On the utilization of Simultaneous Localization and Mapping (SLAM) along with vehicle dynamics in Mobile Road Mapping Systems

Savio J. Pereira

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Mechanical Engineering

John B. Ferris, Chair
Corina Sandu
Saied Taheri
Andrew Kurdila
Mark Psiaki

August 15th, 2019
Blacksburg, Virginia

Keywords: SLAM, EKF, SAM, RANSAC
Copyright 2019, Savio J. Pereira
On the utilization of Simultaneous Localization and Mapping (SLAM) along with vehicle dynamics in Mobile Road Mapping Systems

Savio J. Pereira

(ABSTRACT)

Mobile Road Mapping Systems (MRMS) are the current solution to the growing demand for high definition road surface maps in wide ranging applications from pavement management to autonomous vehicle testing. The focus of this research work is to improve the accuracy of MRMS by using the principles of Simultaneous Localization and Mapping (SLAM). First a framework for describing the sensor measurement models in MRMS is developed. Next the problem of estimating the road surface from the set of sensor measurements is formulated as a SLAM problem and two approaches are proposed to solve the formulated problem. The first is an incremental solution wherein sensor measurements are processed in sequence using an Extended Kalman Filter (EKF). The second is a post-processing solution wherein the SLAM problem is formulated as an inference problem over a factor graph and existing factor graph SLAM techniques are used to solve the problem. For the mobile road mapping problem, the road surface being measured is one the primary inputs to the dynamics of the MRMS. Hence, concurrent to the main objective this work also investigates the use of the dynamics of the host vehicle of the system to improve the accuracy of the MRMS. Finally a novel method that builds off the concepts of the popular model fitting algorithm, Random Sampling and Consensus (RANSAC), is developed in order to identify outliers in road surface measurements and estimate the road elevations at grid nodes using these measurements. The developed methods are validated in a simulated environment and the results demonstrate a significant improvement in the accuracy of MRMS over current state-of-the-art methods.
On the utilization of Simultaneous Localization and Mapping (SLAM) along with vehicle dynamics in Mobile Road Mapping Systems

Savio J. Pereira

(MOBILE ROBOT NAVIGATION ABSTRACT)

Mobile Road Mapping Systems (MRMS) are the current solution to the growing demand for high definition road surface maps in wide ranging applications from pavement management to autonomous vehicle testing. The objective of this research work is to improve the accuracy of MRMS by investigating methods to improve the sensor data fusion process. The main focus of this work is to apply the principles from the field of Simultaneous Localization and Mapping (SLAM) in order to improve the accuracy of MRMS. The concept of SLAM has been successfully applied to the field of mobile robot navigation and thus the motivation of this work is to investigate its application to the problem of mobile road mapping. For the mobile road mapping problem, the road surface being measured is one the primary inputs to the dynamics of the MRMS. Hence this work also investigates whether knowledge regarding the dynamics of the system can be used to improve the accuracy. Also developed as part of this work is a novel method for identifying outliers in road surface datasets and estimating elevations at road surface grid nodes. The developed methods are validated in a simulated environment and the results demonstrate a significant improvement in the accuracy of MRMS over current state-of-the-art methods.
Dedication

To God,

To my family,

To my friends
Acknowledgments

First and foremost I thank God for without His will this work would not have been possible. I would like to express my sincere gratitude and utmost appreciation for my advisor and mentor Dr John Ferris, for his patient guidance, unwavering support and the consistent motivation he instilled in me throughout my graduate career. I would also like to thank my committee members Dr Corina Sandu, Dr Saied Taheri, Dr Andrew Kurdila and Dr Mark Psiaki for their invaluable suggestions and feedback that helped to shape this research work.

My time spent at Virginia Tech working at the Vehicle Terrain Performance Laboratory (VTPL) has been memorable and I would like to express my gratitude to all members of VTPL and especially mention my close colleagues Yong Suk Kang and Craig Altmann for their support and encouragement. I am also grateful for all my friends and colleagues at Virginia Tech for without their love and friendship none of this would have been possible. I would also like to express sincere gratitude towards the Newman Community at Virginia Tech that has been a support system and a second home for me during my graduate career.

I would also like to thank everyone at Robotwits LLC for their immense support and encouragement during the last leg of this research, and also a special thanks to all my friends in Pittsburgh. Finally I will say that all this has been possible and worth the while because of the perpetual love and support from family and I cannot say enough to thank my parents and my siblings for it.
Contents

List of Figures xi

List of Tables xiii

1 Introduction 1

1.1 Mobile Road Mapping 2

1.1.1 Sensors in MRMS 2

1.1.2 Data Acquisition and Processing 3

1.1.3 Applications 4

1.2 Motivation and Problem Definition 6

1.3 Thesis Statement and Scope of Work 8

1.4 Planned contributions 9

1.5 Outline of Dissertation: 10

2 Background 11

2.1 History of Mobile Road Mapping Systems (MRMS) 11

2.2 Sensor Technology in MMS 12

2.2.1 Localization (Navigation) sensors: 12

2.2.2 Mapping sensors: 15
2.3 Sensor measurement models: ........................................ 18

2.3.1 Coordinate Frames: ........................................ 18

2.3.2 Inertial Navigation System (INS) mechanization: ........ 21

2.3.3 Global Navigation Satellite System (GNSS) measurements: ........................................ 24

2.3.4 Mapping sensor measurements: ........................................ 27

2.4 Prediction, Filtering and Smoothing: ........................................ 28

2.4.1 Discrete-Linear Filtering ........................................ 30

2.4.2 Extension to Non-linear systems ........................................ 32

2.4.3 Smoothing: ........................................ 35

2.4.4 Localization using GNSS-INS integration ........................................ 36

2.5 Road Surface Modeling ........................................ 37

2.6 Simultaneous Localization and Mapping (SLAM) ........................................ 39

2.6.1 Data association ........................................ 43

2.7 Identifying outliers in mapping sensor measurements: ........................................ 44

2.7.1 Random Sampling and Consensus (RANSAC): ........................................ 45

3 Filtering based solution to the SLAM problem in mobile road mapping ........................................ 47

3.1 Introduction: ........................................ 47

3.2 Modeling the Mapping system ........................................ 48

3.3 Modeling the road surface: ........................................ 50
3.4 SLAM problem formulation: .............................................. 51
3.5 Localization Sensor Models: ............................................. 53
3.6 Mapping sensor measurement model ..................................... 56
   3.6.1 Mapping sensor deviation model: ..................................... 58
3.7 Dynamic Prediction Model: ................................................. 62
3.8 Extended Kalman Filter (EKF) solution: ................................. 66
   3.8.1 Adding and removing grid nodes: ..................................... 67
3.9 Simulation description: ..................................................... 70
3.10 Simulation results .......................................................... 73
3.11 Discussion, future work and conclusions: ......................... 77

4 Smoothing based solution to the SLAM problem in mobile road mapping 79
4.1 Introduction: ............................................................... 79
4.2 Factor graph problem formulation: ....................................... 80
4.3 Sparse Linear Least Squares: ............................................... 88
   4.3.1 Cholesky Factorization .................................................. 90
   4.3.2 QR Factorization ......................................................... 90
4.4 Simulation results .......................................................... 91
4.5 Discussion, future work and conclusion: .............................. 93

5 Identifying outliers in road surface datasets 96
5.1 Introduction ......................................................... 96

5.2 Identifying Outliers and Estimating Grid Node elevations ..................... 98
  5.2.1 Estimating Grid Node elevations .................................. 99
  5.2.2 Closeness to a Grid Node ........................................ 101
  5.2.3 Membership of points in candidate surfaces .......................... 102
  5.2.4 Validity of candidate surfaces ..................................... 103
  5.2.5 Validity of neighborhood points ................................... 105
  5.2.6 Validity of Surfaces and Points .................................... 106
  5.2.7 Identifying outliers .................................................. 106

5.3 Simulation and Results .............................................. 107
  5.3.1 Proof of concept ................................................... 107

5.4 Discussion, Future work and Conclusions .................................... 109

6 Summary, Conclusions and Future work ........................................... 113
  6.1 Summary .............................................................. 113
  6.2 Conclusions .......................................................... 114
  6.3 Future Work .......................................................... 115

Bibliography ........................................................................ 117

Appendices ........................................................................ 128
# List of Figures

1.1 Sensor Configuration on Mitsubishi Mobile Mapping System K-320 4

2.1 GPS pilot signals and ranging codes. Figure from [15] 14

2.2 Schematics showing the working principles of FOG and RLG. 15

2.3 Schematic of Factor Graph SLAM: Blue circles denote robot states, orange circles denote elements of the map. Black squares denote the factors: 'u' denotes factors for odometry/inertial measurements, 'z' denotes factors for mapping sensor measurements, and 'p' denotes prior factors. 43

3.1 Simulated MRMS schematic 71

3.2 Simulated MRMS motion for baseline simulation scenario 73

3.3 ISO 8608 Class A road profile used in baseline simulation scenario 74

3.4 EKFSLAM - Error plots - Single lap 75

3.5 EKFSLAM - Error plots - Double lap 76

4.1 Factor graph schematic for the mobile road mapping problem 82

4.2 SAM - Error plots - Single lap 92

4.3 SAM - Error plots - Double lap 94

5.1 Grid node height estimates for Stair-4 dataset. \(d_{th} = 0.2\) 109
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Parameters of the dynamic model</td>
<td>71</td>
</tr>
<tr>
<td>3.2</td>
<td>Parameters of the localization and mapping sensors</td>
<td>72</td>
</tr>
<tr>
<td>3.3</td>
<td>Parameters of the mapping sensor measurement model</td>
<td>72</td>
</tr>
<tr>
<td>3.4</td>
<td>EKFSLAM - Comparison of error metrics - Single lap</td>
<td>74</td>
</tr>
<tr>
<td>3.5</td>
<td>EKFSLAM - Comparison of error metrics - Double lap</td>
<td>75</td>
</tr>
<tr>
<td>3.6</td>
<td>Effect of road profile roughness on $\delta \xi_g (mm)$</td>
<td>76</td>
</tr>
<tr>
<td>3.7</td>
<td>Effect of longitudinal velocity on $\delta \xi_g (mm)$</td>
<td>77</td>
</tr>
<tr>
<td>4.1</td>
<td>SAM - Comparison of error metrics - Single lap</td>
<td>92</td>
</tr>
<tr>
<td>4.2</td>
<td>SAM - Comparison of error metrics - Double lap</td>
<td>93</td>
</tr>
<tr>
<td>4.3</td>
<td>SAM - Analysis of computation time</td>
<td>93</td>
</tr>
<tr>
<td>5.1</td>
<td>Parameters of the Simulation dataset</td>
<td>107</td>
</tr>
<tr>
<td>5.2</td>
<td>Parameters of the implementations</td>
<td>108</td>
</tr>
<tr>
<td>5.3</td>
<td>Effect of neighborhood radius on RSPC performance</td>
<td>109</td>
</tr>
<tr>
<td>5.4</td>
<td>Effect of tuning parameters on RSPC performance</td>
<td>111</td>
</tr>
</tbody>
</table>
List of Abbreviations

EKF   Extended Kalman Filter

GNSS Global Navigation Satellite System

INS   Inertial Navigation System

MRMS Mobile Road Mapping System

RANSAC Random Sampling and Consensus

RSPC Random Sampling and Probabilistic Consensus

RTS Rauch-Tung-Streibel

SAM Smoothing and Mapping

SLAM Simultaneous Localization and Mapping
Chapter 1

Introduction

The acquisition and processing of road surface maps using mobile mapping technology is a multi-disciplinary field of research where advancements in sensor technology and data fusion algorithms have stimulated immense progress in this field especially over the last two decades. Large scale High Definition (HD) road surface maps are useful for a variety of applications ranging from pavement management, to urban planning, to simulation and testing of Autonomous vehicles and Advanced Driver Assistance Systems to name just a few [72, 91]. Needless to say there is significant motivation to continually improve the accuracy, robustness and efficiency of Mobile Road Mapping Systems (MRMS). Advances in such systems have come either through improvements in the sensor hardware and technology [59] or through advances in sensor data processing and fusion methods [71]. This dissertation is focused on improving the accuracy and robustness of Mobile Road Mapping Systems (MRMS) by researching the techniques used in sensor data processing and fusion. The objective of this work is to investigate the application of the principle of Simultaneous Localization and Mapping (SLAM) as a framework for sensor data processing in Mobile Road Mapping Systems (MRMS). Within this framework, the use of host vehicle dynamics is investigated as a means to improve the accuracy of the system.

The remainder of this chapter is devoted to briefly describing the Mobile Road Mapping problem (Section 1.1), the challenges encountered while processing sensor data to generate road surface maps, which leads to the motivation for this work (Section 1.2) followed by the
thesis statement and scope of work (1.3). At the end of this chapter, the main contributions of this work and an outline of the entire dissertation are presented.

1.1 Mobile Road Mapping

Mobile Mapping Systems (MMS) emerged as a solution to the problems faced in modern data collection campaigns while using traditional surveying methods such as point-wise GPS and total stations [23]. A good comparison between modern Mobile Mapping methods and traditional methods for acquisition of road-way components is provided in [25]. MMS utilize the measurements from the on-board sensors to generate a map of the environment through which they are moving. Mobile Road Mapping Systems (MRMS) are a special class of MMS in which the emphasis is on specifically mapping the surface of the road in order to determine properties such as roughness and texture of the road.

1.1.1 Sensors in MRMS

In a generic sense, an MRMS basically consists of a number of sensors mounted on a mobile platform, which generate measurements of the road surface relative to the mobile platform. The platform can either be airborne, water-based or land-based. Land-based platforms include vans, trains and in some cases are simply a human carrying the sensory system [84]. Due to the relative nature of the measurements and the fact that the sensor platform is mobile, generating the road surface map in an absolute reference frame entails solving the following two problems:

- **Localization**: The problem of estimating the position and orientation (pose) of the mobile platform in an absolute reference frame.
1.1. Mobile Road Mapping

- **Mapping**: The problem of generating a map of the road surface using the estimated pose of the platform and the sensor measurements relative to the mobile platform.

The sensors in an MRMS can be broadly classified into two categories based on their application:

- **Localization sensors (Navigation Sensors)**: The purpose of the localization sensors is to provide data which can be used to estimate the pose of the mobile platform with respect to a suitable reference frame. Current stat-of-the-art MRMS utilize a combination of Global Navigation Satellite System (GNSS) receivers and an Inertial Navigation System (INS) for localization [71]. In addition some other sensors like odometers, compasses are used sometimes to aid and improve the GNSS-INS localization.

- **Mapping Sensors (Imaging sensors)**: Mapping sensors are used to determine the position of remote road surface points relative to the platform on which these sensors are mounted. The most commonly used mapping sensors in MRMS are Digital Imaging Sensors (Cameras) and Laser Scanners.

A typical configuration of most commonly used sensors in a MRMS is shown in Fig. 1.1

### 1.1.2 Data Acquisition and Processing

The data received from the sensors is processed and fused together in order to generate the road surface map. The first step in the processing pipeline is georeferencing the data from all the mapping sensors relative to a suitable frame of reference. This is achieved by first solving the localization problem to estimate the trajectory of the pose of the mobile sensor platform.
Once the pose trajectory of the mobile platform has been estimated, the relative mapping sensor measurements are transformed into the reference frame to generate a point cloud of data. For convenience the absolute reference frame is usually chosen to be earth fixed such as the curvilinear geodetic coordinate system (latitude, longitude & elevation), Universal Transverse Mercator (UTM) system, or 3 degree Transverse Mercator (3TM) system [23].

1.1.3 Applications

Combining the advances in digital imaging, laser scanning and direct geo-referencing has not only increased the efficiency of mobile mapping considerably, but has also resulted in greater flexibility and lower cost [71]. As a result this technology has found applications in many areas, such as:
1. **Vehicle Chassis and Suspension Development:**
   During a chassis and suspension development program, the design engineer uses predicted chassis and suspension loads. It is clear that the road surface is the main excitation to the vehicle [3]. The non-deformable road surface imposes a unidirectional geometric boundary constraint on rolling tires to which the chassis responds by generating loads, moments, motions, deformations, etc. The road surface remains a consistent excitation to the chassis, even as the chassis design changes. Knowledge of this excitation, when applied in conjunction with high fidelity tire and vehicle models, allows chassis loads to be accurately predicted in vehicle simulations [12].

2. **Pavement Construction, Maintenance and Safety:**
   Evaluating the condition of transportation infrastructure is an expensive, labor intensive, and time consuming process. Remote sensing technologies like MRMS offer non-destructive methods for road condition assessment over a large area. These tools provide an opportunity for frequent, comprehensive, and quantitative surveys of transportation infrastructure. Currently, the utilization of multi-source and multi-spectral data can provide enhanced identification of infrastructure deterioration, as well as timely information regarding the trafficability of road networks after a disaster [70].

3. **Autonomous and semi-autonomous Vehicles:**
   Today’s society is advancing rapidly towards full autonomy of road vehicles. The potential benefits include, but not limited to, increased safety, energy conservation, reduced driver stress, increased mobility of non-drivers and pollution reductions[48]. Intelligent vehicle controllers require an accurate representation of the environment for high-level decision making such as route selection and trajectory planning as well as for low-level control operations such as trajectory tracking and obstacle avoidance. Mobile Road Mapping Systems (MRMS) can generate detailed maps of the local driving
environment which can provide information such as road geometry, obstacle/hazard location, road signage information and location [44] and road characteristics such as friction and roughness.

1.2 Motivation and Problem Definition

During the process of acquiring measurements from the sensors in a MRMS, errors are introduced in the measurements from the individual sensors (localization and mapping). The sources of errors are classified based on the group of sensors from which they originate:

Errors in Localization (Navigation) sensors:

Localization error is defined as the difference in the estimated trajectory of the pose of the mobile platform and its true trajectory. The causes of localization error include, but are not limited to:

- Improper calibration and/or synchronization of localization sensors.
- Bias errors, scaling errors & electronic, mechanical, thermal noise corrupting the measurements from the INS sensors.
- Errors in the GNSS measurements which include satellite clock and ephemeris errors, ionospheric delays, multi-path errors etc.
1.2. Motivation and Problem Definition

Errors in mapping sensors:

Most of the commonly used mapping sensors generate measurements by optically sampling the environment. Optical noise can be introduced in such measurements due to improper lighting conditions and this can even generate gross outliers. In the specific case of measuring road surfaces, deformable debris (e.g., leaves, grass) are sampled in the same manner as non-deformable surface features, thus producing outliers in the datasets. Electronic and thermal noise due to sensor circuitry also contribute to erroneous mapping sensor measurements. Mapping sensor measurements with outliers lead to highly inaccurate road surface maps when combined in the conventional method with the localization solution.

