases. The first type of metric was designed for scenarios where the ground-truth information was accessible. The second metric was developed in the absence of information regarding the true map and was based on the covariance matrix of the position estimates. In contrast to the majority of existing map quality measures, the DIMER metrics is not isotopic, since the properties of the metric varies depending on the direction of the roads corresponding to the map elements.

Although most of the focus in this chapter revolved around maps comprising of road signs, different practical applications (particularly road related applications or applications with similar characteristics) can be inspired by this discussion. The devised metrics are applicable to a variety of mapping frameworks where different user groups employ different mapping profiles and localisation priorities. In principal, the devised metrics can be used by the scientific as well as the business community in order to interactively compare the output of specialised mapping systems such as the multi-vehicle map building system described previously in this thesis.

One of the main objectives of the next chapter (Chapter 5) is to apply and incorporate the DIMER metric in different aspects of mapping systems with the aspiration of generating high-quality maps based on the new metric.

Chapter 5

The DIMER Metric in Mapping Applications

5.1 Introduction

Generally speaking, generating high quality maps is one of the principal objectives of any mapping application. Different mapping frameworks seek to improve the map making process by offering strategies aimed at reducing the error associated with their produced maps. In addition, the demand for accurate mapping solutions for specialised applications has elevated the importance of research on new techniques for improving the quality of generated maps. The main prerequisite for achieving this goal is a reliable measure for evaluating the map quality in different setups and situations[1]. To this aim, Chapter 4 strived to re-define the concept of map quality for road mapping applications and proposed a meaningful way of assessing the accurateness of maps comprised of position and orientation of map elements. A versatile map error metric was devised which could be specifically tailored and adjusted to fit the requirements of a wide range of mapping applications depending on specific user profiles describing the previously mentioned direction and type priorities. This metric was designed to tackle some of the shortcomings of common isotropic measures, especially when they are applied in real-world road mapping applications. One of the main objectives of the current chapter is to apply and incorporate the proposed directional map error (DIMER) metric in different aspects of mapping systems, aiming at performance improvement with respect to the new metric. The structure and components of the multi-vehicle mapping framework introduced in Chapter 3 are the main focus of this discussion.

The structure of this chapter is as follows. Section 5.2 addresses the problem of fusing two or more map estimates obtained from different sources with unknown degree of correlation. The well-known covariance intersection (CI) algorithm will be integrated with the new DIMER metric in order to optimally combine the map estimates to attain the highest accuracy in the resulting map. Moreover, a commonly

[1]This can also be perceived as a chicken and egg problem where on one hand, one needs a suitable measure to evaluate the performance of a mapping process; on the other hand, one wants to utilise an algorithm in order to satisfy a defined quality measure.

used analytical variation of this approach called *fast covariance intersection* will be examined and revised in order to minimise the error with respect to the DIMER criterion. Section 5.3 focuses on targeted modification of the EKF-SLAM algorithm as one of the main cornerstones of a large number of stochastic mapping applications such as the present work. Preliminary effort involves re-deriving the Kalman gain so as to minimise the covariance-based DIMER metric in lieu of the conventional trace metric. Although, as will be seen, the calculated gain will turn out to be identical to the classical Kalman gain (minimal-trace gain), the outcome of this process is an interesting result which is mentioned for completeness. Section 5.4 proposes a systematic methodology for reducing the covariance-based directional error of maps obtained using the state-of-the-art EKF-SLAM algorithm. This method is called the *Criteria-based Covariance Trajectory Perturbation (CTP)* and is inspired by the covariance inflation methods found in the literature. The shortcomings of this method are discussed and qqqq

Despite its seeming potential to reduce the variance of map estimates, the covariance trajectory perturbation method has its shortcomings. Most notably, it suffers from bias problems, especially when large measurement noise is present. To alleviate this problem, a debiasing technique is deployed aimed at restricting the bias effect and reducing the mean squared error of map estimates. This technique is implemented through integration with a method called the *Converted-Measurement Kalman filtering (CMKF)*, a concept partly borrowed from the field of target tracking. The highlight of the result are provided thereafter, while the bulk of the results are demonstrated in Chapter 6.

5.2 Criteria-based Covariance Intersection (CI) Analysis

As discussed succinctly in Subsection 3.2.2.1, covariance intersection (CI) is a conservative technique to consistently consolidate different estimates when the correlation information between them is not available. CI was utilised in Section 3.3 to fuse the information shared between distributed nodes in an extensive environment. The present section revisits a similar scheme where two or more maps, obtained from different sources, are to be merged together to achieve a high quality map. The quality of the maps will be assessed using the covariance-based DIMER metric devised in Chapter 4. The examples provided in this section purposely use road signs as sample landmarks to demonstrate the directional map quality concept.

Consider a realistic scenario where two overlapping maps labeled $\mathcal{A}$ and $\mathcal{B}$, consisting of pose estimates of a large number of landmarks (e.g. thousands) are to be fused together in order to obtain a single, more accurate map[2]. It is assumed that $\mathcal{A}$ and $\mathcal{B}$ are acquired through different sources, using different statistical processes. Each map estimate is represented by a mean vector and a covariance matrix for the position and the orientation of all the landmarks in the map. Since, the correlation between the map estimates in $\mathcal{A}$ and $\mathcal{B}$ is typically non-zero but unknown,

[2]No system dynamics are considered for this section and the assumed maps are stationary.

discarding the common information may lead to over-confident estimates. Hence, the well-known covariance intersection can be effectively used in order to achieve a consistent combined map[3].

Due to its structure (see below), the CI algorithm can be tuned in order to optimise with respect to different performance criteria. The trace and determinant of the resulting covariance matrix are two of the most common criteria for robotics application in the literature. Notwithstanding this, as discussed in detail in Chapter 4, these metrics do not fully reflect the quality of maps in road mapping applications such as AutoMap. We now re-examine the fusion of map estimates with unknown degree of dependency based on the error calculated in relation to the directional map quality metric $\mathcal{M}_C$ described via Equation (4.37). We also provide an analogy between the outcome of this method and the results obtained using the standard trace/determinant minimisation.

To highlight the essence of the results and without loss of generality, we simplify the above map fusion scenario to the problem of combining two pose estimates pertaining to a single road sign (labelled j). We further assume perfect orientation estimation and we focus on calculating the best position estimate for this sample landmark. The two map estimates (of the position of the road sign) are indexed by a and b and are considered to be normally distributed. Therefore, these maps are represented by $a \sim \mathcal{N}(\hat{\mathbf{X}}_{aj}, \mathbf{P}^{aj})$ and $b \sim \mathcal{N}(\hat{\mathbf{X}}_{bj}, \mathbf{P}^{bj})$ respectively[4]. As explained before, since in general the correlation between the two estimates, $\mathbf{P}^{abj}$, is unknown, the CI algorithm may be used to consistently combine a and b, where the CI is defined by the following convex combination:

$$\mathbf{P}^{zj^{-1}} = \omega \mathbf{P}^{aj^{-1}} + (1-\omega)\mathbf{P}^{bj^{-1}} \tag{5.1}$$

$$\mathbf{P}^{zj^{-1}}\hat{\mathbf{X}}_{zj} = \omega \mathbf{P}^{aj^{-1}}\hat{\mathbf{X}}_{aj} + (1-\omega)\mathbf{P}^{bj^{-1}}\hat{\mathbf{X}}_{bj} \tag{5.2}$$

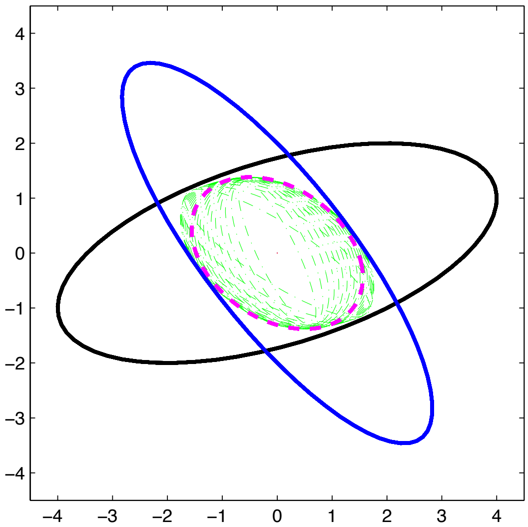
where $z \sim \mathcal{N}(\hat{\mathbf{X}}_{zj}, \mathbf{P}^{zj})$ is an estimate of $\mathbf{X}_j^*$ (true position of landmark j) and $\omega \in [0, 1]$ is calculated according to some performance criteria. We note here simply that for all $\omega \in [0, 1]$, the resulting estimate is consistent and often considerably conservative. Figure 5.1 exemplifies this claim for two sample covariance ellipses. As can be seen from the figure, for all the legitimate values of the free parameter ω, the resulting covariance ellipse encapsulates the space created between the two input ellipses, thus providing consistency. We point to the literature Julier and Uhlmann [1997] for further discussion of the CI algorithm and its consistency. We now seek to examine the effects of minimising the covariance matrix based on the directional map quality metric $\mathcal{M}_C$ described via Equation (4.37). To this aim we propose the following

[3]A similar justification to the one provided in Chapter 3 for using the CI algorithm can be applied in this case.

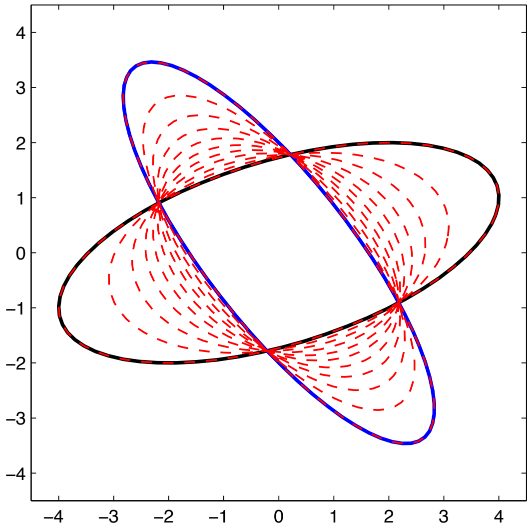
[4]Note that the landmark superscript $'m'$ is dropped for convenience.

optimisation problem for determining the optimal weighting coefficient ω:

$$\begin{aligned} \min_{\omega \in (0,1)} \quad & \mathcal{M}_c^j(\mathbf{P}^{zj}) \\ \text{s.t.} \quad & \mathbf{P}^{zj} = (\omega \mathbf{P}^{aj^{-1}} + (1-\omega)\mathbf{P}^{bj^{-1}})^{-1} \end{aligned} \tag{5.3}$$



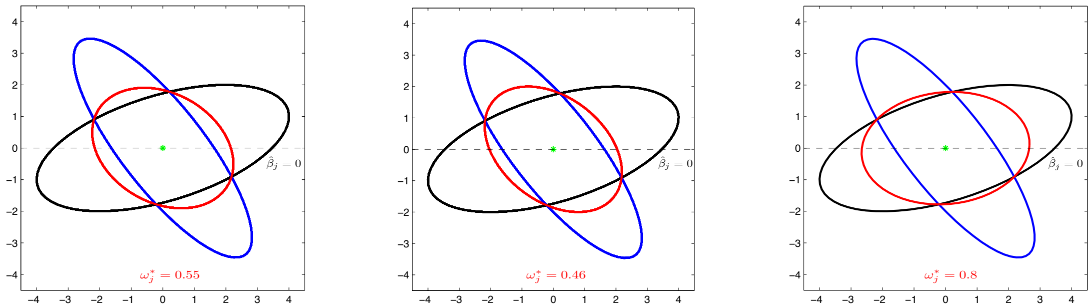
(a) Fusion under known correlation



(b) Fusion using covariance intersection (CI)

Figure 5.1: Fusing two covariance ellipses $\mathbf{P}^{aj}$ and $\mathbf{P}^{bj}$ pertaining to the position estimates of landmark j. The covariance ellipses for the original estimates a and b are depicted using the solid outer ellipses. In the left subfigure, the inner (green) dashed lines show the resulting ellipses for different known correlations between a and b. The slightly ticker dashed line (magenta) shows the case where the correlation between the input estimates is ignored. The right subfigure depicts the resulting CI estimate c for different values of the free parameter ω. As can be seen, all dashed ellipses (red) in this subfigure encapsulate the dashed ellipses (green) in the left subfigure, thus are consistent estimates.

Recalling Eq. (4.37), the computation of $\mathcal{M}_c^j$ requires a reliable estimate for $\hat{\beta}_j$, the heading of the road pertaining to landmark j. Following the discussion provided in Chapter 4, the landmark orientation (which is available for pose maps) can be used in this case to infer the road heading and calculate the directional error metric. Applying the DIMER criterion in calculation of the weighting factor ω in the above equation yield a conservative covariance matrix $\mathbf{P}^{zj}$ with a minimal directional error measure with respect to the road. Figure 5.2 provides an analogy between the resulting covariance ellipse when different minimisation criteria are used for the CI algorithm. Subfigures (a) and (b) use the de-facto trace and determinant criteria, while Subfigure (c) applies the new covariance-based DIMER metric as the minimisation criterion for the covariance intersection method. The input covariance ellipses presented in Figure 5.1 are also used for this example. This example assumes a horizontal road with $\hat{\beta}_j = 0$. In addition, $C_{\parallel}$ and $C_{\perp}$ are set to 0 and 2 respectively. As can be seen, the CI algorithm picks the consistent error ellipse with the lowest perpendicular error amongst all possible results (see Fig.5.1.b). DIMER-based covariance intersection will be studied further in Chapter 6.



(a) Minimal-trace CI (b) Minimal-determinant CI (c) Minimal-DIMER CI

Figure 5.2: Comparison between the minimal-trace, minimal-determinant and minimal-DIMER covariance intersection in fusing the covariance ellipses shown of Fig. 5.1. The road pertaining to the sample landmark j is assumed to be horizontal ($\hat{\beta}_j = 0$) and the directional priorities are set to $C_{\parallel} = 0$ and $C_{\perp} = 2$.

Although numerical methods can theoretically be employed to solve the forgoing optimisation problem stated in 5.3, the use of such methods are deemed costly for extensive applications with a large number of estimates. Alternatively, there are different analytical methods that can be utilised to solve this non-convex optimisation problem and obtain a suboptimal solution. In what follows, we explore one of these methods based on the fast covariance intersection method described in [Niehsen, 2002].

5.2.1 DIMER-based Fast Covariance Intersection

The fast covariance intersection calculation is based on the observation that for the case where $\mathcal{M}_C^j(\mathbf{P}^{aj}) \ll \mathcal{M}_C^j(\mathbf{P}^{bj})$, $\omega \approx 1$, or equivalently, the estimate with the much smaller error measure should be chosen. Thus, we have the following equation that captures the aforementioned observation

$$\omega \mathcal{M}_C^j(\mathbf{P}^{aj}) + (1-\omega)\mathcal{M}_C^j(\mathbf{P}^{bj}) = 0. \tag{5.4}$$

Hence,

$$\omega = \frac{\mathcal{M}_C^j(\mathbf{P}^{bj})}{\mathcal{M}_C^j(\mathbf{P}^{aj}) + \mathcal{M}_C^j(\mathbf{P}^{bj})}. \tag{5.5}$$

Similar results can be obtained straightforwardly for the case where more than two estimates are to be fused. The optimisation problem (5.3) for the case with n_e

estimates becomes

$$\begin{aligned} \min_{\{\omega_l\}} \quad & \mathcal{M}_c^j(\mathbf{P}^{zj}) \\ \text{s.t.} \quad & \mathbf{P}^{zj} = \sum_{l=1}^{n_e} \omega_l \mathbf{P}^{lj^{-1}}, \\ & \sum_{l=1}^{n_e} \omega_l = 1,\ \omega_l \geq 0. \end{aligned} \tag{5.6}$$

Note that a non-iterative way to calculate a suboptimal solution to (5.6) can be obtained in a similar way as the previous case. This calculation is omitted here for the sake of brevity.

5.3 Criteria-based Estimation and Mapping Using EKF-SLAM

As discussed before, producing high quality maps can be accomplished through the incorporation of the newly designed DIMER metric into different aspects of a given mapping system. This section is concerned with targeted modification of one of the key components of the majority of statistical mapping systems, i.e., the filtering algorithm (or the estimator). Broadly speaking, in this context, the filtering algorithm is responsible for combining noisy sensor measurements, given the system's model, to acquire an accurate estimate of the system's unknown states. The Kalman filter along with its nonlinear counterpart, the EKF, have been recognised as two of the most popular filtering structures for mapping systems and are also used in this thesis as the main building-blocks of the proposed large-scale mapping framework. Recalling the multi-vehicle mapping system in Chapter 3, an EKF-based simultaneous localisation and mapping (SLAM) approach was implemented as the nonlinear local filter. This section coupled with Section 5.4 are in pursuit of finding a way to modify the conventional EKF-SLAM algorithm in order to enhance the quality of obtained maps in regards to the DIMER metric.

Typically, algorithm development for this purpose requires a fundamental understanding of different aspects of the mapping system and the applied filtering framework. It is also important to differentiate between the information used in the creation and the factors considered in deployment of the quality metric. Ideally, we want to design an algorithm that performs against all sensible measures. However, realistically, this is not going to be the case for a real world problem due to the inherent complexity. Therefore, inevitably algorithm development must be conducted selectively (with respect to one or a limited number of measures). For example, this chapter endeavours to enhance the quality of map estimates when measured against the covariance-based DIMER metric. Like any other optimisation problem, optimising against a given measure is more likely to deteriorate some of the other measures. The way to get around this problem is to design an algorithm that performs successfully for a single measure or a relevant subset of measures and analyse the effect and possible ramification on other measures. In other words, the relationship between the measure which is used for algorithm development and the (potentially different) measures which are deployed to evaluate the map quality must be carefully studied. This will be discussed later in this chapter and chapter 6.

As a preliminary requirement, we study the structure of the standard discrete-time Kalman filter and its mean-square error minimisation process as it is conventionally used in the literature. Then, the default one-step optimisation criterion will be manipulated appropriately to minimise the covariance-based DIMER metric $\mathcal{M}_c$.

5.3.1 Error Minimisation in the Standard Kalman Filter

Generally speaking, the mean squared error (MSE) is probably one of the most common measures of estimator performance in the estimation community. This measure quantifies the discrepancy between the estimated and the real value of the underlying system's states and is defined by

$$\mathrm{MSE}_X = \mathrm{E}[(\hat{\mathbf{X}} - \mathbf{X})^2] \tag{5.7}$$

where $\mathbf{X}$ and $\hat{\mathbf{X}}$ denote the true and estimated values respectively. Under the Gaussian noise assumption, the Kalman filter algorithm is a minimum mean-square error estimator [Anderson and Moore, 2012]. In other words, no other linear estimator can provide a smaller mean-square error under the same model presumptions. As a recursive Bayes estimator, the discrete-time Kalman filter achieves this optimality through minimising the trace of the a posteriori covariance matrix in every filter step. Therefore, in the linear quadratic Gaussian regime, minimising the trace of the covariance matrix is equivalent to minimising the MSE. The detailed formulation of this minimisation process for the conventional Kalman filter is provided below.

Recalling Subsection 2.2.1 from Chapter 2, in the descrete Kalman filter equations, the *a posteriori* estimate covariance matrix $\mathbf{P}_{k|k}$ in each update step is calculated by

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1}\mathbf{H}_k^\top\mathbf{K}_k^\top + \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^\top \tag{5.8}$$

In Equation (5.8), $\mathbf{P}_{k|k-1}$ is the *a priori* covariance estimate from the prediction step in the Kalman filter. $\mathbf{H}$ is the observation model and $\mathbf{S}_k$ is the innovation covariance defined by

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^\top + \mathbf{R}_k \tag{5.9}$$

with $\mathbf{R}_k$ being the covariance of the observation noise.

The standard optimal Kalman gain $\mathbf{K}_k^{std}$ is derived by minimising the trace of *a posteriori* covariance matrix $\mathbf{P}_{k|k}$ with respect to $\mathbf{K}_k$. The trace $\mathbf{Tr}(\mathbf{P}_{k|k})$ is minimised when its matrix derivative with respect to the gain matrix $\mathbf{K}_k$ is zero. This is equivalent to solving the following differential equation:

$$\frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k})}{\partial \mathbf{K}_k} = 0 \tag{5.10}$$

Starting with the left hand side of the above equation and substituting the *a posteriori* covariance matrix with Equation (5.8), we have

$$\frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k})}{\partial \mathbf{K}_k} = \frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1}\mathbf{H}_k^\top \mathbf{K}_k^\top + \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^\top)}{\partial \mathbf{K}_k} \tag{5.11}$$

Since trace is a linear operator, it commutes with the derivative in the above equation, giving

$$\frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k})}{\partial \mathbf{K}_k} = \frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k-1})}{\partial \mathbf{K}_k} - \frac{\partial \mathbf{Tr}(\mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k|k-1})}{\partial \mathbf{K}_k} - \frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k-1}\mathbf{H}_k^\top \mathbf{K}_k^\top)}{\partial \mathbf{K}_k} + \frac{\partial \mathbf{Tr}(\mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^\top)}{\partial \mathbf{K}_k} \tag{5.12}$$

Applying the gradient matrix rules and using the properties of symmetric matrices we get

$$\frac{\partial \mathbf{Tr}(\mathbf{P}_{k|k})}{\partial \mathbf{K}_k} = 0 - \mathbf{P}_{k|k-1}\mathbf{H}_k^\top - \mathbf{P}_{k|k-1}\mathbf{H}_k^\top + 2\mathbf{K}_k\mathbf{S}_k \tag{5.13}$$

$$= -2\mathbf{P}_{k|k-1}\mathbf{H}_k^\top + 2\mathbf{K}_k\mathbf{S}_k \tag{5.14}$$

Therefore, the standard optimal Kalman gain for the one-step trace minimisation is calculated by

$$\mathbf{K}_k^{std} = \mathbf{P}_{k|k-1}\mathbf{H}_k\mathbf{S}_k^{-1} \tag{5.15}$$

Consequently, the covariance matrix obtained by replacing the above $\mathbf{K}_k^{std}$ into the covariance update formula given by Equation (5.8) is minimal in terms of trace. The resulting covariance can be further simplified according to

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1} \tag{5.16}$$

We now purposely alter the optimisation criterion given by Eq. (5.10) to the covariance-based DIMER metric $\mathcal{M}_C$ described via Equation (4.37) and re-derive the filter gain for minimising the directional error. The initial expectation is that the computed optimal Kalman gain and hence the filter's update equations for this unconventional measure would be different than the standard case described above.

5.3.2 Kalman Gain Derivation for Minimising the DIMER Metric

This part aims at deriving the Kalman gain for minimising the covariance-based DIMER metric rather than the default trace parameter in the Kalman filtering context.

Prior to this derivation, we make a number of preliminary assumptions regarding the EKF-SLAM algorithm for this section. These assumptions aim at highlighting the map estimation process in the EKF-SLAM paradigm. We consider a simplified two-dimensional mapping scenario where a mobile vehicle is observing a single landmark, labelled j, by collecting a series of noisy measurements. Furthermore, since the current discussion is around improving the map quality, without loss of generality, we assume perfect vehicle positioning throughout this analysis. This, in turn, will effectively transform the SLAM problem into a mapping-only problem. As a result, the only unknown state of the filter is the 2D position of the observed landmark x_j. The SLAM algorithm tries to estimate the unknown position of this landmark and provide a measure of uncertainty associated with the state estimation (See Section 2.3 for more details on SLAM). Note that while the above restrictions are essentially applied to simplify the preliminary research, the discussion and results would provide a perspective into the structure of more general systems.

Under the above premise, re-writing the covariance update equation (Eq. (5.8)) for landmark j gives

$$\mathbf{P}^j_{k|k} = \mathbf{P}^j_{k|k-1} - \mathbf{K}^j_k \mathbf{H}^j_k \mathbf{P}^j_{k|k-1} - \mathbf{P}^j_{k|k-1} {\mathbf{H}^j_k}^\top {\mathbf{K}^j_k}^\top + \mathbf{K}^j_k \mathbf{S}^j_k {\mathbf{K}^j_k}^\top$$

Recalling Chapter 2, for a non-linear observation model such as $\mathbf{h}(.,.)$, $\mathbf{H}^j_k$ in the above equation is calculated by evaluating the jacobian matrix $\nabla \mathbf{h}_x(k)$.

We seek the effect of modifying the filter's minimisation criterion to the covariance-based DIMER metric in every filter step. The optimal Kalman gain in this case is denoted by ${\mathbf{K}^j_k}^*$ and is derived by minimising the map error measure pertaining to the *a posteriori* covariance matrix $\mathbf{P}^j_{k|k}$ with respect to $\mathbf{K}^j_k$. This optimisation problem is equivalent to solving the following differential equation:

$$\frac{\partial \mathcal{M}^j_C(\mathbf{P}^j_{k|k})}{\partial \mathbf{K}^j_k} = 0 \tag{5.17}$$

Recalling Equation (4.38) from Chapter 4, the directional error metric $\mathcal{M}^j_C(\mathbf{P}^j_{k|k})$ can be written as

$$\mathcal{M}^j_C(\mathbf{P}^j_{k|k}) = \mathbf{Tr}(\mathbf{C}^j \mathbf{P}^j_{k|k}) \tag{5.18}$$

where

$$\mathbf{C}^j = \begin{bmatrix} C^j_{11} & C^j_{12} \\ C^j_{12} & C^j_{22} \end{bmatrix}$$

is the directional weighting matrix. The elements of the $\mathbf{C}^j$ matrix are evaluated based on the directional priorities and the road heading estimate corresponding to landmark j according to

$$
\begin{aligned}
C_{11}^{j} &= \mathcal{C}_{\parallel}^{j}\left(\frac{1+\cos(2\beta_j)}{2}\right) + \mathcal{C}_{\perp}^{j}\left(\frac{1-\cos(2\beta_j)}{2}\right) \\
C_{22}^{j} &= \mathcal{C}_{\parallel}^{j}\left(\frac{1-\cos(2\beta_j)}{2}\right) + \mathcal{C}_{\perp}^{j}\left(\frac{1+\cos(2\beta_j)}{2}\right) \\
C_{12}^{j} &= \frac{1}{2}(\mathcal{C}_{\parallel}^{j} - \mathcal{C}_{\perp}^{j})\sin(2\beta_j)
\end{aligned}
$$

Substituting Equation (5.18) into the differential equation given by (5.17) yields:

$$
\frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{P}_{k|k}^{j})}{\partial \mathbf{K}_{k}^{j}} = 0 \tag{5.19}
$$

Starting with the left hand side of the above equation and using Equation (5.8), we have

$$
\frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{P}_{k|k}^{j})}{\partial \mathbf{K}_{k}} = \frac{\partial \mathbf{Tr}\left(\mathbf{C}^{j}(\mathbf{P}_{k|k-1}^{j} - \mathbf{K}_{k}^{j}\mathbf{H}_{k}^{j}\mathbf{P}_{k|k-1}^{j} - \mathbf{P}_{k|k-1}^{j}\mathbf{H}_{k}^{j\top}\mathbf{K}_{k}^{j\top} + \mathbf{K}_{k}^{j}\mathbf{S}_{k}^{j}\mathbf{K}_{k}^{j\top})\right)}{\partial \mathbf{K}_{k}^{j}} \tag{5.20}
$$

Since the trace is a linear operator, it commutes with the derivative in the above equation, giving

$$
\begin{aligned}
\frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{P}_{k|k}^{j})}{\partial \mathbf{K}_{k}^{j}} =& \frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{P}_{k|k-1}^{j})}{\partial \mathbf{K}_{k}^{j}} - \frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{K}_{k}^{j}\mathbf{H}_{k}^{j}\mathbf{P}_{k|k-1}^{j})}{\partial \mathbf{K}_{k}^{j}} - \frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{P}_{k|k-1}^{j}\mathbf{H}_{k}^{j\top}\mathbf{K}_{k}^{j\top})}{\partial \mathbf{K}_{k}^{j}} \\
&+ \frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{K}_{k}^{j}\mathbf{S}_{k}^{j}\mathbf{K}_{k}^{j\top})}{\partial \mathbf{K}_{k}^{j}}
\end{aligned} \tag{5.21}
$$

Similar to before, in the light of the linearity of the trace operator and applying the gradient matrix rules and using the properties of symmetric matrices we obtain

$$
\begin{aligned}
\frac{\partial \mathbf{Tr}(\mathbf{C}^{j}\mathbf{P}_{k|k}^{j})}{\partial \mathbf{K}_{k}^{j}} &= 0 - \mathbf{P}_{k|k-1}^{j}\mathbf{H}_{k}^{j\top}\mathbf{C}^{j} - \mathbf{P}_{k|k-1}^{j}\mathbf{H}_{k}^{j\top}\mathbf{C}^{j} + 2\mathbf{K}_{k}^{j}\mathbf{S}_{k}^{j}\mathbf{C}^{j} \\
&= -2\mathbf{P}_{k|k-1}^{j}\mathbf{H}_{k}^{j\top}\mathbf{C}^{j} + 2\mathbf{K}_{k}^{j}\mathbf{S}_{k}^{j}\mathbf{C}^{j}
\end{aligned} \tag{5.22}
$$

Thus, from (5.19) and the above equation, the one-step Kalman gain based on the covariance-based DIMER criteria $\mathcal{M}_{C}^{j}$ is calculated by

$$
\mathbf{K}_{k}^{j*} = \mathbf{P}_{k|k-1}^{j}\mathbf{H}_{k}^{j}\mathbf{S}_{k}^{j-1} \tag{5.23}
$$

As can be seen, the calculated gain in this case is identical to the standard Kalman gain for the optimal-trace Kalman filter in Equation (5.15). This result states that, in addition to the widely accepted error metric $\mathbf{Tr}(\mathbf{P}^j)$, the conventional Kalman filter (used along with the standard Kalman gain) minimises $\mathbf{Tr}(\mathbf{C}^j\mathbf{P}^j)$ in every filter step, where $\mathbf{C}^j$ is any arbitrary positive-semi-definite matrix.

Equivalently, by re-arranging the terms in Equation (5.18), the DIMER metric $\mathcal{M}_C^j(\mathbf{P}^j)$ can be written in terms of the elements of the covariance matrix $\mathbf{P}^j$ according to:

$$\mathcal{M}_C^j(\mathbf{P}^j) = \mathbf{Tr}(\mathbf{C}^j\mathbf{P}^j) = C_{11}^j P_{xx}^j + C_{22}^j P_{yy}^j + 2C_{12}^j P_{xy}^j \tag{5.24}$$

Consequently, in a linear system, in addition to minimising $(P_{xx}^j + P_{yy}^j)$, the standard Kalman gain $\mathbf{K}_k^{j\,*} = \mathbf{P}_{k|k-1}^j \mathbf{H}_k^j \mathbf{S}_k^{j\,-1}$ minimises all the other linear combinations of the elements of the error covariance matrix (including the off-diagonal elements), provided that matrix $\mathbf{C}^j$ is PSD.

To the best of our knowledge, the above Kalman gain derivation for minimising $\mathbf{Tr}(\mathbf{C}^j\mathbf{P}^j)$ has not been carried out explicitly in the literature. Hence, it is presented here as an interesting result which provides an insight into the operation and philosophy of the conventional Kalman filter as an optimal estimator. We also extend this finding to provide the following corollary:

Corollary 1. *Amongst linear filters with the general update equation in the form of*

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \nu_k, \tag{5.25}$$

the optimal choice for minimising the cost function $\mathcal{J} = \mathbf{Tr}(\mathbf{GP})$, with $\mathbf{G}$ being any size-compatible positive-definite matrix, is the classical Kalman filter with the one-step gain equal to the standard Kalman gain $\mathbf{K}_k^* = \mathbf{P}_{k|k-1}\mathbf{H}_k\mathbf{S}_k^{-1}$.

In the next section, we introduce an intuitive approach aimed at enhancing the quality of similar probabilistic mapping techniques.

5.4 Criteria-based Covariance Trajectory Perturbation (CTP)

This section proposes a systematic approach called the criteria-based Covariance Trajectory Perturbation (CTP) for reducing the covariance-based directional errors associated with maps consisting of position and orientation of interesting landmarks. Similar to Section 5.3, the EKF-SLAM algorithm is considered to be at the core of the map estimation framework. We focus on specialised mapping applications, more specifically the AutoMap project (with road signs as landmarks) addressed throughout this thesis.

Similar developments in the literature revolve around efforts to solve an optimal problem associated with a proposed metric or a cost function (often an error minimisation problem). Whereas it is not generally practical to solve the optimisation problem exactly[5], alternative approaches aim at deriving approximations and tweaking the measures until the solution converges. This is the avenue we follow in this section.

As a background, Subsection 5.4.1 provides a brief overview on Covariance Inflation methods as the main inspiration behind the proposed work. Subsection 5.4.2 formulates the problem at hand, followed by a methodological solution aimed at alleviating the error in the map. It is desirable to manipulate the filtering algorithm in order to achieve a better performance in terms of obtaining a lower directional map error measure compared to the standard Kalman filter.

5.4.1 Inspiration and Proximity to the Literature

The work in this section has been inspired by the idea of artificially increasing the estimated covariance in order to control certain aspects and behaviours of statistical system for performance improvement purposes.

Historically, employing variance inflation methods date back as early as the 1970's. For example [Miller, 1971] suggested scaling the variance in order to enhance the filter performance for state estimate filters. This process caused the state estimate to deviate from the assumed system dynamics in order to account for the imperfect knowledge of the system under study. Similarly, injecting artificial noise (as disturbance) can also be utilised to speed up the convergence of estimators. For instance, [Bertuccelli and How, 2008] improves the responsiveness of the estimator for the non-stationary Markov Chain model by adding an artificial pseudo-noise to the system.

In the mapping community, some works have proposed the use of covariance inflation techniques to accelerate the performance of Kalman-filter-based SLAM algorithms. [Julier, 2003] defines covariance inflation as "the process of adding a positive semidefinite matrix to the system covariance matrix to improve the properties of a SLAM algorithm". This process effectively removes the dependency between the vehicle and landmark states in the SLAM paradigm by diagonalising the state error covariance matrix which, in turn, reduces the computational complexity of the algorithm. Examples of covariance inflation methods in SLAM can be found in [Guivant and Nebot, 2001, 2003]. Despite their effectiveness, one of the main drawbacks of such techniques is that the addition of pseudo-noise may potentially lead to instability in the system due to the fact that the covariance can increase without bound. Furthermore, according to [Bailey et al., 2006], the inflation process effectively nullifies all of the convergence properties of SLAM described in [Dissanayake et al., 2001].

Another group of applications in which the covariance may be increased artificially is where it is attempted to compensate for estimation inconsistencies. Although the amount of adequate inflation is not quite clear, such methods have been proven to be effective in providing consistent results. A prevalent example is the practical

[5] Linear Kalman filter is one exception where the optimisation problem is solved exactly.

scenario of fusing two estimates with unknown inter-dependency. A common suboptimal approach is to ignore the existing correlation and avoid the possible generated inconsistency by inflating the covariance of the combined estimates (see [Julier and Uhlmann, 2001a] and [Bailey et al., 2006] for more details).

5.4.2 DIMER-based Covariance Trajectory Perturbation Algorithm

Similar to Section 5.3, we consider a simplified mapping scenario where a single vehicle is observing a landmark (labelled j) by collecting noisy range and bearing measurements at κ different timestamps. Due to the nature of our road mapping application, we assume that κ is a finite, small integer. In practice (for the AutoMap project), we have $\kappa \leq 10$, as road signs are usually detected in a limited number of video frames due to the dynamic of the vehicle and sensor's restrictions.

Without loss of generality, once again, we assume that the vehicle state is known perfectly and the only unknown state is the 2D position of the observed landmark j. We seek to estimate the unknown position of this landmark x_j through employing an EKF-SLAM algorithm. Moreover, a (2×2) covariance matrix $\mathbf{P}^j_{\kappa|\kappa}$ representing the uncertainty associated with the landmark state estimate after κ measurements is to be estimated. Note that the landmark superscript $'m'$ has been dropped for convenience.