As the solution from the localization problem is used in the mapping problem, errors in the localization and mapping sensor measurements are manifest as an overall error in the final road surface map. Since the advent of Mobile Mapping Systems, the localization and mapping problems have conventionally been solved separately and in succession using the sensors specific to the problem. However, recently there has been work investigating the use of the Simultaneous Localization and Mapping (SLAM) as a means to use the mapping sensor measurements and estimated map to improve the estimated localization solution. The estimated localization solution in turn leads to an improved map estimate and this process can be iterated until convergence occurs [49, 65]. The central idea of SLAM is to combine the two independent problems of localizing the mobile platform and then generating a map using relative mapping sensor measurements, into a single estimation problem. The concept of SLAM originated in the field of robotic navigation and mapping, and is broadly defined as the computational problem of localizing a robot within its environment while at the same time generating an accurate map of the environment using data from sensors on-board the robot [11, 21]. Although at the onset this appears to be an inherent ‘chicken and egg’ problem, several solutions to this problem have been developed and successfully tested in the
Chapter 1. Introduction

robotics research community [55, 76]. The application of the concepts and solutions of SLAM specifically in the context of the mobile road mapping problem is developed in this work. An interesting artifact of this specific problem arises due to the fact that the road surface being mapped is one of the primary sources of excitation to the dynamics of the host vehicle of the mapping system. In most of the literature on SLAM, the models used for sensor measurements are based on rigid body kinematics of the robot (mapping system). However, in the specific case of a MRMS there is partial information available through the mapping sensors about the excitation to the dynamics of the mapping system. Thus, it is sought to investigate whether the use of a dynamic model of the mapping system instead of a kinematic one can help to improve the localization solution. It is hypothesized that knowledge of the dynamics of the host vehicle can constrain the estimated pose of the mobile platform along more physically plausible trajectories and thereby improve the localization solution.

1.3 Thesis Statement and Scope of Work

Thesis Statement: The accuracy of a Mobile Road Mapping System (MRMS) can be improved by utilizing the principles of Simultaneous Localization and Mapping (SLAM) along with knowledge of the dynamics of the system.

The overall objective of this dissertation is to develop a framework for the application of Simultaneous Localization and Mapping (SLAM) techniques to the sensor data fusion process in order to improve the accuracy of a Mobile Road Mapping System (MRMS). Within the SLAM framework, utilization of knowledge regarding the dynamics of the system for further improving accuracy of the system is investigated. The developed framework is compared with conventional sensor data fusion techniques used in MRMS. The scope of this work is limited to simulation-based testing & validation of the developed theory and algorithms.
Future work will include testing and validation of the developed framework on a prototype MRMS. Investigation of techniques to improve the sensor hardware and data acquisition technology is beyond the scope of this work.

### 1.4 Planned contributions

The specific contributions of this work in the order of their development are as follows:

1. Development of a framework for the application of the principles of Simultaneous Localization and Mapping (SLAM) to the mobile road mapping problem. Specific focus is given towards utilizing knowledge regarding the dynamics of the system within the developed framework. The solution to the SLAM problem is developed in two stages:

   - **Incremental solution**: In the first stage an Extended Kalman Filter (EKF) is used to process the sensor measurements incrementally in time and estimate the system's current state and the local map of the road surface around the mapping system. The history of the estimated system states and the map of the road surface are stored and utilized as a starting point in the second stage of the solution.

   - **Post-processing solution**: The solution in the second stage is the post-processing solution in which the entire history of the system state and the entire road surface map are simultaneously estimated using the complete set of sensor measurements. The solution to this problem involves inferring the trajectory of the system state and the road surface map from their joint posterior distribution given all of the sensor measurements. The problem is formulated as an inference problem over a factor graph and by exploiting the sparsity of graph is transformed into a sparse
linear least squares problem.

The developed framework is tested and validated on a simulated model of a MRMS in MATLAB and evaluated against current state-of-the-art methods in mobile road mapping.

2. Development of a method for identifying outliers and estimating grid node heights for measured road surface datasets. The developed method draws from the concepts used in Random Sampling and Consensus (RANSAC) algorithm and extends it by using an axiomatic, probabilistic approach. The developed method is validated on a simulated dataset and the results are compared with the existing methods for outlier identification and road surface gridding.

1.5 Outline of Dissertation:

The content of this dissertation is presented as follows: In Chapter 2 a background and literature review on technology used in Mobile Road Mapping Systems (MRMS), Sensor Fusion algorithms, SLAM, outlier identification and other relevant topics is presented. A framework for the application of Simultaneous Localization and Mapping (SLAM) techniques that also incorporates knowledge of the dynamics of a Mobile Road Mapping System (MRMS) is developed in Chapter 3. Chapter 3 also develops the incremental solution to the SLAM problem for MRMS using an Extended Kalman Filter (EKF). The post-processing solution for the SLAM problem for MRMS using factor graph SLAM is developed in Chapter 4. A novel method to identify outliers and estimate grid node heights in road surface measurement datasets is developed in Chapter 5.
Chapter 2

Background

This chapter is intended to present the reader with a theoretical background of the field of mobile road mapping. The topics covered in this chapter are the history of MRMS, sensors used in MRMS and the theoretical estimation techniques for the sensor data processing.

2.1 History of Mobile Road Mapping Systems (MRMS)

The advent of Mobile Mapping Technology dates back to the 1990’s and there has been tremendous progress in the field ever since. Excellent reviews on the technology used in Mobile Mapping Systems (MMS) and their advances over the past few decades can be found in [8, 71, 84]. More recent MMS are characterized by the use of Laser Scanners based on Light Detection and Ranging (LIDAR) technology as the main or complementary mapping subsystem. Based on either the time of flight, phase difference or triangulation measurement principles, LIDAR sensors are able to give a direct measurement of depth (range) to the objects being mapped. This is fundamentally different from the earlier indirect inference of depth using images from multiple cameras. For Mobile Mapping, LIDAR technology offers a number of advantages which include: high speed of data acquisition (saving time and cost), high resolution of data capture (large number of points captured) and direct range measurement. A complete review of modern state-of-the-art MMS based on Laser scanning technology can be found in [61].
Mobile Road Mapping Systems (MRMS) have evolved in conjunction with MMS. Monitoring of road network & railway infrastructure are two of the biggest applications of mobile mapping technology [83, 91]. Automated pavement distress data collected using MRMS helps to support transportation agency officials in making decisions on budget planning and allocation as well as on the design of maintenance and rehabilitation strategies [73]. Scanning lasers based on the triangulation principle are more common in MRMS due to higher achievable accuracy and smaller measurement range [46].

### 2.2 Sensor Technology in MMS

#### 2.2.1 Localization (Navigation) sensors:

Localization is defined as the solution to the problem of estimating the position and orientation (pose) of the sensor platform using measurements from the localization sensors. The current state of the art in MMS & MRMS for localization is a combination of Global Navigation Satellite System (GNSS) receivers and an Inertial Navigation System (INS) due to the high accuracy and precision of such systems [71]. In certain cases the GNSS/INS localization solution is further supplemented by measurements from other localization sensors such as a Distance Measuring Instrument (DMI), magnetometer or odometer.

**Global Navigation Satellite Systems (GNSS):**

The term GNSS is used to refer to a system that uses satellites to provide autonomous geospatial positioning with global coverage. One such system is the 'Navigation Satellite Timing And Ranging Global Positioning System’, or NAVSTAR GPS (more commonly referred to as GPS), a satellite-based radio-navigation system that is capable of providing extremely accurate worldwide, 24-hour, 3-dimensional (latitude, longitude, and elevation) location data.
[90] and velocity data for users with proper receiving equipment. The system was designed and is maintained by the US Department of Defense (DoD) as an accurate, all-weather navigation system. Though initially designed as a military system, it is now available with certain restrictions to civilians. The system has reached its full operational capability, with a complete set of at least 24 satellites orbiting the Earth in a carefully designed pattern [45].

GNSS systems utilize the concept of one-way Time of Arrival (TOA) ranging. In the NAVSTAR GPS system each satellite broadcasts unique ranging codes and satellite navigation data on two distinct carrier signals (L1 at 1575.42 MHz and L2 at 1227.6 MHz) with reference to highly accurate atomic clocks. Two different ranging codes are broadcast, a 1.023 MHz coarse/acquisition code (C/A-code) on L1 and a 10.23 MHz precision code (P-code) on both L1 and L2. A 50 Hz message containing the satellite’s navigation data is superimposed on both the P-code and the C/A-code. The navigation message includes satellite clock-bias data, satellite ephemeris (precise orbital) data for the transmitting satellite, ionospheric signal-propagation correction data, and satellite almanac (coarse orbital) data for the entire constellation [57]. A schematic of the carrier signals and modulated ranging codes is shown in Fig. 2.1.

**Inertial Navigation System (INS):** The most commonplace INS consists of an Inertial Measurements Unit (IMU) and a computer processor. An IMU is a an electronic device that senses a body’s specific forces and angular rates along three orthogonal axes using a combination of accelerometers and gyroscopes. The sensor measurements are then integrated by the processor in a suitable reference frame to keep track of the position and orientation of the body and this process is known as *INS Mechanization.*

The most commonly used IMU’s in MMS can be classified based on the sensing technology employed [59]:
- **IMU’s using Fiber-Optic Gyros (FOG):** FOG utilize the Sagnac effect on counter rotating beams of laser light from an external source passing through a loop of optical fiber. An interferometric phase detector measures the relative phase change between the two beams and this is proportional to the rate of rotation of the fiber loop [30].

- **IMU’s using Ring Laser Gyros (RLG):** RLG also measure rotation rate based on the Sagnac effect, same as the FOG. However, in RLG the laser source emitting counter rotating beams of laser light is located inside a cavity with mirrors at the each of the vertices. One of the mirrors allows enough leakage of light so that the counter-rotating beams can form an interference pattern on a photo-detector array [30].

- **IMU’s based on Micro-Electro Mechanical Systems (MEMS):** MEMS gyros typically utilize tiny quartz tuning forks or vibrating beams as sensors integrated into silicon chips.

FOG and RLG gyroscopes are typically several times more expensive than MEMS gyro-
scopes due to their complex manufacturing and intricate assembly, however they offer much greater accuracy and performance compared to MEMS gyroscopes. A technical study in 2013 however showed that the performance of low-cost MEMS navigation systems is improving rapidly and they are beginning to take the market share away from traditional FOG/RLG systems [28].

![Diagram of FOG gyro](image1)

![Diagram of RLG gyro](image2)

(a) FOG gyro

(b) RLG gyro

Figure 2.2: Schematics showing the working principles of FOG and RLG.

2.2.2 Mapping sensors:

The advancements in MMS are inherently tied to advances in Laser Scanning and Digital Imaging technology. The focus of this work is only on MMS based on Laser Scanning technology.

**Laser Scanning Sensors:**

Laser Scanning using principles of Light Detection and Ranging (LIDAR) is well established in the fields of land-based as well as air-borne mobile mapping. At its core a laser scanning unit consists of a laser light-emitter, an opto-mechanical reflector mechanism and a photodiode to receive the backscattered laser light [32]. Three different principles are commonly
used to compute a range measurement from the received backscattered laser light:

- **Time of Flight (TOF) Principle:** Scanners employing this principle usually emit a series of laser pulses using either a solid state or semiconductor laser. Each pulse has a typical duration of a few nanoseconds. The emitted pulse travels through the atmosphere and interacts with objects in its path. When the beam is emitted a timer counter is started which is stopped once the backscattered beam travels the same path backwards from the reflecting surface and is detected by a system comprising a narrow optical filter and an avalanche photo diode [60]. If the time interval is \( t \), and the average group velocity of light along the path from sensor to object is \( c_g \), the range is computed as,

\[
r = \frac{(c_g \cdot t)}{2}
\]  

(2.1)

The range over which measurements can be made depends on the energy in the laser pulse, the pulse width (in time) and the Pulse Repetition Rate (PRR). Longer range measurements are possible using high energy pulses. With higher pulse rates the emitted energy is lower, thereby reducing the range of measurement. The lower bound on the range of measurement is determined by the pulse duration.

The accuracy and precision of the range measurements depends on the timer measurement accuracy and the accuracy of detecting the backscattered laser light. TOF based scanners are capable of achieving cm-level accuracy under favorable conditions for ranges upto 1 km [60].

- **Phase Shift Measurement Principle:** Laser scanners using this principle emit a continuous wave laser with another signal modulated on it (typically amplitude modulation). If the relative phase difference between the emitted and backscattered received wave is \( \delta \phi \), in radians, then the range is
where, $\lambda$ is the modulation wavelength and $n$ is the unknown number of full wavelengths between the sensor and the reflecting target. Thus, there is an ambiguity in the computed range depending on the value of $n$. For a given modulation wavelength $\lambda$, the distance over which unique range measurements can be obtained is $\lambda$ (determined by substituting $n = 0$ and $\delta \phi = 2\pi$). Typical precision of range measurements are on the order of one percent of the modulation wavelength. Thus, the unique range of measurement can be increased but at the expense of measurement precision. This problem is solved by using multiple modulation wavelengths as shown in Fig where the longest wavelength defines the uniqueness range and the shortest wavelength defines the precision that can be obtained [60]. Phase shift laser scanners are more accurate and precise, but their measurement range is shorter than TOF based scanners.

**Triangulation:** In triangulation the laser emitter projects a plane of laser light as opposed to a beam in the previous methods. The intersection of this plane with the objects of interest in the field of view results in curves of certain shapes. The field of view is simultaneously imaged by a lens which projects the image on a high resolution imaging sensor array. The lens is typically angled with respect to the emitter. The imaging sensor is calibrated in such a way that by knowing the position of the imaged curves on the imaging pixels and the baseline distance between the receiver and emitter, the distance of the objects from the emitter can be estimated. The range over which measurements can be made using this method is limited as the quality of intersection reduces with distance. However, the precision that can be achieved (typically +/-1mm) is much better than TOF and phase shift methods. Modern MRMS typically use triangulation based laser scanners.
2.3 Sensor measurement models:

2.3.1 Coordinate Frames:

The coordinates frames used in inertial navigation are:

1. **Earth Centered Inertial Frame**: *i-frame*

   The *i-frame* is defined to have its origin at the center of the Earth and axes which are non-rotating with respect to the fixed stars. Its z-axis is parallel to the spin axis of the Earth, x-axis pointing towards the mean vernal equinox, and y-axis completing a right-handed orthogonal frame. It is important to note this frame is not truly inertial but it closely approximates an inertial frame [75].

2. **Earth Centered Earth Fixed Frame (ECEF)**: *e-frame*

   The *e-frame* has its origin at the center of mass of the Earth and axes which are fixed with respect to the Earth. Its x-axis points towards the mean meridian of Greenwich, z-axis is parallel to the mean spin axis of the Earth and y-axis completes a right-handed orthogonal frame.

   The rotation rate of the *e-frame* with respect to the *i-frame* with its components resolved in the *e-frame* is given by

   \[
   \omega_{ie} = \begin{bmatrix} 0 & 0 & \omega_e \end{bmatrix}^T
   \]  

   \[ (2.3) \]

   where:

   \( \omega_e \) – Rotation rate of the earth = 7.2921158 \times 10^{-5} \text{ rad/s}

   It is convenient to express the position of the sensor platform in the *e-frame*. Position in the *e-frame* can be expressed either as geodetic co-ordinates of Latitude, Longitude
2.3. Sensor measurement models:

and ellipsoidal height $\mathbf{p}^e = \begin{bmatrix} \varphi & \lambda & h \end{bmatrix}^T$ or as cartesian co-ordinates $\mathbf{p}^e = \begin{bmatrix} x_e & y_e & z_e \end{bmatrix}^T$.

The transformation from geodetic co-ordinates to cartesian co-ordinates is given by:

$$
\begin{bmatrix}
    x_e \\
    y_e \\
    z_e \\
\end{bmatrix} =
\begin{bmatrix}
    (R_N + h)\cos\varphi\cos\lambda \\
    (R_N + h)\cos\varphi\sin\lambda \\
    [(R_N(1-e^2)) + h]\sin\varphi \\
\end{bmatrix}
$$

(2.4)

where,

$R_N$ - Normal radius of the earth

e - Eccentricity.

Transformation from cartesian to geodetic co-ordinates does not have an exact closed form solution and typically an iterative scheme is employed [75].

3. Navigation Frame: n-frame

The n-frame is a locally level geodetic frame. Its X-axis points to geodetic north, its Z-axis is orthogonal to the reference ellipsoid pointing down and Y-axis completing a right-handed orthogonal frame. This convention is called the North-East-Down (NED) system. Another convention for the navigation frame is the East-North-Up (ENU) convention in which the X-axis points towards geodetic east, Y-axis towards geodetic north and Z-axis is orthogonal to the reference ellipsoid pointing up.

The direction cosine matrix from the n-frame to the e-frame is given by

$$
C_n^e =
\begin{bmatrix}
    -\sin\lambda & -\sin\varphi\cos\lambda & \cos\varphi\cos\lambda \\
    \cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi\sin\lambda \\
    0 & \cos\varphi & \sin\varphi
\end{bmatrix}
$$

(2.5)

The earth rotation rate can be expressed in the n-frame using the inverse of the DCM
from Eqn. (2.5)

$$\boldsymbol{\omega}^n_{ie} = \boldsymbol{C}^n_{ie} \boldsymbol{\omega}^e_{ie} = \begin{bmatrix} 0 & \omega_e \cos \varphi & \omega_e \sin \varphi \end{bmatrix}^T$$ (2.6)

The rotation rate of the \textit{n-frame} with respect to the \textit{e-frame} is given by:

$$\omega^n_{en} = \begin{bmatrix} \dot{\lambda} \cos \varphi \\ \phi \\ -\dot{\lambda} \sin \varphi \end{bmatrix}$$ (2.7)

4. \textbf{Body Frame: } \textit{b-frame}

The body frame (b-frame) is the frame on the mapping system in which the specific forces and angular rates measured by the accelerometers and gyroscopes of the IMU are resolved [69]. The axes of the \textit{b-frame} are aligned with those of the IMU.

The gyroscopes in the IMU measure the angular velocity of the body frame (b-frame) with respect to the inertial \textit{i-frame}. The individual gyroscope measurements are along the principal axes of the \textit{b-frame} and thus they can be assembled into a vector of measurements as:

$$\boldsymbol{\omega}^b_{ib} = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$$ (2.8)

Accelerometers sense the specific force in the \textit{b-frame} with respect to the \textit{i-frame} with the components resolved in the \textit{b-frame}

$$\boldsymbol{f}^b = \begin{bmatrix} f_x & f_y & f_z \end{bmatrix}^T$$ (2.9)

It should be mentioned at this point that for the purposes of the simulation-based testing
in this work, an inertial flat earth assumption is made. The effects due to the rotation and curvature of the earth are neglected in the simulation. Hence, for the simulation testing another inertial reference frame is defined.

**Simulation World Frame:** (*w-frame*)

The (*w-frame*) is a stationary inertial frame whose origin is located at a known fixed point on the road surface. The X-axis and Y-axis lie in the horizontal plane of the road surface and are orthogonal to each other. The Z-axis is parallel to the gravity vector and points vertically upwards. The dynamics of the vehicle are derived with respect to the inertial *w-frame*. The IMU measurements are with respect to the inertial *w-frame* with their components resolved in the *b-frame*.

### 2.3.2 Inertial Navigation System (INS) mechanization:

INS mechanization is the process in which the IMU measurements (specific forces and angular velocities) are integrated in the INS processor to predict the position and orientation of the IMU body. The equations of motion are derived through the applications of rigid body kinematics in the inertial *i-frame*. The equations of motion as given in [74] are:

\[
\begin{align*}
\dot{v}^n &= C^n_b f^b - (2\omega^n_{ie} + \omega^n_{en}) \times v^n + g^n & (2.10a) \\
\dot{C}^n_b &= C^n_b (\Omega^b_{ib} - \Omega^b_{in}) & (2.10b)
\end{align*}
\]

where,

- \(v^n = \begin{bmatrix} v_E & v_N & v_U \end{bmatrix}^T\) - Velocity vector in the *n-frame* under the ENU convention

- \(g^n\) - Combined gravity vector in *n-frame* which includes the acceleration due to gravity and the centripetal acceleration due to the rotation of the earth relative to the *i-frame*
\( \omega^n_{en} \times v^n \) - Centripetal acceleration due to the motion of the \( n \)-frame relative to the \( e \)-frame

\( 2\omega^n_{ie} \times v^n \) - Coriolis acceleration

The rotation rate of the \( n \)-frame relative to the \( i \)-frame can be computed as

\[
\omega^b_{in} = C^n_{ib}(\omega^n_{ie} + \omega^n_{en})
\]  

(2.11)

The rate of change of the geodetic co-ordinates in the \( e \)-frame is given by:

\[
\dot{p}^e = \begin{bmatrix} \dot{\varphi} \\ \dot{\lambda} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} \frac{1}{R_M+h}v_N \\ \frac{1}{(R_N+h)\cos\varphi}v_E \\ v_U \end{bmatrix}
\]  

(2.12)

where,

\( R_M \) - Radius of curvature of the earth in the meridian

\( R_N \) - Radius of curvature of the earth in the prime vertical

More detailed descriptions of the INS mechanization equations can be found in \[2, 74, 75\]

The measurements from the IMU contain errors which need to be accounted for in order to perform sensor fusion. The sources of error in IMU measurements are:

- **Sensor Saturation**: The accelerometers and gyro sensors have a maximum range of acceleration and angular rate values that they can sense respectively. Manufacturers typically specify these in \( \text{deg/s} \) or \( \text{deg/hr} \) for angular rates and \( g's \) or \( mg's \) for acceleration.

- **Sensor Non-orthogonality**: Due to manufacturing and mounting tolerances of the
2.3. Sensor measurement models:

sensors, the axes of the sensors may not be perfectly orthogonal to each other. This misalignment can be modeled deterministically and eliminated through careful factory calibration.

- **Scale Factor:** It is defined as the factor by which the sensor in the IMU scales the physical input when producing its output. It has a linear part and non-linear part. The linear part is modeled deterministically and non-linear part modeled using a stochastic process. Scale factor is usually specified in units of parts-per-million (ppm).

- **Bias Error:** For a given physical input (acceleration or angular rate), the sensor measurement is usually offset by a term called the "bias error". The bias error consists of two components:

  - **Constant Bias** This is the constant offset in the sensor measurement from the true angular rate or specific force measurement. It is usually modeled as a deterministic variable and can be estimated by taking a long term average of the sensor output whilst it is stationary.

  - **Bias Drift** The bias error does not remain constant during operation of the IMU. The reasons for its variation include changes in temperature and mechanical stress on the system. This component of the bias error is stochastic and is usually modeled as either a 1st order Gauss Markov Process, Random Walk process, autoregressive process or a combination of them. These models are used to augment the INS/GNSS integration models so that the bias drift can also be estimated during operation. A detailed analysis of the different techniques used to model the bias drift and estimate their parameters is given in [58, 62]. Manufacturers specify a bias stability measurement which describes the change in the bias over a specified period of time, typically 100 seconds [92]. Bias stability is
usually specified as the standard deviation of the bias over the specified time in units of $\text{deg/hr}$ for gyroscopes and $\text{mg}$ for accelerometers.

- **Wide-band Noise**: It is the error caused to thermal, electronic and/or mechanical noise corrupting the inertial sensor measurements. Due to the integration of IMU measurement the wide band noise results in a 1\textsuperscript{st} order random walk in the estimated angles and velocity. Thus, manufacturers typically specify the wideband noise magnitude as an angular random walk measurement ($\text{deg/}\sqrt{\text{hr}}$) for gyroscopes and a velocity random walk measurement ($\text{m/s/}\sqrt{\text{hr}}$) for accelerometers. Alternately, the wideband noise magnitude can also be expressed using the spectral density units ($\text{deg/hr}/\sqrt{\text{Hz}}$ and $\text{m/s}^2/\sqrt{\text{Hz}}$) \cite{92}.

### 2.3.3 Global Navigation Satellite System (GNSS) measurements:

Modern GNSS receivers acquire the carrier signals (L1 and L2) along with the ranging codes and navigation data about the satellites and then process them to compute the position and velocity of the receiver in free space. The GNSS receiver clock does not always remain synchronized with the satellite clock system and hence it is necessary to estimate the receiver clock bias, $\delta t_r$, as well, and for this measurements from a minimum of 4 satellites are required \cite{10,30}. In the case of GPS, the satellite positions and velocities are reported in WGS-84 (World Geodetic System of 1984) co-ordinates whose reference frame is the $e$-frame \cite{78}.

Let,

$$P_i^e = \begin{bmatrix} X_i & Y_i & Z_i \end{bmatrix}^T - \text{Position vector of the } i^{th} \text{ satellite in } e\text{-frame}$$

$$p^e = \begin{bmatrix} x & y & z \end{bmatrix}^T - \text{Position vector of the GNSS receiver in } e\text{-frame}$$

$\delta t_r$ - Receiver clock offset

$c$ - Speed of light
2.3. Sensor measurement models:  

The ideal pseudorange is defined as the sum of the geometric range from the receiver to the transmitting satellite and the range equivalent due to the receiver clock offset from satellite time. The ideal (noiseless) pseudo range from the receiver to the \(i^{th}\) satellite, \(\psi_i\), is given by:

\[
\psi_i = \sqrt{(X_i - x)^2 + (Y_i - y)^2 + (Z_i - z)^2 + c\delta t_r}
\]  

(2.13)

Since the measurement equations are non-linear, they are typically solved by linearizing about a nominal position of the receiver and predicted receiver clock offset [10]. For each signal received from a satellite, a typical GNSS receiver is capable of producing two basic measurements (observables) that both give a measure of the ideal pseudorange:

1. **Code-phase measurement:** In this case the pseudorange is estimated by examining the correlation of the ranging code received form the satellite with a sliding copy of the ranging code generated in the receiver. The measurement equation is:

\[
\rho = \psi + \beta_\rho + \eta_\rho
\]  

(2.14)

where,

- \(\beta_\rho\) - Pseudorange signal-in-space error
- \(\eta_\rho\) - Pseudorange receiver tracking error

2. **Carrier phase measurement:** The phase of the GNSS carrier signal (measured in cycles) also provides a measurement of the ideal pseudorange through the following relation:

\[
\lambda\phi = \psi + \lambda N + \beta_\phi + \eta_\phi
\]  

(2.15)

where,

- \(N\) - Number of cycles of the carrier signal between the satellite and receiver (also known
as integer cycle ambiguity)

\( \lambda \) - Wavelength of the carrier signal

\( \beta_\phi \) - Carrier phase signal-in-space error

\( \eta_\phi \) - Carrier phase receiver tracking error

The signal in space error terms (\( \beta_\rho \) and \( \beta_\phi \)) are independent of the receiver type and are composed of the satellite clock errors, ephemeris errors, tropospheric refraction errors and ionospheric refraction errors. The receiver tracking error terms (\( \eta_\rho \) and \( \eta_\phi \)), are specific to each receiver and consist of the multi-path error and receiver measurement noise. A detailed description of the individual error terms in GNSS receiver measurements is given in [30].

A major difference between the code-phase and carrier phase measurements is the magnitude of the receiver noise. A general rule of thumb is that the accuracy of the measurement is about 1 % of the signal wavelength. Thus for example when using the GPS C/A code for pseudorange measurements the accuracy is around 3 m and for the P-code it is around 30 cm. On the other hand the carrier signals have a much smaller wavelength and the accuracy is on the order of a few centimeters [1]. Another difference is that the code-phase measurements can be directly processed using Eqn.(2.13) to estimate the receiver position. However, due to the integer cycle ambiguity (\( N \)), directly processing the carrier phase measurements to estimate the receiver position is not possible unless the ambiguity can be fully resolved. In order to benefit from the high precision of the carrier phase measurements, they need to be processed using more specialized techniques. One such technique involves using the difference in the carrier phase over short time intervals to approximate the Doppler frequency (shift) due to relative motion between the receiver and satellite. The Doppler shift approximation, \( D \), (in Hz) is given by

\[
D \approx \frac{(\phi_t - \phi_{t-\Delta t})}{\Delta t}
\]  

(2.16)

where, \( \Delta t \) is the GNSS sampling interval. The Doppler shift in turn can be used to estimate
2.3. Sensor measurement models:

the velocity of the receiver by using the equation [17]

\[
D = \mathbf{r} \cdot (\mathbf{V}^e - \mathbf{v}^e) + c\delta t_r
\]  \hspace{1cm} (2.17)

where,

\( \mathbf{r} \) – Unit vector from the receiver to the satellite

\( \mathbf{V}^e \) – Satellite velocity vector

\( \mathbf{v}^e \) – Receiver velocity vector

\( \delta t_r \) – Rate of change of the receiver clock offset

Other specialized techniques of utilizing carrier phase measurements are carrier smoothing of the code-phase measurement, carrier measurement differencing with respect to a fixed known location and standalone position with precise satellite orbital and clock parameters. For details about these methods the reader is referred to [10].

2.3.4 Mapping sensor measurements:

The mapping sensors that are considered in this work are laser scanners which typically measure the geometric range from a reference point on the sensor to the road surface. Let \( \mathbf{m}_0^w \) denote the reference point on the mapping sensor and let \( \mathbf{r}^w \) denote the corresponding measured road surface point (both in the \( w \)-frame). Then, the mapping sensor range measurement denoted as \( m \), is given by

\[
m = \sqrt{(\mathbf{m}_0^w - \mathbf{r}^w)^T(\mathbf{m}_0^w - \mathbf{r}^w)}
\]  \hspace{1cm} (2.18)

The mapping sensors in MRMS typically contain a significant percentage of outlier. For the mapping sensors, outliers are identified as points that do not belong to the local neighborhood
and do not obey the local surface geometry; thereby causing problems in the further steps such as segmentation or mesh generation [79]. The common sources of outliers in laser scanner point clouds are: [79]

- **Boundaries of Surfaces**: Such outliers occur due to the finite size of the laser beam footprint. At a sharp boundary, the beam is split into two or more sections and the measured point is computed as a weighted average of the irradiance reflected from the multiple surfaces.

- **Surface Reflectance**: The ideal surface for laser scanning is one that is opaque and diffusely reflecting. Surfaces whose properties deviate from this ideal case; such as glass, plastics, machined metals, or marble; can generate biases and uncertainties in the measurements as was showed in a study by [9], where vapour-blasted aluminium (VBAI) was used as a reference opaque surface.

- **Multi-path Reflection**: Such outliers occur when the laser beam is reflected from multiple surfaces before reaching the detector thereby increase the round trip time or the relative phase shift of the reflected beam.

### 2.4 Prediction, Filtering and Smoothing:

A mapping system can be considered as a stochastic dynamical system for which noisy measurements are available through the sensors. The field of Estimation Theory provides a formal probabilistic framework for estimating the state of such a system. Consider the
2.4. Prediction, Filtering and Smoothing:

following non-linear stochastic dynamical state space system

\[
\dot{x}(t) = f(x(t), t) + G(t)w(t) \quad (2.19a)
\]
\[
z_k = h(x_k, k) + v_k \quad (2.19b)
\]

where,

\(t\) - Continuous time variable

\(k\) - Discrete time index

\(x \in \mathbb{R}^{n_x}\) - State vector of the system.

\(z \in \mathbb{R}^{n_z}\) - Measurements of the system

\(\{w(t)\}\) - \(n_w\)-dimensional zero mean, white Gaussian process noise with

\[\mathbb{E}[w(t)w(\tau)] = Q(t)\delta(t - \tau)\]

(2.20)

\(G(t) \in \mathbb{R}^{n_x \times n_w}\) - Matrix mapping from process noise to rate of change of system state

\(\{v_k\}\) - \(n_z\)-dimensional zero mean white Gaussian measurement noise sequence with

\[v_k \sim \mathcal{N}(0, R_k) , \quad R_k > 0\]

(2.21)

Such a system is referred to as a Continuous-Discrete system since the state vector evolves continuously over time through a stochastic vector differential equation while measurements of the system are only available at discrete time steps. Let \(Z_l\) denote the sequence of measurements up to discrete time step \(l\)

\[Z_l = \{z_1, z_2, \ldots, z_l\}\]

(2.22)

The estimation problem can be then be defined as that of computing an estimate of \(x_l\) given
If \( t > t_l \), the estimation problem is referred to as the prediction problem, if \( t = t_l \) then it is the filtering problem while if \( t < t_l \) then it is referred to as the smoothing problem. The complete solution to the estimation problem for the stochastic dynamical system in Eqn. (2.19) is the conditional probability density function \( p(x_t|Z_l) \) since it embodies all the statistical information about \( x_t \) contained in the measurements \( Z_l \). Once the conditional probability density function is determined, an estimate of \( x_t \) can be inferred from it. In general, determining the conditional density function may not be tractable in most cases. For certain special cases however, closed form solutions can be obtained.

### 2.4.1 Discrete-Linear Filtering

Consider the special case of a discrete linear stochastic dynamical system

\[
\begin{align*}
x_k &= \Phi_k x_{k-1} + \Gamma_{k-1} w_k \\ z_k &= H_k x_k + v_k
\end{align*}
\]  

(2.23a)

(2.23b)

where,

- \( \Phi_k \in \mathbb{R}^{n_x \times n_x} \) - State transition matrix
- \( \Gamma_{k-1} \in \mathbb{R}^{n_x \times n_w} \) - Process noise mapping matrix
- \( H_k \in \mathbb{R}^{n_z \times n_x} \) - Measurement matrix
- \( \{w_k\} \) - \( n_w \)-dimensional zero mean white Gaussian noise sequence with

\[
w_k \sim \mathcal{N}(0, Q_k) , \quad Q_k > 0
\]  

(2.24)

For the special case of a linear stochastic dynamical system such as the one in Eqn. (2.23), it can be shown that the conditional density \( p(x_k|Z_l) \) is Gaussian, and hence fully characterized by mean vector and covariance matrix of \( x_k \) [34]. The mean and covariance of the conditional
2.4. Prediction, Filtering and Smoothing:

density at time step $k$ given measurements up to time step $l$ are denoted as $\hat{x}_{k|l}$ and $\hat{P}_{k|l}$ respectively. Furthermore it can also be shown that the conditional mean is the optimal estimate of $x_k$, in the minimum mean squared error sense [34] (Theorem 5.3). The well known Kalman Filter [40] is the optimal (minimum mean square error) filter for the discrete linear stochastic dynamical system of Eqn. (2.23). It consists of equations for the evolution of the conditional mean and covariance. Between the measurements, the following difference equations apply:

\[
\begin{align*}
\dot{x}_{k|k-1} &= \Phi_k \hat{x}_{k-1|k-1} \\
\dot{P}_{k|k-1} &= \Phi_k \hat{P}_{k-1|k-1} \Phi_k^T + \Gamma_{k-1} Q_k \Gamma_{k-1}^T
\end{align*}
\] (2.25a)

(2.25b)

Whenever an observation (measurement) is received at time $t_k$, the conditional mean and covariance are updated as:

\[
\begin{align*}
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k (z_k - H_k \hat{x}_{k|k-1}) \\
\hat{P}_{k|k} &= (I - K_k H_k) \hat{P}_{k|k-1} (I - K_k H_k)^T + K_k R_k K_k^T
\end{align*}
\] (2.26a)

(2.26b)

where, $K_k$ is the Kalman gain which is given by

\[
K_k = \hat{P}_{k|k-1} H_k^T \left[ H_k \hat{P}_{k|k-1} H_k^T + R_k \right]^{-1}
\] (2.27)