Assume at each time k, $k \in \{1,\dots,\kappa\}$, a measurement z^j_k is collected. Let $\mathcal{Z}^j_\kappa \triangleq \{z^j_i\}^\kappa_{i=1}$ be the combined vector of all measurements pertaining to landmark j. Moreover, let $\mathcal{F}$ be the set of all recursive functions that at at each time step k take the measurement z^j_k along with $\mathbf{P}^j_{k|k-1}$ and return an updated covariance matrix $\mathbf{P}^j_{k|k}$ such that $\mathbf{P}^j_{k|k-1}$ and $\mathbf{P}^j_{k|k}$ are consistent for a known *a priori* covariance matrix $\mathbf{P}^j_{1|0} = \overline{\mathbf{P}}^j$. Hence, given $\mathcal{Z}^j_\kappa$ the problem of interest is

$$\begin{aligned} \min_{F\in\mathcal{F}} \quad & \mathcal{M}^j_C(\mathbf{P}^j_{\kappa|\kappa}) \\ \text{s.t.} \quad & \mathbf{P}^j_{k|k} = \mathcal{F}(\mathbf{P}^j_{k|k-1}, z^j_k), \quad k \in \{1,\dots,\kappa\} \end{aligned} \tag{5.26}$$

The optimisation problem (5.26) is an infinite-dimension optimisation problem and is typically intractable. To be able to make the problem tractable we make an assumption on the recursive function $\mathcal{F}$. Let $\mathcal{F}$ at time k be of the form

$$\mathbf{P}^j_{k|k} = \mathbf{P}^j_{k|k-1} - \mathbf{K}^j_k\mathbf{H}^j_k\mathbf{P}^j_{k|k-1} - \mathbf{P}^j_{k|k-1}{\mathbf{H}^j_k}^\top{\mathbf{K}^j_k}^\top + \mathbf{K}_k\mathbf{S}^j_k{\mathbf{K}^j_k}^\top \tag{5.27}$$

where $\mathbf{H}^j_k$ is the observation model and $\mathbf{S}^j_k$ is the innovation covariance defined by

$$\mathbf{S}^j_k = \mathbf{H}^j_k\mathbf{P}^j_{k|k-1}{\mathbf{H}^j_k}^\top + \mathbf{R}^j_k. \tag{5.28}$$

Note that for the foregoing recursive update, the only variables that can be chosen independent of the model and $\mathcal{Z}_\kappa$ are $\mathbf{K}^j_k$, $k \in \{1,\dots,\kappa\}$, and $\mathbf{P}^j_{1|0}$. Particularly, $\mathbf{P}^j_{1|0}$

can be chosen to be any conservative estimate of $\overline{\mathbf{P}}^j$ and can be parameterised as

$$\mathbf{P}^j_{1|0} = \overline{\mathbf{P}}^j + \Delta\mathbf{P}^j, \tag{5.29}$$

where $\Delta\mathbf{P}^j$ is any positive definite (PD) matrix that results in $\mathbf{P}^j_{1|0}$ being consistent ($\overline{\mathbf{P}}^j$ is assumed to be consistent as well). Initially, $\Delta\mathbf{P}^j_k$ is defined according to

$$\Delta\mathbf{P}^j_k = \begin{bmatrix} \delta^k_x & 0 \\ 0 & \delta^k_y \end{bmatrix} \tag{5.30}$$

where $\delta^k_x, \delta^k_y > 0$, such that after collecting the κ measurements, the calculated $\mathcal{M}^j_C(\widetilde{\mathbf{P}}^j{}_{\kappa|\kappa})$ is smaller than $\mathcal{M}^j_C(\mathbf{P}^j_{\kappa|\kappa})$, where $\widetilde{\mathbf{P}}^j{}_{\kappa|\kappa}$ denotes the resulting covariance matrix at time κ after applying the perturbation $\Delta\mathbf{P}^j$ as per Eq. (5.29). Then, the optimisation problem (5.26) becomes

$$\begin{aligned}
\min_{\mathbf{K}^j_k, \Delta\mathbf{P}^j} \quad & \mathcal{M}^j_C(\mathbf{P}^j_{\kappa|\kappa}) \\
\text{s.t.} \quad & \mathbf{P}^j_{k|k} = \mathbf{P}^j_{k|k-1} - \mathbf{K}^j_k\mathbf{H}^j_k\mathbf{P}^j_{k|k-1} - \mathbf{P}^j_{k|k-1}\mathbf{H}^{j\top}_k\mathbf{K}^{j\top}_k + \mathbf{K}_k\mathbf{S}^j_k\mathbf{K}^{j\top}_k \\
& \mathbf{S}^j_k = \mathbf{H}^j_k\mathbf{P}^j_{k|k-1}\mathbf{H}^{j\top}_k + \mathbf{R}^j_k \\
& \mathbf{P}^j_{1|0} = \overline{\mathbf{P}}^j + \Delta\mathbf{P}^j, \quad \Delta\mathbf{P}^j \geq 0 \\
& k \in \{1, \ldots, \kappa\}.
\end{aligned} \tag{5.31}$$

In addition, in light of the calculations conducted in Subsection 5.3.2, Eq. (5.31) can be further relaxed. Recalling Eq. (5.23), the Kalman gain given by $\mathbf{K}^j_k \triangleq \mathbf{P}^j_{k|k-1}\mathbf{H}^j_k\mathbf{S}^{j\,-1}_k$, $k \in \{1, \ldots, \kappa\}$, minimises the $\mathcal{M}^j_C(\mathbf{P}^j_{k|k})$ described by Equation (4.38) in every filter step. A summary of the CTP method is provided below.

$$\boxed{\begin{aligned}
\min_{\Delta\mathbf{P}^j} \quad & \mathcal{M}^j_C(\mathbf{P}^j_{\kappa|\kappa}) \\
\text{s.t.} \quad & \mathbf{P}^j_{k|k} = \mathbf{P}^j_{k|k-1} - \mathbf{K}^j_k\mathbf{H}^j_k\mathbf{P}^j_{k|k-1} - \mathbf{P}^j_{k|k-1}\mathbf{H}^{j\top}_k\mathbf{K}^{j\top}_k + \mathbf{K}_k\mathbf{S}^j_k\mathbf{K}^{j\top}_k \\
& \mathbf{S}^j_k = \mathbf{H}^j_k\mathbf{P}^j_{k|k-1}\mathbf{H}^{j\top}_k + \mathbf{R}^j_k \\
& \mathbf{K}^j_k = \mathbf{P}^j_{k|k-1}\mathbf{H}^j_k\mathbf{S}^{j\,-1}_k \\
& \mathbf{P}^j_{1|0} = \overline{\mathbf{P}}^j + \Delta\mathbf{P}^j, \quad \Delta\mathbf{P}^j \geq 0 \\
& k \in \{1, \ldots, \kappa\}.
\end{aligned}} \tag{5.32}$$

Therefore, the problem is to find an appropriate positive definite matrix $\Delta\mathbf{P}^j$ that minimises the resulting $\mathcal{M}^j_C(\mathbf{P}^j_{\kappa|\kappa})$. In this way, by inflating the initial covariance matrix across certain directions, the directional quality of the final obtained map (after κ observations) is improved with respect to the covariance-based DIMER criteria.

The minimisation problem expressed using Equation (5.32) is a non-convex optimisation problem where no exact solution can be mathematically formulated. Therefore, in order to find the best covariance perturbation value for a single landmark j, one may run the filter multiple times, each time with a different value of $\Delta\mathbf{P}^j$ and compare the quality of the final maps after κ observations. The only inputs required for this approach are all the measurements $\mathcal{Z}_k$ for times $k = 1 \ldots \kappa$. In other words, all the measurement up to time $k = \kappa$ are required to find the solution to the optimisation problem in (5.32). Therefore, the nature of this problem can be perceived as small-sized batch processing as opposed to real-time recursive algorithms typically used[6]. The crucial point here is that the optimal $\Delta\mathbf{P}^{j^*}$ is directly dependant on the measurements obtained on that specific filter run and might not be applicable for different set of measurements from the same probability distribution. Moreover, once again, we remark that evaluating the directional metric $\mathcal{M}_C^j(\mathbf{P}^j_{\kappa|\kappa})$ for every filter run requires the availability of the heading parameter for the road corresponding to landmark j. As explained before, the landmark orientation estimate is the best viable candidate for this purpose and will be used in this thesis[7].

Initially, a naive, exhaustive search approach is implemented to solve the optimisation problem set forth by (5.32). The search runs the filter from time $k = 1$ to $k = \kappa$, each time using a pre-defined value of $\delta_x > 0$ and $\delta_y > 0$ and evaluates the map error measure $\mathcal{M}_C$ for the final covariance matrix at time $k = \kappa$. We are interested in the perturbation matrix $\Delta\mathbf{P}^{j^*}$ which yields the lowest covariance-based error $\mathcal{M}_C^j(\mathbf{P}^j{}_{\kappa|\kappa})$ for the final obtained map at time κ. Initial experiments suggest that finding a non-zero $\Delta\mathbf{P}^{j^*}$ in the search cannot be guaranteed. However, in situations in which a valid covariance perturbation is found, the covariance-based quality measure for the obtained map is improved in comparison with the conventional algorithm (with zero covariance perturbation). The proposed naive search process can be expensive depending on the search domain in the $\Delta\mathbf{P}^j$ space and the step size for the search variables. Nevertheless, more intelligent, efficient search methods will be used in the simulations provided in the results chapter to help alleviate the computational load for mapping environments with a large number of landmarks. It will be shown later in Chapter 6 that a more efficient search is confined to testing only four different $\Delta\mathbf{P}^j$ values for any given landmark.

As will be seen, preliminary simulation and analysis suggest that, in case of finding the appropriate perturbation, the CTP algorithm seemingly produces maps with an improved quality, by virtue of reducing the DIMER metric for individual landmarks. At this point, we suggest that the main reason behind the effectiveness of this approach is that the algorithm capitalises on the inherent non-linearities in the system model. In other words, the covariance trajectory perturbation algorithm

[6]With enough available computational power, the CTP algorithm can be applied online (in a delayed batch mode).

[7]As will be seen in Chapter 6 simulations, a pre-filter structure is devised in order to get a preliminary orientation estimate for the existing landmarks in the map.

accounts for some of the exploited approximations in the calculation of the mean and variance of the estimates in the non-linear SLAM framework. These approximations are predominantly caused by the linearisation of system models and the evaluation of Jacobian terms (see Section 2.2.2). As a result, the estimated mean and covariance may not accurately represent the true moments of these probability distributions. This will be discussed further in Chapter 7.

Given the above highlights, the detailed implementation of the covariance trajectory perturbation algorithm will be elaborated in Chapter 6. The effectiveness of this approach in improving the covariance-based DIMER metric will be demonstrated through simulations for both single-vehicle and multi-vehicle systems. Different practical road mapping scenarios will be examined. Furthermore, the average impact of the CTP algorithm on the popular trace metric as well as the ground-truth-based DIMER metric will be examined through implementation of Monte-Carlo simulations.

It is worth mentioning here that the idea of running a Kalman filter multiple times for a given set of measurements has been previously used in the literature for different theoretical or practical objectives. For example, in [Kalandros and Pao, 1998], the authors propose a sensor management strategy in which a separate Kalman filter operates for a series of possible sensor combinations in order to determine the minimal sensor combination that achieves a certain accuracy. Despite introducing additional computational load to the system (resulting from re-running the filter), such schemes are beneficial in examining the effect of different system parameters and in search for optimal values across different dimensions.

The philosophy behind the suboptimal covariance perturbation algorithm to improve the map quality lies in the nature of the practical problem at hand. It is important to note that by virtue of the limited number of observations of each map element, studying the behaviour of the results in the limit is not applicable in this case. More precisely, we are looking at a finite-horizon problem[8]. Therefore, for a limited number of available measurements, applying covariance perturbation can potentially offer a better transient performance compared to the classical Kalman-based SLAM.

One common limitation of algorithm development with respect to a certain cost function (e.g. the above minimisation effort against the DIMER metric) is that minimising the given cost function may potentially result in an increase in other unwanted cost functions associated with the system. As described briefly in Subsection 5.3.1, the Kalman filter is an exception where minimising the variance coincides with minimising some of the other popular measures such as the expected value between the error between the estimated and real values. However, in typical real-world filtering regimes that is not going to be the case. The next section provides a discussion on the trade-off between the bias and variance of an estimator with a direct focus on the criteria-based CTP approach.

[8]It has been shown in the literature that it is typically possible to outperform the infinite-horizon optimal Kalman filter in the transient phase (e.g. unscented Kalman filter and some high-gain observers).

5.5 Discussion on the Trade-off between Bias, Variance and Mean Squared Error

Let $\hat{\theta}$ be a general estimator of the unknown parameter θ. Recalling Equation (5.7), the mean squared error of this estimator is a function of θ and is defined as

$$\mathrm{MSE}(\hat{\theta}) = \mathrm{E}\left[(\hat{\theta} - \theta)^2\right]. \tag{5.33}$$

After applying some mathematical manipulation and using the properties of expected value function (E), the above equation can be expanded according to[9]

$$\mathrm{MSE}(\hat{\theta}) = \mathrm{Var}(\hat{\theta}) + \mathrm{Bias}(\hat{\theta}, \theta)^2 \tag{5.34}$$

Consequently, the mean squared error can be interpreted as the sum of the estimator's variance and the square of its bias, where the variance component is a measure of variability and the bias is an indicator of accuracy. Broadly speaking, for any estimator, it is desirable to minimise its MSE value. In many applications, minimising the MSE is generally done by designing the estimator based on one of the aforementioned components (variance or bias). However, in some occasions, minimising one of these parameters may lead to an increase in the value of the other, resulting in an undesirable increase in the overall MSE value. Consequently, controlling only one of the bias or variance parameters does not guarantee a low mean squared error.

The same situation holds for the criteria-based covariance trajectory perturbation (CTP) algorithm proposed in Subsection 5.4.2. As discussed in detail, the optimisation problem was set up in order to reduce the DIMER-based variance pertaining to the obtained position estimate of the map elements. In other words, the formulation made no specific considerations for controlling the bias associated with the estimates. As will be seen later in this thesis, despite its effectiveness in reducing the variance, applying the CTP algorithm can potentially introduce additional bias into the estimator in some occasions[10]. For this reason, the trade-off between the variance and bias must be carefully analysed. Chapter 6 will provide Monte Carlo simulations to examine the ramification of the covariance trajectory perturbation algorithm on the bias of map estimates for practical road mapping scenarios. Simulations suggest that the bias issue is much more acute in scenarios where the uncertainties in the measurements are large. Under such circumstances, the consistency of the map estimates is endangered due to the generation of biases and the mismatch between the mean and covariance estimates.

The bias problem prompted the implementation of a debiasing procedure in synergy with the CTP method in order to reduce the risk of acquiring inconsistent, biased map estimates. The proposed debiasing compensation used in this work is

[9] For a formal proof see [Wackerly et al., 2008].

[10] Note that due to the nonlinear nature of the EKF-SLAM algorithm, the original (unperturbed) estimates are most probably biased as well. However, as will be seen in the simulations of Chapter 6, the CTP algorithm amplifies this bias in some occasions.

mainly inspired by a target-tracking paper by Lerro and Bar-Shalom [Lerro and Bar-Shalom, 1993][11]. The original applicability of this paper was intended for active sonar systems or long range radar systems. The presented approach is based on the derivation of the expected value of the true bias along with the true covariance for a target-tracking scenario. We now present the debiasing strategy utilised in this work.

5.5.1 A Debiasing Compensation Strategy Combined with Converted Measurement Kalman Filtering (D-CMKF)

As mentioned previously, the debiasing procedure presented in this work is inspired by [Lerro and Bar-Shalom, 1993], a paper borrowed from the field of target tracking. For this reason, appropriate modifications are needed in order to apply the concept to the localisation and mapping applications similar to the work undertaken in this thesis. The debiasing approach is implemented in combination with a converted measurement Kalman filter (CMKF), a filter based on transformation of measurements to a common Cartesian coordinate system. It is worth mentioning that, unlike most mapping systems where the measurements and state variables are represented in both polar and Cartesian frames, tracking systems typically transform the measurements to a common coordinate system.

The motivation behind using the debiasing technique with converted measurements is two-fold. Firstly, it is theoretically expected that the debiasing compensation procedure will help the potential bias/inconsistency problem discussed earlier. Secondly, as will be explained, utilising the Kalman filter with converted, Cartesian measurements should restrict the effect of nonlinearities in the system compared to the extended Kalman filter. We now present the formulation concerning estimation with Debiased Converted Measurement Kalman Filter (D-CMKF) for the practical mapping application in this body of work.

Consider a SLAM system similar to the one studied in Section 3.2 with the following polar measurements, $\mathbf{z}^p(k)$.

$$\mathbf{z}^p(k) = \begin{bmatrix} r_m(k) \\ \theta_m(k) \end{bmatrix} \tag{5.35}$$

The system measurements are in the form of range (r_m) and bearing (θ_m) and are defined with respect to the true range r and bearing θ values:

$$r_m = r + \bar{r} \qquad\qquad \theta_m = \theta + \bar{\theta} \tag{5.36}$$

with $\bar{r}$ and $\bar{\theta}$ being the range and bearing errors. These errors are assumed to be independent with zero-mean and respective standard deviations σ_r and σ_θ. These polar measurements $\mathbf{z}^p(k)$ are related to the vehicle and map state through

[11]Note that other debiasing approaches could also be substituted.

$$\mathbf{z}^p(k) = \begin{bmatrix} r_m(k) \\ \theta_m(k) \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_v(k) - \hat{x}_i(k))^2 + (\hat{y}_v(k) - \hat{y}_i(k))^2} \\ \arctan\left(\frac{\hat{y}_v(k)-\hat{y}_i(k)}{\hat{x}_v(k)-\hat{x}_i(k)} - \hat{\phi}_v(k)\right) \end{bmatrix} + \begin{bmatrix} v_R(k) \\ v_\theta(k) \end{bmatrix}$$

Notice the minor change of notation compared to Eq. (3.8). Subscript m has been used here to indicate the measured values.

In the debiased-CMKF framework, the above polar measurements at time k are converted to a Cartesian coordinate measurement denoted by $\mathbf{z}^c(k)$ through a non-linear transformation η according to

$$\mathbf{z}^c(k) = \eta[\mathbf{z}^p(k)] - \mu_a \tag{5.37}$$

This conversion is given by

$$\mathbf{z}^c(k) = \begin{bmatrix} x_m^c(k) \\ y_m^c(k) \end{bmatrix} = \begin{bmatrix} r_m(k)\cos\left(\theta_m(k) + \hat{\phi}_v(k|k-1)\right) \\ r_m(k)\sin\left(\theta_m(k) + \hat{\phi}_v(k|k-1)\right) \end{bmatrix} - \mu_a(k) \tag{5.38}$$

where x_m^c and y_m^c are used to express the converted Cartesian measurements in the local frame of reference[12].

Notice the incorporation of the new term $\mu_a(k)$ into the above conversion equations. This parameter is the debiasing term that takes into account the correction for the average bias. The authors in [Lerro and Bar-Shalom, 1993] derive the true bias μ_t as well as the true converted measurement covariance $\mathbf{R}_t$. Since these terms are conditioned on true values of range and bearing, the expected value of these true moments (conditioned on the measured range and bearing) are suggested for use in practice.

Since the respective converted measurement equations in [Lerro and Bar-Shalom, 1993] are mainly derived for target tracking applications, some basic amendments ought to be made in order to implement this methodology in the localisation and mapping context studied in this thesis. Most importantly, the heading of the vehicle, ϕ_v, must be incorporated into the original equations from the foregoing paper (see Eq. (5.38)). For simplicity and without loss of generality, we assume negligible error in the heading of the vehicle in this implementation. This section does not dwell on the details of this formulation and we simply provide the final equations for the modified converted measurement filter with bias compensation.

We now temporarily drop the time index k for convenience in the formulation and we provide the revised equations for the EKF-SLAM. The debiasing term, μ_a, is expressed according to the following equation:

[12]These parameters are defined in the local frame and shall not to be confused with the un-superscripted $\hat{x}_m$ and $\hat{y}_m$ parameters used in Chapter 3 to denote the estimated position of map elements in the global frame.

$$\mu_a = \begin{bmatrix} r_m \cos(\theta_m + \hat{\phi}_v)(e^{-\sigma_\theta^2} - e^{-\sigma_\theta^2/2}) \\ r_m \sin(\theta_m + \hat{\phi}_v)(e^{-\sigma_\theta^2} - e^{-\sigma_\theta^2/2}) \end{bmatrix} \tag{5.39}$$

Also, the average covariance of the converted measurements is a 2×2 matrix, $\mathbf{R}_a$, given by

$$\mathbf{R}_a = \begin{bmatrix} R_a^{11} & R_a^{12} \\ R_a^{21} & R_a^{22} \end{bmatrix} \tag{5.40}$$

with the following elements

$$\begin{aligned} R_a^{11} =& r_m^2 e^{-2\sigma_\theta^2}[\cos^2(\theta_m + \hat{\phi}_v)(\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \sin^2(\theta_m + \hat{\phi}_v)(\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] \\ &+ \sigma_r^2 e^{-2\sigma_\theta^2}[\cos^2(\theta_m + \hat{\phi}_v)(2\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \sin^2(\theta_m + \hat{\phi}_v)(2\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] \end{aligned} \tag{5.41a}$$

$$\begin{aligned} R_a^{22} =& r_m^2 e^{-2\sigma_\theta^2}[\sin^2(\theta_m + \hat{\phi}_v)(\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \cos^2(\theta_m + \hat{\phi}_v)(\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] \\ &+ \sigma_r^2 e^{-2\sigma_\theta^2}[\sin^2\theta_m(2\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \cos^2(\theta_m + \hat{\phi}_v)(2\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] \end{aligned} \tag{5.41b}$$

$$R_a^{12} = \sin(\theta_m + \hat{\phi}_v)\cos(\theta_m + \hat{\phi}_v)e^{-4\sigma_\theta^2}[\sigma_r^2 + (r_m^2 + \sigma_r^2)(1 - e^{\sigma_\theta^2})] \tag{5.41c}$$

Given the above-mentioned measurement conversion, the filter equations must be changed accordingly to incorporate the Cartesian measurements $\mathbf{z}^c = [x_m^c, y_m^c]^\top$. This modification will have some positive effects on the nonlinearity of the original filter. More prominently, as a result of using converted measurements, the general observation model $\mathbf{h}(.,.)$ in the original EKF-SLAM equations will no longer be non-linear (see Eq. (2.46)).

The predicted measurement at time k for an arbitrary landmark i is calculated using

$$\begin{aligned} \mathbf{z}_i^c(k|k-1) &= \mathbf{h}(\hat{\mathbf{x}}_v(k), \hat{\mathbf{x}}_{mi}(k)) + \mathbf{v}(k) \\ &= \begin{bmatrix} \hat{x}_i(k|k-1) - \hat{x}_v(k|k-1) \\ \hat{y}_i(k|k-1) - \hat{y}_v(k|k-1) \end{bmatrix} + \mathbf{v}(k) \end{aligned} \tag{5.42}$$

Consequently, the once non-linear jacobian matrix $\nabla\mathbf{h}_\mathbf{x}$, is replaced by the below linear measurement matrix $\mathbf{H}$ in a multi-landmark scenario.

$$\mathbf{H} = \begin{bmatrix} -1 & 0 & 0 & \dots & 1 & 0 & \dots \\ 0 & -1 & 0 & \dots & 0 & 1 & \dots \end{bmatrix} \tag{5.43}$$

Furthermore, the revised update equations for CMKF are given by[13]

$$\nu(k) = \mathbf{z}^c(k) - \mathbf{H}\mathbf{x}(k|k-1) \tag{5.44}$$
$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{H}^T\mathbf{S}^{-1}(k) \tag{5.45}$$
$$\mathbf{S}(k) = \mathbf{H}\mathbf{P}(k|k-1)\mathbf{H}^T + \mathbf{R}_a(k) \tag{5.46}$$

and the covariance matrix is updated according to

$$\mathbf{P}(k|k) = \mathbf{P}(k|k-1) - \mathbf{K}(k)\mathbf{H}\mathbf{P}(k|k-1) - \mathbf{P}(k|k-1)\mathbf{H}^T\mathbf{K}^T(k) + \mathbf{K}(k)\mathbf{S}(k)\mathbf{K}^{-1}(k) \tag{5.47}$$

where $\mathbf{R}_a(k)$ is the converted measurement error covariance given by (5.41).

The effects of using the converted-measurement Kalman filter with bias compensation (D-CMKF) in lieu of the original extended Kalman filter (EKF) will be examined in Chapter 6 for a multi-vehicle mapping scenario. As will be seen, the new filter will produce consistent estimates that are compatible with the calculated covariances, even for high levels of measurement noise. In addition, considerable improvements can be detected for the ground-truth-based directional quality. Furthermore, the resulting mean squared error (MSE) for the converted-measurement filter is reduced on average, compared to the non-linear SLAM using EKF. In addition to these encouraging enhancements, an unexpected result is observed in the simulations. It turns out that the effect of the CTP algorithm (applied for reducing the variance of position estimates) is masked by the D-CMKF implementation. The linearisation of the system model is likely to have a significant bearing on disabling the covariance trajectory perturbation approach. Other possible reasons behind these outcomes will be discussed in Chapter 6.

5.6 Summary

This chapter made an effort to incorporate the newly designed DIMER metric into some of the most common components of mapping systems with an emphasis on the real-world application characterised in this thesis. Section 5.2 addressed the practical problem of merging two or more maps with unknown degree of correlation in order to acquire a single, high-quality, consistent map. To this end, the mainstream covariance intersection (CI) algorithm was modified to adopt the covariance-based DIMER metric rather than the commonly used trace measure as the error minimisation criteria. In addition, a widely used analytical version of this method known as the fast covariance intersection (FCI) was derived so as to minimise the covariance-based DIMER criteria. It was shown that this approach holds the same structure as

[13] See Section 2.3 for the original EKF-SLAM equations.

the trace-based FCI.

The rest of the chapter was concerned with the targeted modification of Kalman-filter-based mapping algorithms in order to improve the quality of generated maps, when judged by the new DIMER metric. The first effort involved deriving the optimal gain in the Kalman filter's update equations to minimise the covariance-based DIMER criteria for the landmarks in the map. This effort yielded a gain identical to the Kalman gain for the conventional optimal-trace filter. Following this attempt, Section 5.4 devised a systematic approach aiming at improving the quality of maps obtained using the SLAM filter in the road application under study. This heuristic algorithm, called the covariance trajectory perturbation (CTP), is a brute-force optimisation method which operates in a batch mode. The method seeks the existence of an appropriate perturbation to the initial covariance matrix of landmark estimates in order to reduce the overall covariance-based DIMER metric pertaining to the map. This suboptimal method requires the filter to be run multiple times (each time with a different value of injected covariance) over a fixed set of measurements, to find the perturbation yielding the highest directional quality in the final map. In essence, this method is designed to improve the transient phase of the mapping process compared to the classical Kalman filter and is conceptually based on the subtle fact that there are only a limited number of observations for each landmark.

Section 5.5 addressed the general problem of trade-off between bias and variance in typical estimation regimes. To reduce the overall MSE and diminish the effect of the amplified bias resulting from the CTP method, a debiasing technique was utilised. The implementation of this technique was carried out through integration with the Converted-Measurement Kalman filtering (CMKF). Chapter 6 will examine the effect of this filter on the covariance trajectory perturbation method in the distributed mapping system proposed in this body of work.

Chapter 6

Simulations and Results

6.1 Introduction

This chapter presents the simulation results of the methods developed throughout previous chapters of this thesis. For this purpose, these results are distinctly divided into three main groups.

The first part of this chapter is concerned with demonstrating the effectiveness and validity of the large-scale mapping solution developed in Chapters 2 and 3. The efficiency of the devised multi-vehicle system in terms of communication and computation will be addressed in Section 6.2 and its performance will be compared against more conventional methods in the literature[1]. We will conclude this part by analysing the solution using a number of illustrative examples.

The second part of the results exclusively concentrates on the proposed directional map quality (DIMER) metric designed in Chapter 4 and the follow-up criteria-based mapping solutions discussed in Chapter 5. In Section 6.3, the effectiveness of the covariance trajectory perturbation (CTP) algorithm in enhancing the quality of landmark localisation will be examined for simplified mapping scenarios comprised of a single vehicle and a single landmark. In addition, the impact of deploying the DIMER-based covariance perturbation method on some of the other quality measures will be inspected. We will further study the behaviour and characteristics of the proposed CTP methodology through various experiments in different realistic scenarios. Monte Carlo simulations constitute the main core of this part as they provide a practical tool for studying the possible outcomes and their occurrence for a range of system parameters. Based on the attained results, the characteristics of this approach will be pointed out and generalised for more complex system structures.

After establishing the above results, Section 6.4 endeavours to incorporate the directional map quality concept with the proposed distributed mapping system from the first part and investigate the results in a large-scale environment. The synergy between different system components and the effects of employing the criteria-based mapping algorithm on the performance of the multi-vehicle mapping system will be studied in Subsection 6.4.1. Moreover, the behaviour of the covariance-based as

[1]Note that this part will use the classical metrics (such as trace of the covariance matrix) for quality assessment rather than the DIMER metric, as the focus is predominantly on efficiency in communication and computation.

well as the ground-truth-based DIMER measures in both local and global maps will be explored. Subsection 6.4.2 presents the results of employing the converted measurement Kalman filtering with debiasing compensation (D-CMKF) in the proposed multi-vehicle map making system. The impact of deploying this methodology on the map estimation process is examined through providing realistic examples and the obtained results are compared against some of the other filtering structures studied in this work.

6.2 Large-scale Distributed Mapping Simulations and Results

Simulations were conducted to evaluate the performance of the communication algorithm proposed in Chapter 3. The simulation consisted of 3 vehicles driving around in overlapping circular trajectories in an environment of 100 landmarks (see Fig. 6.1). A non-linear unicycle motion model was implemented to estimate the 2D position and orientation of each vehicle. In addition to the motion sensors, vehicles were also equipped with a range/bearing sensor which provided observations to landmarks along their line of sight.[2] Refer to Section 3.2.1 for details regarding the implementation of the local SLAM filter (LSF) inside individual vehicles.

The first part of this section's results concentrates on the landmark localisation performance. Two separate runs were performed using identical system configurations, noise parameters and observations. During the first run, none of the vehicles communicated any information to the server and each vehicle simply constructed a local map of its observed landmarks. In the second run, the vehicles communicated their map information at regular intervals with the CFC, according to the bandwidth efficient algorithm described in Section 3.3. At each communication interval, in addition to the information of newly discovered landmarks, the map information of up to $q = 10$ landmarks with the highest information gain (See Equation 3.33) was also transmitted to the CFC. In order to limit the size of the local map within the individual vehicles, a pruning algorithm with $n_{pr} = 15$ was implemented to cut the number of the LSF landmarks to 15 (see Subsection 3.3.4.1). Subsequently, the available regional map information from the CFC was communicated to each vehicle for fusion, as described in Subsection 3.3.1.

The results of the mentioned runs are illustrated in Figures 6.2 and 6.3 respectively. The resulting mean and the 3σ covariance ellipses of the estimated landmark positions are shown in each figure alongside the true location of the landmarks. Figure 6.4 shows the overlaid, enlarged view of the small rectangles shown in Figures 6.2 and 6.3. As can be seen from the figures, the landmark localisation accuracy of the multi-vehicle SLAM with periodic communication outperforms the case where no communication exists[3].

[2]The open source SLAM simulation software by Tim Bailey (available from http://www-personal.acfr.usyd.edu.au/tbailey) was modified and extended to multiple vehicles for use in simulations in this work.

[3]However, due to the selective communication performed here, the obtained map maintained at the server is relatively less accurate than if the entire local map information were to be transmitted (as in

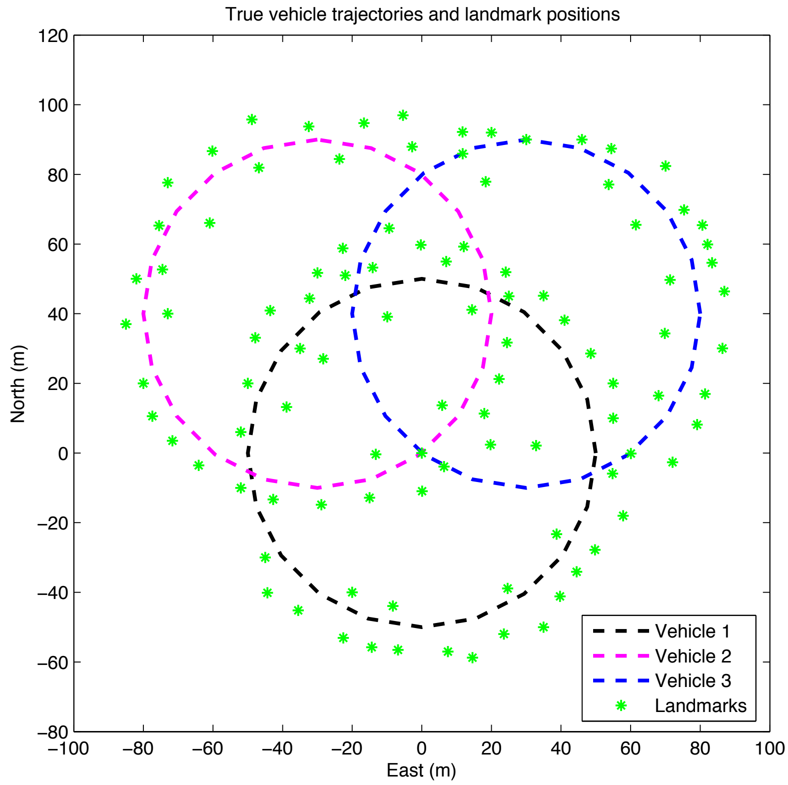


Figure 6.1: The multi-vehicle SLAM simulation consists of 3 vehicles driving around circular trajectories in an environment populated with 100 landmarks. The true vehicle trajectories and the actual position of the landmarks are depicted.

Figure 6.5 shows the uncertainty of the obtained map for different values of maximum communicated map size (q). For these initial simulations, the trace of the map covariance matrix was used as a commonly accepted measure of uncertainty estimation, cf. [Bar-Shalom et al., 2001]. The results suggest that, although the overall uncertainty decreases with the number of communicated local landmarks, the performance quickly converges towards a stable value, as the maximum communicated sub-map size increases. This constant value corresponds to the communication of the entire local maps. This is due to the fact that the algorithm dynamically selects the most informative sub-maps (landmarks with the highest information gain) to transmit. Consequently, if communicated, the landmarks which have not been observed recently and have an insignificant information contribution will have a very small effect on the quality of the global map.

Figure 6.6 shows the changes in the size of the global and local maps for the duration of the exploration when the landmark pruning algorithm from Chapter 3 is applied. As mentioned before, the pruning constant has been set to $n_{pr} = 15$ in this simulation, meaning that after each communication time, the size of the local

the approach in Bryson and Sukkarieh [2007]).

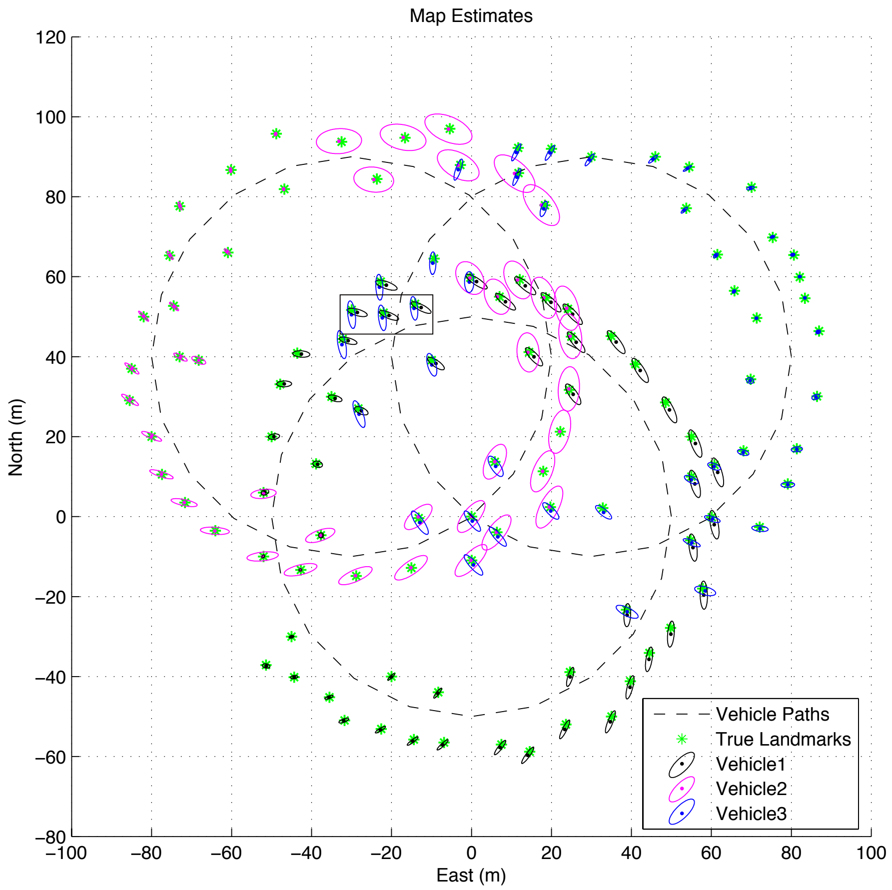


Figure 6.2: The SLAM estimates from individual vehicles (no communication). No map information is communicated between the vehicles and the CFC. The 3σ uncertainty ellipses are shown in the overhead figure.

and channel filters of each vehicle is reduced to 15 landmarks. The discarded landmarks are determined by evaluating their associated information gain. The results are shown in the case where all three vehicles close their respective loops twice. As can be seen, the pruning algorithm effectively prevents the local map from growing unboundedly. Preliminary evidence suggest that the simulation is comparable to the case where no pruning is performed in terms of performance and at some points the landmark localisation accuracy improves by applying the pruning algorithm. We believe that this improvement is likely to occur due to the fact that the CI algorithm employed inside the channel filter of individual vehicles can potentially perform better on a fewer number of landmarks. To elaborate, we argue that reducing the size of the local maps causes the CI algorithm to optimise (with respect to the defined minimisation criteria) against fewer number of map elements, thus achieving better quality for individual landmarks in some occasions. Although the demonstrated results are limited to 3 vehicles and 100 landmarks, the proposed solution is evidently

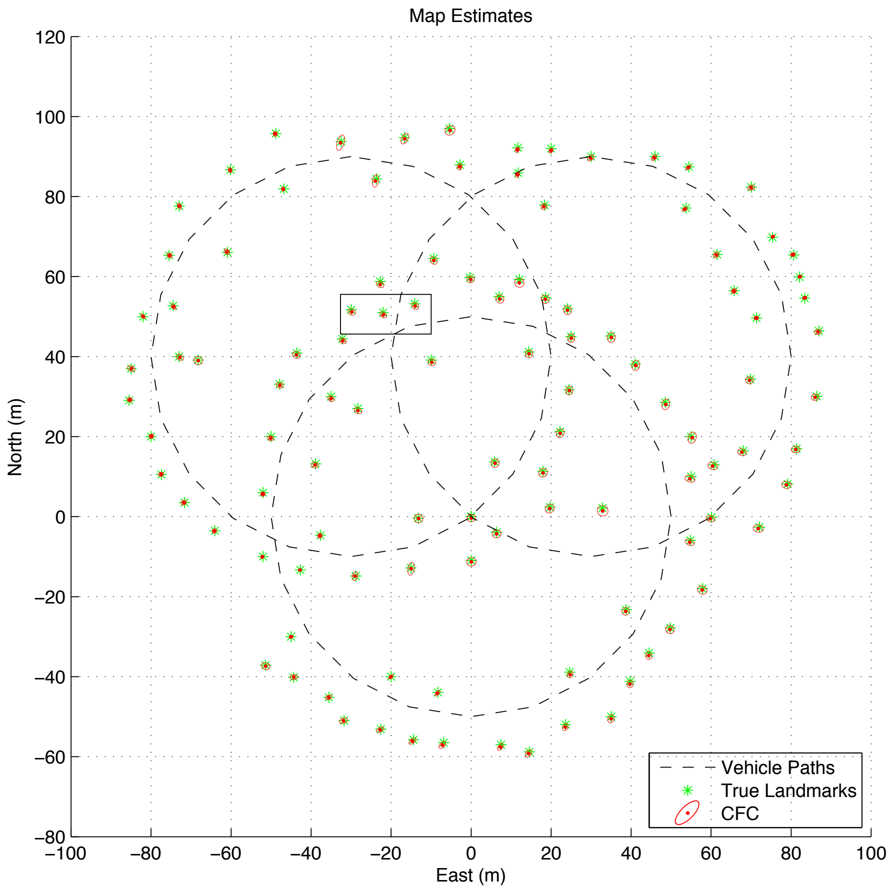


Figure 6.3: The multi-vehicle SLAM estimates obtained by selectively communicating map information to the CFC (using $q = 10$). The 3σ uncertainty ellipses are shown.

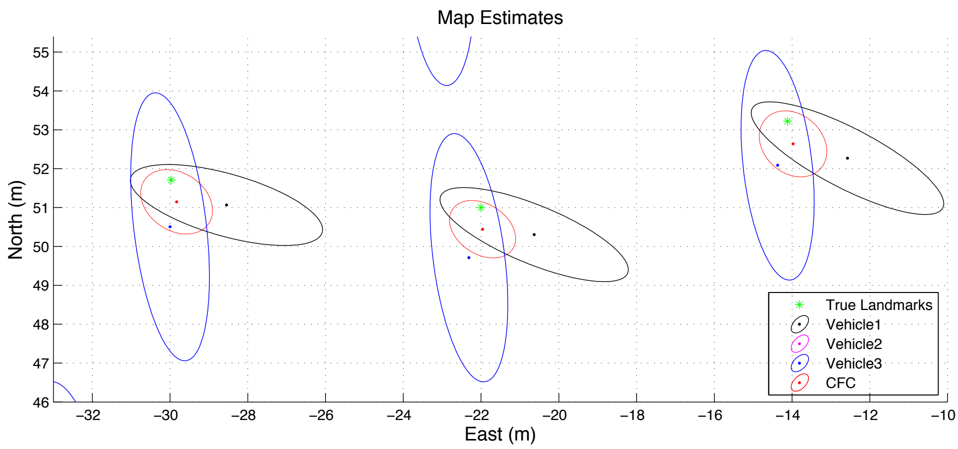


Figure 6.4: The enlarged view of the map estimates. The small rectangular boxes of Figures 6.2 and 6.3 are overlaid together. The 3σ uncertainty ellipses are shown in the figure.

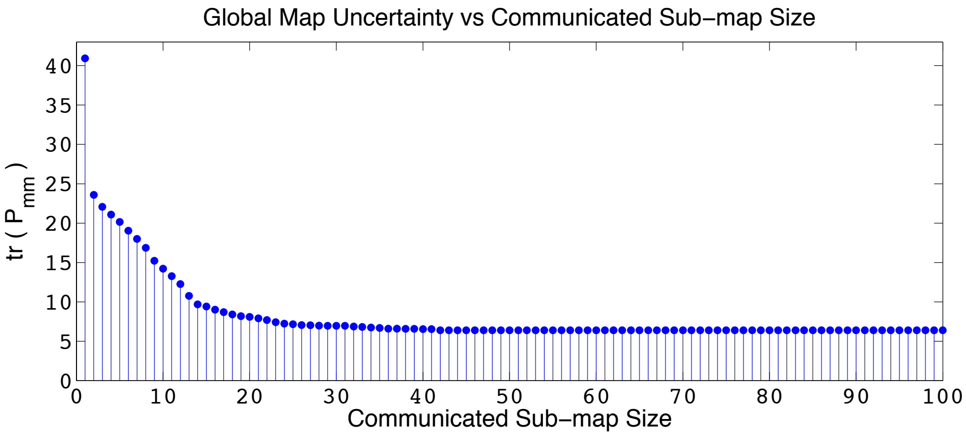


Figure 6.5: The performance of the algorithm in terms of the uncertainty of the obtained global map at the CFC for different values of maximum communicated sub-map sizes.

expandable to large-scale environments with more vehicles and many more map elements.

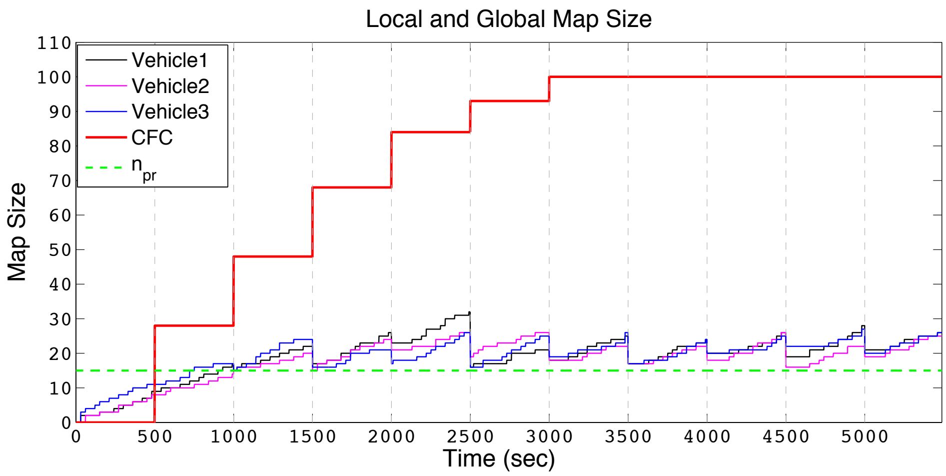


Figure 6.6: The local and global map sizes for $n_{pr} = 15$. The results are shown in the case where the vehicles close their loops twice. The vertical dashed lines demonstrate the communication times.

We conclude this section by providing a cost analysis for the solution provided in this work and a conventional communication algorithm with no selective communication (e.g. [Bryson and Sukkarieh, 2007]). For this purpose, the values provided in the scenario explained in Example 1 from Section 2.7 are used.[4] These values can be found in Table 6.1. Table 6.2 provides a summary of the calculated data regarding each strategy. Although it is assumed that our communication scenario communicates more frequently (20 times more often), the total communication cost is substantially smaller than that of [Bryson and Sukkarieh, 2007]. To reflect the actual network costs, the total communication cost is also compared after considering an overhead cost due to the carrier's flag fall fee.

n	d	m	k	b	c
10 *vehicles*	200 *km*	10 *signs*/*km*	0.1 *com.*/*km*	8 *bytes*	\$30 / *GB*

Table 6.1: Values from Example 1, Section 2.7.

	Total distance per day	Total number of landmarks per day	Landmarks in the LSF of each vehicle	Total number of communications per day	Transmitted sub-map size	Total bytes sent by each vehicle per day	Total communication cost per day (including overhead)
Bryson & Sukkarieh	$n \cdot d$	$n \cdot m \cdot d$	$m \cdot d$	$n \cdot d \cdot k$	$m \cdot d$	$b \cdot d \cdot k \cdot (m \cdot d)^2$	$c \cdot n \cdot b \cdot d \cdot k \cdot (m \cdot d)^2$
	2000	20000	2000	200 (k=0.1)	2000	6.4×10^8	\$192 (\$192)
Algorithm of Sec. 3.3	$n \cdot d$	$n \cdot m \cdot d$	n_{pr}	$n \cdot d \cdot k$	q	$b \cdot d \cdot k \cdot q^2$	$c \cdot n \cdot b \cdot d \cdot k \cdot q^2$
	2000	20000	20	4000 (k=2)	10	320000	\$0.0096 (\$7.68)

Table 6.2: A numerical example to compare the algorithm in [Bryson and Sukkarieh, 2007] with the solution provided in this thesis. The example uses the values provided in Table 6.1 and assumes $q = 10$ and $n_{pr} = 20$ for the selective communication method. For the definition of different variables please see Example 1 in Section 2.7.

6.3 Criteria-based Estimation Simulations and Results

This part of the results chapter pursues two main goals. Firstly, it demonstrates the use of the newly designed directional map error (DIMER) metric introduced in Chapter 4. Although conventional methods (such as trace) were briefly used in Section 6.2 of this chapter for quantifying the estimation error, the rest of this chapter will typically use the DIMER metric as the main tool for map quality assessment. Secondly, this section presents the simulation results of the criteria-based estimation and mapping algorithms outlined in Chapter 5. Subsection 6.3.1 addresses the DIMER-based covariance intersection for the fusion of two maps with unknown correlation. The

[4] Note that the values used in this example are not related to the values used in the simulation.

results are compared against two classical cases where the trace and determinant of the map covariance matrix are used as the minimisation criteria. Subsection 6.3.2 is concerned with the simulations and results pertaining to criteria-based EKF-SLAM algorithm. The validity and effectiveness of the DIMER-based covariance trajectory perturbation (CTP) (devised in Section 5.4) in enhancing the landmark localisation accuracy in road applications is investigated in detail.

The simulations in this section used the same prediction and observation models as those of Section 6.2 (see Section 3.2.1 for more details about this implementation). However, a slight modification was made to incorporate the landmark orientation estimates into the structure of the local SLAM filter. In other words, the originally developed position-only filter was expanded to a pose estimation filter for the remaining simulations in this chapter[5]. Recalling Chapter 4, this information is used to infer the road heading and is essential in evaluating the DIMER metric for a given map element. Like before, an arrow vector is used to depict the landmark orientation in this chapter.

6.3.1 DIMER-based Covariance Intersection (CI) Results

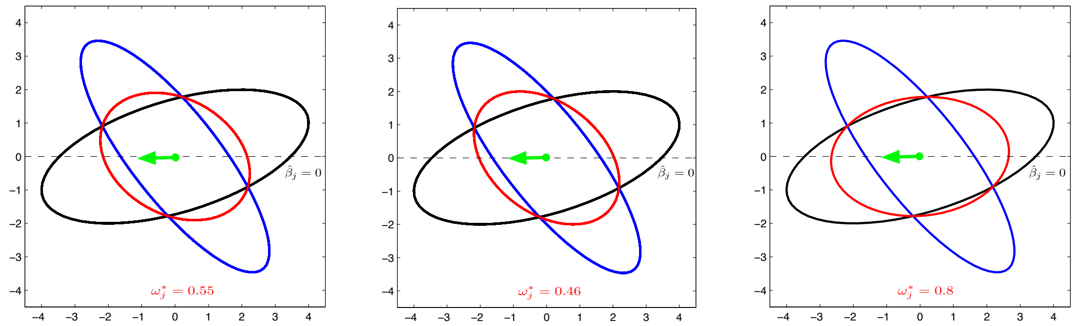
Figure 6.7 illustrates the results of utilising the CI algorithm for fusing two map estimates (corresponding to the same landmark) with unknown degree of dependency. The subfigures compare the obtained maps when different minimisation criteria were used in the calculation of the fused map. Subfigures (a) and (b) demonstrate the results related to the use of the more conventional trace and determinant criteria respectively, while Subfigure (c) shows the outcome when the new covariance-based DIMER metric was applied as the minimisation criterion. This example assumes a horizontal road with $\hat{\beta}_j = 0$ (this can be inferred from the landmark orientation estimate). In addition, $C_{\parallel}$ and $C_{\perp}$ were set to 0 and 2 respectively, meaning the perpendicular error was to be minimised. As can be seen, the CI algorithm picks the consistent error ellipse with the lowest perpendicular error amongst all possible consistent results. The graphs presented in Figure 6.8 show the behaviour of each covariance-based measure for different values of weighting factor ω. It can be seen that the optimal value of ω is different for each minimisation criterion.

Simulations were conducted to extend the above notion to larger environments with several landmarks. Consider the problem of combining two maps (each map is comprised of pose estimates corresponding to a group of landmarks) in order to attain a single, high quality fused map. It is assumed that the degree of dependency among these two maps is not known. Similar to the above analysis for a single landmark, the covariance-based DIMER measure is used as the minimisation criteria in the covariance intersection algorithm. Figure 6.9 provides an example of fusing two maps consisting of 20 landmarks[6]. The original and resulting estimates are illustrated in the figure. Results witnessed a 12 percent decrease in the calculated DIMER metric

[5] We do not dwell on the particulars of landmark orientation estimation in this thesis.

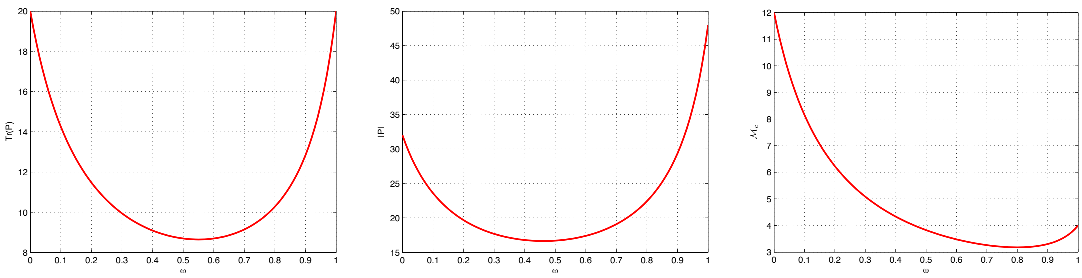
[6] Although the results are shown for a map consisting of 20 landmarks, it can easily be generalised to very-large-scale environments.

associated with the fused map compared to the case where the covariance trace was used as the optimisation criterion. The enlarged view of two of the map elements of Figure 6.9 can be seen in Figure 6.10. A comparison between the trace-based CI and the DIMER-based CI can be seen in the figure. It can be seen that the DIMER-based solution results in a relatively lower perpendicular error.



(a) Minimal-trace CI (b) Minimal-determinant CI (c) Minimal-DIMER CI

Figure 6.7: Comparison between the minimal-trace, minimal-determinant and minimal-DIMER covariance intersection in fusing two sample covariance ellipses. The road pertaining to the sample landmark j is assumed to be horizontal ($\hat{\beta}_j = 0$) and the directional priorities were set to $C_{\parallel} = 0$ and $C_{\perp} = 2$.



(a) Trace metric w.r.t. ω (b) Determinant metric w.r.t. ω (c) DIMER metric w.r.t. ω

Figure 6.8: The behaviour of different covariance-based measures for different values of weighting factor ω. The graphs correspond to the plots shown in Fig. 6.7.

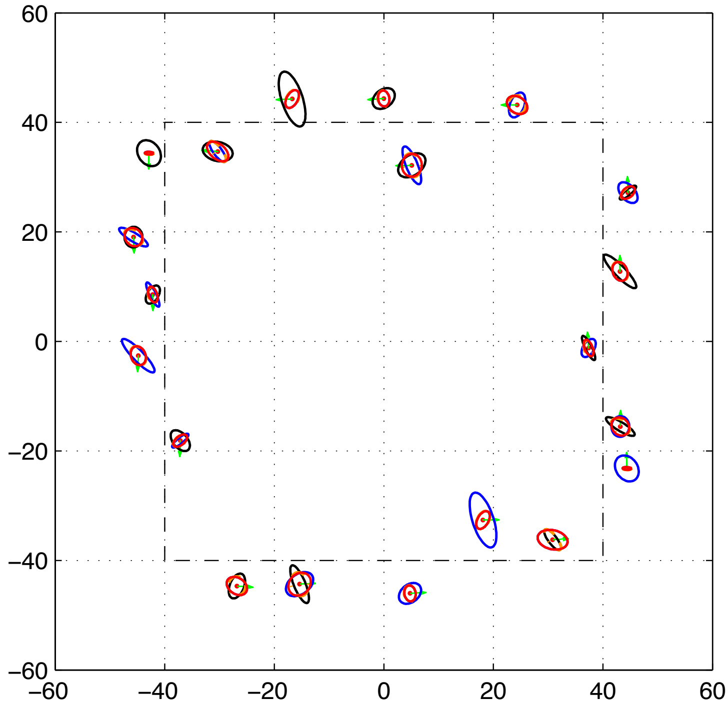


Figure 6.9: DIMER-based Covariance Intersection for fusing two maps with unknown degree of correlation. The directional priorities were set to $C_{\parallel} = 0$ and $C_{\perp} = 2$ for this example.

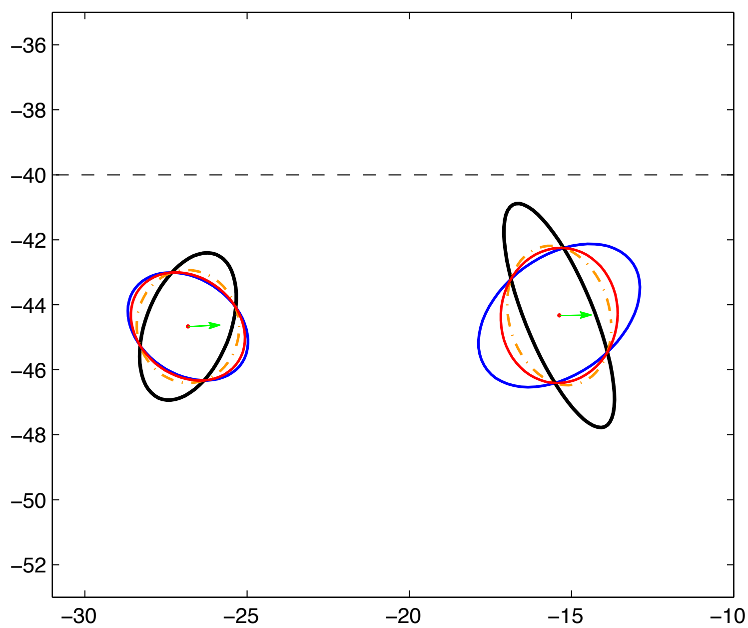


Figure 6.10: DIMER-based Covariance Intersection. The enlarged view of the map estimates in Figure 6.9. The dotted (orange) ellipses represent the optimal-trace and the solid (red) ellipses represent the resulting optimal-DIMER covariance matrix. The DIMER-based solution results in a lower perpendicular error with respect to the road.

6.3.2 Criteria-based EKF-SLAM Simulation Results

Prior to presenting the results of the covariance trajectory perturbation algorithm, we carry out a preliminary experiment in order to clarify the map quality concept in typical SLAM scenarios.

6.3.2.1 Preliminary Map Quality Analysis

This simple simulation which includes a single stationary landmark being observed by a moving vehicle aims at demonstrating the typical manner in which the estimated position mean and its associated covariance ellipse evolve as the number of observations increases. This evolution is shown in Figure 6.11. The vehicle is traversing a horizontal path ($\beta_j = 0$) from left to right and it detects a landmark at 5 occasions by capturing a range and a bearing measurement for each observation. These measurements are processed by the nonlinear EKF-SLAM algorithm to generate the mean and covariance estimates. The solid ellipse in Fig. 6.11 represents the final estimate for the map element after the last observation. The salient point here is that the final mean estimate and covariance ellipse are influenced by various factors. Most notably, the measurement noise has a direct impact on the orientation and size of the resulting covariance ellipses. Furthermore, as can be inferred from the figure, the spatial relativity and geometry of the vehicle and landmark pair with respect to each other at the time of observation is crucial in determining the final map estimate. The individual estimates of Fig. 6.11 after each observation are shown in Figure 6.12. We also monitor the covariance-based and ground-truth-based DIMER measures contained in individual estimates. The directional priorities are assumed to be $C_{\parallel} = 2$ and $C_{\perp} = 0$ for this particular example. The evolution of trace and the covariance-based DIMER metrics are depicted in Figure 6.13, while the classical squared distance and the ground-truth based DIMER metrics are shown in Figure 6.14. As can be seen, both of the covariance-based errors are monotonically diminishing as the number of observations grow. However, that is not the case for the ground-truth-based errors of Fig. 6.14[7], although the general trend seems to be decreasing. This pattern is most probably related to the inherited nonlinearities in the system under study and the infamous bias problem. To elaborate, the EKF algorithm is designed to work around minimising the step-by-step covariance of the estimates with respect to a certain criterion (conventionally the trace and more recently the covariance-based DIMER metric). Although the principal philosophy behind this minimisation is to reduce a form of distance with respect to the true values of the system states (the expected value of this error, to be more precise), the nonlinearities of the system may sometimes prevent the materialisation of this goal in full. In other words, minimising the variance in a nonlinear filter does not necessarily mean that the actual errors are going to be reduced at every filter step and for any given set of measurements[8]. Once again we

[7]Especially since these figures are plotted for a single sample of measurements. It is expected that the expected values of these parameters exhibit a more consistent behaviour.

[8]This was also discussed in Chapter 5 when addressing the general trade-off between bias, variance and mean squared error.

remark that the linear Kalman filter (which is unbiased) is an exception where the variance minimisation coincides exactly with the minimisation of the mean-squared error for a large number of samples.

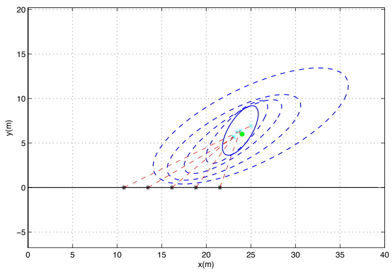
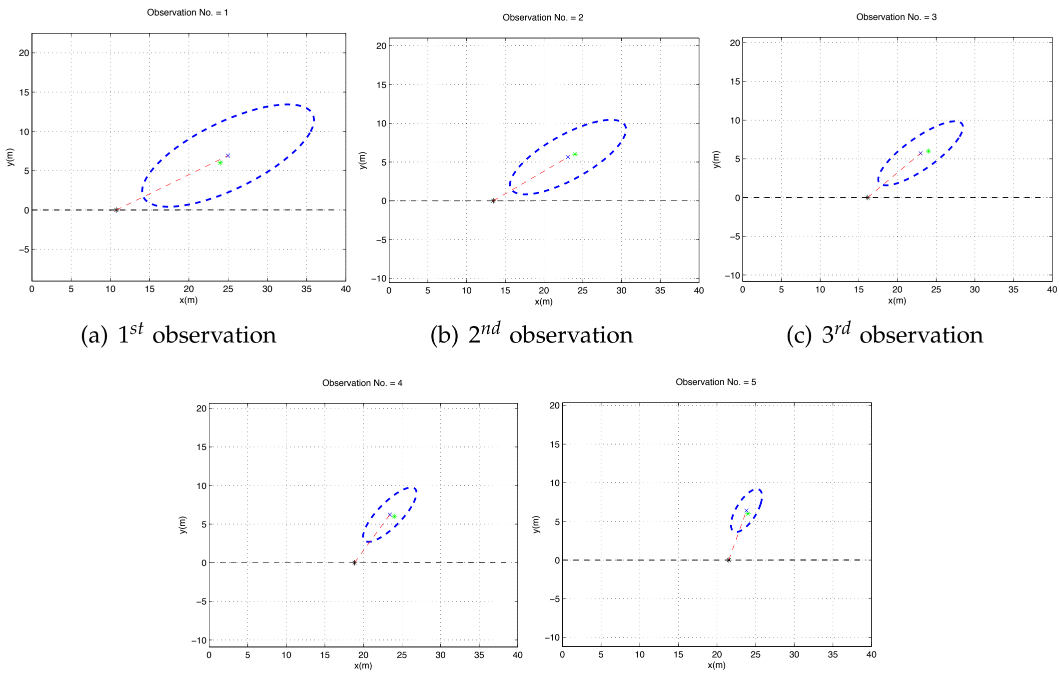


Figure 6.11: The evolution of the estimated position mean and its associated covariance ellipse for a scenario comprising of a moving vehicle and a stationary landmark. Range and bearing with respect to the landmark are measured at each observation. The solid ellipse represents the final map estimate.



(a) 1^{st} observation (b) 2^{nd} observation (c) 3^{rd} observation

(d) 4^{th} observation (e) 5^{th} observation

Figure 6.12: Individual estimates of Fig. 6.11. It can be detected that the orientation and size of the resulting error ellipses depend on the relativity between the moving vehicle and the stationary landmark at observation times.

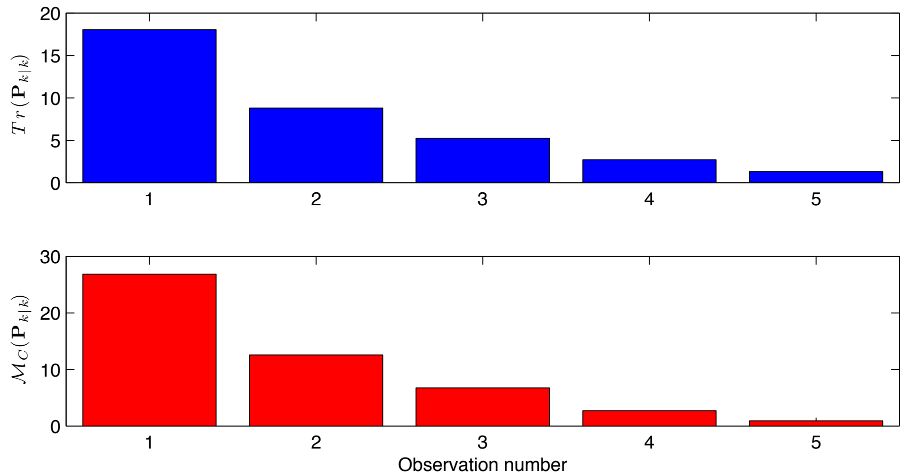


Figure 6.13: The step-by-step behaviour of the covariance-based errors contained in the map estimates of Fig. 6.12. The DIMER metric is calculated based on the assumption that $C_{\|} = 2$ and $C_{\perp} = 0$. Both measures decrease monotonically as the number of landmark observations increase.

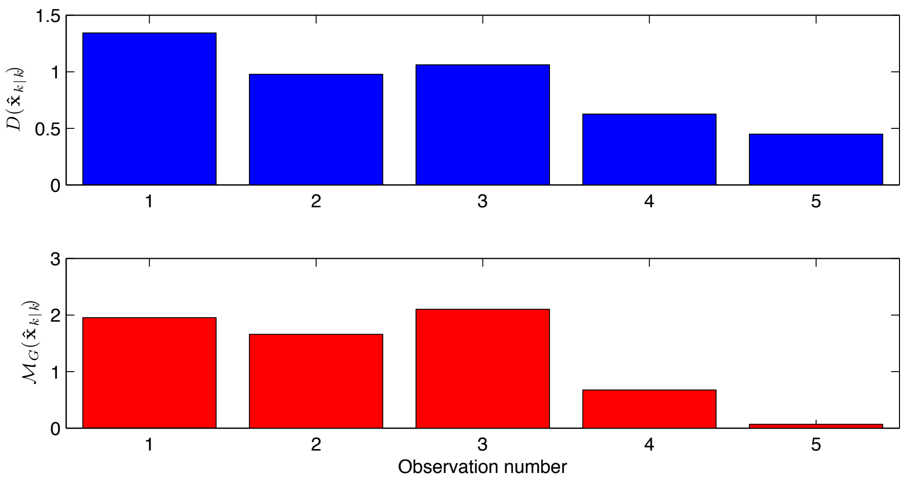


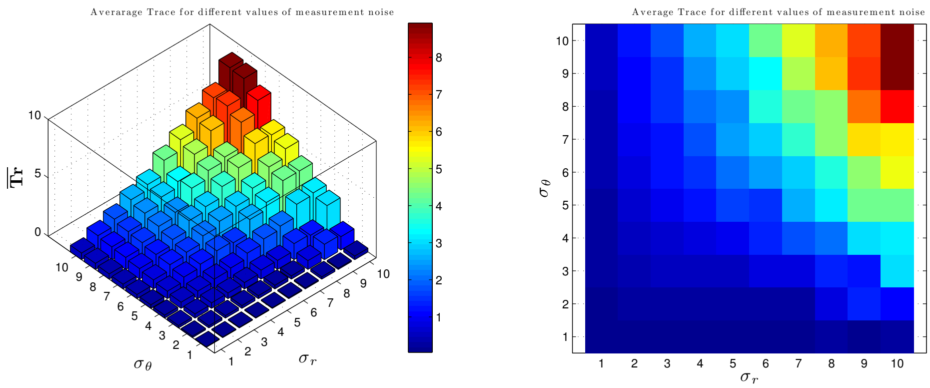
Figure 6.14: The step-by-step behaviour of the ground-truth-based errors contained in the map estimates of Fig. 6.12. The DIMER metric is calculated based on the assumption that $C_{\|} = 2$ and $C_{\perp} = 0$. Although the general trend is of descending order, the behaviour is more random and not monotonic, unlike the covariance-based error metrics.

6.3.2.2 Map quality Analysis Based on Measurement Noise

This subsection examines the general impact of observation noise on landmark localisation accuracy through the implementation of Monte Carlo simulations. The same single-vehicle/single-landmark scenario from Subsection 6.3.2.1 was considered for this experiment. Once again, the directional priorities were set to $C_{\parallel} = 2$ and $C_{\perp} = 0$. The directional map error (DIMER) metrics as well as the state-of-the-art errors pertaining to the final obtained map at time κ were calculated and compared for different values of range and bearing standard deviation. The considered values for these measurement noise levels were $\sigma_r \in \{1, 2, \ldots, 10\}$ and $\sigma_\theta \in \{1, 2, \ldots, 10\}$. For each pair of $(\sigma_r, \sigma_\theta)$ the EKF-SLAM filter was run 1000 times (same level of noise but resampled set of measurements) and the average error in the final map was calculated using four different measures. Figure 6.15 shows the trace of the map covariance matrix (averaged over 1000 samples) for all the possible values of σ_r and σ_θ in the above interval. A heat diagram as well as a 3D bar chart have been used to visualise the simulation results. As can be observed from the graphs, the trace metric increases almost monotonically as the measurement noise level grows. Figure 6.16 illustrates the covariance-based DIMER metric for different levels of measurement noise using the same Monte Carlo simulation. Despite the presence of directional priorities, $\bar{\mathcal{M}}_C$ exhibits a similar behaviour to that of trace.

The behaviours of the average Euclidean distance and the average ground-truth-based DIMER metric were also compared in Figures 6.17 and 6.18 respectively. Note that the squared value of the DIMER metric is shown in the figures in virtue of having a more sensible scaling system which is comparable with the Euclidean distance. As can be seen, both of the plotted ground-truth-based error metrics follow the same pattern and are generally increasing for larger range and bearing noise levels. However, the extremum of these distance errors occur when the range standard deviation is at its highest level, while the bearing standard deviation is small. Based on conjecture, this is probably related to the geometry of the problem and the relativity of the vehicle and the landmark at different observation times.

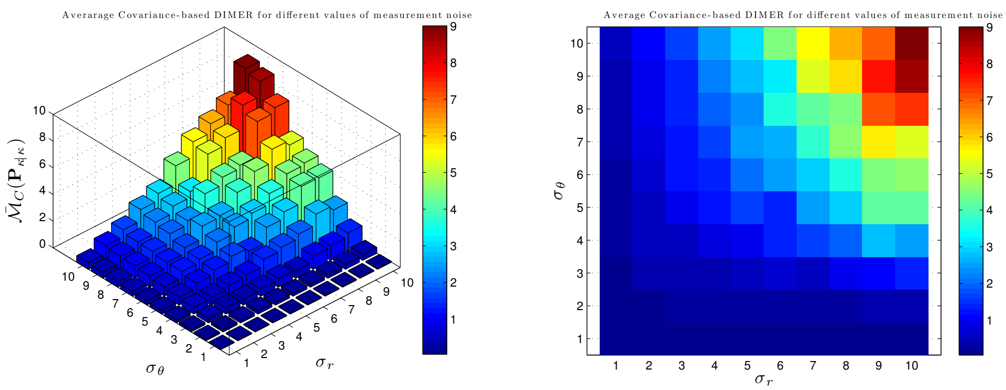
After providing this preliminary map error analysis, we now present the simulation results of covariance trajectory perturbation algorithm in the next part. As mentioned in Chapter 5, this method which is based on targeted modification of the original EKF-SLAM algorithm aims at improving the quality of obtained maps based on particular error criteria.



(a) $\overline{\mathbf{Tr}}(\mathbf{P}_{\kappa|\kappa})$ (3D bar chart)

(b) $\overline{\mathbf{Tr}}(\mathbf{P}_{\kappa|\kappa})$ (Heat diagram)

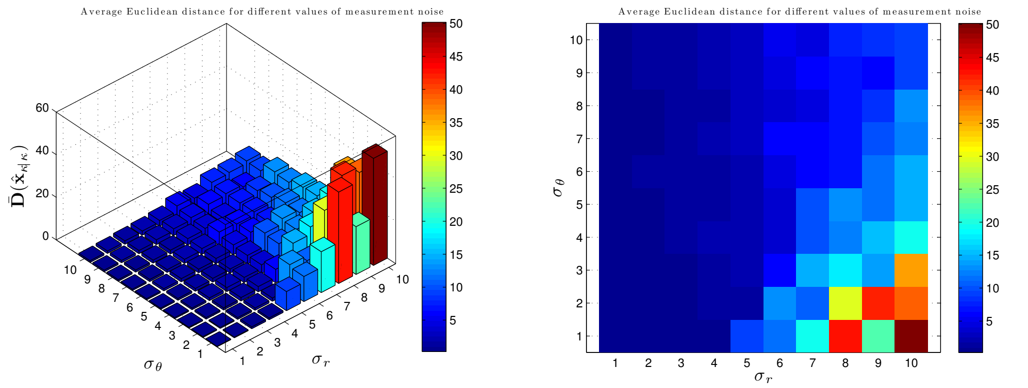
Figure 6.15: Average Trace ($\overline{\mathbf{Tr}}(\mathbf{P}_{\kappa|\kappa})$) for different values of measurement noise



(a) $\bar{\mathcal{M}}_C(\mathbf{P}_{\kappa|\kappa})$ (3D bar chart)

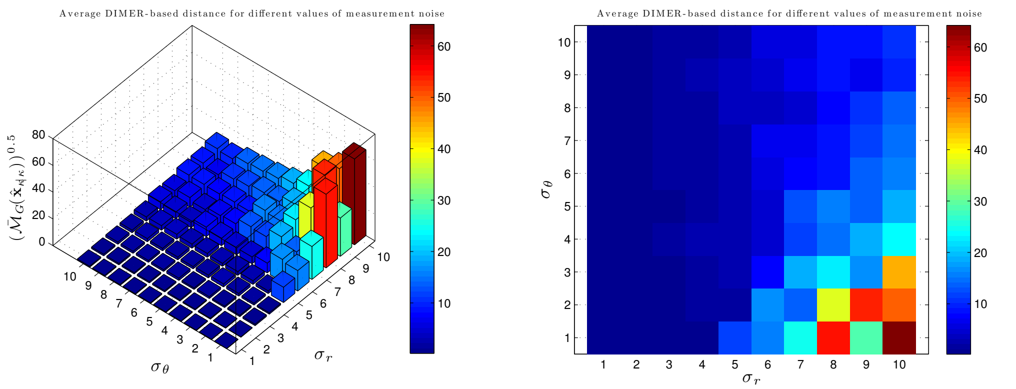
(b) $\bar{\mathcal{M}}_C(\mathbf{P}_{\kappa|\kappa})$ (Heat diagram)

Figure 6.16: Average Covariance-based DIMER ($\bar{\mathcal{M}}_C(\mathbf{P}_{\kappa|\kappa})$) for different values of measurement noise



(a) $\bar{\mathbf{D}}(\hat{\mathbf{x}}_{\kappa|\kappa})$ (3D bar chart) (b) $\bar{\mathbf{D}}(\hat{\mathbf{x}}_{\kappa|\kappa})$ (Heat diagram)

Figure 6.17: Average Ground-truth-based DIMER ($\bar{\mathbf{D}}(\hat{\mathbf{x}}_{\kappa|\kappa})$) for different values of measurement noise



(a) $\sqrt{\bar{\mathcal{M}}_G(\hat{\mathbf{x}}_{\kappa|\kappa})}$ (3D bar chart) (b) $\sqrt{\bar{\mathcal{M}}_G(\hat{\mathbf{x}}_{\kappa|\kappa})}$ (Heat diagram)

Figure 6.18: Average Ground-truth-based DIMER ($\sqrt{\bar{\mathcal{M}}_G(\hat{\mathbf{x}}_{\kappa|\kappa})}$) for different values of measurement noise

6.3.3 Criteria-based Covariance Trajectory Perturbation Simulation Results

This Subsection is concerned with the results of criteria-based EKF-SLAM simulations, particularly the covariance trajectory perturbation algorithm introduced in Section 5.4. The effectiveness of this method in reducing the covariance-based DIMER metric in a typical nonlinear SLAM system will be shown and the consequent impact on its counterpart, the ground-truth-based DIMER measure will be inspected. Finally, through the employment of Monte Carlo simulations, some of the key properties of this approach will be pointed out based on the behaviour of the acquired results.

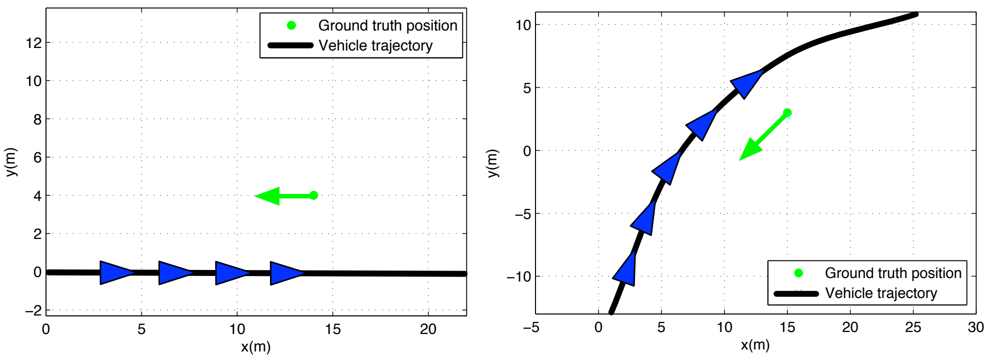
CTP Simulation Setup

Once again, to highlight the essence of the covariance trajectory perturbation results, simplified scenarios consisting of a single vehicle and a single landmark are considered[9]. For this analysis, the focus will be on the real-world road application studied in this thesis. Therefore, two of the most commonly occurring road structures in the real world will be investigated for the trajectory of the mobile vehicle. The nominated trajectory types for this section are straight path and curved path which are shown in Figure 6.19 [10]. The examples in this section have been set up in a way that they resemble real world situations in which a road sign is being detected by a moving vehicle over consecutive observations. The position of the vehicle at each observation time has been depicted in the figure using a triangle. At each observation time k, $k \in \{1, \ldots, \kappa\}$, the vehicle collects a measurement z_k in the form of a range and a bearing value with respect to the landmark. Let $\mathcal{Z}_\kappa \triangleq \{z_i\}_{i=1}^{\kappa}$ be the combined vector of all measurements. Given $\mathcal{Z}_\kappa$ and the system models, the EKF-SLAM algorithm is deployed to estimate a position mean and a covariance matrix corresponding to the detected landmark.

A simulation was implemented to apply the covariance trajectory perturbation algorithm to the original filter structure (outlined above). As elaborated in Section 5.4, the CTP algorithm searches for an appropriate perturbation $\Delta\mathbf{P}$ that minimises the covariance-based DIMER metric and improves the quality of the map compared to the standard filtering scheme with no perturbation. Fundamentally, the perturbation can be applied prior to each filter update. Nonetheless, in the upcoming simulations $\Delta\mathbf{P}$ is designed so that it is applied after the first update equation and prior to the incorporation of the second set of measurements z_2. In this way, the filter has enough time to cope with the injected uncertainty and settle down during the next observations to eventually converge to the final map estimate. The error contained in the final map estimate (after integrating the last set of measurements) is the quantity we aim to minimise in the provided examples. Like before, the covariance-based DIMER metric is utilised for map error calculation in the CTP context. Unless mentioned otherwise, the directional priorities for the simulations of this section are considered to

[9]This will be expanded to multiple vehicle and multiple landmarks later in this chapter.