The form of the covariance update/correction used in Eqn.(2.26b) is widely referred to as the Joseph form and has the advantage of preserving the positive definite property of the covariance matrix in the presence of numerical round-off errors.
2.4.2 Extension to Non-linear systems

In practice most real-world systems like the one in Eqn. (2.19) are non-linear with continuous dynamics and in order to apply the results from discrete linear optimal filtering, the non-linear continuous system must be linearized (approximated) and discretized. Let \( \bar{x}(t) \) denote a reference deterministic trajectory that satisfies

\[
\dot{\bar{x}}(t) = f(\bar{x}(t), t), \quad t \geq t_0 \tag{2.28}
\]

The deviation (perturbation) of the state from the reference trajectory is denoted as

\[
\delta x(t) = x(t) - \bar{x}(t) \tag{2.29}
\]

Taking the time derivative of Eqn. (2.29)

\[
\delta \dot{x}(t) = f(x(t), t) + G(t)w(t) - f(\bar{x}(t), t) \tag{2.30}
\]

By performing a Taylor series expansion of \( f(\cdot) \) about \( \bar{x}(t) \) and neglecting second order and higher terms

\[
f(x(t), t) \approx f(\bar{x}(t), t) + F[t, \bar{x}(t)]\delta x(t) \tag{2.31}
\]

where,

\[
F[t, \bar{x}(t)] = \left. \frac{\partial f}{\partial x} \right|_{x(t) = \bar{x}(t)}
\]

Substituting in Eqn. (2.30)

\[
\delta \dot{x}(t) = F[\bar{x}(t), t]\delta x(t) + G(t)w(t) \tag{2.32}
\]
2.4. Prediction, Filtering and Smoothing:

The system in Eqn. (2.32) is a continuous linear system whose discrete equivalent is given by

\[ \delta x_k = \Phi_k \delta x_{k-1} + w_k \] (2.33)

where,

\[ \Phi_k - \text{State transition matrix of the deviation} \]

\[ w_k = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau)G(\tau)w_\tau d\tau \]

As shown in [34], \( \{w_k\} \) is a zero-mean white Gaussian sequence, \( w_{k+1} \sim N(0, Q(k + 1)) \)

where

\[ Q(k + 1) = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)G(\tau)Q(\tau)G(\tau)\Phi^T(t_{k+1}, \tau) d\tau \] (2.34)

Similarly, one can define the reference measurement and measurement deviation as

\[ z_k = h(x_k, k) \]

\[ \delta z_k = z_k - \bar{z}_k = h(x_k, k) + v_k - h(\bar{x}_k, k) \] (2.35a)

\[ \delta z_k = z_k - \bar{z}_k = h(x_k, k) + v_k - h(\bar{x}_k, k) \] (2.35b)

Once again performing a Taylor series expansion of \( h(\cdot) \) about \( \bar{x}_k \) and neglecting second order and higher terms

\[ h(x_k, k) \approx h(\bar{x}_k, k) + H_k \delta x_k \] (2.36)

where,

\[ H_k = \frac{\partial h}{\partial x} \bigg|_{x_k=\bar{x}_k} \]

Substituting in Eqn. (2.35) we get the linearized measurement model

\[ \delta z_k = H_k \delta x_k + v_k \] (2.37)

From Eqns. (2.33) and (2.37) it is seen that the deviations can be modeled as a linear discrete
stochastic dynamical system and thus the Kalman Filter from Eqns. (2.25) and (2.25) can be applied directly to optimally estimate the deviations. The optimal (upto a linearization) estimate of the state is the computed as

\[ \hat{x}(t_k) = \overline{x}(t_k) + \delta\hat{x}_k \]  

(2.38)

A filter using this approach is known as a \textit{Linearized Kalman Filter}. The choice of the reference trajectory affects the performance of the filter since if the deviation from the true state trajectory from the reference trajectory is large the significance of the neglected higher order terms in the Taylor series expansion increases \[29\]. Another approach is to re-linearize the system about the estimated trajectory at each time step. If the system is sufficiently observable, then the deviations between the true trajectory and the estimated trajectory will remain small. This is the approach used in the \textit{Extended Kalman Filter (EKF)} whose equations are summarized below:

\textbf{Prediction:}

\[ \hat{x}_{k|k-1} = \hat{x}_{k-1|k-1} + \int_{t_{k-1}}^{t_k} f(\hat{x}(t), t) dt \]  

(2.39a)

\[ \hat{P}_{k|k-1} = \Phi_k \hat{P}_{k-1|k-1} \Phi_k^T + Q_k \]  

(2.39b)

\textbf{Observation:}

\[ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (z_k - h(\hat{x}_{k|k-1})) \]  

(2.40a)

\[ \hat{P}_{k|k} = (I - K_k H_k) \hat{P}_{k|k-1} (I - K_k H_k)^T + K_k R_k K_k^T \]  

(2.40b)
2.4. Prediction, Filtering and Smoothing:

The Kalman gain is computed as

\[ K_k = \hat{P}_{k|k-1} H_k^T \left( H_k \hat{P}_{k|k-1} H_k^T + R_k \right)^{-1} \]  

(2.41)

2.4.3 Smoothing:

The problem of smoothing can be classified into three types which are defined below [53]:

- **Fixed-interval smoothing**: The problem of computing the estimate \( \hat{x}_{k|N} \), \( \forall k = 0, 1 \ldots N-1 \), where \( N \) is a fixed positive integer.

- **Fixed-point smoothing**: The problem of computing the estimate \( x_{k|j} \) where \( k \) is a fixed positive integer and \( j = k+1, k+2, \ldots \).

- **Fixed-lag smoothing**: The problem of computing the estimate \( x_{k-L|k} \), \( \forall k = 0, 1 \ldots \), where \( L \) is a fixed positive integer.

The solution to the fixed-interval smoothing problem for the discrete linear stochastic dynamic system of Eqn. (2.23) is the well-known Rauch-Tung-Stribel (RTS) smoothing algorithm [64]. The algorithm consists of two parts: a forward sweep and backward sweep. In the forward sweep a conventional filter is run and the results of the filter are stored for the backward sweep. In the backward sweep, the original filter estimate at each step is updated based on measurements from the future [10]. The recursive equations for the smoothed mean and covariance in the backward sweep are:

\[ \hat{x}_{k|N} = \hat{x}_{k|k} + C_k (\hat{x}_{k+1|N} - \Phi_{k+1} \hat{x}_{k|k}) \]  

(2.42a)

\[ P_{k|N} = P_{k|k} + C_k (P_{k+1|N} - P_{k+1|k}) C_k^T \]  

(2.42b)
where, $C_k$ is the smoother gain which is given by

$$C_k = P_{k|k} \Phi^T (k + 1, k) P_{k+1|k}^{-1}$$

(2.43)

For non-linear systems, the RTS smoother can be applied to the deviations of the state to obtain a smoothed estimate of the deviations which in turn can be then be used to get a smoothed estimate of the state.

### 2.4.4 Localization using GNSS-INS integration

GNSS and INS system have complementary characteristics. Typically the INS measurements are available at a high frequency (100-400 Hz) and are mostly used to predict the system state trajectory. GNSS measurements are usually available at a much lower frequency (1-10 Hz) and are utilized to correct/update the predicted trajectory. The bias and other random errors in INS measurements are relatively small, but they increase rapidly and are unbounded over time (Farrel 2008). Errors in GNSS systems are greater in magnitude but stable in the long-term (Lachapelle 1997). Thus, GNSS systems can be used as external aiding to bound INS errors. Due to these complementary characteristics, localization in state-of-the-art MMS is done by integrating GNSS receiver measurements and INS measurements in an optimal filtering framework. Integration of GNSS/INS measurements is typically done by two approaches:

- **Loosely coupled Integration:** In this approach, the first filter is used to estimate the GNSS receiver position and velocity ($p^e$ and $v^e$) by receiving as input the pseudo-range and delta-range (sometimes even carrier-phase) measurements. The raw measurements from the INS are processed through the Mechanization equations to estimate the pose
2.5 Road Surface Modeling

trajectory. A second filter is then used to correct the estimated INS pose trajectory by using the estimated GNSS position and velocity as measurements. This approach is also referred to as the Decentralized approach.

• **Tightly coupled:** In the tightly coupled approach, a single filter is used to process the raw INS and GNSS measurements. First the INS measurements are used to predicted the pose trajectory through the mechanization equations. Then, the pose trajectory along with the satellite ephemeris information is used to predict the GNSS measurements (pseudo range and delta range). The difference between the predicted GNSS measurements and the true GNSS measurements is used as the input to a filter that estimates the error in the pose trajectory. This approach is also referred to as the Centralized approach.

Detailed explanations of the two approaches can be found in [2].

2.5 Road Surface Modeling

Once the pose trajectory has been estimated, all the measurements from the individual mapping sensors can be transformed into a single reference frame to obtain a point cloud of data. In this work, the reference frame is the w-frame. Denoting the road surface points in the w-frame as \( \mathbf{r}^w \) the transformation of the mapping sensor measurements is given by:

\[
\mathbf{r}^w = \mathbf{C}_b^w \left( \mathbf{C}_s^b \mathbf{z}^s + \mathbf{t}_s^b \right) + \mathbf{p}^w
\] (2.44)
The set of all the road surface points corresponding to all of the mapping sensor measurements forms a point cloud which is denoted as $\mathcal{R}$.

$$
\mathcal{R} = \{r^w_i : \forall i = 1, 2, \ldots N\}
$$

Each point in the cloud is composed of its horizontal location co-ordinates, $q^i$, and the elevation at that location which is denoted as $\xi_{q^i}$. Thus,

$$
r^w_i = [q^i \xi_{q^i}]^T
$$

Point clouds are a dense representation of the road surface and sparsification can be achieved by down-sampling of the point clouds or by creating uniformly gridded road elevation maps. Let $\mathcal{G}$ denote a set of uniformly spaced grid nodes in the horizontal ($x - y$) plane of the $w$-frame. Each grid node in the set is defined as

$$
g_j = [g_x \ g_y]^T \ \forall \ g_j \in \mathcal{G}
$$

The elevation at each grid node, $\xi_g$, is modeled as a random variable that is to be estimated using the measured road surface points in the neighborhood of the grid node. A neighborhood around the $j^{th}$ grid node is defined as

$$
N_j = \{r^w_i : ||q^i - g_j|| \leq r_{th}\} \ \forall \ r^w_i \in \mathcal{R}
$$

Traditional grid node height estimation methods for road surfaces such as Digital Terrain Modeling (DTM) are used to create a regularly spaced grid representation of the road surface. The work in [41] demonstrated that the mean can be used as a statistic for grid node height estimation. Several research studies have shown that Kriging [14] as a suitable method to
estimate heights at uniformly spaced grid nodes [41, 89]. Kriging is a technique originating in the field of geo-statistics that is defined as: "multiple linear regression procedure for arriving at the best linear unbiased predictor" [42].

2.6 Simultaneous Localization and Mapping (SLAM)

SLAM is a technique that combines the individual tasks of localization and mapping into a single estimation problem. The origins of the SLAM problem can be traced back to the mid 1980’s when researcher’s in the field of autonomous robotics were endeavoring to deal with the problem of inaccuracies of sensors used for navigation and mapping. Over the decades there has been tremendous progress on this topic and excellent reviews on the history of SLAM can be found in [5, 21]. Multiple definitions for the term SLAM exist in literature but according to a recent state-of-the-art review paper [11], 'SLAM consists of the simultaneous estimation of the state of a vehicle (robot) equipped with on-board sensors, and the construction of a map of the environment perceived by the sensors’.

The current de-facto formulation of the SLAM problem is as a Maximum A Posteriori (MAP) estimation problem [11]. This formulation has its origins in the seminal work by Lu and Milios [50], followed by Guttman and Konolige [31]. Denoting $S = \{s_k : k = 1, 2, \ldots N\}$ as the discrete state trajectory of the system (robot), $M$ as the map of the environment and $Z = \{z_k : k = 1, 2, \ldots N\}$ as the set of measurements acquired from the sensors, the SLAM problem can be formulated as finding the MAP estimates $S^*$ and $M^*$ such that

$$S^*, M^* = \arg \max_{S, M} p(S, M | Z)$$

(2.49)

Using Bayes Theorem the problem can be rewritten as
\[ S^*, M^* = \arg \max_{S, M} p(\mathcal{Z} \mid S, M) \cdot p(S, M) \]  

(2.50) 

where, \( p(\mathcal{Z} \mid S, M) \) is the likelihood of the sensor measurements and \( p(S, M) \) is the prior probability over the system states and the map in the absence of the sensor measurements. Assuming that the sensor measurements \( \mathcal{Z} \) are independent (i.e., the measurement noises are uncorrelated), the problem can be factorized as

\[ S^*, M^* = \arg \max_{S, M} \prod_{k=1}^{N} p(z_k \mid S, M) \cdot p(S, M) \]  

(2.51) 

The original approaches to the SLAM problem were, however, based on non-linear filtering techniques where only the current system state estimate is retained while the previous state estimates are marginalized out. This approach can be considered as a special case of the general MAP estimation problem and is defined as

\[ \mathbf{s}_k^*, \mathcal{M}^* = \arg \max_{\mathbf{s}_k, \mathcal{M}} p(\mathbf{s}_k, \mathcal{M} \mid \mathbf{z}_{1:k}) \]  

(2.52) 

The problem in Eqn.(2.52) is sometimes referred to as the *Online SLAM* problem, while in contrast the problem in Eqn. (2.49) is referred to as the *full SLAM* problem. The earliest solutions to the *online SLAM* problem in the robotics community were developed by utilizing an Extended Kalman Filter (EKF) in which the state vector of the EKF is augmented with the locations of stationary landmarks (features) in the map \([4, 76]\). The use of the EKF was motivated by the initial work of \([22]\) and \([77]\) which established a statistical basis for describing relationships between landmarks and manipulating geometric uncertainty. Convergence properties for the landmark locations in the map using an EKF are presented in \([18]\). It was also shown in \([18]\) that the motion of the robot through the environment causes
the correlation between estimates of the landmarks to increase monotonically and hence the EKF needs to maintain estimates of all the landmarks. As the dimension of the state vector of the EKF increases linearly in the number of landmarks, the computational complexity of the update step of the EKF increases quadratically \[21\]. Use of the entails EKF linearizing the system by evaluating Jacobians at the current estimates of the system state and landmark locations, and it has been shown in \[33, 37\] that this can lead to inconsistent estimates. Potential solutions to avoid inconsistency include using an Iterated EKF (IEKF) or an Unscented Kalman Filter (UKF) \[36\] which precludes the need to evaluate Jacobians. The use on an EKF is rooted in the assumption that the system state vector is adequately described by a multi-variate Gaussian distribution.

A new approach to the online SLAM problem based on particle filtering was introduced by \[55\] in order to overcome some of the limitations of EKF based SLAM. The dimension of the search space for the SLAM problem makes direct application of particle filters computationally infeasible. However, the approach, called FastSLAM, utilizes a Rao-Blackwellized Particle Filter (RBPF) \[20\]. The central idea of an RBPF is to partition the augmented state vector as

\[
p(S, M | Z) = p(M | S, Z) \cdot p(S | Z)
\] (2.53)

The key insight here that was first provided by \[56\] is that conditioned on the robot state trajectory the estimates of the individual landmarks are conditionally independent. Thus,

\[
p(S, M) = \prod_{j=1}^{M} p(m_j | S, Z) \cdot p(S | Z)
\] (2.54)

The robot state trajectory at any time step \(k\) is represented as a weighted set of samples. The samples, \(s_{i,k}\), are drawn from a proposal distribution given by the non-linear motion model of the robot. The weights of the samples, \(w_k^{i}\), are computed according to the principle
of *Sequential Importance Sampling* [19]. For each sample, a corresponding separate estimate of the map is maintained. For the SLAM problem, the conditional probability of landmark locations given a sample of the state trajectory and sensor measurements can be analytically determined. The original *FastSLAM* formulation in [55] utilizes an EKF to update the location of each landmark in each map corresponding to the state trajectory samples. The joint posterior distribution at any time step $k$ is represented by the set of $P$ particles, $\{p^i\} \forall i = 1, 2, \ldots P$, where each particle is represented as

$$p^i = \left\{ s_{0,k}^i, w_k^i, \prod_{j=1}^{M} p(m_j | s_{0,k}^i, z_{1:k}) \right\}$$

The computational complexity of the *FastSLAM* algorithm is $O(PM)$ which can be further reduced to $O(P \cdot \log M)$ by using an efficient tree-based structure for storing the landmarks [55]. An improved version of *FastSLAM* was introduced in [54] where the proposal distribution is modified such that it also takes into account the most recent sensor measurements. This reduces the number of particles needed for the *FastSLAM* algorithm. One of the drawbacks of *FastSLAM* as shown in [6] is that "the algorithm degenerates with time, regardless of the number of particles used or the density of landmarks within the environment, and will always produce inconsistent estimates of uncertainty in the long-term."

One of the downsides of the *Online* (filtering) approach to the SLAM problem is that marginalization of past state estimates causes the landmark location estimates to become highly correlated. Propagating and updating the joint distribution of the highly correlated landmark location estimates is computationally expensive [80]. On the other hand retaining past state estimates, as in the *full-SLAM* problem causes that state space to increase linearly with time however the system remains sparse. This can be best understood by interpreting the *full-SLAM* problem is an inference problem over a factor graph [43]. Consider the factor
graph shown in Fig. (2.3). The variables to be estimated (system state and map elements) resemble nodes in the factor graph. The terms $p(z_k \mid X)$ and the prior $p(X)$ are called factors, and they provide probabilistic constraints between the nodes of the factor graph. The sparsity of the resulting SLAM problem is determined by the connectivity of the factor graph. A key insight is that for most SLAM problems the matrix appearing in the system of equations to be solved turns out to be sparse [11]. Modern SLAM solvers exploit this fact and by leveraging results from the field of sparse linear algebra, solve large scale full-SLAM problems relatively quickly [16, 38, 39]. Assuming the sensor measurement errors have a Gaussian distribution, the Maximum A Posteriori (MAP) is transformed into a sparse non-linear least squares problem. The system is then linearized and solved using sparse linear least squares solver

![Factor Graph SLAM](image)

**Figure 2.3:** Schematic of Factor Graph SLAM: Blue circles denote robot states, orange circles denote elements of the map. Black squares denote the factors: 'u' denotes factors for odometry/inertial measurements, 'z' denotes factors for mapping sensor measurements, and 'p' denotes prior factors.

### 2.6.1 Data association

Data association is the problem of determining which subset of mapping elements (landmarks) each sensor measurement corresponds to. Data association can be categorized into
two levels: Short term and long term data association [11]. Short term data association determines the correspondence between mapping elements for consecutive sensor measurements, while long term data association (also known as loop closure) determines the correspondence between new sensor measurements and previously measured mapping elements. Usually data association is done by a separate front-end module that pre-processes the sensor measurements and feeds to the correspondences to the core SLAM filter/solver.

2.7 Identifying outliers in mapping sensor measurements:

Outlier identification in point clouds is a challenging task due to geometrical discontinuities, lack of prior information regarding the statistical distribution of points, sensor measurement errors and varying local point densities[79]. The authors in [47] performed outlier identification in both the spatial and temporal domains. In the temporal domain a Moving Fixed Interval Smoother (MFIS) was employed to suppress outliers, while in the spatial domain a quadratic curved surface was fit to points in a neighborhood. Points are classified as outliers if their distance from a fitted surface is greater than a predefined value [35]. The plausibility of road surface measurements can be also checked by using knowledge of the dynamics of the profiling system [68]. The measured host vehicle position is evaluated against the predicted position obtained from the measured road surface and a mathematical model of the system, for which plausibility bounds are placed. A local neighborhood statistical analysis technique was proposed by [66] for outlier identification. This technique computes the mean distance of each point to its nearest neighbors and then forms a distribution of these mean distance over all points in the dataset. Only those points whose mean distance to the nearest neighbors is similar to the one for the rest of the points are considered as inliers.
2.7. Identifying outliers in mapping sensor measurements:

2.7.1 Random Sampling and Consensus (RANSAC):

Random Sampling and Consensus (RANSAC), introduced in 1981, presented a new paradigm of estimating model parameters in order to fit models to experimental data that contains a significant percentage of outliers \cite{26}. One of the contributions in this work builds on the concepts of RANSAC, making it applicable for road surface point clouds obtained using laser scanners. A common technique for smoothing data with outliers is successive least squares approximation wherein for each iteration of the least squares fit, the points with the largest errors are eliminated. However, it was shown in \cite{26} that even a single point with a large error can render this method ineffective. Some other techniques for robust estimation include the M-estimator, L-estimator, and Least Median of Squares. The difficulty with such estimators is the complex non-linear cost functions that they employ, which in turn require computationally intensive numerical optimization algorithms. This computational burden is alleviated in the Random Sampling and Consensus (RANSAC) algorithm in which surfaces are repeatedly fitted to data points until an acceptably small number of outliers is reached. Specifically, each candidate surface model is identified by a random selection of the minimum number of points needed to estimate the model parameters. Consensus on which surface is best is reached by evaluating a cost function applied to each candidate surface that is based on the distance from the points to the candidate surface. Specifically, this hypothesize-and-verify loop is repeated until the probability of finding a model with better consensus than the current best model falls below a predefined threshold (typically 1% - 5%) \cite{63}. Variants of the RANSAC algorithm have been proposed in which the methods by which samples are selected and means by which the consensus is calculated are modified \cite{81, 86, 87}. At its core, the RANSAC algorithm and its variants solve a selection optimization problem that is formulated as:
\[ \hat{m} = \arg\min_{m \in M} \left[ \sum_{p \in P} C(e(p; m)) \right] \]  

where \( P \) is the dataset, \( C \) is a cost function, \( M \) is the set of models and \( e \) represents an error function which characterizes the error of a certain data point \( p \) based on its relation with respect to the estimated model \( m \) [13]. Typically, this error function is the normal distance of the point, \( p \), from the surface, \( m \). Once the best candidate surface has been selected, points are classified as either inliers or outliers under the hypothesis that the single best surface is the true surface representing the dataset. Thus, based on the distance of a point, from the best surface, it is classified, in a binary manner, as either an inlier or outlier. This can lead to incorrect classification when there are multiple candidate surfaces representing different regions of the dataset. In such cases there is critical information in knowing which points are consistently close to high-consensus surfaces and which points are consistently distant. In addition, the results of classification are highly dependent on the appropriate selection of the threshold distance for classification. In this work, an iterative soft-switching (logistic) classification method is proposed based on a logistic function of the distance of each point from each candidate surface.
Chapter 3

Filtering based solution to the SLAM problem in mobile road mapping

3.1 Introduction:

The accuracy of the road surface map generated by a MRMS depends on the accuracy of the individual localization and mapping solutions. Typically the localization and mapping problems are solved independently. Greater accuracy is possible by solving the localization and mapping problems simultaneously as a single estimation problem. Over the past two decades there has been tremendous progress in the area of Simultaneous Localization and Mapping (SLAM) in the field of robotics [11, 21]. This work investigates the application of the principles of SLAM specifically to the problem of mobile road mapping.

The main objective of this chapter is to formulate the mobile road mapping problem as a SLAM problem. In addition a filtering based solution to the SLAM problem is developed using an Extended Kalman Filter (EKF). The road surface is parameterized by a set of regularly spaced grid nodes, where the elevation at each grid node is modeled as a random variable that is to be estimated. In much of the existing literature on SLAM, the models for the sensor measurements are solely based on kinematic spatial relationships. However, in the case of a MRMS the road surface is one of the primary sources of excitation to the dynamics of the host vehicle of the mapping system. The natural question thereby arises:
Chapter 3. Filtering based solution to the SLAM problem in mobile road mapping

Can knowledge of the dynamics of the host vehicle be used in conjunction with the estimated road surface map to improve the accuracy of the localization and mapping solution? Hence, the SLAM framework developed in this work also incorporates the dynamics of the host vehicle of the mapping system.

The remainder of this chapter is organized as follows: modeling of the mapping system and the road surface are described in sections 3.2 and 3.3 respectively. Section 3.4 presents the formulation of mobile road mapping problem as an online SLAM problem. Models for the localization sensor measurements, mapping sensor measurements and dynamic prediction model are developed in sections, 3.5, 3.6 and 3.7 respectively. The EKF based solution to the SLAM problem using the developed models is described in section 3.8. A description of the simulation setup and the results from the simulation of a proof-of-concept MRMS are presented in sections 3.9 and 3.10 respectively. Finally, a discussion on the simulation results along with conclusions and recommendations for future work are given in section 3.11.

3.2 Modeling the Mapping system

In general, a MRMS of a sensor platform rigidly mounted to the sprung mass of host vehicle, that houses the localization and mapping sensors. For the case of a simulated MRMS, the co-ordinate frames used to describe the variables of interest are:

1. World Frame \((w\text{-frame})\)

The \((w\text{-frame})\) is a stationary inertial frame whose origin is located at a known fixed point. The X and Y axes are orthogonal to each other and define the horizontal flat plane of the road surface (the effect of the curvature of the earth’s surface is not modeled). The Z-axis is orthogonal to the X-Y plane, parallel to the gravity vector and points vertically upwards. For the purposes of the simulation based testing in this
3.2. Modeling the Mapping system

work the effects of the rotation and revolution of the earth are neglected thus enabling the w-frame to be considered effectively as an inertial frame.

2. **Body Frame (b-frame)** The body frame is a non-stationary, non-inertial frame that moves along with the sprung mass of the host vehicle of the mapping system. Its origin coincides with the Centre of Mass (C.M) of the sprung mass. Its X-axis points along the lateral axis of the vehicle, Y axis along the longitudinal axis and Z axis completes the right handed orthogonal triad in a left-aft-up manner.

The degrees of freedom considered for the dynamic model of the host vehicle are:

- Translational motion of the sprung mass.
- Rotational motion of the sprung mass
- Vertical motion of each unsprung mass.

The variables chosen to describe the state of the dynamic model are:

- Position vector of sprung mass C.M in the *w-frame* denoted as \( \mathbf{p}^w \)
- Velocity vector of sprung mass C.M in the *w-frame* denoted as \( \mathbf{v}^w \)
- Euler angles parameterizing the orientation of the *b-frame* w.r.t *w-frame*, which are denoted by the vector \( \boldsymbol{\phi}_b^w \)
- Angular velocity vector of sprung mass w.r.t *w-frame* with components resolved in the *b-frame* and denoted as \( \boldsymbol{\omega}_{wb} \)
- Vertical displacements and velocities of each of the unsprung masses in the *w-frame* - \( \mathbf{z}_u^w \) and \( \dot{\mathbf{z}}_u^w \)
Thus, the state vector of the system is given by

\[ s = \{ p^w, v^w, \phi^w_b, \omega^w_{ub}, z^w_u, \ddot{z}^w_u \} \]  \hspace{1cm} (3.1)

### 3.3 Modeling the road surface:

In this work the road surface is modeled as an elevation map over a set of regularly spaced grid nodes in the horizontal plane of the \( w \)-frame denoted as \( G \). The horizontal location of the \( i \)th grid node is denoted as \( g_i \). The elevation at each grid node is modeled as a random variable that is to be estimated. The elevation at the \( i \)th grid node is denoted as \( \xi_{g_i} \). The elevations of any subset of grid nodes, \( S \subseteq G \), is denoted by the vector \( \xi_S \). The neighborhood of a grid node, \( g_i \), is defined as the set of grid nodes that lie within a certain threshold horizontal distance, \( d_{th} \), from it. Denoting the neighborhood of the \( i \)th grid node as \( N_{g_i} \),

\[ N_{g_i} = \{ g_j : ||g_j - g_i|| \leq d_{th} \} \quad \forall \ g_j \in G \]  \hspace{1cm} (3.2)

In a similar manner, the neighborhood of any arbitrary point \( q \in \mathbb{R}^2 \) in the horizontal plane is

\[ N_q = \{ g_j : ||q - g_j|| \leq d_{th} \} \quad \forall \ g_j \in G \]  \hspace{1cm} (3.3)

The elevation at given horizontal location, \( q \), is estimated using a weighted average interpolation of the grid nodes in its neighborhood as

\[ \xi_q = \sum_i \lambda_i \xi_{g_i} \quad \forall \ g_i \in N_q \]  \hspace{1cm} (3.4)
3.4 SLAM problem formulation:

The weights \( \lambda_i \) are computed based on the horizontal euclidean distance of each grid node \( g_i \) from \( q \). A monotonically decreasing logistic function is chosen to compute the weights as

\[
\lambda_i = \frac{\tilde{d}^2}{d^2 + d_i^2}
\]  

(3.5)

The weights are normalized so that they sum to unity. The parameter \( \tilde{d} \) is a measure of the horizontal distance over which grid node elevations influence the elevation of the measured point, and can be set heuristically based on prior knowledge of the roughness of the road surface. The value of \( \tilde{d} \) is the distance at which a grid node has an non-normalized influence weight of \( \frac{1}{2} \).

3.4 SLAM problem formulation:

The objective of the work in this chapter is to simultaneously estimate of the state of the mapping system and the elevations of the road grid nodes by utilizing the localization & mapping sensor measurements along with the dynamic model of the system. Mathematically this can be formulated as the following online SLAM problem for Mobile Road Mapping Systems (MRMS)

\[
\argmax_{s_{1:k}, \xi_G} p(s_{1:k}, \xi_G | l_{1:k}, m_{1:k})
\]  

(3.6)

where, \( l \) and \( m \) denote the localization and mapping sensor measurements respectively, and \( k \) denotes the discrete time step index. In this chapter an Extended Kalman Filter (EKF) is used to solve the online SLAM problem in Eqn.(3.6). For this the system state vector is augmented with the grid node elevations. The augmented state vector at time step \( k \), denoted as \( x_k \), is

\[
x_k = \left\{ s_k, \xi_G \right\}
\]  

(3.7)
The EKF maintains a Gaussian posterior over the augmented state vector which is parameterized by the conditional mean vector, $\hat{x}_k$, and covariance matrix, $\hat{P}_k$. One of the limitations associated with using an EKF for SLAM is that the computational complexity increases quadratically with the size of the state vector. In the case of the mobile road mapping SLAM problem the size of the state vector depends on the number of road grid nodes. When mapping large areas using high resolution grids the number of grid nodes can easily be on the order of hundreds of millions, thus making the use of an EKF impractical in such cases. In order to overcome this issue only a subset of the grid nodes that lie within a bounded neighborhood around the host vehicle are maintained in the EKF state vector. Grid nodes that fall outside of this neighborhood are marginalized out from the state EKF state vector, thus bounding the computational complexity. The trade-off is that this results in a sub-optimal estimate for the elevations of the grid nodes that are marginalized out.

In addition in the event that the MRMS returns to a previously mapped area of the road surface (closes the loop) information from the loop closure event cannot be used to improve the accuracy of the estimates since the correlation between the current system state and previously mapped grid nodes is not maintained. Hence, the approach used in this chapter is to optimally estimate only the system state trajectory over time using the grid nodes within the bounding neighborhood as part of the EKF state vector. Next the mean value of the optimal system state trajectory is used to transform the mapping sensor measurements to the $w$-frame and thus a point cloud of measured road surface points is obtained. The point cloud is then used to estimate the elevation at each of the grid nodes. The procedure of maintaining only those grid nodes that are within the bounded neighborhood in the EKF state vector is described in more detail in Section 3.8.

Let $\mathcal{V}_k \subset \mathcal{G}$ denote the set of grid nodes within the bounding neighborhood around the host
vehicle at time step $k$. The augmented state vector is then given by

$$x_k = \begin{pmatrix} s_k \\ \xi_{vk} \end{pmatrix}$$

(3.8)

### 3.5 Localization Sensor Models:

Consider the localization system of the MRMS as a combination of an Inertial Navigation System (INS) and a Global Navigation Satellite System (GNSS) receiver. The accelerometers of the INS measure the specific force on the sprung mass with components resolved in the $b$-frame and the measurement model is

$$a_k = C_{wb}^b[k] \left( M^{-1} \cdot f_{sk}^w + g_{wk}^w \right) + b_{ak} + \nu_{ak}$$

(3.9)

where,

$C_{wb}^b$ – Direction cosine matrix from $w$-frame to $b$-frame

$f_{sk}^w$ – Vector sum of the individual forces acting on the sprung mass in $w$-frame.

$M$ – Mass matrix

$g_{wk}^w$ – Acceleration due to gravity vector in $w$-frame

$b_{ak}$ – Bias error in the accelerometer measurements.

$\nu_{ak}$ – Noise in the accelerometer measurements modeled as a zero mean white Gaussian sequence

The gyroscopes measure the angular velocity of the sprung mass w.r.t the inertial $w$-frame with the components resolved in the $b$-frame.

$$\omega_k = \omega_{wbk}^b + b_{\omega k} + \nu_{\omega k}$$

(3.10)

where,
Bias error in the gyroscope measurement  

\( b_{\omega k} \) — Bias error in the gyroscope measurement  

\( \nu_{\omega k} \) — Noise in the gyroscope measurements modeled as a zero mean white Gaussian sequence  

As is common in inertial navigation literature, the bias errors in the accelerometer and gyroscopes measurements are estimated as part of the system state, by defining a stochastic process to model the time evolution of the bias errors. Some of the stochastic processes that are typically used to model bias errors are Random Walk processes, 1st order Gauss-Markov processes and Auto-regressive processes \[62\]. The system state vector is augmented with the accelerometer and gyroscope biases

\[
\mathbf{s} \leftarrow \left\{ \mathbf{s} \ b_a \ b_w \right\} 
\]  

(3.11)

The GNSS receiver is rigidly attached to the sprung mass and it measures the pseudo range to the visible satellites via the code-phase and carrier phase measurements. For the simulation based testing in this work, stationary satellites have been simulated at fixed known locations in the \( \mathbf{w-frame} \). The position and velocity vector of the \( j^{th} \) satellite in the \( \mathbf{w-frame} \) are denoted as \( \mathbf{P}_j^w \) and \( \mathbf{V}_j^w \) respectively. The position vector of the GNSS receiver antenna in the \( \mathbf{w-frame} \) is given by

\[
\mathbf{p}_g^w = \mathbf{p}^w + \mathbf{C}_b^w \mathbf{t}_g^b 
\]  

(3.12)

where,

\( \mathbf{t}_g^b \) — Lever arm offset from the sprung mass C.M to the GNSS antenna
3.5. Localization Sensor Models:

The velocity vector, $\mathbf{v}_g^w$ of the GNSS receiver antenna in the $w$-frame is given by

$$\mathbf{v}_g^w = \mathbf{v}^w + \mathbf{t}_g^b \times \mathbf{\omega}_{wb}^b$$

(3.13)

The measurement models in section 2.3.3 describe the GNSS measurements as non-linear functions of the receiver antenna position and velocity. Using those models and Eqns. (3.12) and (3.13), the GNSS measurements can be expressed as a non-linear function of the system state. The GNSS receiver clock bias and bias rate are modeled as stochastic processes and estimated as part of the system state. The measurements from all the localization sensors are combined into a single vector denoted as $\mathbf{l}$, and the noise terms are combined into a single vector denoted as $\mathbf{v}_l$. At each discrete time step $k$, the localization sensor measurements can then be summarized through the following non-linear measurement model

$$\mathbf{l}_k = g(\mathbf{x}_k) + \mathbf{v}_{lk}$$

(3.14)

where \{\mathbf{v}_{lk}\} is a vector zero mean white Gaussian sequence with $\mathbf{v}_{lk} \sim \mathcal{N}(\mathbf{0}, R_l)$. It is assumed that the covariance $R_l$ is time invariant and can be determined from the localization sensor specifications. Given a nominal value of the augmented state, $\bar{\mathbf{x}}_k$, the nominal localization measurements are given by

$$\bar{\mathbf{l}}_k = g(\bar{\mathbf{x}}_k)$$

(3.15)

The deviation in the localization measurements from this nominal value is defined as

$$\delta \mathbf{l}_k = \mathbf{l}_k - \bar{\mathbf{l}}_k = g(\mathbf{x}_k) + \mathbf{v}_{lk} - g(\bar{\mathbf{x}}_k)$$

(3.16)

Performing a 1st order Taylor series expansion of the function $g(\cdot)$ about the nominal value
where,
\[ G_k = \left. \frac{\delta g}{\delta x} \right|_{x_k = x_k} \]

Substituting Eqn.(3.17) in Eqn.(3.16)

\[ \delta l_k \approx G_k \delta x_k + \nu_{lk} \]  

### 3.6 Mapping sensor measurement model

The mapping sensors in a MRMS typically measure the range (distance) from the sensor to the road surface. The range measurement is defined as the straight line distance from the origin of the sensor to the corresponding road surface point. Let \( m^w_0 \) denote the origin of the mapping sensor which can be computed as

\[ m^w_0 = p^w + C_b t^b_s \]  

where, \( t^b_m \) denotes the lever arm offset from sprung mass C.M to the mapping sensor origin.