[10]Although other road geometry structures can be seen in the real world, analysing the mentioned trajectories has been deemed sufficient to highlight the results of this section.



(a) Straight Trajectory

(b) Curved Trajectory - this figure needs to be changed

Figure 6.19: Test scenarios for analysing the covariance trajectory perturbation method. These scenarios are two of the most commonly occurring situations in the real world. The true landmark orientation in each figure has been shown using a green arrow.

be $C_{\|} = 2$ and $C_{\perp} = 0$.

The two-dimensional perturbation matrix $\Delta\mathbf{P}$ needs to be positive definite in order not to sabotage the consistency of the system. Initially, $\Delta\mathbf{P}$ is parametrised according to

$$\Delta\mathbf{P} = \begin{bmatrix} \delta_x & 0 \\ 0 & \delta_y \end{bmatrix} \tag{6.1}$$

and the search is carried out along the matrix diagonal elements with $\delta_x^k, \delta_y^k > 0$. For the simulations of this section, the search domain for the free diagonal variables was set to $\delta_x \in (0, 100]$ and $\delta_y \in (0, 100]$. This interval which ensures the positive definiteness of $\Delta\mathbf{P}$ was initially chosen based on intuition. It was observed that searching in wider intervals had negligible impact on localisation quality and yielded similar outcomes. We infer that the relative value of the search parameters δ_x^k and δ_y^k (rather that their absolute values) is decisive in determining the error contained in the final map.

For each eligible perturbation matrix $\Delta\mathbf{P}$ in the above search domain the EKF-SLAM filter was run (in a batch mode) using the given measurements $\mathcal{Z}_\kappa$ under the same model parameters and initial values. The covariance-based DIMER metric $\mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})$ was evaluated for each of the final map estimates (after the last set of measurements at time κ). The $\Delta\mathbf{P}$ matrix yielding the lowest $\mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})$ is sought in this search process. We introduce the following notations pertaining to the covariance trajectory perturbation algorithm which are used throughout this section:

- $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa})$ denotes the value of the covariance-based DIMER metric associated with the final map (after κ measurements) when no perturbation is considered ($\Delta\mathbf{P} = 0$). The final map obtained using this method is equivalent to the results of the original EKF-SLAM algorithm. Therefore,

$$\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) \triangleq \mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})\big|_{\Delta\mathbf{P}=0} \tag{6.2}$$

The ground-truth-based DIMER measure corresponding to this map is denoted $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa})$. Therefore we have

$$\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) \triangleq \mathcal{M}_G(\hat{\mathbf{x}}_{\kappa|\kappa})\big|_{\Delta\mathbf{P}=0} \tag{6.3}$$

- $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa})$ denotes the optimal value of the covariance-based DIMER metric associated with the final map (after κ measurements). This value is obtained by applying the optimal perturbation matrix $\Delta\mathbf{P}^*$ to the filter update equations yielding the lowest map error. Therefore,

$$\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) \triangleq \mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})\big|_{\Delta\mathbf{P}=\Delta\mathbf{P}^*} \tag{6.4}$$

The ground-truth-based DIMER measure corresponding to this map is denoted $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa})$. Therefore we have

$$\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) \triangleq \mathcal{M}_G(\hat{\mathbf{x}}_{\kappa|\kappa})\big|_{\Delta\mathbf{P}=\Delta\mathbf{P}^*} \tag{6.5}$$

□

Conducting numerous experiments with different system parameters suggest that finding an optimal, non-zero $\Delta\mathbf{P}^*$ cannot always be guaranteed and the perturbation search is dependant on various factors such as the noise levels, geometry of the road, system model and the nature of the captured measurements. Nevertheless, in situations in which a valid covariance perturbation is found, the covariance-based quality measure for the generated map (i.e. $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa})$) is enhanced compared to the standard EKF-SLAM solution without the presence of covariance perturbation (i.e. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa})$). Therefore, the worst case scenario (the largest error when judged using the covariance-based DIMER metric) is always identical to the result obtained using the standard filter.

DIMER-based CTP Analysis and Results

We now present the results of utilising the covariance perturbation algorithm in the real-world mapping situations shown in Fig. 6.19. For each nominated trajectory (straight and curved) we designed a series of experiments to examine the behaviour of the covariance-based DIMER metric (as well as the ground-truth based DIMER metric) of the acquired map for different values of the perturbation matrix $\Delta\mathbf{P}$.

In general, three different scenarios might occur when applying the CTP algorithm to a mapping problem:

Case (I) - The CTP algorithm fails to improve the quality of the map.

Case (II) - The CTP algorithm decreases the covariance-based DIMER, but increases the ground-truth-based DIMER measure.

Case (III) - The CTP algorithm decreases both the covariance-based and the ground-truth-based DIMER measures.

For each of the nominated trajectories shown in Fig. 6.19, we provide an example of the above potential cases. For all the examples in this section, the range and bearing standard deviations have been set to $\sigma_r = 7m$ and $\sigma_\theta = 5^\circ$ respectively. The measurements for each case are re-sampled from zero-mean normal distributions given by these standard deviations.

Straight Path Analysis

Case $(\mathbf{I})$: CTP fails to improve the quality of the map

Figure 6.20 shows the resulting position mean and the 3σ covariance ellipse obtained by employing the standard Kalman filter (with no covariance perturbation) for an example set of measurements collected by a moving vehicle on the straight path shown in Figure 6.19(a). These noisy range and bearing measurements are artificially generated by adding random noise values sampled from normal distribution functions with $\sigma_r = 7m$ and $\sigma_\theta = 5^\circ$ to the true values.

Figure 6.21 visualise the $\Delta\mathbf{P}$ search process for this example. This 3D graph shows the calculated $\mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})$ for different values of δ_x and δ_y. The flat (blue) surface represents the DIMER measure for the standard filter (i.e. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa})$), while the other (red) surface is comprised of the DIMER measures corresponding to non-zero $\Delta\mathbf{P}$ values in the search interval.

As can be observed, all valid perturbation values yield higher DIMER measures compared to the original EKF-SLAM filter (where $\Delta\mathbf{P} = 0$). Therefore,

$$\forall \ \Delta\mathbf{P}\Big|_{\delta_y \in (0,100]}^{\delta_x \in (0,100]} : \mathcal{M}_C^{\Delta\mathbf{P}}(\mathbf{P}_{\kappa|\kappa}) > \mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) \tag{6.6}$$

In other words, for this particular instance of range and bearing measurements (which are instantaneous samples of a normally distributed function), the CTP algorithm is unable to improve the directional quality of the generated map. Consequently, the non-perturbed error ellipse shown in Fig. 6.20 has the lowest directional error compared to the perturbed results, effectively making the CTP algorithm a fruitless effort in this case.

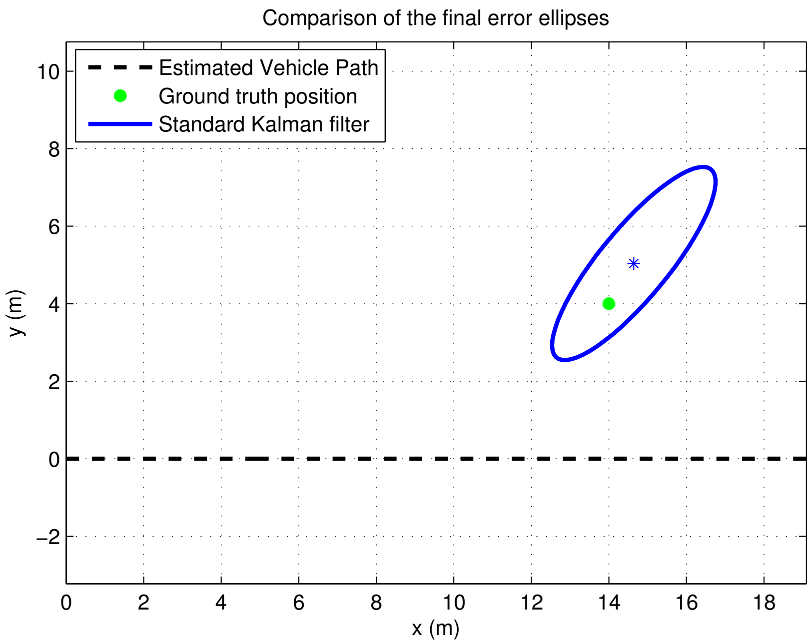


Figure 6.20: Case (**I**) - Straight trajectory: The 3σ covariance ellipse obtained by applying the standard EKF-SLAM filter (with no perturbation) to an example scenario. The DIMER measures associated with this estimate are $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 0.55973$ and $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.83641$.

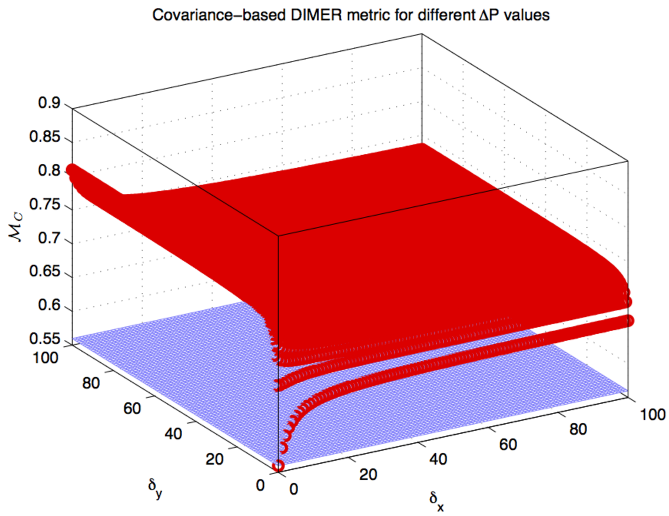


Figure 6.21: Case (**I**) - Straight trajectory: The behaviour of the covariance-based DIMER measure $\mathcal{M}_C$ for different values of $\Delta\mathbf{P}$ in the search interval. No non-zero $\Delta\mathbf{P}^*$ could be found in this case. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 0.55973$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.55973$. $C_{\|} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

Case (II): CTP reduces the covariance-based DIMER, but deteriorates the ground-truth-based DIMER measure.

A different set of range and bearing values (compared to Case (I) explained above) are examined in this example for the scenario shown in Fig. 6.19(a). Like before, these range and bearing noises are artificially generated by sampling normal distribution functions with $\sigma_r = 7m$ and $\sigma_\theta = 5^\circ$ respectively.

Figure 6.22 visualises the $\Delta\mathbf{P}$ search process for this example. Unlike Case (I), there are numerous values for the pair (δ_x, δ_y) where the resulting covariance-based DIMER metric is below the blue surface representing $\mathcal{M}^0_C(\mathbf{P}_{\kappa|\kappa})$. Recalling the optimisation problem expressed using Eq. (5.31), we seek a $\Delta\mathbf{P}^*$ that achieves the lowest directional error. In this example, the optimal DIMER value, $\mathcal{M}^*_C(\mathbf{P}_{\kappa|\kappa})$, occurs at point $\delta^*_x = 100$ and $\delta^*_y = 0$.

Figure 6.23 examines the behaviour of the ground-truth-based DIMER measure for different values of $\Delta\mathbf{P}$. The flat (blue) surface in this graph represents the ground-truth-based DIMER measure for the standard filter, i.e. $\mathcal{M}^0_G(\hat{\mathbf{x}}_{\kappa|\kappa})$. Despite the effectiveness of the CTP method in reducing the covariance-based DIMER measure, the presence of the injected perturbation deteriorates the ground-truth-based directional error compared to the standard EKF-SLAM filter. Notwithstanding this, as can be seen from the figure, the error associated with the optimal perturbation matrix (i.e. $\mathcal{M}^{\Delta\mathbf{P}^*}_G(\hat{\mathbf{x}}_{\kappa|\kappa})$) is only somewhat worse than the error contained in the non-perturbed mean estimate $\mathcal{M}^0_G(\hat{\mathbf{x}}_{\kappa|\kappa})$.

The relation between the estimation results of the standard EKF-SLAM filter (zero perturbation) and the perturbed Kalman filter (perturbed by $\Delta\mathbf{P}^*$) is shown in Figure 6.24. The decrease in the covariance-based parallel error and the slight increase in the ground-truth-based parallel error caused by applying the CTP method are observable in the figure (since the directional priorities are set to $C_\parallel = 2$ and $C_\perp = 0$, the aim is to minimise the error associated with the parallel direction).

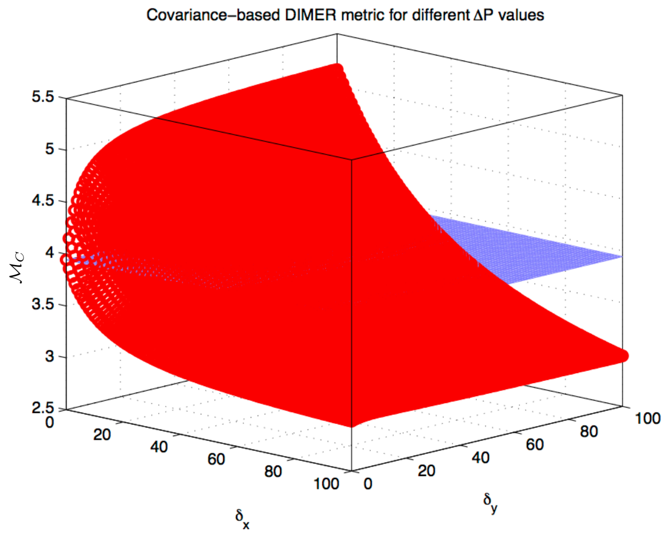


Figure 6.22: Case (**II**) - Straight trajectory: The behaviour of the covariance-based DIMER measure $\mathcal{M}_C$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ occurred at $\delta_x^* = 100$ and $\delta_y^* = 0$. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 3.9441$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 2.9812$. $C_{\parallel} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

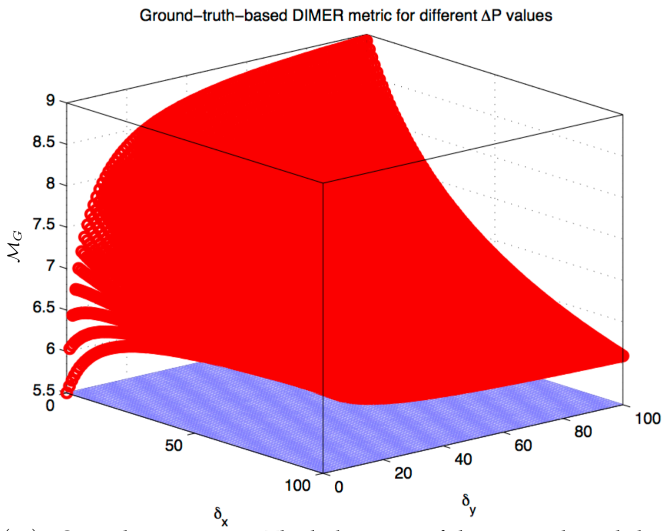


Figure 6.23: Case (**II**) - Straight trajectory: The behaviour of the ground-truth-based DIMER measure $\mathcal{M}_G$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ has had a deteriorating effect on the $\mathcal{M}_G$ metric compared to the standard case. $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 5.5021$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 6.5359$. $C_{\parallel} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

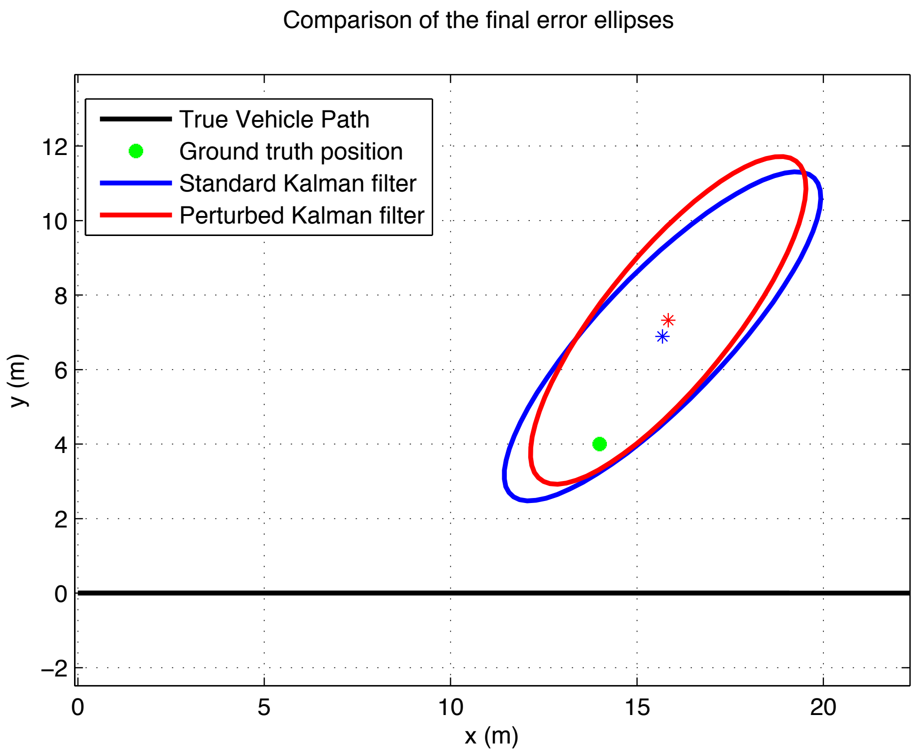


Figure 6.24: Case (**II**) - Straight trajectory: The effect of applying the CTP algorithm on the mean and the 3σ ellipse associated with the map estimate. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 3.9441$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 2.9812$; $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 5.5021$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 6.5359$. $\delta_x^* = 100$; $\delta_y^* = 0$. $C_\parallel = 2$; $C_\perp = 0$.

Case (III): CTP reduces both the covariance-based and the ground-truth-based DIMER measures.

We now provide an example of the third potential case. This case, which is the ideal scenario, demonstrates the improvement in both covariance-based and ground-truth-based DIMER measures. Similar to Cases I & II, the range and bearing measurements are samples from same normally distributed functions with $\sigma_r = 7m$ and $\sigma_\theta = 5°$.

Figure 6.25 visualises the $\Delta\mathbf{P}$ search process for this example. Similar to Case (II), there are several values for the pair (δ_x, δ_y) where the resulting covariance-based DIMER metric ($\mathcal{M}_c$) is below the blue surface representing $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa})$. In this example, the optimal DIMER value, $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa})$, occurs at point $\delta_x^* = 100$ and $\delta_y^* = 100$.

Figure 6.26 examines the behaviour of the ground-truth-based DIMER measure for different values of $\Delta\mathbf{P}$. In this particular example, this error exhibits the same behaviour as the covariance-based metric shown in Fig. 6.25. Also, as can be seen, the error corresponding to the optimal perturbation, i.e. $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa})$) is smaller than the error contained in the non-perturbed mean estimate $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa})$. Therefore, both types of DIMER metrics have shown improvements in this example.

The relation between the estimation results of the standard EKF-SLAM filter (zero perturbation) and the perturbed Kalman filter (perturbed by $\Delta\mathbf{P}^*$) is shown in Figure 6.27. The decline in the covariance-based as well as ground-truth-based parallel errors caused by applying the CTP method are observable in the figure (since the directional priorities are set to $C_\parallel = 2$ and $C_\perp = 0$, the parallel direction is the dominant axis in the directional error calculation).

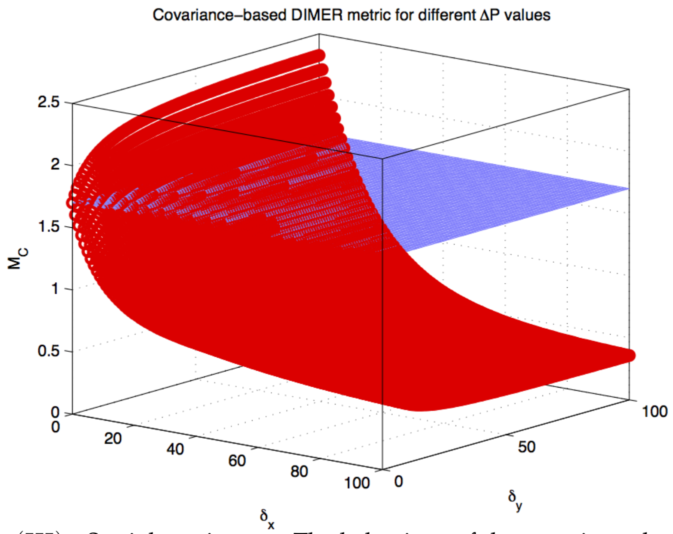


Figure 6.25: Case (III) - Straight trajectory: The behaviour of the covariance-based DIMER measure $\mathcal{M}_C$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ occurred at $\delta_x^* = 100$ and $\delta_y^* = 100$. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 1.70034$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.3569$. $C_\parallel = 2$; $C_\perp = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

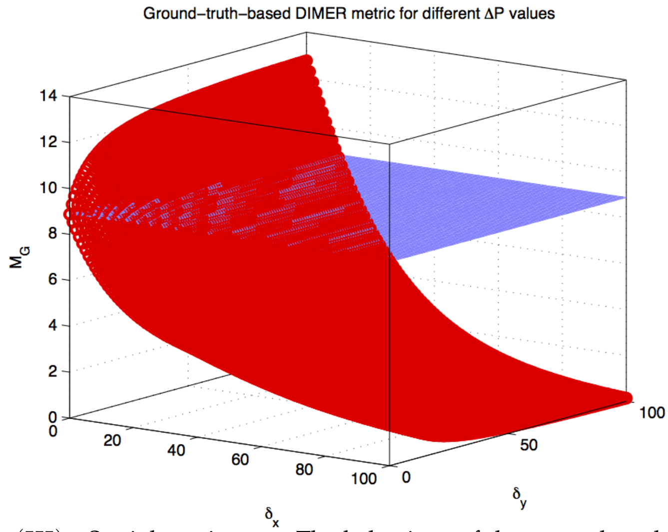


Figure 6.26: Case (**III**) - Straight trajectory: The behaviour of the ground-truth-based DIMER measure $\mathcal{M}_G$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ at $\delta_x^* = 100$ and $\delta_y^* = 100$ managed to significantly decrease the $\mathcal{M}_G$ metric compared to the standard case. $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 8.8796$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.1390$. $C_{\parallel} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

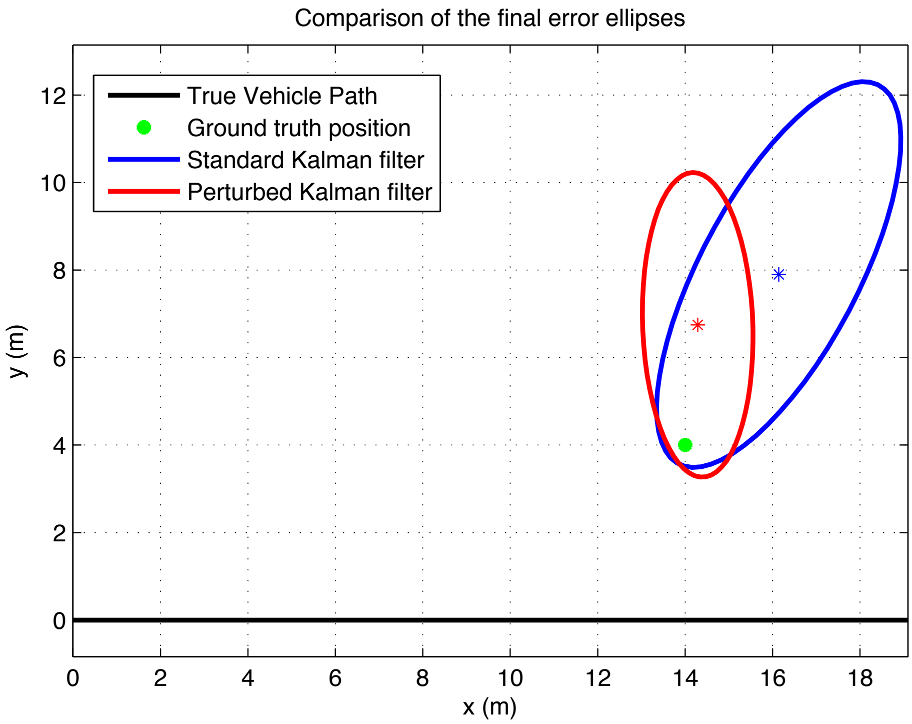


Figure 6.27: Case (**III**) - Straight trajectory: The effect of applying the CTP algorithm on the mean and 3σ ellipse associated with the map estimate. Both DIMER metrics have improved compared to the standard case (with no perturbation). $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 1.70034$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.3569$; $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 8.8796$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.1390$. $\delta_x^* = 100$; $\delta_y^* = 100$. $C_{\parallel} = 2$; $C_{\perp} = 0$.

Curved Path Analysis

This section presents the results of applying the CTP algorithm to the example scenario shown in Figure 6.19(b) in which a vehicle traverses a curved path and observes a stationary landmark at different time-steps. Similar to the straight path analysis conducted above, we designed a series of experiments to investigate the behaviour of the covariance-based DIMER metric (as well as the ground-truth based DIMER metric) of the acquired map for different values of the perturbation matrix $\Delta\mathbf{P}$.

An example for each of the aforementioned potential cases (Cases I-III) is provided here. The system setup and the details of this analysis are exactly the same as those of the straight path. Hence, for brevity, we do not dwell on the description of these examples and merely present the final results.

Case $(\mathbf{I})$: CTP fails to improve the quality of the map

The results related to this example are shown in Figures 6.28 and 6.29.

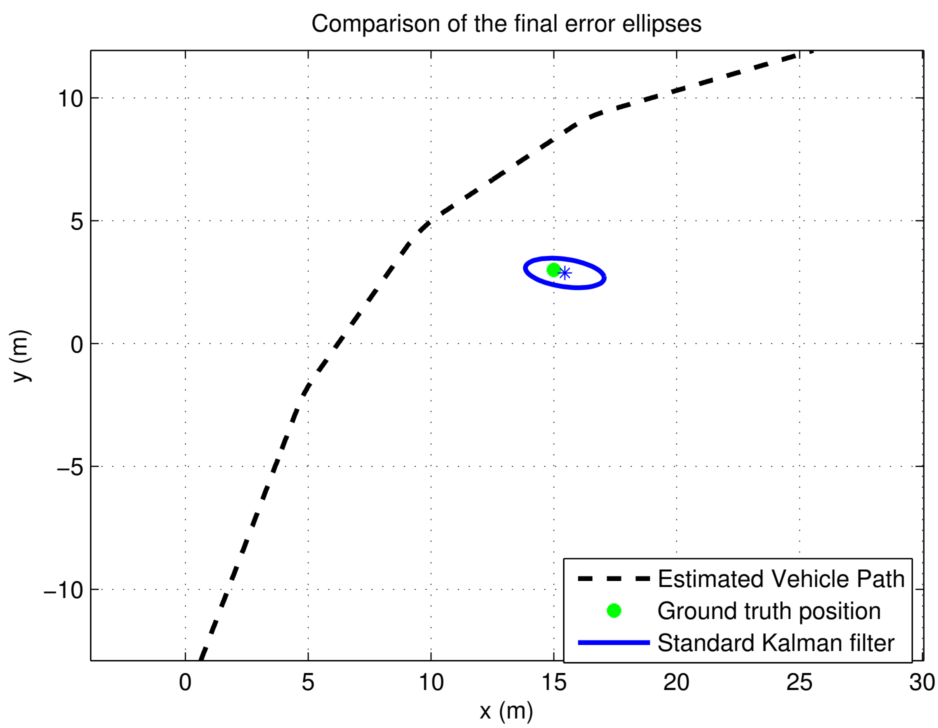


Figure 6.28: Case $(\mathbf{I})$ - Curved trajectory: The 3σ covariance ellipses obtained by applying the standard EKF-SLAM filter (with no perturbation) to an example scenario. The DIMER measures associated with this estimate are $\mathcal{M}^0_C(\mathbf{P}_{\kappa|\kappa}) = 0.55973$; $\mathcal{M}^0_G(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.18202$.

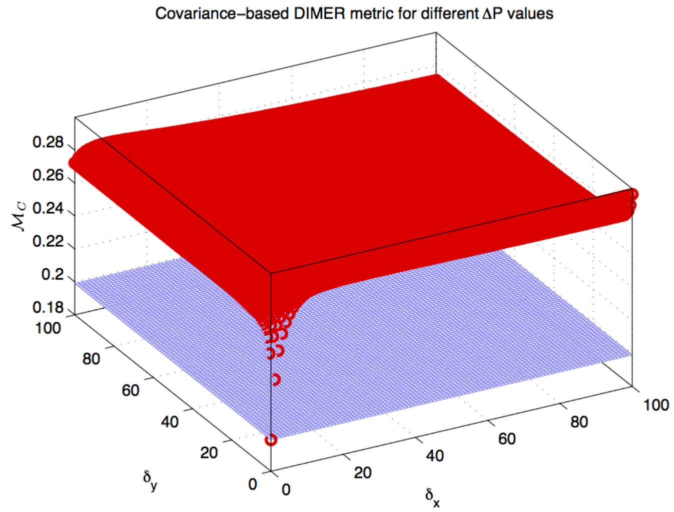


Figure 6.29: Case $(\mathbf{I})$ - Curved trajectory: The behaviour of the covariance-based DIMER measure $\mathcal{M}_C$ for different values of $\Delta\mathbf{P}$ in the search interval. No non-zero $\Delta\mathbf{P}^*$ could be found in this case. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 0.19912$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.19912$. $C_{\|} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

Case $(\mathbf{II})$: CTP reduces the covariance-based DIMER, but deteriorates the ground-truth-based DIMER measure.

The results related to this example are shown in Figures 6.30 to 6.32.

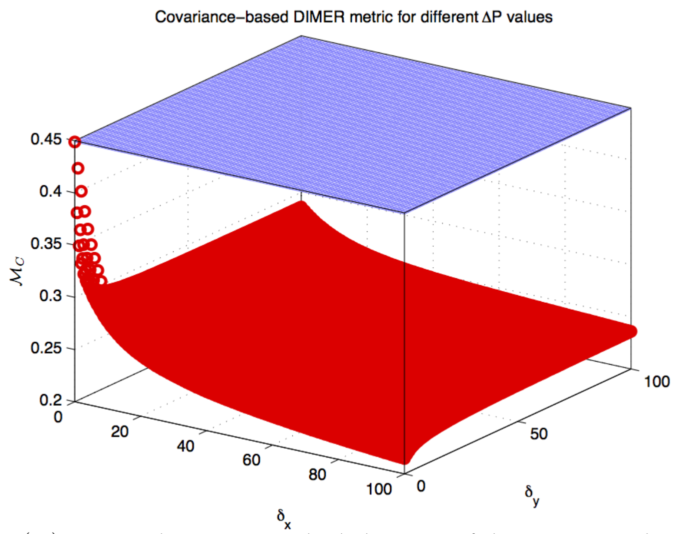


Figure 6.30: Case $(\mathbf{II})$ - Curved trajectory: The behaviour of the covariance-based DIMER measure $\mathcal{M}_C$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ occurred at $\delta_x^* = 100$ and $\delta_y^* = 0$. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 0.44885$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.21473$. $C_{\|} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

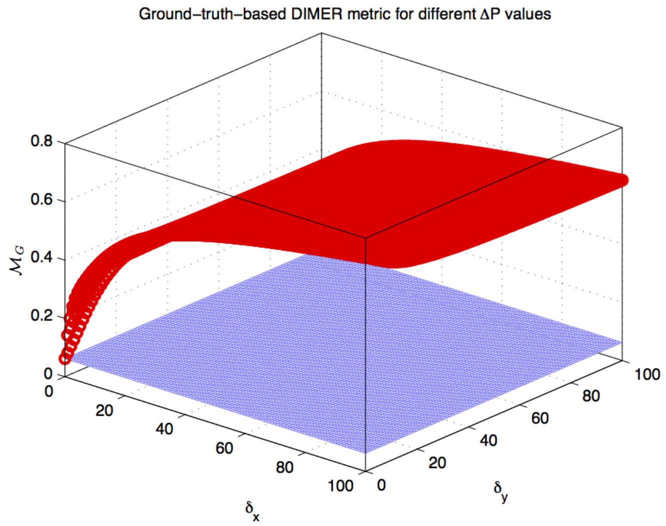


Figure 6.31: Case (**II**) - Curved trajectory: The behaviour of the ground-truth-based DIMER measure $\mathcal{M}_G$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ has had a deteriorating effect on the $\mathcal{M}_G$ metric compared to the standard case. $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.062181$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.78965$. $C_\parallel = 2$; $C_\perp = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

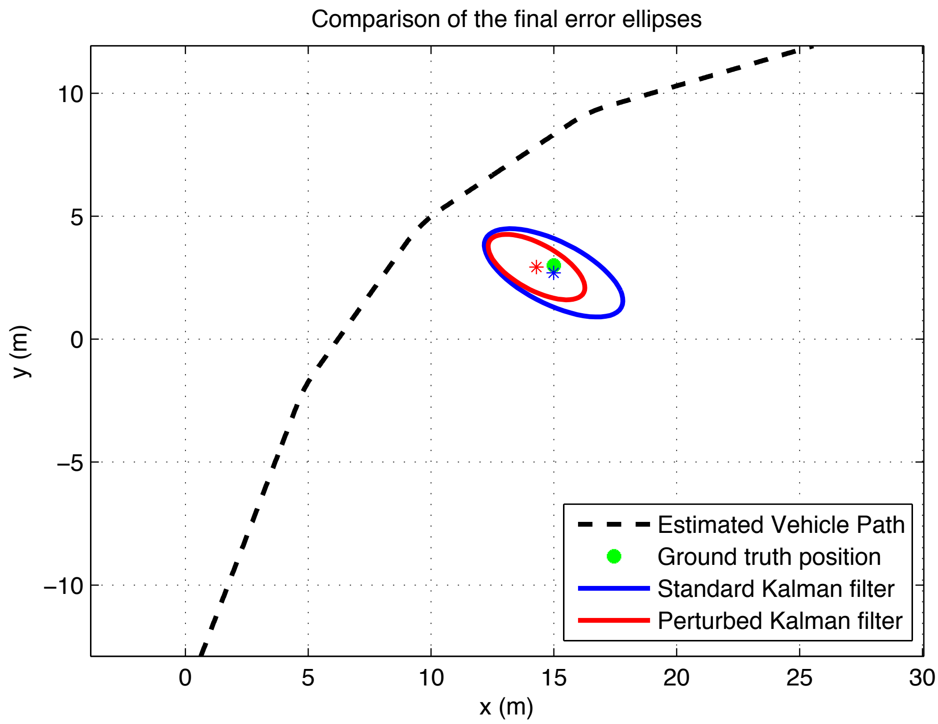


Figure 6.32: Case (**II**) - Curved trajectory: The effect of applying the CTP algorithm on the mean and the 3σ ellipse associated with the map estimate. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 0.44885$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.21473$; $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.062181$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 0.78965$. $\delta_x^* = 100$; $\delta_y^* = 0$. $C_\parallel = 2$; $C_\perp = 0$.

Case (III): CTP reduces both the covariance-based and the ground-truth-based DIMER measures.

The results related to this example are shown in Figures 6.33 to 6.35.

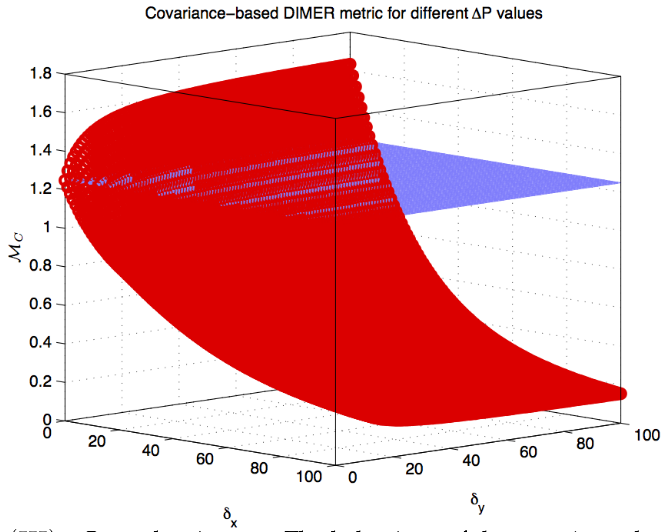


Figure 6.33: Case (**III**) - Curved trajectory: The behaviour of the covariance-based DIMER measure $\mathcal{M}_C$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ occurred at $\delta_x^* = 100$ and $\delta_y^* = 100$. $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 1.2478$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.1534$. $C_{\parallel} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

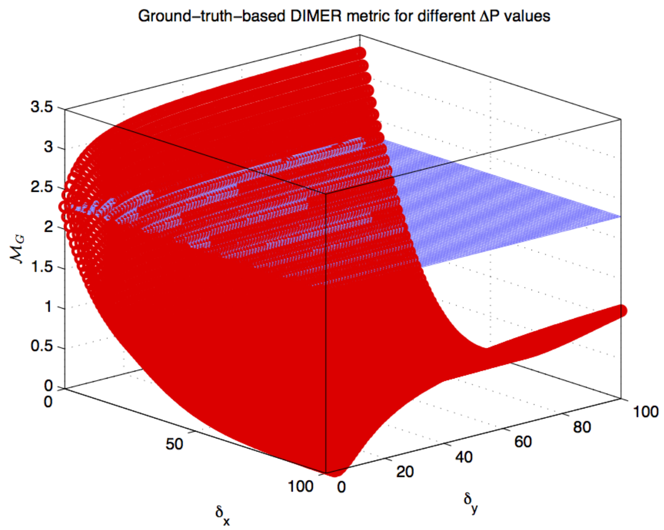


Figure 6.34: Case (**III**) - Curved trajectory: The behaviour of the ground-truth-based DIMER measure $\mathcal{M}_G$ for different values of $\Delta\mathbf{P}$ in the search interval. The optimal $\Delta\mathbf{P}^*$ at $\delta_x^* = 100$ and $\delta_y^* = 100$ managed to effectively decrease the $\mathcal{M}_G$ metric compared to the standard case. $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 2.2804$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 1.1002$. $C_{\parallel} = 2$; $C_{\perp} = 0$. $\delta_x \in (0, 100]$; $\delta_y \in (0, 100]$.