The range measurement, \( m \), is given by

\[ m = \sqrt{(r^w - m^w_0)^T(r^w - m^w_0)} \]

where, \( r^w \) denotes the corresponding road surface point. Each measured road surface point, \( r^w \), is composed of its horizontal location, denoted as \( q \) and the elevation at that location,
3.6. Mapping sensor measurement model

denoted as $\xi_q$.

$$r^w = \{q, \xi_q\}$$  \hspace{1cm} (3.21)

Combining Eqns.(3.20) and (3.21) the range measurement at time step, $k$, can be written as

$$m_k = h_m(s_k, q_k, \xi_{q_k}) + \epsilon_{mk}$$  \hspace{1cm} (3.22)

In this work the errors in the range measurement are modeled as a zero mean white Gaussian sequence with

$$\epsilon_{mk} \sim \mathcal{N}(0, \sigma^2_{mk})$$  \hspace{1cm} (3.23)

where, $\sigma^2_{mk}$ is the variance in the errors which can be determined from sensor specifications. It is assumed that outliers from the mapping sensors have been identified and eliminated. A potential method to eliminate outliers from mapping sensor range measurements is developed in Chapter 5.

Given the system state, the horizontal location of the road surface that a mapping sensor is measuring can be estimated by performing a co-ordinate transformation to the $w$-frame with a reference mapping sensor measurement. Thus,

$$q_k = h_q(s_k) + \epsilon_{qk}$$  \hspace{1cm} (3.24)

where, $h_q(\cdot)$ is a non-linear function representing the co-ordinate transformation. The random error in the estimate, $\epsilon_{qk}$, is modeled as additive white Gaussian noise with $\epsilon_{qk} \sim \mathcal{N}(0, \Sigma_{qk})$. Given the elevations at neighboring grid nodes, the elevation at the measured horizontal location can be estimated using the weighted average interpolation model from
section 3.3. Thus, the elevation estimation model at time step $k$ can be written as

$$ \xi_{q_k} = h_{\xi}(q_k, \xi_{Nq_k}) + \epsilon_{\xi_k} $$

where, $h_{\xi}(\cdot)$ is a function representing the weighted average interpolation. The error in the interpolated elevation, denoted as $\epsilon_{\xi_k}$, is assumed to be additive and modeled as a zero mean white Gaussian sequence with $\epsilon_{\xi_k} \sim \mathcal{N}(0, \sigma^2_{\xi_k})$.

### 3.6.1 Mapping sensor deviation model:

In this subsection a model of the mapping sensor measurement deviation conditioned on the current system state deviation and the deviation in the elevations of neighboring grid nodes is developed. The deviations are defined with respect to nominal values as:

$$ \delta s_k = s_k - \bar{s}_k $$

$$ \delta \xi_{Nq_k} = \xi_{Nq_k} - \bar{\xi}_{Nq_k} $$

where, $\bar{s}_k$ and $\bar{\xi}_{Nq_k}$ are nominal values of the system state and neighboring grid node elevations respectively at time step $k$. The deviation model is developed by linearizing the measurement model equations. Given a nominal value for the system state, a nominal value for the horizontal location of the measured road surface point is given by

$$ q_k = h_q(s_k) $$

and the corresponding deviation is defined as

$$ \delta q_k = q_k - \bar{q}_k = h_q(s_k) + \epsilon_{q_k} - h_q(s_k) $$
3.6. Mapping sensor measurement model

Performing a 1st order Taylor series expansion of the function \( h_q(\cdot) \) about the nominal value \( \bar{s}_k \).

\[
\begin{align*}
    h_q(s_k) - h_q(\bar{s}_k) &\approx H_{qsk}\delta s_k \\
\end{align*}
\]

where,

\[
H_{qsk} = \frac{\partial h_q}{\partial s_k} \bigg|_{s_k = \bar{s}_k}
\]

Substituting Eqn.(3.29) in Eqn.(3.28)

\[
\begin{align*}
    \delta q_k &\approx H_{qsk}\delta s_k + \epsilon_q \tag{3.30}
\end{align*}
\]

Given the nominal value of the horizontal location of the measured road surface point, \( \bar{q}_k \), and nominal values of the neighboring grid node elevations, \( \bar{\xi}_{N_qk} \), the nominal value of the interpolated elevation is given by

\[
\begin{align*}
    \bar{\xi}_{qk} &= h_\xi(q_k; \bar{\xi}_{N_qk}) \\
\end{align*}
\]

and the corresponding deviation from the nominal value is defined as

\[
\begin{align*}
    \delta \xi_{qk} &= \xi_{qk} - \bar{\xi}_{qk} = h_\xi(q_k; \xi_{N_qk}) + \epsilon_\xi - h_\xi(q_k; \bar{\xi}_{N_qk}) \\
\end{align*}
\]

Performing a 1st order Taylor series expansion of the function \( h_\xi(\cdot) \) about the nominal values \( \bar{q}_k \) and \( \bar{\xi}_{N_qk} \)

\[
\begin{align*}
    h_\xi(q_k; \xi_{N_qk}) - h_\xi(\bar{q}_k; \bar{\xi}_{N_qk}) &\approx H_{\xi qk}\delta q_k + H_{\xi \xi qk}\delta \xi_{N_qk} \\
\end{align*}
\]

where,

\[
\begin{align*}
    H_{\xi qk} &= \frac{\partial h_\xi}{\partial q_k} \bigg|_{q_k = \bar{q}_k, \xi_{N_qk} = \bar{\xi}_{N_qk}} \\
    H_{\xi \xi qk} &= \frac{\partial h_\xi}{\partial \xi_{N_qk}} \bigg|_{q_k = \bar{q}_k, \xi_{N_qk} = \bar{\xi}_{N_qk}}
\end{align*}
\]
Substituting Eqn.(3.33) in Eqn.(3.32)

\[
\delta \xi_{\mathbf{q}_k} \approx H_{\xi q_k} \delta \mathbf{q}_k + H_{\xi \xi q_k} \delta \xi_{\mathbf{q}_k} + \epsilon_{\xi q_k} \tag{3.34}
\]

Substituting the nominal values \( \bar{s}_k \), \( \bar{q}_k \) and \( \bar{\xi}_{\mathbf{q}_k} \) in the mapping sensor measurement model the nominal value of the mapping sensor measurement is given by

\[
\bar{m}_k = h_m(\bar{s}_k, \bar{q}_k, \bar{\xi}_{\mathbf{q}_k}) \tag{3.35}
\]

The deviation in the mapping sensor measurement is defined as

\[
\delta m_k = m_k - \bar{m}_k = h_m(\mathbf{s}_k, \mathbf{q}_k, \xi_{\mathbf{q}_k}) + \epsilon_{m_k} - h_m(\bar{s}_k, \bar{q}_k, \bar{\xi}_{\mathbf{q}_k}) \tag{3.36}
\]

Performing a 1st order Taylor series expansion of the function \( h_m(\cdot) \) about the nominal values of its arguments

\[
h_m(\mathbf{s}_k, \mathbf{q}_k, \xi_{\mathbf{q}_k}) - h_m(\bar{s}_k, \bar{q}_k, \bar{\xi}_{\mathbf{q}_k}) \approx H_{msk} \delta \mathbf{s} + H_{mqk} \delta \mathbf{q}_k + H_{m\xi q_k} \delta \xi_{\mathbf{q}_k} \tag{3.37}
\]

where,

\[
H_{msk} = \left. \frac{\partial h_m}{\partial \mathbf{s}_k} \right|_{\mathbf{s}_k = \bar{s}_k, \mathbf{q}_k = \bar{q}_k, \xi_{\mathbf{q}_k} = \bar{\xi}_{\mathbf{q}_k}}
\]

\[
H_{mqk} = \left. \frac{\partial h_m}{\partial \mathbf{q}_k} \right|_{\mathbf{s}_k = \bar{s}_k, \mathbf{q}_k = \bar{q}_k, \xi_{\mathbf{q}_k} = \bar{\xi}_{\mathbf{q}_k}}
\]

\[
H_{m\xi q_k} = \left. \frac{\partial h_m}{\partial \xi_{\mathbf{q}_k}} \right|_{\mathbf{s}_k = \bar{s}_k, \mathbf{q}_k = \bar{q}_k, \xi_{\mathbf{q}_k} = \bar{\xi}_{\mathbf{q}_k}}
\]
3.6. Mapping sensor measurement model

Substituting Eqn\((3.37)\) in Eqn\((3.36)\)

\[
\delta m_k \approx H_{msk}\delta s_k + H_{mqk}\delta q_k + H_{m\xi k}\delta \xi q_k + \epsilon_{mk}
\]  

\(3.38\)

Substituting Eqns.\((3.30)\) and \((3.34)\) in Eqn.\((3.38)\)

\[
\delta m_k \approx (H_{msk} + H_{mqk}H_{qsk} + H_{m\xi k}H_{\xi qk}H_{qsk})\delta s_k + H_{m\xi k}H_{\xi qk}\delta \xi q_k + \nu_{mk}
\]  

\(3.39\)

The cumulative error term, \(\nu_{mk}\), is a linear combination of the errors in the range sensor measurement, the horizontal location estimate and the elevation interpolation

\[
\nu_{mk} = \epsilon_{mk} + H_{m\xi k} \cdot H_{\xi qk}\epsilon_{\xi k} + (H_{m\xi k} \cdot H_{\xi qk} + H_{mqk})\epsilon_{qk}
\]  

\(3.40\)

and thus \(\{\nu_{mk}\}\) can also be modeled as a white Gaussian sequence with \(\nu_{mk} \sim \mathcal{N}(0, \tilde{\sigma}_{mk}^2)\).

The conditional probability model for the mapping sensor range measurement deviation can be written as

\[
p(\delta m_k \mid \delta s_k, \delta \xi q_k) \approx \mathcal{N}(H_{sk}\delta s_k + H_{\xi qk}\delta \xi q_k, \tilde{\sigma}_{mk}^2)
\]  

\(3.41\)

The range measurements from all the mapping sensors are combined into a single vector denoted as, \(\mathbf{m}_k\). Based on Eqns. \((3.22)\), \((3.24)\) and \((3.25)\) the nominal value of the range measurements can be expressed as a non-linear function of the nominal value of the augmented state vector as

\[
\tilde{\mathbf{m}}_k = h(\tilde{x}_k)
\]  

\(3.42\)

The deviations of all the mapping sensor measurements are combined into a single vector denoted as, \(\delta \mathbf{m}_k\), and the individual terms are combined into a single error vector denoted as \(\mathbf{v}_{mk}\). Based on Eqn.\((3.39)\), the mapping sensor deviation vector can be approximated as
a linear function of the augmented state vector deviation as

$$\delta m_k \approx H_k \delta x_k + \nu_{mk} \quad (3.43)$$

where, $H_k$ is the assembled Jacobian matrix and $\{\nu_{mk}\}$ is a zero mean white Gaussian vector sequence with $\nu_{mk} \sim \mathcal{N}(0, R_{mk})$

### 3.7 Dynamic Prediction Model:

In this work a multi-body dynamic model of the host vehicle of the MRMS is used to predict the trajectory of the system state in between the availability of sensor measurements. The forcing inputs to a vehicle dynamics model are the tire-road interaction forces and other external the forces acting on the vehicle. Estimates of the vertical forces at each tire can be obtained by utilizing the estimated grid node elevations along with a suitable tire force model. This is of course based on the assumption that the mapping sensors measure the road surface ahead of the tire contact patch locations. The simplest of these models is a point follower spring tire model in which the tire vertical force is computed as

$$f_{tire}(t) = k_{tire} \cdot (z_w(t) - \bar{z}(t)) \quad (3.44)$$

where,

- $z_w$ – Elevation of the unsprung mass
- $k_{tire}$ – Vertical spring stiffness of the tire
- $\bar{z}(t)$ – Average of the elevations of the grid nodes within the tire contact patch.

Models for estimating the lateral and longitudinal tire forces is more complex and requires knowledge of the micro-texture properties of the road surface. In a similar manner estimating
other external force inputs to the dynamic model necessitates instrumenting the system with additional force sensors. However, the approach taken in this work is not to assume that additional force sensors are available. Rather, the more general approach taken in this work is to model the unknown lateral and longitudinal tire forces and other external force inputs as stochastic processes and estimate their values using measurements from the existing localization and mapping sensors. This approach is based in principle on existing methods used to estimate the biases in the IMU measurements. The stochastic process model used in this work is a 1st order Gauss-Markov process due to its mathematical simplicity and proven ability to model a large variety of physical processes [10]. The continuous time representation for a 1st order Gauss-Markov process is:

\[ \dot{\zeta}_i(t) = -\frac{1}{\tau} \zeta_i(t) + w_i(t) \] (3.45)

where,

- \( \zeta_i \) - \( i^{th} \) unknown force input
- \( \tau \) - Correlation time (secs)
- \( w_i \) - Driving zero-mean white noise process

The stochastic forcing inputs are collectively represented by the vector \( \zeta(t) \) where

\[ \zeta(t) = \{\zeta_i(t)\} \quad \forall i = 1, 2, \ldots, n_\zeta \] (3.46)

where, \( n_\zeta \) is the total number of unknown force inputs to the dynamic model. The driving zero mean white noise processes are combined into a single vector white noise process denoted as \( w_\zeta(t) \),

\[ w_\zeta(t) = \{w_i(t)\} \quad \forall i = 1, 2, \ldots, n_\zeta \] (3.47)
with
\[
\mathbb{E}[w(t)w(\tau)^T] = Q(t)\delta(t - \tau)
\] (3.48)

where \(Q(t)\) is the spectral density matrix. The stochastic force inputs are appended to the system state vector as
\[
s(t) \leftarrow \begin{pmatrix} s(t) & \zeta(t) \end{pmatrix}
\] (3.49)

Combining the equations of motion of the dynamic model along with the vertical tire force model and the stochastic process models, a non-linear continuous state space model for the MRMS is
\[
\dot{s}(t) = f_s(s(t), \zeta_{\mathcal{C}(t)}) + B_s w(t)
\] (3.50)

where, \(B_s \in \mathbb{R}^{n_s \times n_u}\) is the matrix mapping from the process noise to the rate of change of the system state and \(\mathcal{C}(t)\) is the subset of grid nodes within the tire contact patches at time \(t\). The detailed derivation of the equations of motion of the dynamic model used for the simulation based proof-of-concept MRMS used in this work are given in Appendix A. The deviation model for dynamic prediction is developed by linearizing the dynamic equations of motion and will be utilized in the prediction step of the EKF. Let \(\bar{s}(t)\) and \(\bar{\xi}_{\mathcal{N}_C(t)}\) are nominal values of the system state and tire contact patch grid node elevations respectively at time \(t\). Then the deviations with respect to the nominal values are defined as:

\[
\delta s(t) = s(t) - \bar{s}(t)
\] (3.51a)

\[
\delta \xi_{\mathcal{N}_C(t)} = \xi_{\mathcal{N}_C(t)} - \bar{\xi}_{\mathcal{N}_C(t)}
\] (3.51b)

Taking the derivative of Eqn.(3.51a) with respect to time
\[
\dot{\delta s}(t) = f_s \left( s(t), \xi_{\mathcal{N}_C(t)} \right) + B_s \omega(t) - f_s \left( \bar{s}(t), \bar{\xi}_{\mathcal{N}_C(t)} \right)
\] (3.52)
Performing a 1st order Taylor series expansion of the function \( f_s(\cdot) \) about the nominal values \( \bar{s}(t) \) and \( \bar{\xi}_{N_C(t)} \).

\[
f_s \left( s(t), \xi_{N_C(t)} \right) - f_s \left( \bar{s}(t), \bar{\xi}_{N_C(t)} \right) \approx F_s(t) \delta s(t) + F_\xi(t) \delta \xi_{N_C(t)} \tag{3.53}
\]

where,

\[
F_s(t) = \frac{\partial f_s}{\partial s(t)} \bigg|_{s(t)=\bar{s}(t), \xi_{N_C(t)}=\bar{\xi}_{N_C(t)}} \quad F_\xi(t) = \frac{\partial f_s}{\partial \xi_{N_C(t)}} \bigg|_{s(t)=\bar{s}(t), \xi_{N_C(t)}=\bar{\xi}_{N_C(t)}}
\]

Thus, the rate of change of the system state deviation is given by

\[
\delta \dot{s}(t) = F_s(t) \delta s(t) + F_\xi(t) \delta \xi_{N_C(t)} + B_s w_s(t) \tag{3.54}
\]

According to [34](Sec 8.3) the equivalent discrete model for the state deviation model in Eqn.(3.54) is given by

\[
\delta s_k = \Phi_{s_k} \delta s_{k-1} + \Phi_{\xi_k} \delta \xi_{N_C_{k-1}} + w_k \tag{3.55}
\]

where, \( \Phi_{s_k} \) is the state transition matrix, \( \Phi_{\xi_k} \) is the transition matrix mapping from grid elevation deviations to system state deviation and \( \{w_k\} \) is a zero-mean white Gaussian sequence with \( w_k \sim \mathcal{N}(0, Q_k) \), where \( Q_k \) is the discrete process noise covariance matrix. In this work, the transition matrices and the discrete process noise covariance matrix are computed using the numerical integration technique described in [10](Chap 3) which is based on the original work in [88]. It is assumed that the deviation model parameters and the process noise remain constant during the discrete time intervals. The road grid node elevations are invariant with respect to time and hence the prediction model for the grid node elevations is

\[
\dot{\xi}_v = 0 \tag{3.56}
\]
This also implies that the rate of change in deviation of the elevations from their nominal values is also zero and thus

\[
\delta \xi_{\nu_k} = \delta \xi_{\nu_{k-1}} \tag{3.57}
\]

Combining together the dynamic prediction model from Eqn. (3.50) and the grid node elevation prediction model, the non-linear function to predict the nominal trajectory of the augmented state vector is

\[
\dot{\bar{x}}(t) = f(\bar{x}(t)) \tag{3.58}
\]

Assembling together the dynamic prediction deviation model from Eqn. (3.55) and the deviation model for grid node elevations from Eqn. (3.57), the deviation model for the augmented state vector can be written as

\[
\delta \bar{x}_k = \Phi_k \delta \bar{x}_{k-1} + \Gamma_k w_k \tag{3.59}
\]

where, \( \Phi_k \) denotes the augmented state transition matrix and \( \Gamma_k \) the mapping from the discrete process noise to the augmented state vector.