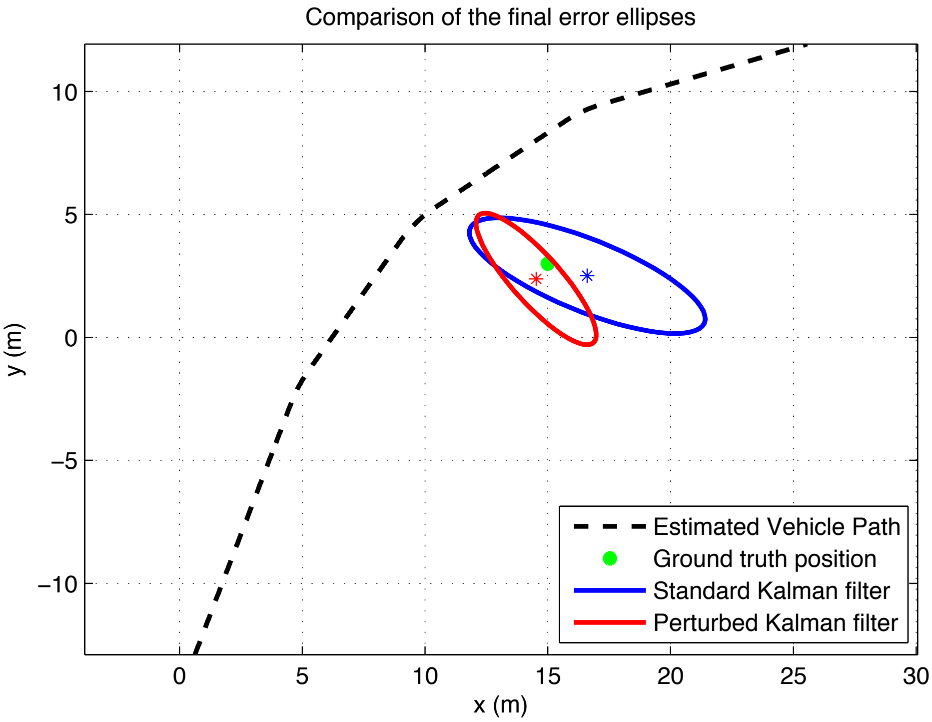


Figure 6.35: Case (**III**) - Curved trajectory: The effect of applying the CTP algorithm on the mean and 3σ ellipse associated with the map estimate. Both DIMER metrics have improved compared to the standard case (with no perturbation). $\mathcal{M}_C^0(\mathbf{P}_{\kappa|\kappa}) = 1.2478$; $\mathcal{M}_C^*(\mathbf{P}_{\kappa|\kappa}) = 0.1534$; $\mathcal{M}_G^0(\hat{\mathbf{x}}_{\kappa|\kappa}) = 2.2804$; $\mathcal{M}_G^{\Delta\mathbf{P}^*}(\hat{\mathbf{x}}_{\kappa|\kappa}) = 1.1002$. $\delta_x^* = 100$; $\delta_y^* = 100$. $C_{\|} = 2$; $C_{\perp} = 0$.

6.3.4 Further Analysis on Covariance Trajectory Perturbation Algorithm

This section provides supplementary analysis on the CTP algorithm and the simulations conducted in Subsection 6.3.3. The experiments carried out previously aimed at exemplifying the potential effectiveness of the CTP method in improving the directional map error metrics in different mapping scenarios. As shown, three different cases could be detected when applying a positive-definite covariance perturbation to the EKF-SLAM algorithm depending on the geometry of the collected measurements. We now examine the tentative probability for the occurrence of each case through the utilisation of Monte Carlo simulations. These simulations were designed to verify the CTP search in the following scenarios:

1. Parallel error minimisation for the straight-path scenario of Fig. 6.19(a)
2. Perpendicular error minimisation for the straight-path scenario of Fig. 6.19(a)
3. Parallel error minimisation for the curved-path scenario of Fig. 6.19(b)
4. Perpendicular error minimisation for the curved-path scenario of Fig. 6.19(b).

For each of the four scenarios outlined above, the criteria-based EKF-SLAM was performed for 1000 runs of Monte Carlo simulations. The noisy measurements were artificially generated by adding a random noise to the true range and bearing values. The measurement noise values for the initial simulations were sampled from normal distribution functions with $\sigma_r = 7m$ and $\sigma_\theta = 5°$. In each filter run the CTP algorithm was applied to minimise the covariance-based DIMER metric. Similar to the CTP simulations of Subsection 6.3.3, $\Delta\mathbf{P}$ was presumed diagonal with $\delta_x \in (0, 100]$ and $\delta_y \in (0, 100]$. For parallel error minimisation, the directional priorities were chosen as $C_\parallel = 2$ and $C_\perp = 0$, whereas for perpendicular error minimisation these priorities were set to $C_\parallel = 0$ and $C_\perp = 2$.

Prior to presenting the relevant results, we define the following notations. Although some of these notations have been used earlier, they are presented once again for completeness.

- $\hat{\beta}_j$: an estimate of the heading of the road pertaining to landmark j.
- N_{mc}: number of total Monte Carlo runs
- N_f: number of runs where a non-zero $\mathbf{\Delta P}^*$ is found ($\mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})$ is improved).
- N_{cg}: number of runs where both ground-truth-based and covariance-based DIMER metrics are improved.
- $\Delta\mathcal{M}_{G,avg}$: average change in ground-truth based DIMER metric for the samples where a non-zero $\mathbf{\Delta P}^*$ is found. A negative value indicates improvement, whereas a positive value shows an increase in this error (compared to the standard EKF-SLAM filter with no covariance perturbation).

Table 6.3 summarises the Monte Carlo simulation results for the above scenarios. As can be seen, in the parallel error minimisation problem for the straight-path scenario (No.1), in just over half the runs the CTP algorithm managed to find an optimal perturbation $\Delta\mathbf{P}^*$ that reduced the covariance-based DIMER. Furthermore, in 38% of those runs (194/508) the ground-truth-based DIMER metric was also improved. The obtained value for $\Delta\mathcal{M}_{G,avg}$ shows that on average, the ground-truth-based DIMER improved compared to the standard filter with no perturbation (positive $\Delta\mathcal{M}_{G,avg}$ indicates that the CTP has increased the average ground-truth-based error). As a result, the covariance trajectory perturbation was successful in improving both the average covariance-based and the average ground-truth-based measures over a large number of samples. However, for the perpendicular error minimisation (No.2), that is not the case. In spite of improving the covariance-based DIMER metric in about 38% of the total runs (of which 37% of them - or 14% of the total runs - exhibit ground-truth-based error reduction), the average ground-truth-based DIMER has become worse, in relation to the unperturbed filter. This deterioration can be interpreted as the introduction of an additional bias term into the estimator. To elaborate, the map estimate obtained using the nonlinear EKF-SLAM filter (with no perturbation) is originally biased with respect to the ground-truth, due to the difference between the estimator's expected value and the true map value. In some situations (such as scenario No.2), the integration of the CTP algorithm amplifies the already existing bias. Recalling Section 5.5, a large bias value may result in an increased MSE value and a mismatch between the mean and covariance estimates. In Chapter 5, we proposed the use of a debiasing technique known as D-CMKF to address the unwanted bias problem. Section 6.4.2 examines the effect of employing this strategy in a more general setting when incorporated into a multi-vehicle SLAM system.

The simulation results related to the curved trajectory (No.3 and No.4) demonstrated similar behaviour to those of the straight trajectory discussed above. The main difference was that the percentage of the runs in which an optimal, non-zero $\Delta\mathbf{P}^*$ could be found was lower than that of their straight-path scenario counterparts (i.e., No.1 and No.2). This might tentatively suggest that the CTP method is more likely to be effective in situations where the vehicle is moving in a straight trajectory compared to a curved path. Another observed difference was that both minimisation problems related to the curved trajectory exhibited a slight increase in the average ground-truth-based DIMER metric (similar to Experiment No.2). An important remark regarding these experiments and similar cases is that although the CTP method does not improve the ground-truth-based DIMER metric at 100 percent of the occasions (and even though the average ground-truth is also deteriorated), there is still value in reducing the covariance-based DIMER in a certain direction. Since the ground-truth information in not always accessible in real-world applications, the covariance matrix is the only available tool for performance assessment. Reducing the uncertainty of the estimate in a desirable direction is still beneficial because it provides a higher degree of confidence in the position estimate (because of the smaller error ellipse). Thus, the CTP method can potentially offer more credibility to a given map estimate.

No.	Vehicle Trajectory	σ_r	σ_θ	$\hat{\beta}_j$	Directional Priorities	N_{mc} (percentage)	N_f (percentage)	N_{cg} (percentage)	$\Delta\mathcal{M}_{G,avg}$
1	Straight Fig. 6.19(a)	7.0 m	5.0°	0.24°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	508 (50.8%)	194 (19.4%)	−10.81
2	Straight Fig. 6.19(a)	7.0 m	5.0°	0.24°	$C_\parallel = 0$ $C_\perp = 2$	1000 (100%)	378 (37.8%)	140 (14.0%)	+7.11
3	Curved Fig. 6.19(b)	7.0 m	5.0°	−166.7°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	363 (36.3%)	188 (18.8%)	+1.45
4	Curved Fig. 6.19(b)	7.0 m	5.0°	−166.7°	$C_\parallel = 0$ $C_\perp = 2$	1000 (100%)	238 (23.8%)	96 (9.6%)	+1.72

Table 6.3: CTP analysis: Examining the CTP search in different scenarios.

No.	$\Delta\mathbf{P}$ Search Domain	N_f (percentage)	$N^{(\delta_x^*=100)}_{(\delta_y^*=100)}$ (percentage)	$N^{(\delta_x^*=100)}_{(\delta_y^*=0)}$ (percentage)	$N^{(\delta_x^*=0)}_{(\delta_y^*=100)}$ (percentage)
1	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	508 (100%)	181 (35.6%)	198 (39.0%)	129 (25.4%)
2	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	378 (100%)	158 (41.8%)	65 (17.2%)	155 (41.0%)
3	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	363 (100%)	107 (29.4%)	182 (50.1%)	74 (20.4%)
4	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	238 (100%)	63 (26.5%)	150 (63.0%)	25 (10.5%)

Table 6.4: CTP analysis: Statistics about the optimal perturbation $\Delta\mathbf{P}^*$ for the scenarios of Table 6.3.

In order to study the behaviour of the CTP algorithm, we carried out further analysis regarding the $\Delta\mathbf{P}$ search in each of the scenarios shown in Table 6.3. Table 6.4 presents some statistics around the optimal $\Delta\mathbf{P}^*$ point in the pre-set search interval in situations where the CTP method successfully reduces the covariance-based directional error. As can be seen in all the studied scenarios in this section, the minimum value of $\mathcal{M}_C(\mathbf{P}_{\kappa|\kappa})$ occurs at one of the 4 corners corresponding to the maximum or minimum values of the search parameters δ_x and δ_y (i.e. 0 or 100 in these examples), of which one belongs to the point associated with no perturbation ($\delta_x = 0$ and $\delta_y = 0$). Therefore, based on the evidence, it is likely that when $\Delta\mathbf{P}$ is parametrised according to Eq. (6.1), evaluating and comparing the DIMER metrics at only 4 points is sufficient in determining the optimal point in the CTP search process[11]. This can make the CTP algorithm computationally far more efficient compared to the exhaustive search approach performed initially.

To examine the effects of different levels of measurement noise on the CTP search we conducted a separate Monte Carlo experiment for the straight-path scenario of Fig. 6.19(a). For each value of the nominated σ_r and σ_θ pair, the simulation was run

[11] The same observation can be made in the results presented in 6.3.3 (e.g. Figs. 6.21 and 6.22).

1000 times. Once again, CTP was incorporated with the original EKF-SLAM algorithm to improve the directional quality of the obtained map. Table 6.5 presents the results of these simulations. Five different pairs of measurement noise are considered in this table ($N1$ to $N5$). According to this table, the likelihood of finding a non-zero $\Delta\mathbf{P}^*$ that reduces the covariance-based DIMER measure increases with the level of measurement noise. Furthermore, the trend seen in the values of N_{cg} and $\Delta\mathcal{M}_{G,avg}$ suggests that the likelihood of improvement (as well as the degree of improvement) in the ground-truth-based metric increases for higher levels of measurement noise. Table 6.6 provides some statistics on the $\Delta\mathbf{P}$ search for these scenarios. It can be seen that for this particular studied scenario, the likelihood of the success of the CTP method increases by inflating the covariance matrix in both x and y directions.

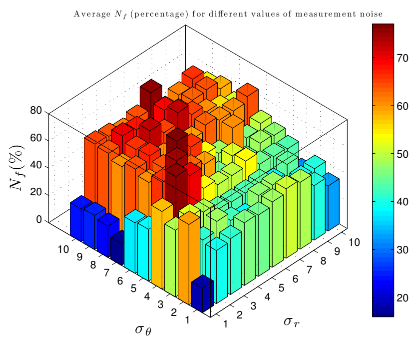
A more thorough analysis on the relationship between the CTP success and measurement noise level is performed in Figure 6.36. The figure uses a 3-D bar chart and a heat diagram to visualise this search process for the straight-path scenario shown in Fig. 6.11, when the values of range and bearing standard deviation vary between 1 and 10 (only the integer values are shown). The pattern shown in this figure also confirms that in general the CTP algorithm is more effective in scenarios where the measurement noise level is high. More extensive simulations were performed for different scenarios (such as curved path, etc.) that yielded similar outcomes with identical behaviour for the CTP search. However, to avoid repetitiveness, the results of these simulations are not presented here.

No.	Vehicle Trajectory	σ_r	σ_θ	$\hat{\beta}_j$	Directional Priorities	N_{mc} (percentage)	N_f (percentage)	N_{cg} (percentage)	$\Delta\mathcal{M}_{G,avg}$
$N1$	Straight Fig. 6.19(a)	1.0 m	0.5°	0.24°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	185 (18.5%)	48 (4.80%)	+0.12
$N2$	Straight Fig. 6.19(a)	2.0 m	1.0°	0.24°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	378 (37.8%)	122 (12.2%)	+0.69
$N3$	Straight Fig. 6.19(a)	4.0 m	3.0°	0.24°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	456 (45.6%)	144 (14.4%)	−3.59
$N4$	Straight Fig. 6.19(a)	5.0 m	4.0°	0.24°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	482 (48.2%)	154 (15.4%)	−5.30
$N5$	Straight Fig. 6.19(a)	6.0 m	5.0°	0.24°	$C_\parallel = 2$ $C_\perp = 0$	1000 (100%)	518 (51.8%)	195 (19.5%)	−7.74

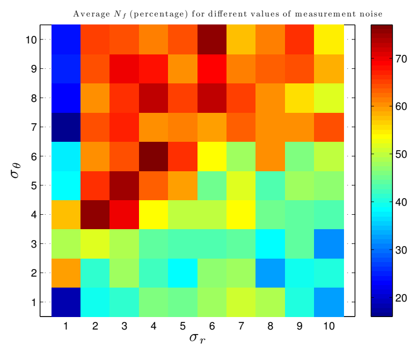
Table 6.5: CTP analysis: The performance of the CTP algorithm for different values of range and bearing noise (straight trajectory).

No.	$\Delta\mathbf{P}$ Search Domain	N_f (percentage)	$N_{(\delta_y^*=100)}^{(\delta_x^*=100)}$ (percentage)	$N_{(\delta_y^*=0)}^{(\delta_x^*=100)}$ (percentage)	$N_{(\delta_y^*=100)}^{(\delta_x^*=0)}$ (percentage)
$N1$	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	185 (100%)	158 (85.4%)	5 (2.7%)	22 (11.9%)
$N2$	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	378 (100%)	307 (81.2%)	15 (4.0%)	56 (14.8%)
$N3$	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	456 (100%)	321 (70.4%)	49 (10.7%)	86 (18.9%)
$N4$	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	482 (100%)	258 (53.5%)	117 (24.3%)	107 (22.2%)
$N5$	$\delta_x \in (0, 100]$ $\delta_y \in (0, 100]$	518 (100%)	221 (42.7%)	182 (35.1%)	115 (22.2%)

Table 6.6: CTP analysis: Statistics about the optimal perturbation $\Delta\mathbf{P}^*$ for the scenarios presented in Table 6.5.



(a) $\bar{N}_f(\%)$ (3D bar chart)



(b) $\bar{N}_f(\%)$ (Heat diagram)

Figure 6.36: CTP analysis: Average N_f (percentage) for different values of measurement noise.

6.4 Criteria-based Multi-vehicle SLAM Simulations and Results

In the previous section we demonstrated the effectiveness of the criteria-based covariance trajectory perturbation algorithm in reducing the covariance-based DIMER metric for a scenario involving a single vehicle and a single landmark. The present section expands this solution by integrating the CTP method into the multi-vehicle distributed mapping system proposed in Chapter 3. The mapping performance will be investigated by evaluating the quality of obtained maps inside individual vehicles as well as the central fusion centre (CFC). Both types of quality measures, i.e. the covariance-based and the ground-truth-based DIMER metrics will be examined.

6.4.1 DIMER-based Covariance Trajectory Perturbation for Multi-vehicle SLAM

The simulation consisted of three vehicles driving around overlapping trajectories in a sample environment populated with 25 landmarks (as shown in Figure 6.37). The paths and the movements of the vehicles were chosen so as to resemble a real-world situation. Like before, each vehicle collects and processes range and bearing measurements from the visible landmarks in their respective line of sight. The vehicles transmit their most informative local information to the server, where a global map is maintained (see Section 3.3).

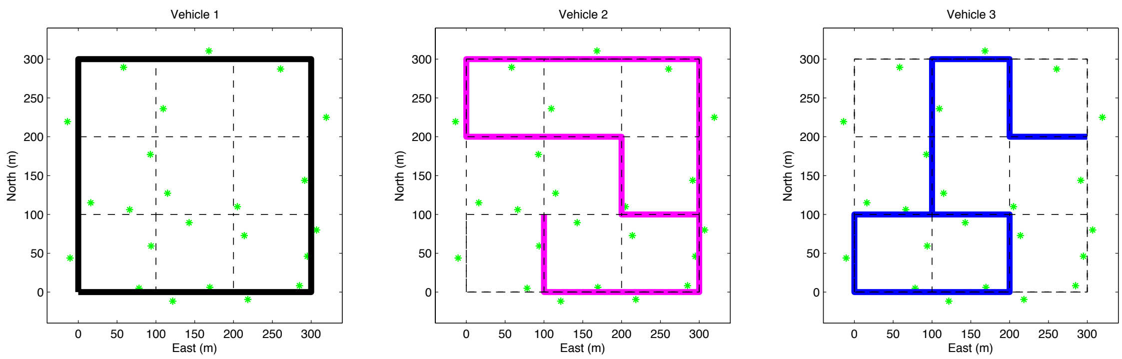


Figure 6.37: Individual vehicle tracks. The vehicle trajectories have been chosen so that every landmark in the grid is covered by at least one vehicle.

Since the CTP algorithm was originally developed for a simple case including a single vehicle and a single map element (see Section 5.4), some partial modifications had to be made in order to integrate this strategy into the multi-vehicle system with several landmarks. Consider an environment populated with N landmarks. A naive way of generalising the CTP method would be to solve a batch optimisation problem including N free variables (one unknown $\Delta\mathbf{P}$ for each landmark). The implementation of such a method would require re-running the mapping algorithm on the entire trajectory for a large number of times (each time with a different set of $\Delta\mathbf{P}$ parameters and comparing the resulting maps). This would be unfeasible for a large number of landmarks due to its computational complexity. This prompted the application of a segmentation technique in which the path of each vehicle is divided into smaller segments in an intelligent way. Each "chunk" (or horizon) is basically comprised of a small number of landmarks (usually $N_i < 4$) in order to mathematically facilitate the $\Delta\mathbf{P}$ search algorithm. Such a systematic approach can effectively render this process computationally tractable in large-scale environments.

For this purpose, we devise the use of a *pre-filter* structure as a pre-phase system component. For each vehicle, the pre-filter runs a separate EKF-SLAM (with no covariance trajectory perturbation) on the respective trajectory to obtain an initial, tentative map. Note that since no information is shared between the vehicles, the obtained estimates using this process are most likely less accurate compared to the case where the multi-vehicle SLAM is implemented. Based on the initial map estimates for individual vehicles, the previously mentioned landmark segmentation can be carried out prior to the implementation of the CTP method[12]. This is performed in a way that the landmarks in vicinity of each other are placed in the same segment. The main reason behind this is that there is generally a stronger correlation between adjacent landmarks compared to spatially distant landmarks in the SLAM context. Therefore, artificially inflating the covariance of one landmark in a specific segment would probably affect the other landmarks in that same segment more than the rest of the landmarks in the map. Consequently, due to this existing correlation, it is sensible to apply the CTP algorithm to a segment of the map consisting of adjacent landmarks.

After operating the pre-filter and assigning the landmarks to their specific segments, the distributed multi-vehicle SLAM with efficient communication can be run to construct the local and global maps. However, for each pre-defined road segment (determined from the pre-filter phase) the CTP algorithm is run to find the appropriate $\Delta\mathbf{P}^*$ values that minimise the DIMER metric associated with the landmarks in that segment. During the perturbation search, all side activities related to the multi-vehicle system are essentially paused temporarily until the $\Delta\mathbf{P}$ search is completed. The operation of the filter is resumed upon the completion of this search.

Two different sets of simulations were conducted to exemplify the impact of the CTP algorithm when incorporated into the distributed mapping system with efficient communication. The standard deviations of range and bearing measurements were set to $\sigma_r = 4m$ and $\sigma_\theta = 4^\circ$ for both simulations. Figure 6.38 illustrates the resulting map when no covariance trajectory perturbation is applied. The local map estimates inside individual vehicles (including the position and orientation of the landmarks) as well as the global map at the server are shown in the figure. Figure 6.39 shows the obtained map when the CTP algorithm is integrated into the mapping system (an enlarged view will be provided shortly to demonstrate the effectiveness of the CTP method). The directional priorities for this experiment were set to $C_\parallel = 2$ and $C_\perp = 0$ and the search limits were set as $\delta_x \in (0, 100]$ and $\delta_y \in (0, 100]$.

[12]Once again, we assumed perfect data association for the simulations presented in this chapter.

Note that the generated measurements and other system parameters were identical in both simulations to provide a credible comparison platform. Table 6.7 provides the covariance-based ($\mathcal{M}_C$) and ground-truth-based ($\mathcal{M}_G$) DIMER metrics associated with the global map as well as the local maps inside the vehicles for each of the above runs. As can be observed from this table, deploying the CTP algorithm for this example reduced the covariance-based local map errors in each vehicle. Moreover, as an indirect consequence of the CTP integration, the total covariance-based DIMER metric associated with the global map at the CFC witnessed a decrease compared to the normal case with no perturbation. On the other hand, changes in the overall ground-truth-based DIMER metric were negligible, meaning that the covariance-reduction was performed with no severe ramification on the ground-truth-based measure (this implies a small increase in the estimator's bias). Notwithstanding this, a few inconsistent map estimates were observed in both cases (Figs. 6.38 and 6.39). In other words, some of the covariance ellipses did not match the mean estimates. This issue will be addressed later in this Chapter.

The impact of applying the DIMER-based CTP on an example landmark can be seen in Figure 6.40. Subfigures (a) and (b) depict the enlarged views of Box No.1 in Figures 6.38 and 6.39 respectively. As can be seen, the local CTP algorithm employed in the third vehicle has managed to reduce the parallel error in the covariance ellipse pertaining to that landmark (notice the change in the solid ellipse). In addition, the same error has been decreased for the constructed global map at the central fusion centre (after the incorporation of information from all the vehicles).

The pattern reported here was also consistent with the results obtained from other experiments performed in similar multi-vehicle settings[13]. Consequently, the results support the initial hypothesis that, in addition to single-vehicle/single-landmark scenarios, the CTP algorithm is also effective in the multi-vehicle mapping system devised in this thesis. We remark that it is not generally trivial that decreasing the DIMER error for each vehicle deceases the DIMER error at the CFC. This uncertainty arises from the complex nature of the distributed system and the impact of sub-map fusion inside local, channel and server structures.

Fig.	CTP	$\mathcal{M}_C^1$	$\mathcal{M}_C^2$	$\mathcal{M}_C^3$	$\mathcal{M}_C^{CFC}$	$\mathcal{M}_G^1$	$\mathcal{M}_G^2$	$\mathcal{M}_G^3$	$\mathcal{M}_G^{CFC}$
6.38	**0**	168.10	121.25	85.48	**47.62**	158.67	88.44	31.83	**56.45**
6.39	**1**	165.49	119.85	82.9	**44.77**	159.06	91.26	44.71	**58.70**

Table 6.7: The effect of applying the CTP algorithm to the multi-vehicle SLAM system with efficient communication. The covariance-based ($\mathcal{M}_C$) and the ground-truth-based ($\mathcal{M}_G$) DIMER metrics associated with the final obtained maps (for both local and global estimates) are shown. Directional priorities were set to $C_{\parallel} = 2$ and $C_{\perp} = 0$.

[13]These results are not shown in this thesis to avoid repetitiveness.

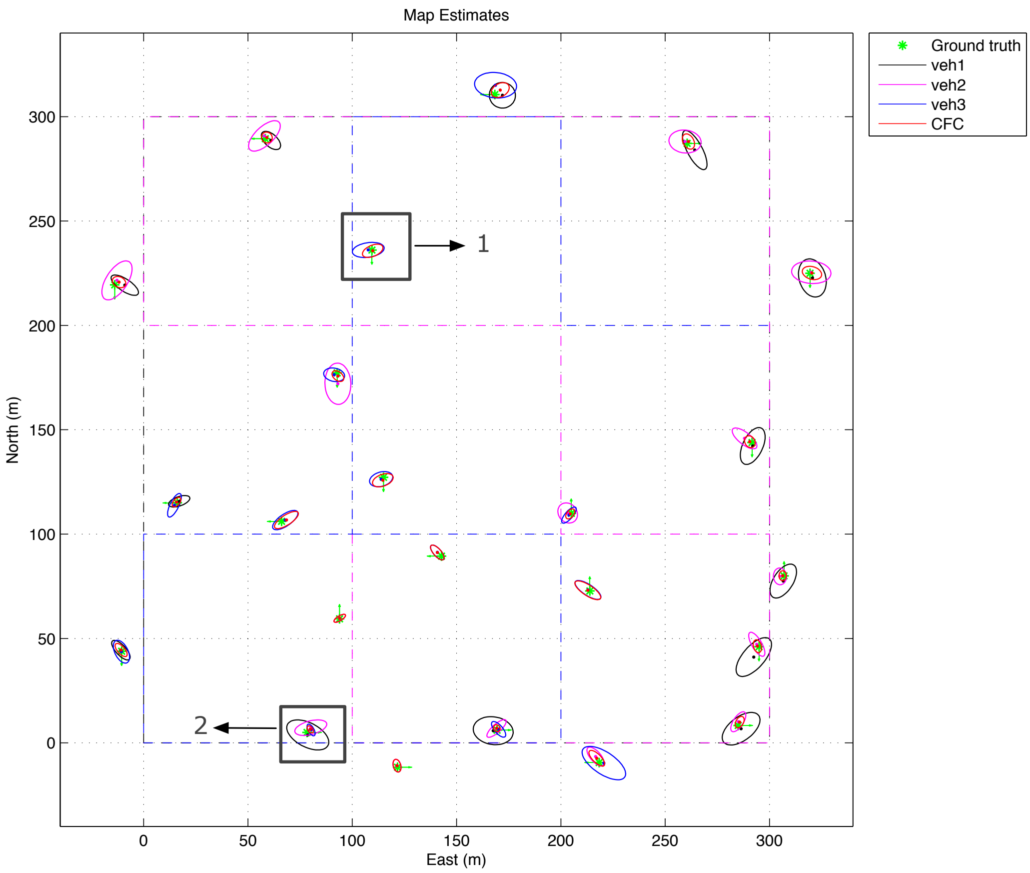


Figure 6.38: The local and global map estimates obtained by running the efficient multi-vehicle SLAM algorithm (with no perturbation). The 3σ uncertainty ellipses are shown in the overhead figure.

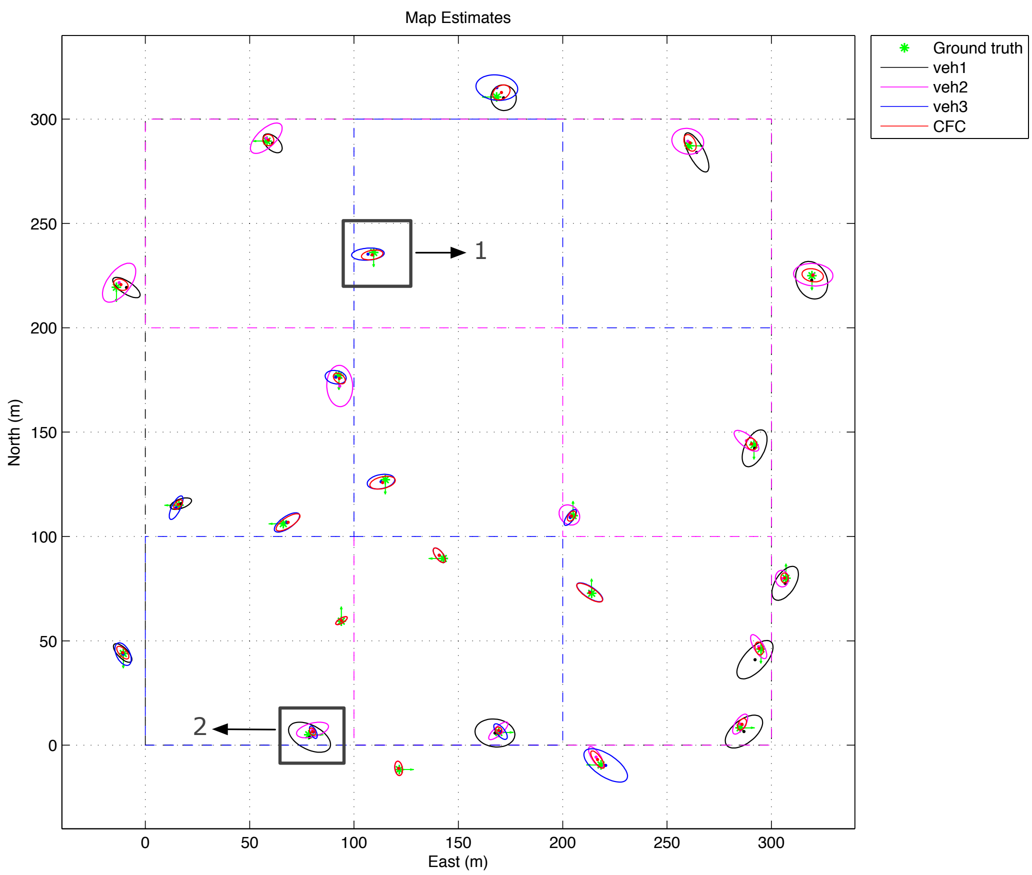
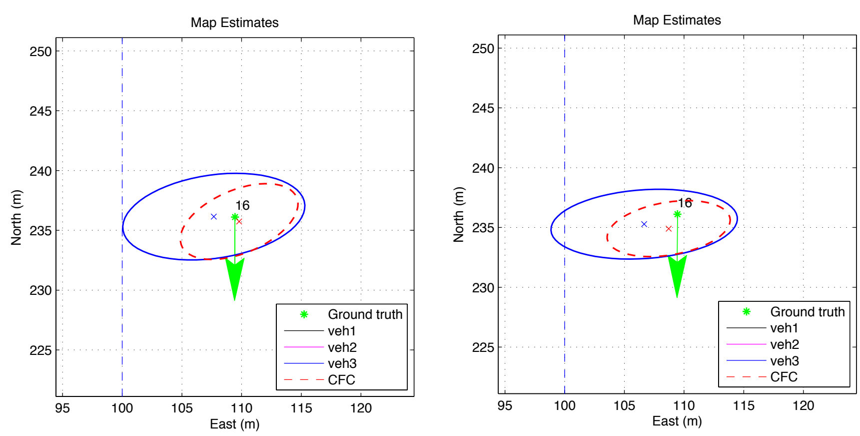


Figure 6.39: The resulting local and global map estimates obtained by applying the CTP algorithm to the efficient multi-vehicle SLAM algorithm of Fig. 6.38. Directional priorities were set to $C_{\|} = 2$ and $C_{\perp} = 0$ in this example. See Fig. 6.40 for an enhanced view.



(a) CTP = 0 (b) CTP = 1

Figure 6.40: The impact of deploying the CTP algorithm on the local and global map estimates pertaining to a sample landmark. Subfigures (a) and (b) are the enlarged views of Box No.1 in Figures 6.38 and 6.39 respectively. Directional priorities were set to $C_{\|} = 2$ and $C_{\perp} = 0$.

6.4.2 Results on Converted Measurement Kalman Filtering with De-biasing Compensation (D-CMKF)

In most of the experiments carried out previously in this section, in addition to analysing the effect of deploying the CTP algorithm on the covariance of the obtained maps, it was set out to examine the behaviour of the *actual* directional mapping error through the use of the $\mathcal{M}_G$ metric. The results (e.g. Monte Carlo simulations of Table 6.3) indicated that in some occasions, perturbing the initial covariance matrix can introduce additional bias into the estimator. This can potentially create a mismatch between the mean and covariance estimates and result in inconsistent estimates. Recalling Section 5.5, we proposed a debiasing algorithm coupled with converted measurement Kalman filtering (CMKF) to diminish the existing bias and avoid the inconsistency problem in the system. This subsection analyses the effect of D-CMKF integration on the multi-vehicle SLAM system shown in Fig. 6.37.

To demonstrate the overall impact of the D-CMKF technique and to compare the generated maps with the formerly obtained results where no debiasing approach was applied (i.e., Figures 6.38 and 6.39), four different experiments were designed and carried out. The same system model and identical measurements were used for these simulations. Each experiment (as shown below), utilised a different filtering structure inside the vehicles.

Experiment A - The original EKF-SLAM filter with non-linear observation model

Experiment B - The EKF-SLAM filter in synergy with DIMER-based CTP algorithm

Experiment C - The D-CMKF filter which uses a converted measurement system

Experiment D - The D-CMKF filter in synergy with DIMER-based CTP algorithm.

Table 6.8 compares the performance of the above four cases. Like before, the covariance-based as well as the ground-truth-based DIMER measures have been calculated for the final global and local maps. Based on the table, it appears that the D-CMKF algorithm has managed to reduce the ground-truth-based DIMER metric in both local and global maps. No consistent pattern could be detected for the covariance-based error associated with the local estimates, however D-CMKF seems to have marginally increased the overall covariance-based DIMER corresponding to the global map at the server (compared to the case where no D-CMKF is used). The second experiment (in which the EKF-SLAM algorithm was combined with the CTP method) still delivers the best quality in terms of the covariance-based DIMER. Other experiments with different set of measurements and system parameters exhibited the same type of behaviour for the above four cases. In other words, in all the conducted simulations the D-CMKF algorithm reduced the directional error with respect to the actual map in the local and global maps. We provide two hypothesis on reasons behind this improvement. Firstly, this enhancement is attributed to the employment of the debiasing structure in the D-CMKF filter (this was in fact the original motivation behind using this method). Secondly, this might also have occurred as a result of the

Exp.	Fig.	CTP	D-CMKF	$\mathcal{M}_C^1$	$\mathcal{M}_C^2$	$\mathcal{M}_C^3$	$\mathcal{M}_C^{CFC}$	$\mathcal{M}_G^1$	$\mathcal{M}_G^2$	$\mathcal{M}_G^3$	$\mathcal{M}_G^{CFC}$
A	6.38	**0**	**0**	168.10	121.25	85.48	**47.62**	158.67	88.44	31.83	**56.45**
B	6.39	**1**	**0**	165.49	119.85	82.9	**44.77**	159.06	91.26	44.71	**58.70**
C	6.41	**0**	**1**	163.99	120.50	90.43	**51.32**	153.33	80.33	35.76	**47.41**
D	6.41	**1**	**1**	163.99	120.50	90.43	**51.32**	153.33	80.33	35.76	**47.41**

Table 6.8: This table shows the impact of employing the D-CMKF structure on the multi-vehicle mapping system. The DIMER metric associated with the final obtained maps (local and global) have been shown. $\sigma_r = 4m$ and $\sigma_\theta = 4°$. Directional priorities were set to $C_\parallel = 2$ and $C_\perp = 0$.

using a linear observation model in the CMKF structure. Linearising the measurement model (and avoiding Jacobian calculations and other approximations normally used in an EKF-based system) may have caused the generation of more accurate estimates with respect to the true map.

Figure 6.41 shows the local and global map estimates obtained by applying the D-CMKF structure to the multi-vehicle scenario of Fig. 6.37 (i.e. Experiments 3 & 4 from above). In this experiment and other conducted experiments using the D-CMKF structure, inconsistent map estimates reported previously were no longer observed. We believe that this is also attributed to the structure of the D-CMKF filter and the fact that it provides more accurate estimates through using a linear observation model. Figure 6.42 exemplifies this claim for a single landmark in the above dataset. Subfigures (a), (b) and (c) depict the close-up views of Box No.2 shown in Figures 6.38, 6.39 and 6.41 respectively.

Another interesting, though surprising observation was that in the experiments where the D-CMKF structure was used as the local mapping algorithm, the incorporation of the CTP method failed to enhance the directional quality of the existing landmarks in the map. The data presented in Table 6.8 complies with this observation. It can be seen that the implementation of the D-CMKF filter for this experiment yielded identical test results regardless of whether the CTP was deployed or not. In other words, based on this evidence, applying the covariance trajectory perturbation to the converted measurement filter with bias compensation offers no advantage, since the effect of the CTP method is overshadowed by the D-CMKF filter. Our tentative hypothesis for the reason behind this unexpected result is that the developed CTP structure operates through exploiting the non-linearities of the mapping system. As mentioned previously, the proposed D-CMKF filter linearises the original EKF-SLAM filter model by converting polar measurements (range and bearing) to measurements in the Cartesian frame. As a result of this process, the CTP algorithm can no longer perform successfully to reduce the covariance-based DIMER of the estimates. Notwithstanding this, it is important to note that the CTP algorithm can still be effective in other filtering algorithms and mapping systems where the measurement model cannot be linearised.

Once again, we note that the conducted experiments were not limited to the

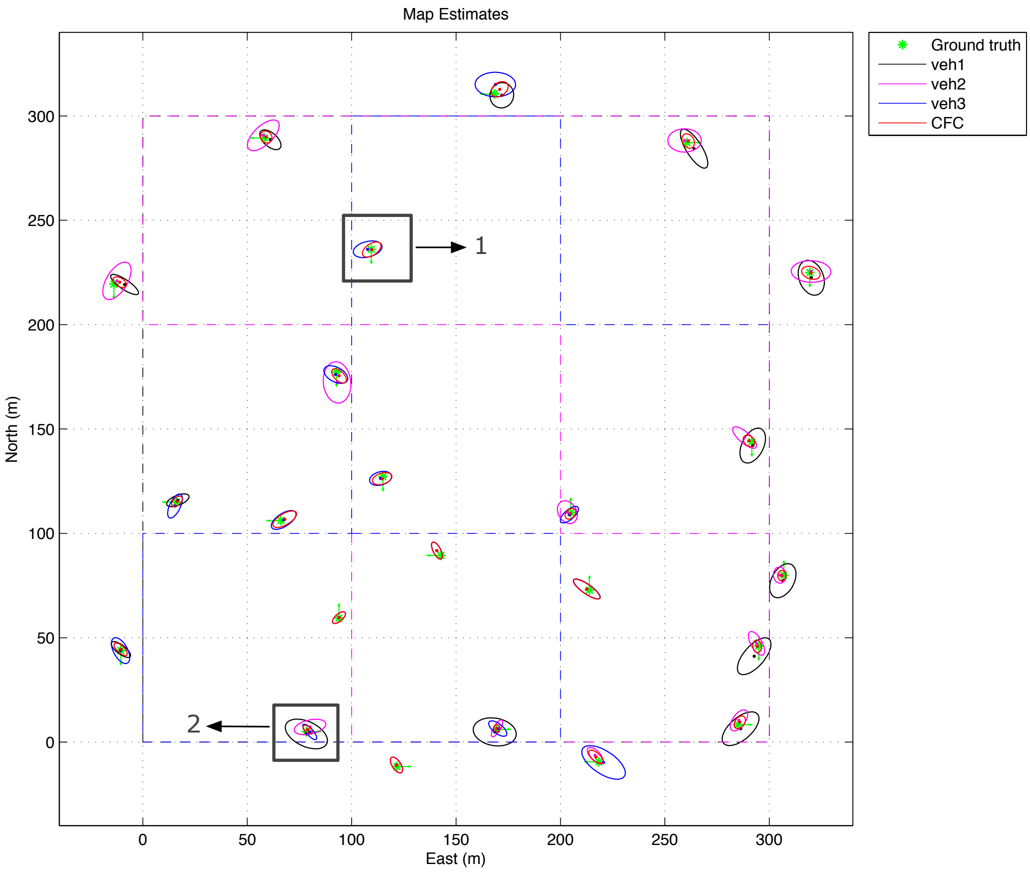
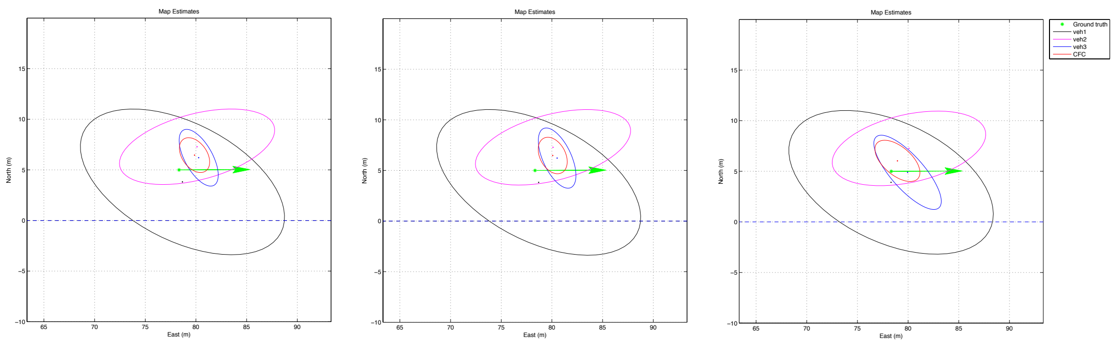


Figure 6.41: The resulting local and global map estimates obtained by applying the D-CMKF algorithm to the original SLAM filter inside individual vehicles. Directional priorities were set to $C_{\parallel} = 2$ and $C_{\perp} = 0$ in this example.



(a) CTP = 0, CMKF = 0 (b) CTP = 1, CMKF = 0 (c) CTP = 1, CMKF = 1

Figure 6.42: Subfigures (a), (b) and (c) depict the close-up views of Box No.2 in Figs. 6.38, 6.39 and 6.41 respectively. In (a) and (b), the global map estimate at the CFC as well as the local estimate from one of the vehicles (veh3) are inconsistent. The inconsistency issue has been resolved by applying the converted measurement filtering in Subfigure (c).

results presented in this chapter. Several other mapping scenarios were tested with different system parameters (different trajectories, set of measurements, noise levels, vehicle parameters, etc) which exhibited similar behaviour as reported here.

6.5 Summary

This chapter presented the simulations and results related to the methods proposed in this thesis. Section 6.2 showed the effectiveness of the developed multi-vehicle mapping architecture in a large-scale environment. It was shown that selectively communicating the landmarks with the highest information gain would significantly reduce the required transmission bandwidth in this architecture. Furthermore, the applied pruning algorithm (also based on information gain) limited the size of the local maps inside individual vehicles, which in turn made the system computationally feasible in large-scale environments.

Section 6.3 mainly focused on the application results of the newly designed DIMER metric in Chapter 4. The results related to the incorporation of this metric into the structure of two of the most widely used map estimation techniques, namely the covariance intersection and the EKF-SLAM algorithms were established. In Section 6.3.1, the results of applying DIMER-based CI for fusing maps with unknown degree of correlation were demonstrated and compared against more conventional methods in the literature. Subsection 6.3.3 presented the results of utilising the covariance trajectory perturbation (CTP) algorithm in a simplified scenario including a vehicle and a landmark. Three fundamentally different cases were identified and exemplified when applying a covariance perturbation to the EKF-SLAM algorithm. Detailed examples demonstrated how the integration of the CTP algorithm into the EKF-SLAM structure could be potentially constructive in reducing the covariance-based DIMER metric of the obtained map. Nevertheless, the results suggest that this method is situation-dependent and is impacted by several factors, thus could not guarantee success (in reducing the covariance-based DIMER) in 100% of the occasions. In addition to the studied straight and curved vehicle trajectories, the application of the CTP method was tested in other scenarios and yielded similar results. Based on the Monte Carlo results, it was made probable that the search process can be performed more efficiently than originally perceived by confining the search points for any given landmark to merely the four edge points in the search interval.

Section 6.4 was concerned with the simulation results of DIMER-based estimation in the previously developed multi-vehicle SLAM settings. The results indicated that deploying the CTP algorithm inside the local SLAM filter of individual vehicles could effectively enhance the directional error of the produced global map at the central fusion centre. The impact of utilising the D-CMKF filter on the performance of the mapping system was studied in Subsection 6.4.2. It was discovered that applying the covariance trajectory perturbation to the converted measurement filter with bias compensation offers no advantage, since the effect of the CTP method is overshadowed by the D-CMKF filter. It was also observed that the proposed methodology managed to improve the ground-truth-based DIMER metric (in both local and global maps) in analogy with the normal EKF-SLAM filter. Nevertheless, the formerly discussed synergy between the EKF-SLAM and the CTP algorithm (where no D-CMKF is applied) yielded better maps in terms of the covariance-based DIMER metric.

Chapter 7

Conclusions and Future Work

This chapter provides the conclusion of this work and presents suggestions on areas of future research.

7.1 Conclusions

This work devised an efficient data fusion framework for the problem of multi-vehicle SLAM for very-large-scale road mapping applications. The proposed solution conforms to the enforced practical restrictions and fundamental requirements of the AutoMap project, as the real-world inspiration behind this work. The deployed hierarchical architecture with selective communication significantly reduces the communication bandwidth in this setup. In addition, utilising the landmark pruning algorithm overcomes the problem of growing map sizes at the local nodes, thus rendering the local filter computationally feasible. The developed mapping solution is potentially scalable to environments with thousands of vehicles and many millions of landmarks.

A new concept of map quality for specialised road mapping applications such as AutoMap has been established. A particular type of error measure is derived in this work which is capable of reflecting the accuracy of landmark maps in a more meaningful way. The devised DIMER measure assimilates a number of significant practical factors (such as spatial orientation and type of map elements), which are not typically accounted for in traditional measures, to capture the accurateness of a given map in a variety of mapping applications. The DIMER metric can be deployed by both scientific and business communities to serve as a tool for comparing the performance of different mapping algorithms. It has been shown that the proposed covariance-based metric is a natural generalisation of the popular trace metric and it successfully accounts for the major deficiencies of conventional methods. Moreover, in a single-vehicle/single-landmark SLAM scenario, investigating the impact of measurement noise on the error contained in the resulting map demonstrates that the average covariance-based DIMER-metric exhibits comparable behaviour to that of the trace metric (both error measures increase as the measurement noise increases). Nevertheless, this is not necessarily true for the ground-truth-based DIMER metric in general (particularly for high levels of measurement noise).

The results have demonstrated that the devised covariance trajectory perturbation (CTP) solution is capable of enhancing the quality of landmark localisation when consolidated into the non-linear EKF-SLAM algorithm. Nonetheless, the proposed methodology is situation-dependent and cannot guarantee quality improvements in 100 percent of the occasions. Simulations suggest that there are various factors that attribute to the success of this method. More specifically, the geometry of the landmark observations should have a significant bearing on the performance of the CTP method. Monte-Carlo simulations conducted in this thesis indicate that statistically the CTP algorithm is effective in reducing the covariance-based DIMER measure in approximately 25-50% of the occasions. Furthermore, this method achieves enhancements in both ground-truth-based and covariance-based DIMER measures in 10-20% percent of the cases. It is also observed that an increase in the level of measurement noise would effectively increase the prospect of CTP success as well as the likelihood of reducing the ground-truth-based DIMER measure.

Although the original CTP algorithm was initially designed to operate in a single vehicle/single landmark environment, it is evidentially effective when expanded to the distributed multi-vehicle setting with intermittent communications developed in this thesis. Deploying the CTP algorithm inside the local SLAM filter of individual vehicles effectively reduces the covariance-based directional error of the local maps as well as the generated global map at the server. Despite its proven effectiveness in reducing the covariance-based directional error, the CTP implementation can bring about a number of undesirable consequences. The main ramifications include the creation of bias and inconsistent map estimates, both phenomena relatively more prevalent when the measurement noise is large. It turns out that the implementation of the D-CMKF filter, originally intended as a debiasing compensation structure, improves the ground-truth-based DIMER metric in both local and global maps. Moreover, the D-CMKF solution successfully prevents the issue of inconsistent map estimates from occurring by virtue of the linearised observation model. It has also been shown that applying CTP to the D-CMKF-SLAM filter renders the covariance search futile, since the D-CMKF filter effectively superseded the covariance perturbation method.

7.2 Future Work

This chapter concludes by providing suggestions on areas of future work. We outline three potential research problems related to the current thesis that merit further investigation and research.

Computer simulations were used in this thesis to demonstrate the effectiveness of the proposed methods. Due to the data collection platforms not being ready, the majority of these solutions were not tested on real-world data. Confirming the effectiveness of the devised methods and analysing their performance using real-world sensory data is a potential avenue worth pursuing.

This thesis developed a number of strategies for efficient, large-scale, distributed mapping using multiple vehicles in the presence of different practical constraints.

Although these methods are interconnected through the hierarchical mapping architecture, they were mostly addressed individually throughout this work. An alternative approach is to set up and solve a single joint-optimisation problem consisting of all the existing constraints dictated by the problem. This off-line, batch-optimisation problem can include a number of different factors such as the available computational power and communication bandwidth, the maximum size of local maps, cost parameters, targeted map quality, etc. Solving such intricate optimisation problem requires a well-defined mathematical formulation encompassing all the practical and theoretical aspects.

Finally, the last area of future research envisioned in this work is related to the problem of criteria-based estimation and mapping in the distributed mapping system. The development of other stochastic/deterministic techniques or filtering structures to enhance the DIMER metric associated with the obtained local and global map estimates can be investigated. In addition, the incorporation of the DIMER metric into the existing filtering/batch processing techniques (e.g. Graph-SLAM, Particle filtering, Unscented Kalman filtering, etc.) can be studied to provide an insight into more complex systems.

Chapter 8

Appendices

Appendix A : Applicability and Analysis of Covariance Intersection (CI)

A.1 introduction

In this appendix we examine the conditions in which data fusion can be performed by neglecting the unmodeled correlation between two information sources without compromising the consistency of the system. More specifically, we explore those situations in which one can disregard the correlation information and achieve a consistent estimate by simply adding the respective estimates′ information matrices. This estimate will deliver considerably better performance than the widely employed Covariance Intersection (CI) algorithm in terms of estimation uncertainty.

This work is motivated by a practical project with the aim of developing a distributed fusion system to map a large-scale environment. The data fusion algorithm is distributed across multiple vehicles, each given the task of producing and updating a local map. The vehicles are equipped with a range of sensors and selectively communicate maps to and from a central station [Amirsadri et al., 2012a]. The local maps obtained from different vehicles are not independent; e.g. all vehicles share information obtained from the central station. Hence, an appropriate fusion strategy must be deployed to tackle the problem of correlated submaps.

The rest of this section is arranged as follows: Section A.1.1 provides some preliminaries on the data fusion problem under study. Subsection A.1.2 outlines three classical fusion methods given correlated estimates. In Section A.1.3 conditions on consistent fusion, while ignoring the unknown correlation, will be derived for fusing two unbiased estimates. Simulations are provided in Section A.2 and Conclusions are drawn in Section A.3.

A.1.1 Preliminaries

We consider two (random variable) estimates $\mathbf{a} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{aa})$ and $\mathbf{b} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{bb})$ of some fixed parameter $\mathbf{c}^*$. The estimation error of $\mathbf{a}$ and $\mathbf{b}$ are defined by the random variables

$$\tilde{\mathbf{a}} = \mathbf{a} - \mathbf{c}^* \ , \ \tilde{\mathbf{b}} = \mathbf{b} - \mathbf{c}^* \tag{A.1}$$

where, in this case,

$$\mathrm{E}[\tilde{\mathbf{a}}] = 0 \ , \ \tilde{\mathbf{P}}_{aa} = \mathrm{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^\top] \tag{A.2}$$

$$\mathrm{E}[\tilde{\mathbf{b}}] = 0 \ , \ \tilde{\mathbf{P}}_{bb} = \mathrm{E}[\tilde{\mathbf{b}}\tilde{\mathbf{b}}^\top] \tag{A.3}$$

Although the true values $\tilde{\mathbf{P}}_{aa}$ and $\tilde{\mathbf{P}}_{bb}$ may not be known, consistent approximations $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ are assumed available where [1]

$$\mathbf{P}_{aa} \geq \tilde{\mathbf{P}}_{aa} \ , \ \mathbf{P}_{bb} \geq \tilde{\mathbf{P}}_{bb} \tag{A.4}$$

The cross-correlation matrix between the two estimates is denoted by $\tilde{\mathbf{P}}_{ab}$ and is defined by

$$\tilde{\mathbf{P}}_{ab} = \mathrm{E}[(\mathbf{a} - \mathbf{c}^*)(\mathbf{b} - \mathbf{c}^*)^\top] = \mathrm{E}[\tilde{\mathbf{a}}\tilde{\mathbf{b}}^\top] \tag{A.5}$$

This matrix may be known or unknown and may even be zero in some applications.

Let $\mathbf{c} \sim \mathcal{N}(\mathbf{c}^*, \mathbf{P}_{cc})$ denote a third estimate of $\mathbf{c}^*$ obtained via a linear combination of $\mathbf{a}$ and $\mathbf{b}$. That is

$$\mathbf{c} = \mathbf{K}_1\mathbf{a} + \mathbf{K}_2\mathbf{b} \tag{A.6}$$

where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ and $\mathbf{K}_1, \mathbf{K}_2 \in \mathbb{R}^{n\times n}$. The error in this estimate is

$$\tilde{\mathbf{c}} = \mathbf{c} - \mathbf{c}^* \tag{A.7}$$

and obeys $\mathrm{E}[\tilde{\mathbf{c}}] = 0$ when $\mathbf{K}_1 + \mathbf{K}_2 = \mathbf{I}$.

The true covariance $\tilde{\mathbf{P}}_{cc} = \mathrm{E}[\tilde{\mathbf{c}}\tilde{\mathbf{c}}^\top]$ is calculated by

$$\tilde{\mathbf{P}}_{cc} = \mathbf{K}_1\tilde{\mathbf{P}}_{aa}\mathbf{K}_1^\top + \mathbf{K}_2\tilde{\mathbf{P}}_{bb}\mathbf{K}_2^\top + \mathbf{K}_1\tilde{\mathbf{P}}_{ab}\mathbf{K}_2^\top + \mathbf{K}_2\tilde{\mathbf{P}}_{ba}\mathbf{K}_1^\top \tag{A.8}$$

and calculation of this term requires $\tilde{\mathbf{P}}_{ab} = \tilde{\mathbf{P}}_{ba}^\top$ be known (when it is non-zero).

In this paper we are mainly interested in the construction of an estimate $\mathbf{P}_{cc}$ of $\tilde{\mathbf{P}}_{cc}$ when the cross-correlation $\tilde{\mathbf{P}}_{ab}$ is non-zero but unknown. We are further interested in certain properties of the resulting $\mathbf{P}_{cc}$. In particular, we are interested in the property of consistency

$$\mathbf{P}_{cc} \geq \tilde{\mathbf{P}}_{cc} \tag{A.9}$$

where $\tilde{\mathbf{P}}_{cc}$ is given by (A.8). In this case, (A.8) holds for any estimator defined by the linear combination (A.6) but the computation (A.8) requires knowledge of the cross-correlation $\tilde{\mathbf{P}}_{ab}$ or some estimation thereof.

In many cases, one is not interested in the class of estimators defined by arbitrary

[1] This inequality is in the sense of matrix positive definiteness.

parameters $\mathbf{K}_1 + \mathbf{K}_2 = \mathbf{I}$ but rather in some optimal estimator. In this case, we note the following estimator defined by

$$(\mathbf{K}_1^*, \mathbf{K}_2^*) = \underset{(\mathbf{K}_1, \mathbf{K}_2)}{\operatorname{argmin}} \ \operatorname{tr}(\tilde{\mathbf{P}}_{cc}) \quad \text{s.t.} \quad \mathbf{K}_1 + \mathbf{K}_2 = \mathbf{I} \tag{A.10}$$

$$\tilde{\mathbf{P}}_{cc} = \begin{bmatrix} \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{P}}_{aa} & \tilde{\mathbf{P}}_{ab} \\ \tilde{\mathbf{P}}_{ab}^T & \tilde{\mathbf{P}}_{bb} \end{bmatrix} \begin{bmatrix} \mathbf{K}_1^\top \\ \mathbf{K}_2^\top \end{bmatrix} \tag{A.11}$$

where the pair $\mathbf{K}_1$ and $\mathbf{K}_2$ are chosen to minimise the trace of $\tilde{\mathbf{P}}_{cc}$. Solving the above constrained optimisation problem for $\mathbf{K}_1$ and $\mathbf{K}_2$ yields an optimal value for $\tilde{\mathbf{P}}_{cc}$ in the form of

$$\begin{aligned} \tilde{\mathbf{P}}_{cc}^{*^{-1}} = \tilde{\mathbf{P}}_{aa}^{-1} + (\tilde{\mathbf{P}}_{aa}^{-1}\tilde{\mathbf{P}}_{ab} - \mathbf{I})(\tilde{\mathbf{P}}_{bb} - \tilde{\mathbf{P}}_{ab}^\top \tilde{\mathbf{P}}_{aa}^{-1} \tilde{\mathbf{P}}_{ab})^{-1} \cdot \\ (\tilde{\mathbf{P}}_{ab}^\top \tilde{\mathbf{P}}_{aa}^{-1} - \mathbf{I}) \end{aligned} \tag{A.12}$$

As noted, in this paper we are concerned, primarily with the construction of a consistent estimate $\mathbf{P}_{cc}$ of $\tilde{\mathbf{P}}_{cc}$ when the cross-correlation $\tilde{\mathbf{P}}_{ab}$ is non-zero but unknown. To this end we define consistency against the optimal value $\tilde{\mathbf{P}}_{cc}^*$ which in turn is defined as that $\tilde{\mathbf{P}}_{cc}$ with the minimum trace over all estimators of the form (A.6).

Definition 3. *Suppose $\tilde{\mathbf{P}}_{aa}$ and $\tilde{\mathbf{P}}_{bb}$ are given along with $\tilde{\mathbf{P}}_{ab} = \tilde{\mathbf{P}}_{ba}^\top$. Suppose $\tilde{\mathbf{P}}_{ab} = \tilde{\mathbf{P}}_{ba}^\top$ is non-zero. An estimate $\mathbf{P_{cc}}$ of $\tilde{\mathbf{P}}_{cc}$ is said to be consistent if*

$$\mathbf{P}_{cc} \geq \tilde{\mathbf{P}}_{cc}^* \tag{A.13}$$

where $\tilde{\mathbf{P}}_{cc}^*$ is an optimal value for $\tilde{\mathbf{P}}_{cc}$ given by (A.12).

This definition of consistency is particularly useful for the purposes of studying information fusion algorithms as it relates practical estimators (particularly their uncertainty estimate) with an ideal estimator that could be constructed if the cross-correlation between individual estimators were known (and it was known that individual estimates were not over-confident).

It is generally true that ignoring the correlation information $\tilde{\mathbf{P}}_{ab}$ when fusing **a** and **b** can lead to overly confident results; i.e. the resulting estimate of $\mathbf{P}_{cc}$ will be inconsistent as per Definition 3. Some algorithms, such as covariance intersection (CI), on the other hand are designed to generate consistent estimates when the cross-correlation is unknown. In many cases, the resulting estimators are considerably conservative. We explore those situations in which one can simply ignore the correlation information and achieve a consistent estimate by simply adding the respective estimates' information matrices. This estimate will deliver considerably better performance than the suboptimal covariance intersection. The specific details of the estimators in question will become clear as the paper progresses.

A.1.2 Three Classical Fusion Algorithms

In this section we outline three well-known estimation algorithms given the setup provided in the previous section. Each estimator assumes different information to be available for computation. We are mainly focused on the computation of the estimator's covariance in this paper as we will later be concerned with consistency.

A.1.2.1 Minimum Trace Fusion of Two Normally Distributed Estimators with a Known Degree of Correlation

We consider two estimates $\mathbf{a} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{aa})$ and $\mathbf{b} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{bb})$ of some fixed parameter $\mathbf{c}^*$. Suppose two consistent estimates of $\mathbf{a}$ and $\mathbf{b}$ with $\mathbf{P}_{aa} \geq \tilde{\mathbf{P}}_{aa}$ and $\mathbf{P}_{bb} \geq \tilde{\mathbf{P}}_{bb}$ are available and the cross-correlation $\tilde{\mathbf{P}}_{ab}$ is known. Replacing $\tilde{\mathbf{P}}_{aa}$ and $\tilde{\mathbf{P}}_{bb}$ in (A.12) by $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ respectively, automatically generates a consistent estimate $\mathbf{P}^*_{cc} \geq \tilde{\mathbf{P}}^*_{cc}$ if $\tilde{\mathbf{P}}_{ab} = \tilde{\mathbf{P}}_{ba}^\top$ is known. This is a consequence of Eq. (A.8). Therefore, when $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ are consistent and $\tilde{\mathbf{P}}_{ab}$ is known then the combined estimate

$$\begin{aligned} \mathbf{P}^{*\,-1}_{cc} = \mathbf{P}_{aa}^{-1} + (\mathbf{P}_{aa}^{-1}\tilde{\mathbf{P}}_{ab} - \mathbf{I})(\mathbf{P}_{bb} - \tilde{\mathbf{P}}_{ab}^\top \mathbf{P}_{aa}^{-1}\tilde{\mathbf{P}}_{ab})^{-1} \cdot \\ (\tilde{\mathbf{P}}_{ab}^\top \mathbf{P}_{aa}^{-1} - \mathbf{I}) \end{aligned} \tag{A.14}$$

is by definition consistent (as per Definition 3). As noted, the problem in practice is that $\tilde{\mathbf{P}}_{ab}$ is typically unknown.

A.1.2.2 Fusion of Two Normally Distributed Estimators with an Unknown Degree of Correlation: Covariance Intersection

In many practical applications the degree of correlation between different information sources is not available. A common solution in this case is to use the well-known covariance intersection (CI) algorithm. Suppose again we have two estimates $\mathbf{a} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{aa})$ and $\mathbf{b} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{bb})$ of some fixed parameter $\mathbf{c}^*$. Suppose consistent estimates $\mathbf{P}_{aa} \geq \tilde{\mathbf{P}}_{aa}$ and $\mathbf{P}_{bb} \geq \tilde{\mathbf{P}}_{bb}$ are available. The cross-correlation $\tilde{\mathbf{P}}_{ab}$ is unknown (cannot be used in the fusion algorithm) and may be non-zero. Then CI is defined by a convex combination

$$\mathbf{P}_{cc}^{\mathrm{CI}\,-1} = \omega \mathbf{P}_{aa}^{-1} + (1-\omega)\mathbf{P}_{bb}^{-1} \tag{A.15}$$

$$\mathbf{P}_{cc}^{\mathrm{CI}\,-1}\mathbf{c} = \omega \mathbf{P}_{aa}^{-1}\mathbf{a} + (1-\omega)\mathbf{P}_{bb}^{-1}\mathbf{b} \tag{A.16}$$

where $\mathbf{c} \sim \mathcal{N}(\mathbf{c}^*, \mathbf{P}_{cc})$ is an estimate of $\mathbf{c}^*$ and where $\omega \in (0,1)$ is calculated according to some criteria; e.g. such as minimising the trace of the resulting covariance matrix $\mathbf{P}_{cc}^{\mathrm{CI}}$.

We note here simply that for all $\omega \in (0,1)$, CI is guaranteed consistent as per Definition 3; i.e. $\mathbf{P}_{cc}^{\mathrm{CI}} \geq \tilde{\mathbf{P}}^*_{cc}$ and is often considerably conservative. We point to the literature [Julier and Uhlmann, 1997] for further discussion of the CI algorithm and its consistency.

A.1.2.3 Fusion of Two Normally Distributed Estimators with an Unknown Degree of Correlation: Assuming Zero Correlation

Suppose again we have two estimates $\mathbf{a} \sim \mathcal{N}(\mathbf{c}^*, \mathbf{P}_{aa})$ and $\mathbf{b} \sim \mathcal{N}(\mathbf{c}^*, \mathbf{P}_{bb})$ of some fixed parameter $\mathbf{c}^*$ and each estimate is consistent; i.e. $\mathbf{P}_{aa} \geq \tilde{\mathbf{P}}_{aa}$ and $\mathbf{P}_{bb} \geq \tilde{\mathbf{P}}_{bb}$. The cross-correlation $\tilde{\mathbf{P}}_{ab}$ is unknown (cannot be used in the fusion algorithm) and may be non-zero. Let $\mathbf{c} \sim \mathcal{N}(\mathbf{c}^*, \mathbf{P}_{cc})$ denote an estimate of $\mathbf{c}^*$.

Now if $\mathbf{a}$ and $\mathbf{b}$ were in fact uncorrelated, then substituting $\tilde{\mathbf{P}}_{ab} = 0$ into (A.14) yields

$$\mathbf{P}^{0\ -1}_{cc} = \mathbf{P}^{-1}_{aa} + \mathbf{P}^{-1}_{bb} \tag{A.17}$$

which can be computed and is subsequently (by definition) consistent as per Definition 3. We also have

$$\mathbf{P}^{0\ -1}_{cc}\mathbf{c} \;\;=\;\; \mathbf{P}^{-1}_{aa}\mathbf{a} + \mathbf{P}^{-1}_{bb}\mathbf{b} \tag{A.18}$$

for completeness. This solution is optimal (in the sense of a minimum trace) when $\tilde{\mathbf{P}}_{ab}$ is indeed zero.

The main question motivating the subsequent work in this paper is summarised in the following.

Question 1. *If one computes $\mathbf{P}^{0\ -1}_{cc} = \mathbf{P}^{-1}_{aa} + \mathbf{P}^{-1}_{bb}$ when $\tilde{\mathbf{P}}_{ab}$ is non-zero, is $\mathbf{P}^0_{cc}$ consistent as per Definition 3?*

It is easily observed that $\mathbf{P}^0_{cc} \leq \mathbf{P}^{\mathrm{CI}}_{cc}$. Thus, if $\mathbf{P}^0_{cc} \geq \tilde{\mathbf{P}}^*_{cc}$, i.e. if $\mathbf{P}^0_{cc}$ is consistent as per Definition 3, then it follows that estimation via $\mathbf{P}^0_{cc}$ is typically more desirable than estimation via $\mathbf{P}^{\mathrm{CI}}_{cc}$. It will turn out that the inequality $\mathbf{P}^0_{cc} \geq \tilde{\mathbf{P}}^*_{cc}$ holds for only some values of $\tilde{\mathbf{P}}_{ab}$. In those cases, it so happens that one may simply ignore (set to zero) the cross-correlation and perform optimal (minimum trace) fusion. The result will be sub-optimal (as expected) but better (in terms of the trace) than covariance intersection. The result, as per the definition of consistency, will be conservative (non-optimistic) as desired.

A.1.3 Condition on consistent estimation under unknown correlation

It is well known that the CI algorithm guarantees the combined estimate to be consistent as per Definition 3. However, the consistency of $\mathbf{P}^{0\ -1}_{cc} = \mathbf{P}^{-1}_{aa} + \mathbf{P}^{-1}_{bb}$, i.e. simply ignoring the correlation, when $\tilde{\mathbf{P}}_{ab}$ is non-zero has yet to be established. As per Definition 3 consistency requires

$$\mathbf{P}^0_{cc} \geq \tilde{\mathbf{P}}^*_{cc} \tag{A.19}$$

where $\tilde{\mathbf{P}}^*_{cc}$ is computed by (A.12).

Now given consistent estimates $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ and a known cross-correlation $\tilde{\mathbf{P}}_{ab}$, a consistent representation of the combined estimate $\mathbf{P}^*_{cc}$ can be computed using Eq. (A.14). As explained in Subsection A.1.2.1, the resulting estimate automatically generates a consistent estimate, i.e.

$$\mathbf{P}^*_{cc} \geq \tilde{\mathbf{P}}^*_{cc} \tag{A.20}$$

As a consequence of (A.19) and (A.20), if the inequality

$$\mathbf{P}^0_{cc} \geq \mathbf{P}^*_{cc} \tag{A.21}$$

holds, the consistency of $\mathbf{P}^0_{cc}$ can be guaranteed as per Definition 3.

A.1.3.1 Consistency Analysis in One-Dimension

Suppose we have two estimates $a \sim \mathcal{N}(c^*, \tilde{P}_{aa})$ and $b \sim \mathcal{N}(c^*, \tilde{P}_{bb})$ of some fixed parameter $c^* \in \mathbb{R}$. Consistent estimates of a and b with $P_{aa} \geq \tilde{P}_{aa}$ and $P_{bb} \geq \tilde{P}_{bb}$ are available. The cross-correlation $\tilde{P}_{ab}$ is unknown (cannot be used in the fusion algorithm) and is non-zero. The following is the main result of this subsection.

Theorem 1. *Suppose one computes*

$$\begin{aligned} P^{0\ -1}_{cc} &= P^{-1}_{aa} + P^{-1}_{bb} && \text{(A.22)} \\ P^{0\ -1}_{cc} c &= P^{-1}_{aa} a + P^{-1}_{bb} b && \text{(A.23)} \end{aligned}$$

Then,

$$P^0_{cc} \geq P^*_{cc} \tag{A.24}$$

if and only if

$$\begin{aligned} -\sqrt{P_{aa}P_{bb}} &\leq \tilde{P}_{ab} \leq 0, \quad \text{or} && \text{(A.25)} \\ \left(\frac{P^{-1}_{aa} + P^{-1}_{bb}}{2}\right)^{-1} &\leq \tilde{P}_{ab} \leq \sqrt{P_{aa}P_{bb}} && \text{(A.26)} \end{aligned}$$

where P^*_{cc} is computed via (A.14) using the consistent $P_{aa} \geq \tilde{P}_{aa}$ and $P_{bb} \geq \tilde{P}_{bb}$ and the true $\tilde{P}_{ab}$.

That is in particular, P^0_{cc} is consistent as per Definition 3 when $\tilde{P}_{ab}$ obeys one of the theorem's stated inequalities.

Proof. The inequality (A.24) can be written as

$$\begin{aligned}
&{P^0_{cc}}^{-1} \leq {P^*_{cc}}^{-1} \\
&P_{aa}^{-1}+P_{bb}^{-1} \leq P_{aa}^{-1}+ \\
&\qquad (P_{aa}^{-1}\tilde{P}_{ab}-1)(P_{bb}-\tilde{P}_{ab}^{\top}P_{aa}^{-1}\tilde{P}_{ab})^{-1}(\tilde{P}_{ab}^{\top}P_{aa}^{-1}-1) \\
&P_{aa}^{-1}+P_{bb}^{-1} \leq \frac{P_{aa}+P_{bb}-2\tilde{P}_{ab}}{P_{aa}P_{bb}-\tilde{P}_{ab}^2} \\
&(P_{aa}+P_{bb})(P_{aa}P_{bb}-\tilde{P}_{ab}^2) \leq P_{aa}P_{bb}(P_{aa}+P_{bb}-2\tilde{P}_{ab})
\end{aligned}$$

Rearranging gives

$$\tilde{P}_{ab}\left[(P_{aa}+P_{bb})\tilde{P}_{ab}-2P_{aa}P_{bb}\right] \geq 0$$

and thus

$$\tilde{P}_{ab} \leq 0, \text{ or } \tilde{P}_{ab} \geq \frac{2P_{aa}P_{bb}}{(P_{aa}+P_{bb})}$$

However, the joint covariance matrix

$$\mathbf{P} = \begin{bmatrix} P_{aa} & \tilde{P}_{ab} \\ \tilde{P}_{ab}^{\top} & P_{bb} \end{bmatrix}$$

must be positive definite which yields the upper and lower bounds on $\tilde{P}_{ab}$ and gives

$$\begin{aligned}
-\sqrt{P_{aa}P_{bb}} \leq \tilde{P}_{ab} \leq 0, \quad \text{or} \\
\left(\frac{P_{aa}^{-1}+P_{bb}^{-1}}{2}\right)^{-1} \leq \tilde{P}_{ab} \leq \sqrt{P_{aa}P_{bb}}
\end{aligned}$$

This completes the proof. □

This theorem suggests that if the ignored correlation $\tilde{P}_{ab}$ obeys the inequalities stated in the theorem then the solution provided by P^0_{cc} will still deliver a consistent estimate. An important point here is that P^0_{cc} is always smaller than P^{CI}_{cc} regardless of the correlation and thus offers a higher quality estimate. We state an equivalent result in a different way via the following corollary.

Corollary 2. *Consider the same one-dimensional problem setup as applied in the preceding theorem. For all consistent P_{aa} and P_{bb} there exists a choice of $\tilde{P}_{ab} \neq 0$ such that $P^0_{cc} > P^*_{cc}$ holds with strict inequality. Similarly, for all P_{aa} and P_{bb} there exists a different choice of $\tilde{P}_{ab} \neq 0$ such that $P^0_{cc} < P^*_{cc}$ holds with strict inequality.*

A.1.3.2 Consistency Analysis in Higher Dimensions

Consider two n-dimensional estimates ($n \in \mathbb{N}$) $\mathbf{a} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{aa})$ and $\mathbf{b} \sim \mathcal{N}(\mathbf{c}^*, \tilde{\mathbf{P}}_{bb})$ of some fixed parameter $\mathbf{c}^* \in \mathbb{R}$. We consider a special case where consistent estimates $\mathbf{P}_{aa} \geq \tilde{\mathbf{P}}_{aa}$ and $\mathbf{P}_{bb} \geq \tilde{\mathbf{P}}_{bb}$ are available and are defined in the form of:

$$\mathbf{P}_{aa} = \gamma_a \cdot \mathbf{I}_n \tag{A.27}$$
$$\mathbf{P}_{bb} = \gamma_b \cdot \mathbf{I}_n \tag{A.28}$$

where γ_a and γ_b are scalars and $\mathbf{I}_n$ denotes the $(n \times n)$ identity matrix. The cross-correlation $\tilde{\mathbf{P}}_{ab}$ is unknown but assumed to be in the form of

$$\tilde{\mathbf{P}}_{ab} = \rho \cdot \mathbf{I}_n \tag{A.29}$$

where ρ is the scalar correlation coefficient. The following theorem summarises the main result of this subsection.

Theorem 2. *Suppose one computes*

$$\mathbf{P}^{0\ -1}_{cc} = \mathbf{P}^{-1}_{aa} + \mathbf{P}^{-1}_{bb} \tag{A.30}$$
$$\mathbf{P}^{0\ -1}_{cc}\mathbf{c} = \mathbf{P}^{-1}_{aa}\mathbf{a} + \mathbf{P}^{-1}_{bb}\mathbf{b} \tag{A.31}$$

Then,

$$\mathbf{P}^{0}_{cc} \geq \mathbf{P}^{*}_{cc} \tag{A.32}$$

if and only if

$$-\sqrt{\gamma_a\gamma_b} \leq \rho \leq 0, \quad \text{or} \tag{A.33}$$
$$\left(\frac{\gamma_a{}^{-1} + \gamma_b{}^{-1}}{2}\right)^{-1} \leq \rho \leq \sqrt{\gamma_a\gamma_b} \tag{A.34}$$

where $\mathbf{P}^{*}_{cc}$ is computed via (A.14) using the consistent $\mathbf{P}_{aa} \geq \tilde{\mathbf{P}}_{aa}$ and $\mathbf{P}_{bb} \geq \tilde{\mathbf{P}}_{bb}$ and the true $\tilde{\mathbf{P}}_{ab} = \rho \cdot \mathbf{I}_n$.

That is, $\mathbf{P}^{0}_{cc}$ is consistent as per Definition 3 when ρ obeys one of the inequalities in Equations (A.33) and (A.34). The proof for theorem 2 is fundamentally similar to the proof provided for the one-dimensional case in theorem 1, thus not provided here to avoid repetition.

This theorem suggests that if ρ in (A.29) obeys the inequalities stated in the theorem then the solution provided by $\mathbf{P}^{0}_{cc}$ will still deliver a consistent estimate. An important point here is that $\mathbf{P}^{0}_{cc}$ is always smaller than $\mathbf{P}^{\mathrm{CI}}_{cc}$ of (A.15) regardless of the correlation and thus offers a higher quality estimate. Similar to the one-dimensional case, we state an equivalent result via the following corollary.