### 3.8 Extended Kalman Filter (EKF) solution:

Recall from section 2.4 the equations for computing the predicted mean and covariance for the EKF are given by

\[
\hat{x}_{k|k-1} = \hat{x}_{k-1|k-1} + \int_{t_{k-1}}^{t_k} f(\hat{x}(t))dt
\]
\[
\hat{P}_{k|k-1} = \Phi_k \hat{P}_{k-1|k-1} \Phi_k^T + \Gamma_k Q_k \Gamma_k^T
\]
3.8. Extended Kalman Filter (EKF) solution:

The equations for the correction step of the EKF are given by

\[ \dot{x}_{k|k} = \dot{x}_{k|k-1} + K_k \left( \begin{bmatrix} l_k \\ m_k \end{bmatrix} - \begin{bmatrix} g(\dot{x}_{k|k-1}) \\ h(\dot{x}_{k|k-1}) \end{bmatrix} \right) \]  

\[ \hat{P}_{k|k} = (I - K_k J_k) \hat{P}_{k|k-1} (I - K_k J_k)^T + K_k R_k K_k^T \]  

(3.61a)

(3.61b)

where,

\[ J_k = \begin{bmatrix} G_k \\ H_k \end{bmatrix} \]

\[ R_k = \begin{bmatrix} R_{lk} & 0 \\ 0 & R_{mk} \end{bmatrix} \]

The Kalman gain for the correction step, \( K_k \) is computed as:

\[ K_k = \hat{P}_{k|k-1} J_k^T \left( J_k \hat{P}_{k|k-1} J_k^T + R_k \right)^{-1} \]  

(3.62)

3.8.1 Adding and removing grid nodes:

For notational convenience, at a given time step \( k \), the following sets of grid nodes are defined:

\[ \mathcal{U}_k \subset \mathcal{G} \] — Set of grid nodes within the bounding neighborhood around the host vehicle

\[ \mathcal{V}_k \subset \mathcal{G} \] — Set of grid nodes currently in the augmented state vector.

\[ \mathcal{W}_k \subset \mathcal{G} \] — Set of grid nodes currently in the scanning area of the mapping sensors.

The set \( \mathcal{W}_k \) is the union of the neighborhoods at the mapping sensor measurement locations.

\[ \mathcal{W}_k = \bigcup_j \mathbb{N}_{q_j} \]  

(3.63)
where, $q_j$ is the horizontal location of the road surface point measured by the $j^{th}$ mapping sensor measurement. The grid nodes whose elevations are to be added to the EKF state vector are those that are within the scanning area of the mapping sensors but not currently in the augmented state vector. Formally this set is defined as

$$\mathcal{A}_k = \{g_j : g_j \notin \mathcal{V}_k\} \forall g_j \in \mathcal{W}_k$$ (3.64)

In order to add the elevations of the grid nodes in set to the EKF state vector, it is necessary to predict the elevations of the new grid nodes given the elevations of grid nodes currently in the augmented state vector. A weighted average extrapolation model is utilized to estimate the elevations of the new grid nodes. Thus, the extrapolation model is given by

$$\xi_{g_j} = \sum_i \lambda_{ji} \xi_{g_i} + w_p \forall g_j \in \mathcal{A}_k, g_i \in \mathcal{V}_k$$ (3.65)

The extrapolation error, $w_{aj}$, is assumed to be normally distributed with zero mean and a variance $\sigma_{aj}^2$. The extrapolation weights are computed based on the horizontal Euclidean distance and once again a monotonically decreasing logistic function parameterized by $\tilde{d}$ is used to compute the extrapolation weights. The weights are computed as

$$\lambda_{ji} = \begin{cases} \frac{d^2}{d^2 + d^2_{ji}}, & \text{if } g_i \in \mathbb{N}_{g_j} \\ 0, & \text{otherwise} \end{cases}$$ (3.66)

Thus, the elevations of the new grid nodes can be expressed as a linear function of the predicted EKF state

$$\xi_{\mathcal{A}_k} = \Lambda_k x_k + w_a$$ (3.67)

where, $\Lambda_k$ is a matrix containing the extrapolation weights and $w_a$ is the vector of extrapolation errors.
3.8. Extended Kalman Filter (EKF) solution:

motion errors with $w_a \sim \mathcal{N}(0, \Sigma_a)$. The mean value of the new grid node elevations is computed as

$$\hat{x}_{A_k} = \mathbb{E}[\xi_{A_k}] = \mathbb{E}[\Lambda_k x_k + w_a] = \Lambda_k \hat{x}_{k|k-1}$$

(3.68)

The covariance of the elevations of the new grid nodes is computed as

$$\hat{P}_{A_k} = \mathbb{E}\left[ (\xi_{A_k} - \hat{\xi}_{A_k})(\xi_{A_k} - \hat{\xi}_{A_k})^T \right] = \Lambda_k \hat{P}_{k|k-1} \Lambda_k^T + \Sigma_a$$

(3.69)

The cross-covariance between the elevations of the new grid nodes and the predicted EKF state vector is given by

$$\hat{P}_{x_k A_k} = \mathbb{E}\left[ (x_k - \hat{x}_{k|k-1})(\xi_{A_k} - \hat{\xi}_{A_k})^T \right] = \hat{P}_{k|k-1} \Lambda_k^T$$

(3.70)

The mean of the EKF state vector is then augmented with the mean of the new grid node elevations as

$$\hat{x}_{k|k-1} \leftarrow \{ \hat{x}_{k|k-1} \hat{\xi}_{A_k} \}$$

(3.71)

and the covariance matrix of the EKF state vector is augmented as

$$\hat{P}_{k|k-1} \leftarrow \begin{bmatrix} \hat{P}_{k|k-1} & \hat{P}_{x_k A_k} \\ \hat{P}_{x_k A_k}^T & \hat{P}_{A_k} \end{bmatrix}$$

(3.72)

The grid nodes to be removed from the EKF state vector are those that are currently in the state vector but not within the bounding neighborhood around the vehicle. Formally this set is defined as

$$B_k = \{ g_j : g_j \notin \mathcal{U}_k \} \quad \forall \ g_j \in \mathcal{V}_k$$

(3.73)

Since the EKF represents the distribution over the state vector as a multivariate normal distribution parameterized by the mean vector and covariance matrix, marginalizing out
a subset of grid nodes is relatively easy. This simply corresponds to eliminating the corresponding rows from the mean vector and the corresponding rows and columns from the covariance matrix. The set of grid nodes in the EKF state vector at the next time step is

$$V_{k+1} = V_k | B_k + A_k$$  \hfill (3.74)

### 3.9 Simulation description:

The developed framework is tested on a simulated Mobile Road Mapping System (MRMS) in Matlab. The simulated MRMS model is chosen to mimic the MRMS used at the Vehicle Terrain Performance Laboratory (VTPL) at Virginia Tech. The sensor platform is rigidly mounted on the sprung mass of the a trailer which acts as the host vehicle. The trailer is in turn towed by another vehicle. A schematic of the simulated MRMS is shown in Fig.(3.1). Without loss of generality to the theory the motion of the simulated system is constrained by modeling the following 4 degrees of freedom of the host vehicle:

- Translation of motion of the sprung mass along x and z axes of \textit{w-frame} \((p_x \text{ and } p_z)\)
- Pitching motion of sprung mass about \textit{y-axis of \textit{b-frame}}\((\theta)\)
- Vertical motion of the unsprung mass in \textit{w-frame} \((z_u)\)

The parameters of the dynamic model of the host vehicle (trailer) have been set using the golden quarter car model as a reference and are listed in Table 3.1.

The sensor platform of the MRMS houses an INS and a GNSS receiver as the localization system. The INS measures the specific forces on the sprung mass along the \(x\) and \(z\) axes of the \textit{b-frame} along with the pitch rate of the sprung mass. The GNSS receiver measures the
3.9. Simulation description:

Table 3.1: Parameters of the dynamic model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sprung mass ((m_s))</td>
<td>1000 kg</td>
</tr>
<tr>
<td>Unsprung mass ((m_u))</td>
<td>75 kg</td>
</tr>
<tr>
<td>Sprung mass inertia ((I))</td>
<td>4097 kg-m²</td>
</tr>
<tr>
<td>Suspension spring stiffness ((k_s))</td>
<td>3.165e04 N/m</td>
</tr>
<tr>
<td>Suspension damping coefficient ((c_s))</td>
<td>3e03 N-s/m</td>
</tr>
<tr>
<td>Tire spring stiffness ((k_t))</td>
<td>3.265e05 N/m</td>
</tr>
<tr>
<td>Distance from C.M. to hitch point ((a))</td>
<td>2.25 m</td>
</tr>
<tr>
<td>Distance from C.M. to suspension ((b))</td>
<td>0.75 m</td>
</tr>
</tbody>
</table>

pseudorange from the GNSS receiver to the simulated stationary satellites in the \(w\)-frame.

It also measures the doppler shift between the receiver and the satellites. The platform also house 2 single spot laser scanners, which measure the range (depth) to the road profile as shown in Fig. 3.1.

The parameters of interest for the sensors of the simulated MRMS are given in Table 3.2.
The road surface is modeled as a single longitudinal road profile. The longitudinal road profiles are generated according to the specifications in the standard ISO 8608 [27], in which the road profile is described by its displacement Power Spectral Density (PSD). In order to simulate the MRMS driving in a loop and returning back to map the same section of road, the simulated longitudinal profile is repeated at fixed intervals.

Table 3.2: Parameters of the localization and mapping sensors

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>INS frequency</td>
<td>100 Hz</td>
</tr>
<tr>
<td>GNSS frequency</td>
<td>5 Hz</td>
</tr>
<tr>
<td>Laser frequency</td>
<td>1000 Hz</td>
</tr>
<tr>
<td>Velocity random walk</td>
<td>2.189e-05 m/s²/√Hz</td>
</tr>
<tr>
<td>Angular random walk</td>
<td>2.424e-06 rad/s/√Hz</td>
</tr>
<tr>
<td>Accel bias instability</td>
<td>9.81e-05 m/s³/√Hz</td>
</tr>
<tr>
<td>Gyro bias instability</td>
<td>4.8286e-06 rad/s²/√Hz</td>
</tr>
<tr>
<td>GNSS codephase receiver noise std dev</td>
<td>1.5 m</td>
</tr>
<tr>
<td>GNSS doppler shift receiver noise std dev</td>
<td>5 Hz</td>
</tr>
<tr>
<td>Laser range measurement noise</td>
<td>1.0 mm</td>
</tr>
</tbody>
</table>

The parameters used for the mapping sensor measurement model are listed in Table 3.3. The baseline used to evaluate the developed framework is based on the current state-of-the-art in mobile road mapping. In the state-of-the-art method the INS and GNSS measurements are fused together using an Extended Kalman Filter (EKF) to estimate the trajectory of the pose of the sprung mass (localization solution). The localization solution is further improved by running an RTS smoother on the estimated pose trajectory. The refined localization solution is then combined with the mapping sensor data to produce a raw point cloud. The raw point
cloud is then used to estimate the elevations at the nodes of the regularly spaced grid using a suitable gridding method. In this work the gridding method used is developed in Chapter 5. From here on the state-of-the-art method is denoted as 'SOA', while the method developed in this chapter is denoted as 'EKFSLAM'.

### 3.10 Simulation results

As a baseline case the motion of MRMS is simulated over an ISO 8608 class A longitudinal profile. The MRMS is traveling at a maximum longitudinal velocity of 30 m/s (65 mph) and makes one lap of the track. The motion of the simulated MRMS is shown in Fig. 3.2 and the simulated ISO 8608 road profile is shown in Fig. 3.3. A comparison of the errors in the localization and mapping solutions from the two methods being evaluated is shown in Fig. 3.4. The localization error is characterized by the error in the sprung mass position over time.
Chapter 3. Filtering based solution to the SLAM problem in mobile road mapping

\((\delta p_x \text{ and } \delta p_z)\). The mapping error is characterized by the error in the estimated elevation at each grid node \((\delta \xi_g)\). The metrics used to compare the accuracy of the two methods are the Root Mean Square Error (RMSE) and the Maximum Absolute Error (MAE). The two methods are also compared based on the required computation time. Each simulation case is repeated 15 times and the average RMSE, MAE and computation time values over the 15 runs are computed and summarized in Table 3.4.

<table>
<thead>
<tr>
<th>Variable</th>
<th>SOA</th>
<th>EKFSLM</th>
<th>% improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>MAE</td>
<td>RMSE</td>
</tr>
<tr>
<td>(\delta p_x \text{ (mm)})</td>
<td>8.57</td>
<td>13.65</td>
<td>6.09</td>
</tr>
<tr>
<td>(\delta p_z \text{ (mm)})</td>
<td>4.47</td>
<td>7.24</td>
<td>2.93</td>
</tr>
<tr>
<td>(\delta \xi_g \text{ (mm)})</td>
<td>3.88</td>
<td>8.94</td>
<td>2.69</td>
</tr>
<tr>
<td>Time (secs)</td>
<td>9.15</td>
<td>100.13</td>
<td></td>
</tr>
</tbody>
</table>

In a second simulation scenario the MRMS is simulated to make two laps of the longitudinal track in order to simulate a loop closure event for each of the grid nodes. The error plots
3.10. Simulation results

Figure 3.4: EKFSLAM - Error plots - Single lap

are shown in Fig. 3.5 and a summary of the error metrics is given in Table 3.5.

Table 3.5: EKFSLAM - Comparison of error metrics - Double lap

<table>
<thead>
<tr>
<th>Variable</th>
<th>SOA</th>
<th>EKFSLAM</th>
<th>% improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE MAE</td>
<td>RMSE MAE</td>
<td>RMSE MAE</td>
</tr>
<tr>
<td>$\delta p_x$ (mm)</td>
<td>6.25 10.15</td>
<td>5.06 9.89</td>
<td>19.04 2.56</td>
</tr>
<tr>
<td>$\delta p_z$ (mm)</td>
<td>4.27 7.62</td>
<td>2.78 6.34</td>
<td>34.89 16.79</td>
</tr>
<tr>
<td>$\delta \xi_g$ (mm)</td>
<td>3.09 7.23</td>
<td>2.28 6.38</td>
<td>26.21 11.75</td>
</tr>
<tr>
<td>Time (secs)</td>
<td>15.24</td>
<td>156.80</td>
<td>-</td>
</tr>
</tbody>
</table>

The baseline simulation scenario is repeated with longitudinal profiles of classes B, C and D as specified in the ISO 8608 standard in order to investigate the effect of road profile roughness. The variation in the error metrics for the road grid node elevations with different classes of road profiles is presented in Table 3.6. The baseline simulation scenario is also repeated a maximum longitudinal velocities ranging from 35 mph to 65 mph. The effect of
Figure 3.5: EKFSLAM - Error plots - Double lap

the MRMS longitudinal velocity on the road grid node elevation error metrics for the SOA and EKFSLAM is presented in Table 3.7

<table>
<thead>
<tr>
<th>Variable</th>
<th>SOA RMSE</th>
<th>SOA MAE</th>
<th>EKFSLAM RMSE</th>
<th>EKFSLAM MAE</th>
<th>% improvement RMSE</th>
<th>% improvement MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class A</td>
<td>3.88</td>
<td>8.94</td>
<td>2.69</td>
<td>8.05</td>
<td>30.67</td>
<td>9.95</td>
</tr>
<tr>
<td>Class B</td>
<td>4.07</td>
<td>9.34</td>
<td>3.18</td>
<td>8.12</td>
<td>21.86</td>
<td>13.06</td>
</tr>
<tr>
<td>Class C</td>
<td>3.97</td>
<td>9.41</td>
<td>3.47</td>
<td>8.26</td>
<td>12.59</td>
<td>12.22</td>
</tr>
<tr>
<td>Class D</td>
<td>4.57</td>
<td>9.37</td>
<td>3.33</td>
<td>8.57</td>
<td>27.13</td>
<td>8.53</td>
</tr>
</tbody>
</table>
3.11. Discussion, future work and conclusions:

From the results of the simulation it can be seen that for the baseline simulation scenario there is an improvement in the accuracy of the localization and mapping solutions when using the EKFSLAM method over the State-of-the-Art (SOA) method. The accuracy improvement can be attributed to both the use of the mapping sensor measurements and a dynamic model of the system within the SLAM framework. The dynamic model of the system constrains the localization solution along physically plausible trajectories in between the availability of sensor measurements. For the case where the MRMS makes two laps of the track there is an improvement in the grid node elevation accuracy for both methods. This can be attributed to the higher number of mapping sensor measurements per grid node available during grid node elevation estimation. However, as seen in Table 3.5 the accuracy of the EKFSLAM relative to the SOA method is almost the same as compared to the baseline simulation case (single lap). This is because in the EKFSLAM method loop closure information cannot be utilized to improve the accuracy of the estimates. It can be seen from the results in Table 3.7 that at lower longitudinal velocities the accuracy of the EKFSLAM method relative to the SOA method increases. At lower longitudinal velocities there are higher number of mapping sensor measurements per grid node available for estimating the localization solution within the EKFSLAM algorithm. Hence, an improvement in the accuracy of localization solution results in an improvement in the accuracy of the point cloud used to estimate the grid node

<table>
<thead>
<tr>
<th>Variable</th>
<th>SOA RMSE</th>
<th>SOA MAE</th>
<th>EKFSLAM RMSE</th>
<th>EKFSLAM MAE</th>
<th>% improvement</th>
<th>RMSE</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>65 mph</td>
<td>3.88</td>
<td>8.94</td>
<td>2.69</td>
<td>8.05</td>
<td>30.67</td>
<td>9.95</td>
<td></td>
</tr>
<tr>
<td>55 mph</td>
<td>3.55</td>
<td>8.14</td>
<td>2.47</td>
<td>7.38</td>
<td>30.42</td>
<td>9.33</td>
<td></td>
</tr>
<tr>
<td>45 mph</td>
<td>3.12</td>
<td>7.75</td>
<td>2.13</td>
<td>6.87</td>
<td>31.73</td>
<td>11.35</td>
<td></td>
</tr>
<tr>
<td>35 mph</td>
<td>2.96</td>
<td>7.12</td>
<td>2.02</td>
<td>5.23</td>
<td>31.75</td>
<td>26.54</td>
<td></td>
</tr>
</tbody>
</table>
One of the demerits of the EKFSLAM method is that the time for computation in the EKFSLAM method is almost 10x the time required by the SOA method. The reason is the increased dimension of the state vector due to augmented it with the grid node elevations. The number of grid node elevations added to the state vector is proportional to the size of the bounding neighborhood and the spacing of the grid nodes in the horizontal plane. Another limitation is the need to manually tune the parameters of the algorithm viz the noise standard deviations and the characteristic length of the logistic function. The resulting accuracy of the method is dependent on the proper tuning of the parameters. Future work will address this issue with the possibility of estimating the parameters as well within the EKF-SLAM method.