Corollary 3. *Consider the same n-dimensional problem setup as applied in the preceding theorem. For all consistent $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ there exists a choice of $\tilde{\mathbf{P}}_{ab} \neq 0$ such that $\mathbf{P}^{0}_{cc} > \mathbf{P}^{*}_{cc}$ holds with strict inequality. Similarly, for all $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ there exists a different choice of $\tilde{\mathbf{P}}_{ab} \neq 0$ such that $\mathbf{P}^{0}_{cc} < \mathbf{P}^{*}_{cc}$ holds with strict inequality.*

A.2 Simulations and Results (Consistency Analysis)

We now provide two simulations to exemplify the theorems stated in Section A.1.3. The first simulation considers the fusion of two unbiased one-dimensional estimates $a \sim \mathcal{N}(0, \tilde{P}_{aa})$ and $b \sim \mathcal{N}(0, \tilde{P}_{bb})$ into estimate c. The covariances of the input estimates are given by $P_{aa} = 1$ and $P_{bb} = 0.3$. Fig. A.1 compares the covariance of the combined estimate c as a function of the cross-correlation $\tilde{P}_{ab}$ for the three classical methods outlined in Section A.1.2. If the ignored correlation $\tilde{P}_{ab}$ satisfies the inequalities (A.25) and (A.26), the covariance of the obtained estimate P^0_{cc} is greater than the covariance of the optimal estimate provided by (A.14), thus guaranteeing a consistent estimate. The covariance of the solution obtained by using CI is always greater than both P^*_{cc} and P^0_{cc}.

Fig. A.2 shows the fusion of two unbiased two-dimensional estimates **a** and **b** represented by $\mathbf{P}_{aa}$ and $\mathbf{P}_{bb}$ where

$$\mathbf{P}_{aa} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } \mathbf{P}_{bb} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

These estimates are represented by their corresponding 2σ uncertainty ellipsoids. The combined estimate **c** using CI and the method ignoring the correlation have been shown. The dashed ellipsoids (green) are the calculated $\mathbf{P}^*_{cc}$ estimates using different values of the cross-correlation matrix (obtained using equal sampling) defined by $\tilde{\mathbf{P}}_{ab} = \rho \cdot \mathbf{I}$ as described in Subsection A.1.3.2. For those values of the cross-correlation $\tilde{\mathbf{P}}_{ab}$ in which the exact optimal value $\mathbf{P}^*_{cc}$ is enclosed by the ellipsoid defined by $\mathbf{P}^0_{cc}$, it is safe to ignore the cross-correlation and still be consistent. However, if the ellipsoid representing the optimal $\mathbf{P}^*_{cc}$ encloses the $\mathbf{P}^0_{cc}$ ellipsoid then ignoring the correlation generates an inconsistent estimate. The CI algorithm achieves a consistent, yet conservative estimate.

A.3 Conclusions

This paper analysed the consistency and applicability of three notable fusion algorithms for combining correlated random variables. It was shown that, although ignoring the non-zero correlation can cause inconsistency in the general case, there are cases where the consistency of the combined estimate can be achieved by simply neglecting the correlation. We derived conditions on the correlation under which one may simply ignore the correlation (as if it were zero) and apply an optimal fusion algorithm. Such conditions were given in the one-dimensional case and in a special case of high-dimensional estimation. This method of fusion will be considerably less conservative than covariance intersection.

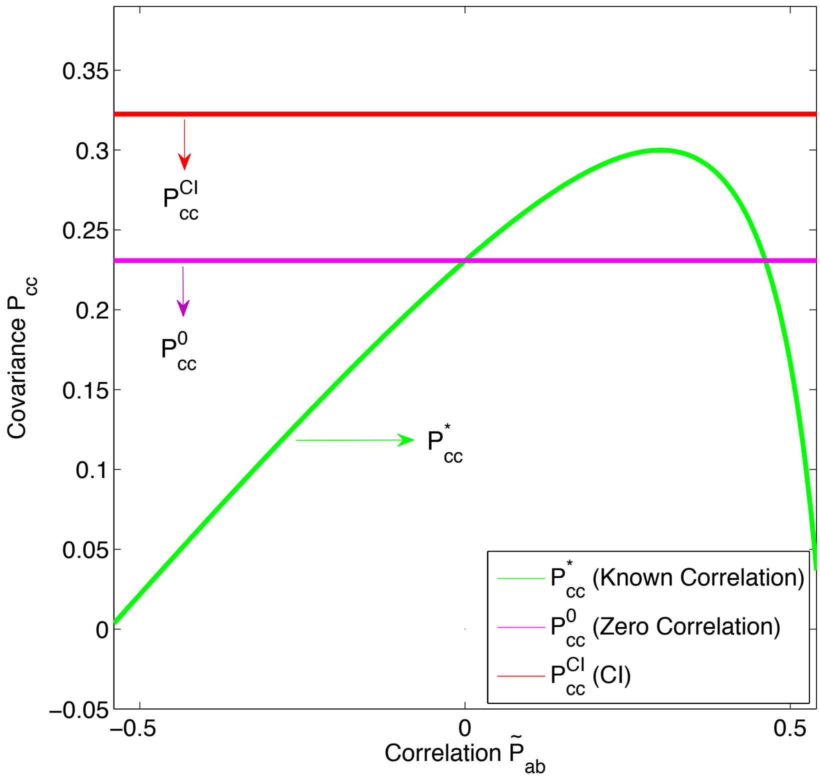


Figure A.1: Comparison of the covariance of the combined estimate as a function of the true cross-correlation $\tilde{P}_{ab}$. For those values of $\tilde{P}_{ab}$ where P_{cc}^{0} is larger than the optimal value P_{cc}^{*}, consistent estimates can be achieved when the correlation is ignored. The intersection points of P_{cc}^{0} and P_{cc}^{*} can be found by looking at the boundaries in (A.25) and (A.26). As expected, the conservative CI estimate P_{cc}^{CI} is always larger than both P_{cc}^{*} and P_{cc}^{0}.

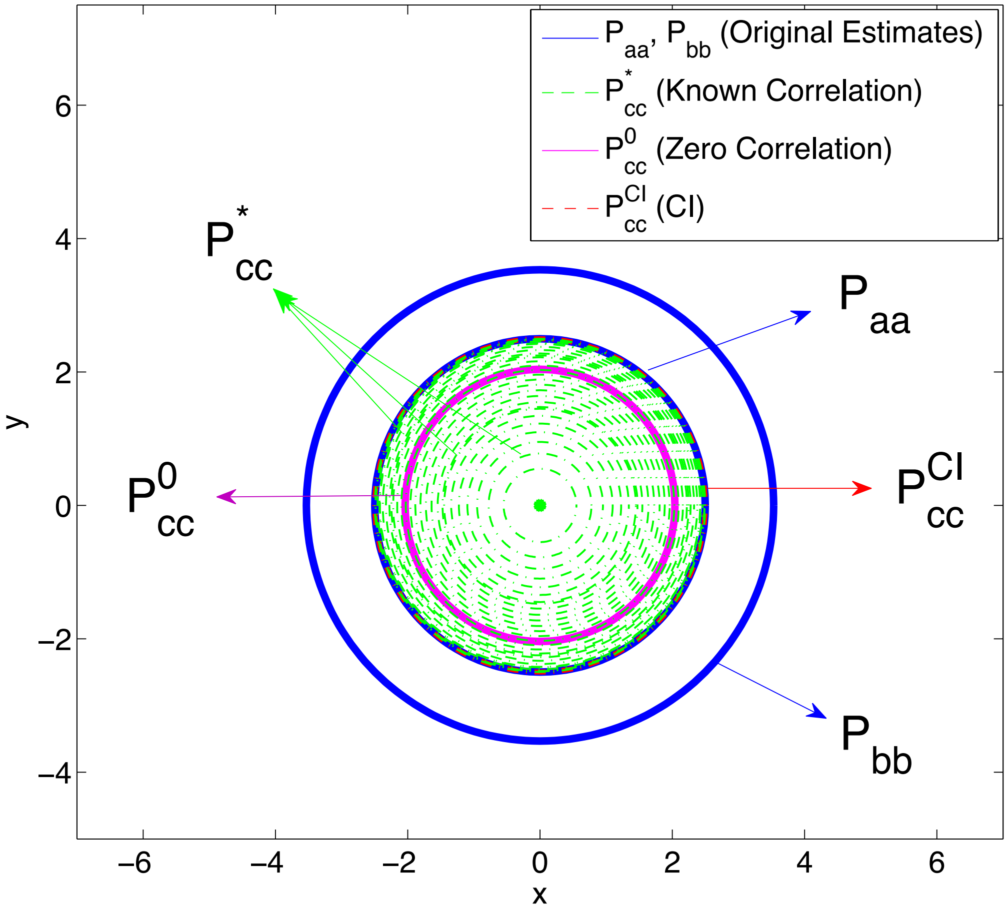


Figure A.2: Comparison of the obtained estimate **c** resulting from fusing 2-D estimates **a** and **b** using different fusion techniques. 2σ uncertainty bounds have been shown using the covariance ellipsoids (circles here).

Appendix B: Practical Considerations in Precise Calibration of a Low-cost MEMS IMU for Road-Mapping Applications

This paper addresses the theoretical and experimental development of a calibration scheme to overcome the intrinsic limitations of a low-cost Micro-Electrical-Mechanical System (MEMS) based Inertial Measurement Unit (IMU). The two-stage calibration algorithm was developed and tested successfully on a six-degree of freedom prototype MEMS IMU to determine the deterministic and stochastic errors of the sensor. This paper makes use of artificial observations known as pseudo-velocity measurements resulting from a specific scheme of rotation to calibrate the IMU in the laboratory environment. The proposed structure is then modified and utilised as a basis for the IMU's error estimation in outdoor navigation applications. For this purpose, the designed calibration method is applied to an integrated GPS/MEMS IMU system, showing improved navigational and road sign positioning performance in a test vehicle.

B.1 Introduction

The last two decades have witnessed an increasing trend towards the use of navigation and positioning technologies in land vehicle applications. The demand for high quality navigation information on one hand and the well-known limitations of the Global Positioning System (GPS) on the other hand have driven the research into employment of Inertial Navigation Systems (INS) in positioning and mapping applications. Probably one of the most widely used inertial sensor assemblies is the Inertial Measurement Unit (IMU). A tri-axial IMU, like the one implemented in this work, includes a triad of gyroscopes and accelerometers, all placed in an orthogonal arrangement with respect to each other.

In spite of their widespread utilisation, the high cost and complexity of traditional inertial navigation systems create some constraints on their use in general purpose civilian applications. The advent of Micro-Electrical-Mechanical Systems (MEMS) has enabled the manufacturing of low-cost inertial sensors. MEMS-based IMUs have proved their use in a myriad of applications from robotics to integrated navigation systems. These sensors are capable of providing reasonably accurate navigation data over short intervals of time. Nevertheless, the major challenge in dealing with them is their notorious error characteristic which leads to degraded performance in the long term [Godha, 2006]. Consequently, determining the associated errors (such as noises, biases, drifts and scale factor instabilities) becomes indispensable in the utilisation of these sensors in real-world navigation applications.

The work in this paper is strongly motivated by a project called AutoMap which aims at developing cost effective methods for automatic creation of digital maps [Petersson, 2010]. The project exploits computer vision algorithms for road sign extraction from video footage captured by land vehicles. The AutoMap project requires continuous positioning information of the fleet vehicle in order to localise the detected landmarks. For this purpose, inertial sensors are used alongside the GPS to obtain synergetic observation effects. The utilisation of low-cost MEMS IMUs enables the development of data collection sensor platforms at a reasonable cost. Consequently, the study and calibration of the MEMS IMU in this work is of vital importance to the future development of the AutoMap technology.

Over the years, several calibration techniques have been developed in different works to address the problem of accumulative errors presented in inertial sensors. Grewal et al. [Grewal et al., 2002] and Foxlin & Naimark [Foxlin and Naimark, 2003] designed a Kalman filter with precise maneuvers to calibrate low-cost IMU sensors for less demanding applications. Their methods have difficulties generating accurate external calibration values such as bias and scale factor errors and they often require costly and high-precision equipment which may not be available to researchers for general orientation measurement applications. Nebot & Durrant-Whyte [Nebot and Durrant-Whyte, 1999] implemented an algorithm for online initial calibration and alignment of an IMU with six degrees-of-freedom (DoF) for land vehicle navigation applications. Kim & Golnaraghi [Kim and Golnaraghi, 2004] studied a calibration process using an optical tracking system. Park & Gao [Park and Gao, 2002] and Syed et al. [Syed et al., 2007] investigated the lab calibration of MEMS-based IMUs by developing a turn table test procedure. Hall & Williams [Hall and Williams II, 2000] developed an electromechanical system for automated calibration of IMUs using GPS antennas. Titterton & Weston [Weston and Titterton, 1997], Farrel [Farrell and Barth, 1999] and Shin & Sheimy [Shin and El-Sheimy, 2002] used a velocity matching alignment method where the attitude of the IMU was being initialised by the GPS velocity information. A common finding of the majority of prior works on calibration is that they do not account for the time-varying errors associated with inertial sensors. We will address this issue by constantly and continuously estimating and compensating for the IMU's stochastic errors. Although online error estimation and fault-detection techniques were previously investigated in other publications such as [Sukkarieh et al., 1999], the context in which they appear in this work is different in implementation and application. The method in [Sukkarieh et al., 1999] investigates the fault detection of IMU and GPS in a GPS/INS fusion system for outdoor applications. However, unlike our method, it does not address the stand-alone calibration of the inertial sensors for removing the time-varying errors in the lab environment.

This paper provides a systematic framework for both lab and in-field calibration of a 6-DoF MEMS-based inertial measurement unit. The proposed calibration scheme is comprised of two distinct phases which are implemented sequentially. Firstly, fixed errors are removed from the stationary IMU during a designed laboratory test in a process referred to as static calibration. Secondly, the extended Kalman filter (EKF) is utilised in a context known as dynamic calibration to estimate the time-varying errors. An intuitive concept called a pseudo-measurement based approach was taken to tackle the dynamic calibration problem in the lab environment. The pseudo-measurement method is closely related, but not identical to the concept of zero velocity update (ZUPT) in [14] and [15]. ZUPT is often used for outdoor applications and it involves performing calibration and resetting the sensor's errors while the vehicle is stationary. However, as will be seen later in this paper, the pseudo-measurement methodology dynamically and continuously estimates and removes the time-varying errors of inertial sensors. In addition, the relaxed rotational scheme in the lab environment, which will be described later, provides a more general type of motion for the calibration experiment compared to the ZUPT method.

Although the pseudo-measurement concept is mainly designed for lab calibration, it can easily be expanded to incorporate GPS measurements to calibrate low-grade IMUs in outdoor navigation applications. Unlike Park and Gao [2002]; Syed et al. [2007]; Winkler et al., the calibration solution described here is independent of any advanced equipment such as turn tables and it does not require precise maneuvers explained in [Grewal et al., 2002; Foxlin and Naimark, 2003]. It provides a simpler operational solution than [Hadfield and Leiser, 1988; Rogers et al., 2002] in that it does not require the frequent stoppage of the vehicle to perform calibration. Moreover, in contrast to [Aggarwal et al., 2006, 2008] it does not require a thermal model and a thermal calibration of the sensor.

The rest of this paper is arranged as follows. Section B.2 starts with a brief overview on the calibration procedure and providing the preliminary definitions. Subsection B.2.1 explains a methodical solution for static calibration of an IMU. Section B.2.2 employs the state space representation to formulate the dynamic calibration algorithm in the extended Kalman filtering context. This section introduces the pseudo-measurement concept and a specific scheme of rotational movement as the main contribution of this paper. Section B.3 provides the IMU calibration results followed by the results of the GPS/MEMS-based IMU fusion system. This system is a real-world navigation application of the pseudo-measurement based calibration framework provided earlier. Finally, the conclusions are drawn from the results in Section B.4.

B.2 Calibration Scheme

Calibration is widely defined as the process of comparing instrument outputs with known reference information. In this process, the coefficients are determined that force the output to agree with the reference information for any range of output values. The error characteristic of MEMS components is often highly nonlinear and temperature dependent. In addition, MEMS-based IMUs are typically not compensated for errors such as biases and scale factors. To achieve the desired accuracy, it is therefore crucial to model the dominating errors and analyse their effects in navigation applications.

Accelerometer bias is defined as an offset in the output that varies randomly from time to time after removing the gravitational term. Gyro bias offset is the measured angular velocity when no rotational motion is present. On the other hand, the scale factor errors of the accelerometer and gyro are errors which are proportional to the sensed quantities. The errors caused by the bias and scale factor in the inaccurate sensor reading accumulate with time and will subsequently lead to the systematic error known as the integration drift in the velocity, position and attitude provided by the unit. The calibration model used in this paper is a simple linear model where the scale factor (S_A) and bias (β_A) are, respectively, the multiplicative and additive factors of the generic variable A. That is,

$$A(t) = S_A \tilde{A}(t) + \beta_A(t), \tag{B.1}$$

where A denotes the real value of the quantity which is being calibrated, and $\tilde{A}$ is the direct reading from the sensor. Other error sources such as axis misalignment errors are not taken into account in this work[1]. Interested readers are referred to [Sukkarieh et al., 1999] and Skog and Händel for axis misalignment estimation.

The measurement errors in an IMU can be categorised into deterministic and stochastic errors [Salychev and University, 1998]. The term deterministic errors refers to fixed biases and scale factor errors. In contrast, stochastic errors vary randomly from time to time which is an intrinsic nature of MEMS sensors. In this paper, the calibration is performed through two consecutive steps. At first, the deterministic errors presented in the raw sensor measurements will be compensated using controlled experimental methods to calculate the *conditioned* inertial quantities. This is called the static calibration procedure. The method employs a variation of the six-position static and rate tests which are discussed in different works [Syed et al., 2007; Weston and Titterton, 1997]. In the second stage, the conditioned accelerations and angular rates from the first step are fed into the designed calibration module as inputs. This structure estimates the stochastic errors presented in the sensor. This step is referred to as the dynamic calibration. The pseudo-measurement concept provided

[1]Since, in the MEMS quality sensor under study, all components are assembled in a single, automated PCB assembly step, misalignment errors can be kept to a minimum.

by a relaxed rotational movement is introduced in this phase as a tool for estimating the time-varying errors of the implemented IMU during a mission.

B.2.1 Static Calibration

The deterministic type of errors in an IMU can usually be determined in controlled laboratory tests. Several procedures (e.g. [Ferraris et al., 1995; Lötters et al., 1998; Won and Golnaraghi, 2010]) have been proposed in the literature to remove the fixed errors of inertial sensors. Typically, for obtaining the biases of inertial sensors, the simplest method is to measure the output reading while the sensor is stationary. The methodology which has been used in this paper is described here[2]. The scale factor of the gyro is determined by using the information from the sensor's data sheet. Using the employed Analog-Digital Converter (ADC) specifications and sensor's sensitivity, a rough estimate, S_{0g}, of the scale factor is calculated:

$$S_{0g} = 2^{(n_b-1)}(\frac{V_{\text{ref}}}{S_{\text{gyro}}}), \tag{B.2}$$

where V_{ref}, S_{gyro} and n_b are the ADC reference voltage, the gyroscope's sensitivity and the number of ADC bits respectively. After taking into account the calculated scale factor, the bias value, $\beta_{0\text{g}}$, is simply determined using the average value of gyro reading over a sufficiently large period (e.g. 2 minutes) so that Equation (B.1) leads to a zero value for a static IMU (a gyro at rest experiences an angular rotation equal to the Earth rotation rate, which is considered negligible for our application). Therefore,

$$\beta_{0\text{g}} = -S_{0\text{g}}\tilde{\omega}_{\text{avg}}, \tag{B.3}$$

where $\tilde{\omega}_{\text{avg}}$ is the average sensor reading for angular rate and $\beta_{0\text{g}}$ and $S_{0\text{g}}$ denote the gyro's constant bias and scale factor, respectively.
Determining the unknown errors of an accelerometer is more subtle than for a gyro. In this paper, the Earth's gravity is used as a physical standard for calibrating the IMU. An accelerometer at rest on the Earth's surface will indicate $1g$ along the vertical axis,

$$\overrightarrow{a_x} + \overrightarrow{a_y} + \overrightarrow{a_z} = \overrightarrow{g}. \tag{B.4}$$

Taking the ℓ^2 Norm of Equation (B.4) yields:

$$\sqrt{||\overrightarrow{a_x}||^2 + ||\overrightarrow{a_y}||^2 + ||\overrightarrow{a_z}||^2} = g. \tag{B.5}$$

Similar to the structure of Equation (B.1), we define the following equation for

[2]Expert readers may skip the static calibration section.

each axis:

$$\bar{a} = S_{0a}\tilde{a}_{\mathrm{avg}} + \beta_{0a}, \tag{B.6}$$

where $\tilde{a}_{\mathrm{avg}}$ denotes the average sensor reading and $\bar{a}$ is the resulting conditioned acceleration after compensating for the deterministic bias $\beta_{0\mathrm{a}}$ and scale factor $S_{0\mathrm{a}}$. Substituting the above equations into Equation (B.5) and squaring both sides of the equation yields:

$$\begin{aligned} &\left(S_{0\mathrm{a,x}}\tilde{a}_{\mathrm{x,avg}} + \beta_{0\mathrm{a,x}}\right)^2 + \\ &+ \left(S_{0\mathrm{a,y}}\tilde{a}_{\mathrm{y,avg}} + \beta_{0\mathrm{a,y}}\right)^2 + \\ &+ \left(S_{0\mathrm{a,z}}\tilde{a}_{\mathrm{z,avg}} + \beta_{0\mathrm{a,z}}\right)^2 = g^2. \end{aligned} \tag{B.7}$$

In theory, at least 6 equations are required to solve for the 6 unknown errors of Equation (B.7). In this work, to be prudent, 12 equations are formed by placing the IMU at 12 different tilt angles, and measuring the accelerations while the sensor is at rest. At each tilt angle, the corresponding reading for each axis is measured and averaged over a random period of time. The average values are then fed into Equation (B.7) to constitute the required set of equations. By taking advantage of regression analysis and curve fitting techniques on the obtained polynomials, the unknown errors of the accelerometer can be successfully computed.

B.2.2 Dynamic Calibration

Due to the nature of low-cost MEMS inertial units, the deterministic errors from Section B.2.1 tend to vary from time to time. Drifts in angle measurements pertaining to the gyro errors, cause the gravity vector to be incorrectly subtracted from the acceleration vector, producing a virtual bias in the predicted acceleration. On the other hand, changes in the environmental conditions, especially the ambient temperature can change the bias and scale factor values. These errors are integrated into progressively larger errors in velocity, which are accumulated into even greater errors in position. It is well known that the bias terms affect the estimated velocity and attitude linearly with time, while they affect the estimated position quadratically [Sukkarieh et al., 2002]. For these reasons, the MEMS-based IMU sensors need to be calibrated frequently during a mission to avoid the accumulation of error and the integration drift phenomena.

The purpose of the designed dynamic calibration process is to statistically estimate the stochastic errors such as turn-on bias and in-run bias, by augmenting them into the state of a stochastic observer [Kim, 2004]. The EKF is used in this paper as the nonlinear state estimator to determine the IMU's stochastic errors. This structure receives a set of measured data from the IMU and estimates the unknown biases and scale factors of the components embedded in the sensor. However, prior to the implementation, the fixed errors are removed from the raw measurements using the static calibration procedure explained in Section B.2.1. Therefore,

$$\bar{a}(t) = S_{0\mathrm{a}}\tilde{a}(t) + \beta_{0\mathrm{a}} \tag{B.8}$$

$$\bar{\omega}(t) = S_{0\mathrm{g}}\tilde{\omega}(t) + \beta_{0\mathrm{g}}, \tag{B.9}$$

where similar to the notation used in Equation (B.6), $\bar{a}$ and $\bar{\omega}$ denote the statically conditioned IMU measurements after the removal of deterministic errors. For the dynamic calibration, the acceleration and angular velocity equations for each axis are defined as:[3]

$$a^b(t) = \left(1 + S_a^b(t)\right)\bar{a}(t) + \beta_a^b(t) \tag{B.10}$$

$$\omega^b(t) = \left(1 + S_g^b(t)\right)\bar{\omega}^b(t) + \beta_g^b(t). \tag{B.11}$$

The two previous equations link the conditioned values $\bar{a}$ and $\bar{\omega}$ from (B.8) and (B.9) to the real acceleration and angular rate values in the body-fixed frame (a^b and ω^b). The t index is used to represent the time-varying nature of the error terms. However, for the sake of simplicity in the notation, continuous or discrete time index (t and k) of these errors are dropped for most equations from now on.

Although this paper will not dwell on the detailed Kalman filtering equations, it provides the required steps to construct the filter model. Equations (B.25) and (B.28) below provide the full discretised system model used with a standard EKF construction. The first step in designing the filter is to identify the state vector x for equations of the model. For a tri-axial IMU, there are 3 orthogonally mounted accelerometers and 3 orthogonal gyroscopes. Since each axis has an unknown bias and an unknown scale factor, the calibration process consists of determining a total number of 12 unknowns. These unknowns are used as a part of the state vector to be estimated directly by the filter. In addition to the above unknown variables, linear velocities of the IMU in the Earth-fixed navigation frame, and the four-component quaternion vector constitute the state vector. A quaternion vector has been preferred over Euler angles to describe the attitude of the sensor in different maneuvering situations[4]. As a result, the state x can be constructed as a 19×1 vector, consisting of velocity, body attitude and stochastic errors (biases and scale factors) according to the following discrete-time representation:

$$x(k) = \begin{bmatrix} v^n(k) & q(k) & \beta_a^b(k) & \beta_g^b(k) & S_a^b(k) & S_g^b(k) \end{bmatrix}^T. \tag{B.12}$$

State Transition Model: The next step is to construct a discrete-time state transition

[3]This is the model which is used for the dynamic calibration. Note that Equations (B.8) and (B.9) are related to the static calibration phase and should not be confused with Equations (B.10) and (B.11) introduced here.

[4]Attitude parameterization using the quaternion is more computationally efficient and numerically accurate than the Euler angle method.

model in the form of the following equation:

$$x(k) = f(x(k-1), u(k), w(k)), \tag{B.13}$$

where $u(k)$ and $w(k)$ denote the control input and the process noise respectively. The conditioned values obtained from Equations (B.8) and (B.9) are used as control inputs for the process model. Therefore, u is a 6×1 vector such that,

$$u(k) = \begin{bmatrix} \bar{a}_x^b(k) & \bar{a}_y^b(k) & \bar{a}_z^b(k) & \bar{\omega}_x^b(k) & \bar{\omega}_y^b(k) & \bar{\omega}_z^b(k) \end{bmatrix}^T. \tag{B.14}$$

Since the conditioned sensor measurements are still in the body-fixed frame, superscript 'b' is used for the vector components in (B.14). For constructing the discrete-time model, we first present the continuous-time equation models followed by the discretization process. The first set of equations of the state transition model (Equation (B.13)), links the rate of change of velocity to the state vector x, and the control inputs u according to:

$$\dot{v^n} = a^n = C_b^n a^b + g^n, \tag{B.15}$$

where $\bar{a}^b = \begin{bmatrix} \bar{a}_x^b & \bar{a}_y^b & \bar{a}_z^b \end{bmatrix}^T$ is formed by extracting the first three components of the control input vector in (B.14) and the 3×3 matrix C_b^n transforms the acceleration quantities in the body-fixed frame to the navigation frame. The gravity vector $g^n = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T$ with g denoting gravity, is used to compensate the effect of the local gravity on the measured acceleration along the Earth's z-axis. Substituting Equation (B.10) into (B.15) results in the velocity equations for the state transition model, that is,

$$\dot{v^n} = C_b^n \left((1 + S_a^b) \circ \bar{a}^b \right) + C_b^n \beta_a^b + g^n, \tag{B.16}$$

where $\circ$ represents the Hadamard product as in [Million, 2011]. This type of product (also known as the component-wise product) is between two matrices or vectors with the same dimensions[5]. C_b^n should be expressed in terms of the filter states (in this case, the quaternions). This is done by using the quaternion transformation as:

$$C_b^n = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & -2(q_0q_3 - q_1q_2) & 2(q_0q_2 + q_1q_3) \\ 2(q_0q_3 + q_1q_2) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & -2(q_0q_1 - q_2q_3) \\ -2(q_0q_2 - q_1q_3) & 2(q_0q_1 + q_2q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{B.17}$$

The second set of equations for the state transition model expresses the orientation of the IMU platform using gyro measurements,

$$\dot{q} = \frac{1}{2} [q \otimes] \check{\omega}_{nb}^b \tag{B.18}$$

[5] Let A and B be $m \times n$ matrices with entries in C. The Hadamard product of A and B is defined by $[A \circ B]_{ij} = [A]_{ij}[B]_{ij}$ for all $1 \leq i \leq$ m, $1 \leq j \leq$ n.

where,

$$[q\otimes] = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \quad and \quad \check{\omega}^b_{nb} = \begin{bmatrix} 0 \\ \omega^b_x \\ \omega^b_y \\ \omega^b_z \end{bmatrix}. \tag{B.19}$$

Equation (B.18) expresses the rate of change of the quaternion in terms of the quaternion and angular velocities from the gyro. In this way, $\dot{q}$ is indirectly linked to the state vector x and control input u. Substituting Equation (B.11) into (B.18) results in the second set of equations for the state transition model, that is,

$$\dot{q} = \frac{1}{2}[q\otimes]\left((1+\check{S}^b_g)\circ\bar{\omega}^b_{nb} + \check{\beta}^b_g\right), \tag{B.20}$$

where $\bar{\omega}^b_{nb}$ is the modified angular velocity (extracted from the control input vector in (B.14)), according to:

$$\bar{\omega}^b_{nb} = \begin{bmatrix} 0 & \bar{\omega}^b_x & \bar{\omega}^b_y & \bar{\omega}^b_z \end{bmatrix}^T; \tag{B.21}$$

and similar to the notation used in Equation (B.19),

$$\check{S}^b_g = \begin{bmatrix} 0 & S^b_{g,x} & S^b_{g,y} & S^b_{g,z} \end{bmatrix} \tag{B.22}$$

$$\check{\beta}^b_g = \begin{bmatrix} 0 & \beta^b_{g,x} & \beta^b_{g,y} & \beta^b_{g,z} \end{bmatrix}. \tag{B.23}$$

The differential equations for the last 12 states in Equation (B.12) are simply:

$$\begin{bmatrix} \dot{\beta}^b_a & \dot{\beta}^b_g & \dot{S}^b_a & \dot{S}^b_g \end{bmatrix}^T = 0. \tag{B.24}$$

This set of equations is based on the assumption that the stochastic errors of inertial sensors vary slowly compared to the dynamics of the moving vehicle. Hence, they are considered as constant values between two consecutive IMU samples throughout the EKF's prediction stage. As will be shown in the next section, these varying errors are estimated during each iteration of the filter. Equations (B.16), (B.18) and (B.24) are the fundamental equations that enable the computation of the state x of the sensor from an initial state $x(0)$ and a series of measurements $\tilde{a}^b$ and $\tilde{\omega}^b$. The salient point here is that these equation are valid for general motion of the IMU in 3D space, regardless of the motion. Since the discrete form of the EKF is used in this paper, the above continuous-time state transition model is discretised [Kim, 2004] using the forward Euler method [Butcher and Corporation, 2008]:

$$\begin{bmatrix} v^n(k) \\ q(k) \\ \beta_a^b(k) \\ \beta_g^b(k) \\ S_a^b(k) \\ S_g^b(k) \end{bmatrix} = \begin{bmatrix} v^n(k-1) \\ q(k-1) \\ \beta_a^b(k-1) \\ \beta_g^b(k-1) \\ S_a^b(k-1) \\ S_g^b(k-1) \end{bmatrix} +$$

$$+ \begin{bmatrix} C_b^n(k)\left((1+S_a^b(k))\circ \bar{a}^b(k)\right) + (\Delta T)C_b^n(k)\beta_a^b(k) + g^n(k) \\ \frac{(\Delta T)}{2}[q\otimes(k)]\left((1+\breve{S}_g^b(k))\circ\bar{\omega}_{nb}^b(k) + \breve{\beta}_g^b(k)\right) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{B.25}$$

where ΔT is the sampling time of the IMU.

Observation Model: The observation model is generally constructed in order to provide a relationship between the observations, the state vector and the control input according to the following equation:

$$z(k) = h(x(k), u(k), v(k)). \tag{B.26}$$

where $v(k)$ denotes the observation noise. The sensor's velocity in the earth-fixed navigation frame is used as the basis for constructing the observation model, that is,

$$z(k) = \begin{bmatrix} v_x^n(k) & v_y^n(k) & v_z^n(k) \end{bmatrix}^T. \tag{B.27}$$

In the proposed dynamic calibration method, the raw IMU data is collected for rotational movements of the sensor about several arbitrary axes where no translational movement is imposed on the sensor's center of mass[6]. Moreover, the calibration starts from the stationary mode with zero initial velocity. As a result of this specific scheme of motion (assuming the linear acceleration caused by manual rotation is negligible), velocity in the navigation frame can be considered equal to zero throughout the calibration process. Since in reality no measuring instrument is used to directly measure the velocity of the sensor in the navigation frame, the term "pseudo-velocity" is used for referring to the mentioned measurements. The pseudo-velocity measurements are used as the filter's observation, therefore,

$$\forall k: \qquad z(k) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T + v(k). \tag{B.28}$$

It is important to note that Equation (B.28) is just the simple case of the general observation model described by Equation (B.27). In applications where the velocity of the sensor is known at each time, this velocity can be used to form the observation model in order to correct the estimated states from the prediction phase of the EKF. This will be shown in Section B.3 where the designed calibration procedure is used as a basis to form the structure of a GPS/INS integrated system.

[6]The method does not require advanced maneuvers of the sensor. The IMU is simply held by hand and rotated around several arbitrary axes by delicate wrist movements.

After the filter is initialised, it enters a loop as long as IMU measurements exist. At any given time (k), the EKF first predicts the state based upon the state estimate from the previous time $(k-1)$. Subsequently, pseudo-velocity observation at the current time is used for further correction of this prediction and to provide a better estimate of the system states. The estimated state and covariance are augmented with the mean and covariance of the process noise. Through this recursive solution of prediction and update, the EKF efficiently estimates the bias and scale factor errors presented in the inertial sensors.

B.3 Experimental Results

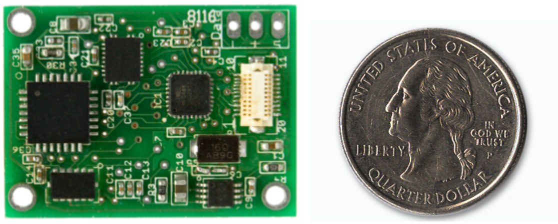


Figure B.1: ThinIMU Micro consists of three accelerometers and three gyros in an orthogonal arrangement. (Dimensions: 31.5×25×5 (mm)).

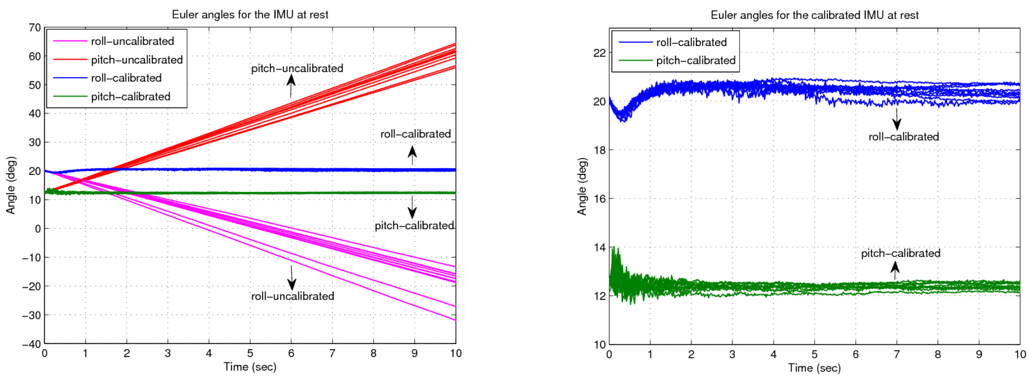


Figure B.2: The roll and pitch angles of a stationary sensor with and without the dynamic calibration (left). The figure on the right is the enlarged view of the Euler angles for the dynamically calibrated sensor.

The inertial data was collected from a prototype IMU known as ThinIMU Micro[7]which is an extremely small and very thin inertial measurement unit with an

[7] Designed and developed by Felix Schill at the School of Engineering, Australian National University.

on-board processor. ThinIMU Micro chip includes an integrated dual-axis gyro [IDG-300 Datasheet, InvenSense, Inc.] for X and Y axes, and a yaw-rate gyro [ADXRS610 Datasheet, Analog Devices Inc. 2007] for Z axis. It also includes a three axis accelerometer [MMA7340L Datasheet, Freescale Semiconductor, Inc., 2007]. Due to the miniature footprint and low price, it is ideal for applications like the AutoMap project with size, weight and cost constraints and it can be easily integrated into a motion capturing suit or a navigation platform. This IMU is depicted in Figure B.1.

The static calibration outlined in Section B.2.1 was performed to obtain the deterministic errors of the sensor. Removing the stochastic errors associated with the IMU was tested for both a stationary and a rotating IMU using the dynamic calibration algorithm described in Section B.2.2.

The left graph in Figure B.2 illustrates the change in the roll and pitch angles for 10 data sequences before and after carrying out the dynamic calibration process. The measurements for all the sequences were collected from a static IMU, while the sensor was left unchanged on the table between two sequences. As can be seen from the figure, the attitude for the uncalibrated IMU diverges with time due to the bias terms presented in the sensor. Furthermore, there is a considerable difference between the attitude results of the uncalibrated IMU from sequence to sequence. This might be due to the variations of the turn-on bias which is an undesirable characteristic of MEMS IMUs. Consequently, it is crucial to remove the bias and scale factor errors associated with the sensor. As can be seen from the figure, the estimated Euler angles after performing the dynamic calibration phase using the pseudo-velocity concept are approximately fixed during the filter run. Figure B.2 (right) is the enlarged view of the calibrated roll and pitch for all the data sequences.

As the second contribution of this work, the calibration scheme described in this paper is applied to a designed GPS/INS integrated system comprising the ThinIMU Micro IMU and an ordinary GPS receiver. The development of the GPS/INS navigation system is enabled through the augmentation of the dynamic calibration method described in Section B.2.2. The integration system is designed by incorporating the developed EKF structure used to estimate the dynamic states of an IMU, with GPS velocity measurements. The main difference between this system and the calibration structure described in Section B.2.2 is the use of GPS outputs instead of the so-called pseudo-velocity measurements as the EKF observed quantities according to:

$$\forall k: \quad z(k) = \begin{bmatrix} V_x^{\mathrm{GPS}}(k) & V_y^{\mathrm{GPS}}(k) & V_z^{\mathrm{GPS}}(k) \end{bmatrix}^T + v_k. \tag{B.29}$$

The described GPS/INS algorithm was run on a data sequence collected by driving around a test vehicle on the trajectory shown in Figure B.3, with the average speed of 50 km/h. The path was chosen to include interesting types of vehicle motion for our navigation application (e.g. straight line, slight turn and sharp turn). The test vehicle is equipped with ThinIMU Micro and a Ublox 5 GPS antenna. Please note that since the physical distance between the two sensors is negligible in our setup, the velocity experienced by the IMU is considered to be the same as the GPS velocity. The GPS velocity updates of Equation (B.29), which are calculated directly from the GPS positioning information, correct and estimate the biases and scale factor errors presented in the IMU. In addition, the structure is capable of estimating the position, velocity and attitude (PVA) of the moving platform. The tuning process of the EKF is a crucial step in the fusion implementation. Tuning was performed by assigning appropriate values to the state covariance matrix (Q) and the observation covariance matrix (R)[8]. The effectiveness of the tuning process was verified by monitoring the velocity innovations and the normalised innovation square (NIS) as a measure of the filter's consistency[9]. Figure B.4 compares the fusion system's performance for the calibrated and uncalibrated inertial sensors for a segment of the nominated trajectory. The behaviour of the accelerometer bias is illustrated in Figure B.5. Other estimated errors are not shown here but they follow the same type of behaviour. Figure B.6 shows an example map output which is acquired by running the sign detection and the GPS/INS fusion algorithms on real data captured by the test vehicle. The estimated trajectory of the vehicle and the location of the detected road signs are illustrated.

B.4 Conclusions and Future Work

A simple and effective calibration procedure was developed and tested successfully on a low-cost 6-DoF MEMS IMU. Pseudo-velocity measurements were used as the virtual observations for estimating the sensor's stochastic errors in the lab. The proposed method overcomes the most important deficiencies associated with previous work in the area.

[8]The exact value of all the tuning parameters are available from the authors on request.

[9]Since the true state values were not available in the experiment, the innovations were used as a statistical measure of consistency, cf. [Castellanos et al., 2007].

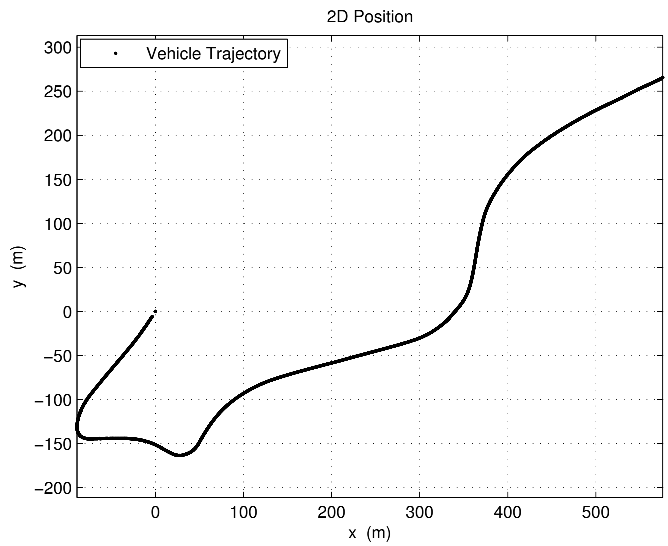


Figure B.3: Test Vehicle's Trajectory

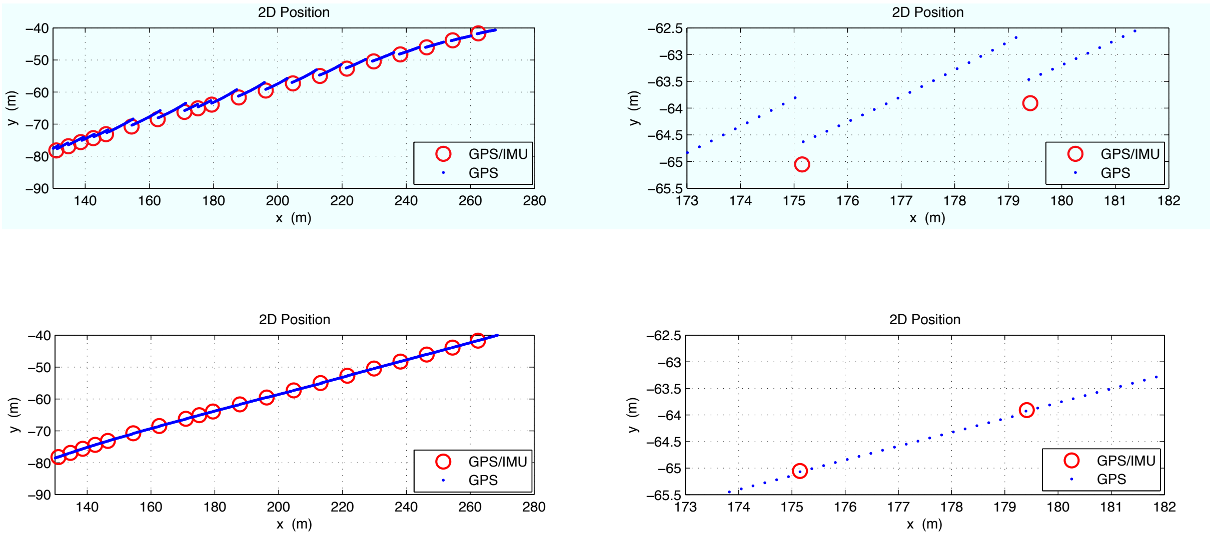


Figure B.4: The GPS/INS integration results for uncalibrated (top) and calibrated (bottom) IMU. The figures at the right are the enlarged view of the left figures.

The effectiveness of the calibration method was investigated through designing a GPS/MEMS-based IMU fusion system for outdoor applications. Although not presented in this paper, promising navigation results were attained for the GPS/INS integration for land vehicles under deliberate GPS dropout. The performance of the fusion system during GPS outage periods can be further improved using a nonlinear

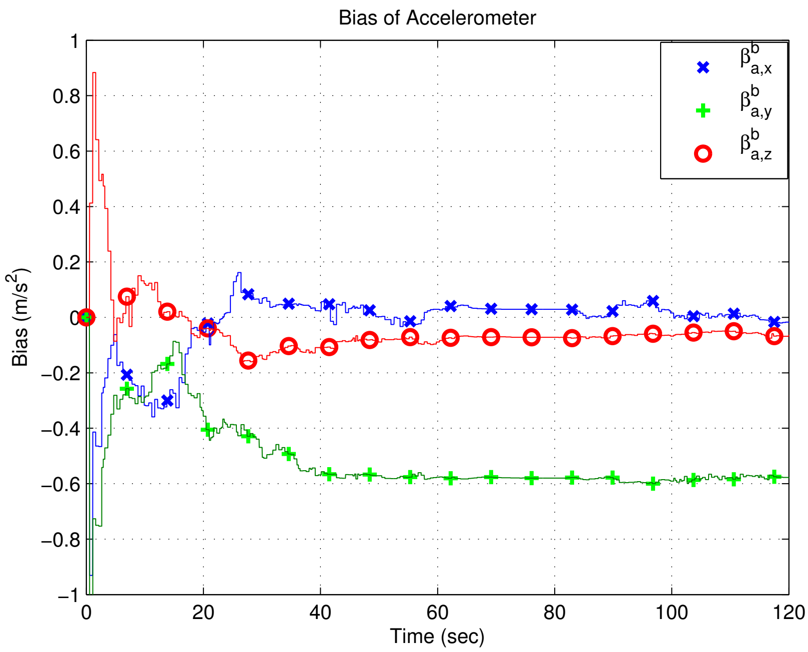


Figure B.5: Stochastic biases of the accelerometer

Figure B.6: Example map output. The resulted trajectory has been obtained using the GPS/MEMS IMU integration described in Section B.3. Location: Canberra, Australia (Source: Google Earth).

smoothing method [Seo et al., 2005]. Running the filtering algorithm both in forward and backward directions and combining the results using a smoother enables the fusion system to alleviate the sensor drifts.

Finally, the utilisation of ThinIMU Micro with the developed calibration procedure has enabled the AutoMap project to accurately localise survey vehicles and geolocate the road signs of interests. The encouraging results merit further investigation into other application domains of the low-cost IMU under study. The calibration methodology discussed in this paper can potentially be used in other field applications with parsimonious consumption of resources.

Bibliography

2002. EPFL and IROS Cleaning Robot Contest. http://robotika.cz/competitions/cleaning2002/en. (cited on page 59)

2008. ESA Lunar Robotics Challenge. http://www.esa.int/Our_Activities/Human_Spaceflight/Research/Lunar_Robotics_Challenge_A_successful_cooperation_within_ESA. (cited on page 59)

2013. DARPA | CyberGrand Challenge. http://www.darpa.mil/cybergrandchallenge.org. (cited on page 59)

2013. RoboCup. http://www.robocup.org. (cited on page 59)

Aggarwal, P.; Syed, Z.; and El-Sheimy, N., 2008. Thermal Calibration of Low Cost MEMS Sensors for Land Vehicle Navigation System. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, 2859–2863. IEEE. (cited on page 175)

Aggarwal, P.; Syed, Z.; Niu, X.; and El-Sheimy, N., 2006. Cost-effective testing and calibration of low cost MEMS sensors for integrated positioning, navigation and mapping systems. In *XXIII FIG (International Federation of Surveyors) Congress. Munich, Germany*. (cited on pages 36 and 175)

Alriksson, P. and Rantzer, A., 2006. Distributed kalman filtering using weighted averaging. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, 1–60. (cited on page 60)

Amirsadri, A.; Bishop, A.; Kim, J.; Trumpf, J.; and Petersson, L., 2012a. A computationally efficient low-bandwidth method for very-large-scale mapping of road signs with multiple vehicles. In *Information Fusion, 2012. To be published in Proceedings. Fusion'12. IEEE International Conference on*. IEEE. (cited on pages 61 and 161)

Amirsadri, A.; Kim, J.; Petersson, L.; and Trumpf, J., 2012b. Practical considerations in precise calibration of a low-cost MEMS IMU for road-mapping applications. In *Proceedings of the 2012 American Control Conference*. IEEE. (cited on page 47)

Anderson, B. D. and Moore, J. B., 2012. *Optimal filtering*. DoverPublications. com. (cited on pages 19 and 93)

Andrade-Cetto, J.; Vidal-Calleja, T.; and Sanfeliu, A., 2005. Stochastic state estimation for simultaneous localization and map building in mobile robotics. *Cutting Edge Robotics*, (2005), 3. (cited on page 61)

Bailey, T.; Nieto, J.; Guivant, J.; Stevens, M.; and Nebot, E., 2006. Consistency of the ekf-slam algorithm. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 3562–3568. IEEE. (cited on pages 98 and 99)

Bar-Shalom, Y.; Li, X.; Kirubarajan, T.; and Wiley, J., 2001. *Estimation with applications to tracking and navigation*. Wiley Online Library. (cited on page 111)

Bar-Shalom, Y.; Li, X. R.; and Kirubarajan, T., 2004. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons. (cited on page 19)

Bertuccelli, L. F. and How, J. P., 2008. Estimation of non-stationary markov chain transition models. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 55–60. IEEE. (cited on page 98)

Blackman, S. and Popoli, R., 1999. *Design and analysis of modern tracking systems*, vol. 685. Artech House Noorwood, MA. (cited on pages 31 and 47)

Braun, J.; Glina, Y.; Hess, A.; Dasey, T.; and Wack, E., 2009. Information fusion for cb defense applications: Challenge and opportunity. In *Technologies for Homeland Security, 2009. HST'09. IEEE Conference on*, 493–502. IEEE. (cited on page 2)

Brooks, R., 1986. A robust layered control system for a mobile robot. *Robotics and Automation, IEEE Journal of*, 2, 1 (1986), 14–23. (cited on page 28)

Brown, R. G.; Hwang, P. Y.; et al., 1992. *Introduction to random signals and applied Kalman filtering*, vol. 1. John Wiley & Sons New York. (cited on page 13)

Bryson, M. and Sukkarieh, S., 2005. An information-theoretic approach to autonomous navigation and guidance of an uninhabited aerial vehicle in unknown environments. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, 3770–3775. IEEE. (cited on pages 2 and 28)

Bryson, M. and Sukkarieh, S., 2007. Co-operative localisation and mapping for multiple uavs in unknown environments. In *Aerospace Conference, 2007 IEEE*, 1–12. IEEE. (cited on pages xxiii, 3, 39, 60, 111, and 115)

Burgard, W.; Moors, M.; Fox, D.; Simmons, R.; and Thrun, S., 2000. Collaborative multi-robot exploration. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 1, 476–481. IEEE. (cited on pages 2, 25, and 26)

Butcher, J. and Corporation, E., 2008. *Numerical methods for ordinary differential equations*. Wiley Online Library. ISBN 0470723351. (cited on page 181)

Cao, Y.; Fukunaga, A.; and Kahng, A., 1997. Cooperative mobile robotics: Antecedents and directions. *Autonomous robots*, 4, 1 (1997), 7–27. (cited on pages 2, 25, and 29)

CASSANDRA, A.; KAELBLING, L.; AND KURIEN, J., 1996. Acting under uncertainty: Discrete bayesian models for mobile-robot navigation. In *Intelligent Robots and Systems' 96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, vol. 2, 963–972. IEEE. (cited on page 26)

CASTELLANOS, J.; MARTINEZ-CANTIN, R.; TARDÓS, J.; AND NEIRA, J., 2007. Robocentric map joining: Improving the consistency of ekf-slam. *Robotics and Autonomous Systems*, 55, 1 (2007), 21–29. (cited on page 185)

CASTELLANOS, J.; MONTIEL, J.; NEIRA, J.; AND TARDÓS, J., 2000. Sensor influence in the performance of simultaneous mobile robot localization and map building. *Experimental Robotics VI*, (2000), 287–296. (cited on pages 2 and 21)

CHEN, L.; ARAMBEL, P.; AND MEHRA, R., 2002. Fusion under unknown correlation-covariance intersection as a special case. In *Information Fusion, 2002. Proceedings of the Fifth International Conference on*, vol. 2, 905–912. IEEE. (cited on page 47)

CHONG, C.; MORI, S.; AND CHANG, K., 1990. Distributed multitarget multisensor tracking. *Multitarget-multisensor tracking: Advanced applications*, 1 (1990), 247–295. (cited on page 33)

DAI, X. AND DU, J., 2009. Hierarchical simultaneous localization and mapping based on discrete event systems. In *Measuring Technology and Mechatronics Automation, 2009. ICMTMA'09. International Conference on*, vol. 2, 234–237. IEEE. (cited on page 31)

DISSANAYAKE, G.; DURRANT-WHYTE, H.; AND BAILEY, T., 2000a. A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2, 1009–1014. IEEE. (cited on page 33)

DISSANAYAKE, M.; NEWMAN, P.; CLARK, S.; DURRANT-WHYTE, H.; AND CSORBA, M., 2001. A solution to the simultaneous localization and map building (SLAM) problem. *Robotics and Automation, IEEE Transactions on*, 17, 3 (2001), 229–241. (cited on pages 21, 32, 47, and 98)

DISSANAYAKE, M.; NEWMAN, P.; DURRANT-WHYTE, H.; CLARK, S.; AND CSORBA, M., 2000b. An experimental and theoretical investigation into simultaneous localisation and map building. *Experimental Robotics VI*, (2000), 265–274. (cited on pages 2 and 21)

DOUCET, A.; DE FREITAS, N.; AND GORDON, N., 2001. *Sequential Monte Carlo methods in practice*. Springer Verlag. (cited on page 28)

DURRANT-WHYTE, H. AND BAILEY, T., 2006. Simultaneous localisation and mapping (SLAM): Part i the essential algorithms. *Robotics and Automation Magazine*, 13, 2 (2006), 99–110. (cited on pages 2 and 21)

Durrant-Whyte, H. and Henderson, T., 2006. Multisensor data fusion. *Australian Center for Field Robotics*, (2006). (cited on pages 26, 29, 30, and 31)

Durrant-Whyte, H. and Henderson, T. C., 2008. Multisensor d. (2008). (cited on page 19)

Durrant-Whyte, H.; Rye, D.; and Nebot, E., 1996. Localization of autonomous guided vehicles. In *Robotics Research*, 613–625. Springer. (cited on page 59)

Eustice, R.; Walter, M.; and Leonard, J., 2005. Sparse extended information filters: Insights into sparsification. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, 3281–3288. IEEE. (cited on page 28)

Farrell, J. and Barth, M., 1999. *The global positioning system and inertial navigation*. McGraw-Hill Professional. ISBN 007022045X. (cited on page 174)

Feder, H.; Leonard, J.; and Smith, C., 1999. Adaptive mobile robot navigation and mapping. *The International Journal of Robotics Research*, 18, 7 (1999), 650. (cited on pages 2, 21, and 61)

Fenwick, J., 2001. *Collaborative concurrent mapping and localization*. Ph.D. thesis, MIT Dept. of Electrical Engineering and Computer Science. (cited on page 27)

Fenwick, J.; Newman, P.; and Leonard, J., 2002. Cooperative concurrent mapping and localization. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 2, 1810–1817. IEEE. (cited on pages 3, 27, and 29)

Ferraris, F.; Grimaldi, U.; and Parvis, M., 1995. Procedure for effortless in-field calibration of three-axis rate gyros and accelerometers. *Sensors and Materials*, 7 (1995), 311–311. (cited on page 177)

Fox, D.; Burgard, W.; Kruppa, H.; and Thrun, S., 2000. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 8, 3 (2000), 325–344. (cited on pages 2, 25, and 26)

Foxlin, E. and Naimark, L., 2003. Miniaturization, calibration & accuracy evaluation of a hybrid self-tracker. In *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on*, 151–160. IEEE. (cited on pages 174 and 175)

Godha, S., 2006. Performance evaluation of low cost MEMS-based IMU integrated with GPS for land vehicle navigation application. *Master's thesis, Department of Geomatics Engineering, University of Calgary, Calgary, Canada*, (2006). (cited on pages 36 and 174)

Grewal, M.; Henderson, V.; and Miyasako, R., 2002. Application of Kalman filtering to the calibration and alignment of inertial navigation systems. *Automatic Control, IEEE Transactions on*, 36, 1 (2002), 3–13. (cited on pages 174 and 175)

GRISETTI, G.; STACHNISS, C.; AND BURGARD, W., 2007. Improved techniques for grid mapping with rao-blackwellized particle filters. *Robotics, IEEE Transactions on*, 23, 1 (2007), 34–46. (cited on page 28)

GUIVANT, J. AND NEBOT, E., 2002. Improving computational and memory requirements of simultaneous localization and map building algorithms. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 3, 2731–2736. IEEE. (cited on page 33)

GUIVANT, J. AND NEBOT, E., 2003. Solving computational and memory requirements of feature-based simultaneous localization and mapping algorithms. *Robotics and Automation, IEEE Transactions on*, 19, 4 (2003), 749–755. (cited on pages 33 and 98)

GUIVANT, J.; NEBOT, E.; AND BAIKER, S., 2000. Autonomous navigation and map building using laser range sensors in outdoor applications. *Journal of robotic systems*, 17, 10 (2000), 565–583. (cited on page 59)

GUIVANT, J. E. AND NEBOT, E. M., 2001. Optimization of the simultaneous localization and map-building algorithm for real-time implementation. *Robotics and Automation, IEEE Transactions on*, 17, 3 (2001), 242–257. (cited on page 98)

GUTMANN, J. AND KONOLIGE, K., 1999. Incremental mapping of large cyclic environments. In *Computational Intelligence in Robotics and Automation, 1999. CIRA'99. Proceedings. 1999 IEEE International Symposium on*, 318–325. IEEE. (cited on pages 2, 21, and 27)

HADFIELD, M. AND LEISER, K., 1988. Ring laser gyros come down to earth: field test results on the rlg modular azimuth position system (maps). In *Position Location and Navigation Symposium, 1988. Record. Navigation into the 21st Century. IEEE PLANS'88., IEEE*, 61–72. IEEE. (cited on page 175)

HALL, J. AND WILLIAMS II, R., 2000. Case study: Inertial measurement unit calibration platform. *Journal of Robotic Systems*, 17, 11 (2000), 623–632. (cited on page 174)

HASHEMIPOUR, H.; ROY, S.; AND LAUB, A., 1988. Decentralized structures for parallel kalman filtering. *Automatic Control, IEEE Transactions on*, 33, 1 (1988), 88–94. (cited on page 31)

J. GUIVANT, J. N. AND NEBOT., E., 2007. VictoriaParkDataset. www.acfr.usyd.edu.au/homepages/academic/enebot/dataset.htm. (cited on page 60)

JAULMES, R.; MOLINÉ, E.; AND OBRIET-LECLEF, J., 2009. Towards a quantitative evaluation of simultaneous localization and mapping methods. In *Control Architecture of Robots national conference*. (cited on pages 8 and 58)

JULIER, S., 2003. The stability of covariance inflation methods for slam. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 3, 2749–2754. IEEE. (cited on page 98)

JULIER, S. AND UHLMANN, J., 1997. A non-divergent estimation algorithm in the presence of unknown correlations. In *American Control Conference, 1997. Proceedings of the 1997*, vol. 4, 2369–2373. IEEE. (cited on pages 47, 89, and 165)

JULIER, S. AND UHLMANN, J., 2001a. General decentralized data fusion with covariance intersection (ci). (2001). (cited on page 99)

JULIER, S. AND UHLMANN, J., 2007. Using covariance intersection for SLAM. *Robotics and Autonomous Systems*, 55 (2007). (cited on page 60)

JULIER, S. J. AND UHLMANN, J. K., 2001b. Building a million beacon map. In *Intelligent Systems and Advanced Manufacturing*, 10–21. International Society for Optics and Photonics. (cited on pages 27 and 34)

KALANDROS, M. AND PAO, L. Y., 1998. Controlling target estimate covariance in centralized multisensor systems. In *American Control Conference, 1998. Proceedings of the 1998*, vol. 5, 2749–2753. IEEE. (cited on page 102)

KIM, A. AND GOLNARAGHI, M., 2004. Initial calibration of an inertial measurement unit using an optical position tracking system. In *Position Location and Navigation Symposium, 2004. PLANS 2004*, 96–101. IEEE. (cited on page 174)

KIM, J., 2004. *Autonomous navigation for airborne applications*. Ph.D. thesis, Dept. of Aerospace, Mechanical and Mechatronic Engineering, Graduate School of Engineering, University of Sydney. (cited on pages 2, 25, 28, 60, 178, and 181)

KIM, J. AND SUKKARIEH, S., 2003. Airborne simultaneous localisation and map building. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 1, 406–411. IEEE. (cited on pages 2, 21, and 27)

KIM, J.; SUKKARIEH, S.; AND WISHART, S., 2006. Real-time navigation, guidance, and control of a UAV using low-cost sensors. In *Field and Service Robotics*, 299–309. Springer. (cited on page 47)

KLEINER, A.; PREDIGER, J.; AND NEBEL, B., 2006. Rfid technology-based exploration and slam for search and rescue. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 4054–4059. IEEE. (cited on page 59)

KUIPERS, B. AND BYUN, Y., 1991. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and autonomous systems*, 8, 1-2 (1991), 47–63. (cited on page 28)

KÜMMERLE, R.; STEDER, B.; DORNHEGE, C.; RUHNKE, M.; GRISETTI, G.; STACHNISS, C.; AND KLEINER, A., 2009. On measuring the accuracy of slam algorithms. *Autonomous Robots*, 27, 4 (2009), 387–407. (cited on pages 8, 58, and 59)

LÁZARO, M. AND CASTELLANOS, J., 2010. Localization of probabilistic robot formations in SLAM. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 3179–3184. IEEE. (cited on pages 2 and 21)

Leonard, J. and Durrant-Whyte, H., 1992. *Directed sonar sensing for mobile robot navigation*. Springer. (cited on page 21)

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

Lerro, D. and Bar-Shalom, Y., 1993. Tracking with debiased consistent converted measurements versus ekf. *Aerospace and Electronic Systems, IEEE Transactions on*, 29, 3 (1993), 1015–1022. (cited on pages 104 and 105)

Levitt, T. and Lawton, D., 1990. Qualitative navigation for mobile robots. *Artificial Intelligence*, 44, 3 (1990), 305–360. (cited on page 28)

Liggins, M.; Chong, C.; Kadar, I.; Alford, M.; Vannicola, V.; Thomopoulos, S.; et al., 1997. Distributed fusion architectures and algorithms for target tracking. *Proceedings of the IEEE*, 85, 1 (1997), 95–107. (cited on pages 2 and 28)

Liu, Y. and Thrun, S., 2003. Results for outdoor-SLAM using sparse extended information filters. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 1, 1227–1233. IEEE. (cited on page 28)

Lötters, J.; Schipper, J.; Veltink, P.; Olthuis, W.; and Bergveld, P., 1998. Procedure for in-use calibration of triaxial accelerometers in medical applications. *Sensors and Actuators A: Physical*, 68, 1-3 (1998), 221–228. (cited on page 177)

Lu, F. and Milios, E., 1997. Robot pose estimation in unknown environments by matching 2d range scans. *Journal of Intelligent and Robotic Systems*, 18, 3 (1997), 249–275. (cited on page 27)

Makarenko, A. and Durrant-Whyte, H., 2004. Decentralized data fusion and control in active sensor networks. In *Proceedings of the Seventh International Conference on Information Fusion*. Citeseer. (cited on page 26)

Maybeck, P., 1979. *Stochastic models, estimation and control*, vol. 141. Academic press. (cited on pages 16 and 19)

Menegatti, E.; Zanella, A.; Zilli, S.; Zorzi, F.; and Pagello, E., 2009. Range-only SLAM with a mobile robot and a wireless sensor networks. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, 8–14. IEEE. (cited on pages 2 and 21)

Miller, R., 1971. Asymptotic behavior of the kalman filter with exponential aging. *AIAA Journal*, 9, 3 (1971), 537–539. (cited on page 98)

Million, E., 2011. The hadamard product. http://buzzard.ups.edu/courses/2007spring/projects/million-paper.pdf. (cited on page 180)

Mornhinweg, D.; Shapiro, D. B.; and Valente, K., 1993. The principal axis theorem over arbitrary fields. *American Mathematical Monthly*, (1993), 749–754. (cited on page 78)

Mourikis, A. and Roumeliotis, S., 2004. Performance bounds for cooperative simultaneous localization and mapping (C-SLAM). *Dept. of Computer Science and Engineering, University of Minnesota, Tech. Rep*, (2004). (cited on page 29)

Mourikis, A. I. and Roumeliotis, S. I., 2006. Analytical characterization of the accuracy of slam without absolute orientation measurements. In *Robotics: Science and Systems*. (cited on pages 8 and 58)

Nebot, E. and Durrant-Whyte, H., 1999. Initial calibration and alignment of low-cost inertial navigation units for land vehicle applications. *Journal of Robotic Systems*, 16, 2 (1999), 81–92. (cited on page 174)

Nerurkar, E.; Roumeliotis, S.; and Martinelli, A., 2009. Distributed maximum a posteriori estimation for multi-robot cooperative localization. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, 1402–1409. IEEE. (cited on pages 2 and 28)

Nettleton, E., 2003. Decentralised architectures for tracking and navigation with multiple flight vehicles. (2003). (cited on pages 2, 3, 28, 39, and 45)

Nettleton, E.; Thrun, S.; Durrant-Whyte, H.; and Sukkarieh, S., 2006. Decentralised SLAM with low-bandwidth communication for teams of vehicles. In *Field and Service Robotics*, 179–188. Springer. (cited on pages 3, 20, 28, 30, 39, 46, and 60)

Niehsen, W., 2002. Information fusion based on fast covariance intersection filtering. In *Information Fusion, 2002. Proceedings of the Fifth International Conference on*, vol. 2, 901–904. IEEE. (cited on page 91)

Ong, L.; Ridley, M.; Kim, J.; Nettleton, E.; and Sukkarieh, S. Six dof decentralised slam. In *Australasian Conf. on Robotics and Automation*, 10–16. Citeseer. (cited on pages 20 and 28)

Ong, L.; Ridley, M.; Kim, J.; Nettleton, E.; and Sukkarieh, S., 2003. Six dof decentralised SLAM. In *Proceedings of the Australasian Conference on Robotics and Automation*, 10–16. (cited on pages 2, 3, 28, and 30)

Overett, G.; Petersson, L.; Andersson, L.; and Pettersson, N., 2009. Boosting a heterogeneous pool of fast hog features for pedestrian and sign detection. In *Intelligent Vehicles Symposium, 2009 IEEE*, 584–590. IEEE. (cited on pages 4 and 34)

Papoulis, A. and Probability, R., 1991. *Stochastic processes*, vol. 3. McGraw-Hill, New York. (cited on page 47)

Park, M. and Gao, Y., 2002. Error analysis of low-cost MEMS-based accelerometers for land vehicle navigation. In *ION GPS 2002: 15 th International Technical Meeting of the Satellite Division of The Institute of Navigation; Portland, OR*. Institute of Navigation, 3975 University Drive, Suite 390, Fairfax, VA, 22030, USA,. (cited on pages 174 and 175)

Petersson, L., 2010. NICTA | AutoMap. http://www.nicta.com.au/research/projects/AutoMap. (cited on page 174)

Petersson, L., 2014. NICTA | AutoMap. http://www.nicta.com.au/research/projects/AutoMap. (cited on page 34)

Piniés, P.; Tardós, J. D.; and Neira, J., 2006. Localization of avalanche victims using robocentric slam. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 3074–3079. IEEE. (cited on page 59)

Rogers, R.; Wit, J.; Crane III, C.; Armstrong, D.; et al., 2002. Integrated INU/DGPS for autonomous vehicle navigation. In *Position Location and Navigation Symposium, 1996., IEEE 1996*, 471–476. IEEE. (cited on page 175)

Rosencrantz, M.; Gordon, G.; and Thrun, S., 2003. Decentralized sensor fusion with distributed particle filters. In *Proc. of UAI*. Citeseer. (cited on page 26)

Salychev, O. and University, B. M. S. T., 1998. *Inertial systems in navigation and geophysics*. Bauman MSTU Press. ISBN 5703813468. (cited on page 176)

Schneider-Fontan, M. and Mataric, M., 1998. Territorial multi-robot task division. *Robotics and Automation, IEEE Transactions on*, 14, 5 (1998), 815–822. (cited on page 26)

Seo, J.; Lee, J.; Park, C.; Lee, H.; and Kim, S., 2005. Application of nonlinear smoothing to integrated GPS/INS navigation system. *Journal of Global Positioning Systems*, 4, 1-2 (2005), 88–94. (cited on page 187)

Sharon, L.; Ling, L.; et al., 2003. Six dof decentralised SLAM. (2003). (cited on pages 3 and 30)

Shin, E. and El-Sheimy, N., 2002. A new calibration method for strapdown inertial navigation systems. *Z. Vermess*, 127 (2002), 1–10. (cited on page 174)

Sim, R. and Roy, N., 2005. Global a-optimal robot exploration in slam. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, 661–666. IEEE. (cited on page 60)

Simmons, R.; Apfelbaum, D.; Burgard, W.; Fox, D.; Moors, M.; Thrun, S.; and Younes, H., 2000. Coordination for multi-robot exploration and mapping. In *Proceedings of the National Conference on Artificial Intelligence*, 852–858. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. (cited on page 26)

Singh, K. and Fujimura, K., 1993. Map making by cooperating mobile robots. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, 254–259. IEEE. (cited on page 26)

Skog, I. and Händel, P. Calibration of a mems inertial measurement unit. In *XVII IMEKO World Congress on Metrology for a Sustainable Development, September*, 17–22. Citeseer. (cited on page 176)

Smith, R. and Cheeseman, P., 1986. On the representation and estimation of spatial uncertainty. *The international journal of Robotics Research*, 5, 4 (1986), 56. (cited on pages 2 and 21)

Smith, R.; Self, M.; and Cheeseman, P., 1990. Estimating uncertain spatial relationships in robotics. *Autonomous robot vehicles*, 1 (1990), 167–193. (cited on page 21)

Stipes, J.; Hawthorne, R.; Scheidt, D.; and Pacifico, D., 2006. Cooperative localization and mapping. In *Networking, Sensing and Control, 2006. ICNSC'06. Proceedings of the 2006 IEEE International Conference on*, 596–601. IEEE. (cited on pages 2 and 28)

Sukkarieh, S.; Nebot, E.; and Durrant-Whyte, H., 1999. A high integrity imu/gps navigation loop for autonomous land vehicle applications. *Robotics and Automation, IEEE Transactions on*, 15, 3 (1999), 572–578. (cited on pages 175 and 176)

Sukkarieh, S.; Nebot, E.; and Durrant-Whyte, H., 2002. Achieving integrity in an INS/GPS navigation loop for autonomous land vehicle applications. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 4, 3437–3442. IEEE. (cited on page 178)

Syed, Z.; Aggarwal, P.; Goodall, C.; Niu, X.; and El-Sheimy, N., 2007. A new multi-position calibration method for MEMS inertial navigation systems. *Measurement Science and Technology*, 18 (2007), 1897. (cited on pages 174, 175, and 176)

Tay, T.-T.; Mareels, I.; and Moore, J. B., 1998. *High performance control*. Springer. (cited on page 17)

Thrun, S., 2001. A probabilistic on-line mapping algorithm for teams of mobile robots. *The International Journal of Robotics Research*, 20, 5 (2001), 335. (cited on pages 2, 26, and 28)

Thrun, S., 2002. Robotic mapping: A survey. *Exploring artificial intelligence in the new millennium*, (2002), 1–35. (cited on page 21)

Thrun, S.; Burgard, W.; and Fox, D., 1998. A probabilistic approach to concurrent mapping and localization for mobile robots. *Autonomous Robots*, 5, 3 (1998), 253–271. (cited on page 27)

THRUN, S.; KOLLER, D.; GHAHMARANI, Z.; AND DURRANT-WHYTE, H., 2002. SLAM updates require constant time. In *Workshop on the Algorithmic Foundations of Robotics*. (cited on pages 20 and 28)

THRUN, S.; KOLLER, D.; GHAHRAMANI, Z.; DURRANT-WHYTE, H.; AND NG, A., 2003. Simultaneous mapping and localization with sparse extended information filters: Theory and initial results. *Algorithmic Foundations of Robotics V*, (2003), 363–380. (cited on page 20)

THRUN, S. AND LIU, Y., 2005. Multi-robot SLAM with sparse extended information filers. *Robotics Research*, (2005), 254–266. (cited on pages 3 and 29)

THRUN, S.; LIU, Y.; KOLLER, D.; NG, A.; GHAHRAMANI, Z.; AND DURRANT-WHYTE, H., 2004. Simultaneous localization and mapping with sparse extended information filters. *The International Journal of Robotics Research*, 23, 7-8 (2004), 693. (cited on page 28)

THRUN, S. AND MONTEMERLO, M., 2006. The graph slam algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25, 5-6 (2006), 403–429. (cited on page 27)

UHLMANN, J.; JULIER, S.; KAMGAR-PARSI, B.; LANZAGORTA, M.; AND SHYU, H., 1999. Nasa mars rover: A testbed for evaluating applications of covariance intersection. In *Proceedings of SPIE*, vol. 3693, 140. (cited on page 47)

VIDAL-CALLEJA, T.; ANDRADE-CETTO, J.; AND SANFELIU, A., 2004. Conditions for suboptimal filter stability in slam. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1, 27–32. IEEE. (cited on page 61)

WACKERLY, D. D.; MENDENHALL, W.; AND SCHEAFFER, R. L., 2008. *Mathematical statistics with applications*. Cengage Learning. (cited on page 103)

WALTER, M. AND LEONARD, J., 2004. An experimental investigation of cooperative SLAM. In *Proceedings of the Fifth IFAC/EURON Symposium on Intelligent Autonomous Vehicles,(Lisbon, Portugal)*. (cited on page 29)

WANG, Y. AND LI, X., 2010. Distributed estimation fusion under unknown cross-correlation: An analytic center approach. In *Information Fusion (FUSION), 2010 13th Conference on*, 1–8. IEEE. (cited on page 47)

WELCH, G. AND BISHOP, G., 1995. An introduction to the kalman filter. (cited on page 14)

WESTON, D. AND TITTERTON, D., 1997. *Strapdown inertial navigation technology*. Lavenham Press Ltd, Lavenham. (cited on pages 174 and 176)

WILLIAMS, S., 2001. *Efficient solutions to autonomous mapping and navigation problems*. Ph.D. thesis, Citeseer. (cited on pages 2, 25, 28, and 33)

WILLIAMS, S.; DISSANAYAKE, G.; AND DURRANT-WHYTE, H., 2002. Towards multi-vehicle simultaneous localisation and mapping. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 3, 2743–2748. IEEE. (cited on pages 26 and 27)

WILLIAMS, S. B.; NEWMAN, P.; DISSANAYAKE, G.; AND DURRANT-WHYTE, H., 2000. Autonomous underwater simultaneous localisation and map building. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2, 1793–1798. IEEE. (cited on page 27)

WINKLER, S.; BUSCHMANN, M.; KORDES, T.; SCHULZ, H.; AND VORSMANN, P. MEMS-based IMU development, calibration and testing for autonomous MAV navigation. In *Proceedings of the ION 59th Annual Meeting and the CIGTF 22nd Guidance Test Symposium (Albuquerque, NM, 23-25 Juni, 2003)*, 128–134. (cited on page 175)

WON, S. AND GOLNARAGHI, F., 2010. A triaxial accelerometer calibration method using a mathematical model. *Instrumentation and Measurement, IEEE Transactions on*, 59, 8 (2010), 2144–2153. (cited on page 177)

WULF, O.; NÜCHTER, A.; HERTZBERG, J.; AND WAGNER, B., 2008. Benchmarking urban six-degree-of-freedom simultaneous localization and mapping. *Journal of Field Robotics*, 25, 3 (2008), 148–163. (cited on page 59)

XIONG, N. AND SVENSSON, P., 2002. Multi-sensor management for information fusion: issues and approaches. *Information fusion*, 3, 2 (2002), 163–186. (cited on page 39)

YAN, L.; LIU, B.; AND ZHOU, D., 2007. Asynchronous multirate multisensor information fusion algorithm. *Aerospace and Electronic Systems, IEEE Transactions on*, 43, 3 (2007), 1135–1146. (cited on page 2)