The EKFSLAM method is sensitive to the presence of outliers in the mapping sensor measurements and even a tiny fraction of outliers can have a detrimental effect. This is because this work assumes a Gaussian distribution over the mapping sensor measurement errors whereas a heavy-tailed distribution is more suitable to model mapping sensor measurements with outliers. The reason for choosing a Gaussian distribution is that the EKF theory is derived based on a Gaussian error distribution assumption. It is proposed in the future to investigate approximation techniques that can implement the Bayesian recursion step with non-Gaussian error distributions and thus ensure that the method is robust to outliers in the mapping sensor measurements.

The EKFSLAM method is a filtering based approach and thus only utilizes sensor measurements up to the current time step. Further improvement in the accuracy can be achieved by using sensor measurements from the future through a smoothing based approach. Such a smoothing based approach to the mobile road mapping problem is explored in the next chapter.
Chapter 4

Smoothing based solution to the
SLAM problem in mobile road
mapping

4.1 Introduction:

The work in this chapter builds off from the theoretical framework developed in the previous chapter. So far the approach to solving the mobile road mapping SLAM problem has been an incremental one wherein the measurements are incorporated one step at a time in conjunction with a filtering based SLAM solution. Thus the estimates of the system state and grid node elevations only utilize sensor measurements (information) up to the current time step. Thus, from a post-processing perspective this can be considered as a "locally" optimal solution to the problem. In this chapter the "globally" optimal solution is developed wherein the complete set of measurements (past, present and future) is utilized to estimate the system state trajectory and road grid node elevations. This corresponds to the smoothing version of solutions to the SLAM problem.

The biggest challenge for the smoothing problem is the large dimension of the search space. This is because in the smoothing formulation one seeks to jointly optimize over
the entire system state trajectory and all the road grid node elevations. Hence, for long
mapping runs the dimension of the search space increases linearly over time thus rendering
the problem computationally infeasible. In order to overcome this challenge, a factor graph
SLAM formulation is explored for the mobile road mapping problem. One of the key insights
is that, similar to many other SLAM problems, the connectivity matrix of the factor graph
turns out to be sparse for the mobile road mapping problem. As shown in [16], under certain
assumptions inference over a sparsely connected factor graph corresponds to solving a linear
least squares problem that is sparse. By taking advantages of the advances in sparse linear
algebra, the sparse system can be solved orders of magnitude faster than the corresponding
dense one. This drastic reduction in computation time makes solving the smoothing problem
for mobile road mapping computationally feasible.

The remainder of this chapter is presented as follows: The formulation of the smoothing
version of the mobile road mapping problem as an inference problem over a factor graph
is developed in section 4.2. Simulation results for a proof-of-concept MRMS are presented
in Section 4.4. Finally, a discussion on the simulation results along with conclusions and
recommendations for future work are given in section 4.5.

4.2 Factor graph problem formulation:

In this section, the mobile road mapping problem is formulated as an inference problem
over a factor graph. The formulation is based on the seminal work of [16] wherein it was
originally developed for the SLAM problem in the field of robot navigation and mapping.
In SLAM literature the formulation is referred to as 'Smoothing and Mapping' (SAM) since
in essence it is solving the smoothing problem as opposed to the filtering problem. Let \( s_{1:T} \)
denote the dynamic model state trajectory over \( T \) discrete time steps, and \( l_{1:T} \) and \( m_{1:T} \)
4.2. Factor graph problem formulation:

denote the history of localization and mapping sensor measurements respectively. The follow-
notations are defined for the factors in the factor graph:
\( \phi^h \) - Mapping sensor factor in the factor graph
\( \phi^g \) - Localization factor in the factor graph
\( \phi^f \) - Dynamic model factor in the factor graph
\( \phi^c \) - Prior factor over road grid node elevations

A schematic of the factor graph representing the mobile road mapping problem is shown in Fig 4.1. The problem of inference over the factor graph is transformed into a sparse linear least squares problem. In order to do this it is necessary to linearize the non-linear sensor measurement and dynamic prediction models. For this a nominal estimate of the system state trajectory and the grid node elevations is required a priori as a linearization point. In this work the estimated state trajectory and grid node elevations from Chapter 3 are utilized as the linearization point. Once the system has been linearized the objective then is to estimate the deviations from the nominal estimate that maximize the posterior probability given the sensor measurements. The deviations are defined as:

\[
\delta s_{1:T} = s_{1:T} - \bar{s}_{1:T} \tag{4.1a}
\]
\[
\delta \xi_G = \xi_G - \bar{\xi}_G \tag{4.1b}
\]
\[
(4.1c)
\]

where, \( \bar{s}_{1:T} \) and \( \bar{\xi}_G \) are the nominal estimates for the system state trajectory and road grid node elevations respectively. For notational brevity, deviations are combined into a single vector and denoted as
\[
\mathcal{X} = \{\delta s_{1:T} \quad \delta \xi_G\} \tag{4.2}
\]
The measurement deviations are defined as the difference between the observed measurements and the measurements predicted using the nominal estimates of the state trajectory and grid node elevations. The history of mapping sensor and localization sensor measurement deviations are denoted as $\delta m_{1:T}$ and $\delta l_{1:T}$ respectively and for notational purposes are combined into a single vector as

$$Z = \{\delta m_{1:T} \quad \delta l_{1:T}\}$$  \hspace{1cm} (4.3)

The objective in this chapter is to solve the linearized smoothing problem for the MRMS which corresponds to the following Maximum A Posteriori (MAP) inference problem over a factor graph

$$x^* = \arg\max_{x} p(x | Z)$$  \hspace{1cm} (4.4)

Using Bayes theorem the posterior probability density function in Eqn.(4.4) can be written
4.2. Factor graph problem formulation:

as

\[ p(\mathcal{X} \mid \mathcal{Z}) \propto p(\mathcal{Z} \mid \mathcal{X}) \cdot p(\mathcal{X}) \]  

(4.5)

The errors in the measurements from the mapping and localization sensors are independent of each other and thus,

\[ p(\mathcal{X} \mid \mathcal{Z}) \propto p(\delta m_{1:T} \mid \mathcal{X}) \cdot p(\delta l_{1:T} \mid \mathcal{X}) \cdot p(\mathcal{X}) \]  

(4.6)

The errors in the localization sensor measurements and the individual mapping sensor measurements at each time step are modeled as additive white Gaussian sequences and thus,

\[ p(\mathcal{X} \mid \mathcal{Z}) \propto \prod_{k=1}^{T} \left\{ \prod_{j=1}^{D} [p(\delta m_{jk} \mid \mathcal{X}) \cdot p(\delta l_{k} \mid \mathcal{X})] \right\} \cdot p(\mathcal{X}) \]  

(4.7)

The mapping sensor measurement residuals at any time step are only dependent on the current system state deviation and the deviation in the elevations of grid nodes in the neighborhood being scanned. Similarly, the localization sensor measurement residuals at any time step only depend on the current system state deviation. Thus,

\[ p(\mathcal{X} \mid \mathcal{Z}) \propto \prod_{k=1}^{T} \left\{ \prod_{j=1}^{D} \left[ p(\delta m_{jk} \mid \delta s_{k}, \delta \xi_{N_{q_{jk}}} \mid \mathcal{X}) \right] \cdot p(\delta l_{k} \mid \mathcal{X}) \right\} \cdot p(\mathcal{X}) \]  

(4.8)

where, \( N_{q_{jk}} \) is the neighborhood of grid nodes in the scanning area of the \( j^{th} \) mapping sensor measurement at time step \( k \). Conditional probability models for the localization and mapping sensor measurement residuals were developed in sections 3.5 and 3.6.1. These
models correspond to the sensor measurement factors in the factor graph and are given as

\[
p(\delta m_{jk} | \delta s_k, \delta \xi_{\eta_{jk}}) = \phi_{jk}^h \approx N(H_{sjk} \delta s_k + H_{\xi_{jk}} \delta \xi_{\eta_{jk}}, \sigma_{mjk}^2) \tag{4.9a}
\]

\[
p(\delta l_k | \delta s_k) = \phi_k^g \approx N(G_{sk} \delta s_k, R_{lk}) \tag{4.9b}
\]

The last term in the R.H.S of Eqn. (4.8) can be further factored as

\[
p(\mathcal{X}) \propto p(\delta s_{T-1} | \delta s_{1:T-1}, \delta \xi_G) \cdot p(\delta s_{1:T-1}, \delta \xi_G) \tag{4.10}
\]

In section 3.7, an approximate continuous linear dynamic model for the system state deviation was developed as

\[
\delta s(t) = F_s(t) \delta s(t) + F_\xi(t) \delta \xi_{\eta_{c(t)}} + B_s \omega_c(t) \tag{4.11}
\]

The equivalent discrete model is given by

\[
\delta s_k \approx \Phi_{sk} \delta s_{k-1} + \Phi_{\xi k} \delta \xi_{\eta_{c(k-1)}} + w_k \tag{4.12a}
\]

where,

\[\Phi_{sk} \] – State transition matrix

\[\Phi_{\xi k} \] – Transition matrix mapping from grid node elevations to rate of change of system state

\[w_k \approx N(0, Q_k) \] – Discrete model process noise

Note that we use \(\delta \bar{s}_k\) to denote the deviation of the system state from the nominal state obtained by running the previous system state and grid node elevations through the non-
4.2. Factor graph problem formulation: linear dynamic prediction model. Thus,

\[ \delta \tilde{s}_k = s_k - f(s_{k-1}, \tilde{\xi}_{N_{Ck-1}}) \] (4.13a)

\[ \delta \tilde{s}_k = \bar{s}_k + \delta s_k - f(\bar{s}_{k-1}, \tilde{\xi}_{N_{Ck-1}}) \] (4.13b)

Substituting Eqn.(4.13) in Eqn.(4.12)

\[ \delta s_k \approx f(\bar{s}_{k-1}, \tilde{\xi}_{N_{Ck-1}}) - \bar{s}_k + \Phi_{sk} \delta s_{k-1} + \Phi_{\xi k} \delta \xi_{N_{Ck-1}} + w_k \] (4.14a)

\[ \delta s_k \approx \delta f_k + \Phi_{sk} \delta s_{k-1} + \Phi_{\xi k} \delta \xi_{N_{Ck-1}} + w_k \] (4.14b)

where, \( \delta f_k = f(\bar{s}_{k-1}, \tilde{\xi}_{N_{Ck-1}}) - \bar{s}_k \). Thus, the conditional probability for the system state deviation conditioned on the previous state deviation and the deviation in the grid node elevations in the tire contact patch areas is given by

\[ p(\delta s_k | \delta s_{k-1}, \delta \xi_{N_{Ck-1}}) = \phi_k^{f} \approx N\left( \delta f_k + \Phi_{sk} \delta s_{k-1} + \Phi_{\xi k} \delta \xi_{N_{Ck-1}}; Q_k \right) \] (4.15)

From the conditional probability model it can be seen that the system state deviation at any time instant is only dependent on the system state deviation at the previous time step and the deviation in the elevations of the road grid nodes in the tire contact patch areas. Thus,

\[ p(\mathcal{X}) = p(\delta s_T | \delta s_{T-1}, \delta \xi_{N_{C_{T-1}}}) \cdot p(s_{1:T-1}, \delta \xi_{G}) \] (4.16)

where, \( N_{C_{T-1}} \) is the neighborhood of grid nodes around the tire contact patch locations at time step \( T - 1 \). Eqn.(4.16) provides a recursive factorization for \( p(\mathcal{X}) \). Thus,

\[ p(\mathcal{X}) = \prod_{k=1}^{T} \left[ p(\delta s_k | \delta s_{k-1}, \delta \xi_{N_{Ck-1}}) \right] \cdot p(\delta s_0) \cdot p(\delta \xi_{G}) \] (4.17)
The initial system state is assumed to have a normal distribution centered about the nominal estimate with a known covariance, \( \Sigma_0 \) and thus,

\[
p(\delta s_0) = \phi_0 = \mathcal{N}(0, \Sigma_0) \tag{4.18}
\]

In this work the grid node elevation deviations are assumed to be independent and each grid node elevation is assumed to have a prior normal distribution centered about the nominal estimate with a fixed known variance, \( \sigma_g^2 \). Thus,

\[
p(\delta \xi_g) = \prod_{i=1}^{G} p(\delta \xi_{g_i}) = \prod_{i=1}^{G} \mathcal{N}(0, \sigma_g^2) \tag{4.19}
\]

where, \( \phi_i \) represent the prior factors of the grid node elevation deviations. Substituting Eqns. (4.18) and (4.19) in Eqn.(4.17)

\[
p(X) = \prod_{k=1}^{T} \left[ p(\delta s_k | \delta s_{k-1}, \delta \xi_{N_{k-1}}) \right] \cdot p(\delta s_0) \cdot \prod_{i=1}^{G} p(\delta \xi_{g_i}) \tag{4.20}
\]

Finally substituting Eqn.(4.20) in Eqn.(4.8)

\[
p(X | \mathcal{Z}) \propto \prod_{k=1}^{T} \left\{ \prod_{j=1}^{D} \left[ p(\delta m_{jk} | \delta s_k, \delta \xi_{\mathcal{N}_{jk}}) \right] \cdot p(\delta l_{jk} | \delta s_k) \cdot p(\delta s_k | \delta s_{k-1}, \delta \xi_{\mathcal{N}_{jk}}) \right\} \ldots
\]

\[
\ldots \cdot p(\delta s_0) \cdot \prod_{i=1}^{G} p(\delta \xi_{g_i}) \tag{4.21a}
\]

The corresponding factor graph expression for the posterior density is given by

\[
p(X | \mathcal{Z}) \propto \prod_{k=1}^{T} \left\{ \prod_{j=1}^{D} \left[ \phi_{jk}^h \cdot \phi_{jk}^g \cdot \phi_{jk}^f \right] \right\} \cdot \phi_0 \cdot \prod_{i=1}^{G} \phi_i^\xi \tag{4.22}
\]
4.2. Factor graph problem formulation:

Taking the negative natural logarithm on both sides of Eqn. (4.21)

\[- \log p(\mathcal{X} \mid \mathcal{Z}) \propto \sum_{k=1}^{T} \left\{ - \sum_{j=1}^{D} \log \left[ p(\delta m_{jk} \mid \delta s_{k}, \delta \xi_{\mathcal{N} \setminus jk}) \right] \right\} - \log [p(\delta l_{k} \mid \delta s_{k})] \ldots \]

\[\ldots - \log \left[ p(\delta s_{k} \mid \delta s_{k-1}, \delta \xi_{\mathcal{N} \setminus k-1}) \right] \right\} - \log p(\delta s_{0}) - \sum_{i=1}^{G} \log p(\delta \xi_{g_{i}})\]  

(4.23)

Since the conditional probability models (factors) for the sensor measurements, dynamic prediction and prior probabilities are all be normal distributions the logarithm of the probability models can be simplified as

\[- \log p(\mathcal{X} \mid \mathcal{Z}) \propto \sum_{k=1}^{T} \left\{ \sum_{j=1}^{D} \left\| \delta m_{jk} - H_{s_{jk}} \delta s_{k} - H_{c_{jk}} \delta \xi_{\mathcal{N} \setminus jk} \right\|_{\Sigma^{-1}}^{2} \right\} + \left\| \delta l_{k} - G_{sk} \delta s_{k} \right\|_{R_{sk}}^{2} \ldots \]

\[\ldots + \left\| \delta s_{k} - \delta f_{k} - \Phi_{sk} \delta s_{k-1} - \Phi_{c_{jk}} \delta \xi_{\mathcal{N} \setminus c_{k-1}} \right\|_{Q_{k}}^{2} \right\} + \left\| \delta s_{0} \right\|_{\Sigma_{0}}^{2} + \sum_{i=1}^{G} \left\| \delta \xi_{g_{i}} \right\|_{\Sigma_{g}}^{2} \]

(4.24a)

where, \( \| \cdot \|_{\Sigma}^{2} \) is the squared Mahalanobis distance given a covariance matrix \( \Sigma \). Since the natural logarithm is a monotonically increasing function, maximizing the posterior probability is equivalent to minimizing its negative natural logarithm. The problem in Eqn. (4.4) can be reformulated as

\[\mathcal{X}^{*} = \arg\min_{\mathcal{X}} - \log \left[ p(\mathcal{X} \mid \mathcal{Z}) \right] \]  

(4.25)

From Eqns. (4.25) and (4.24) it can be seen that the problem being solved is essentially a weighted least squares problem. Each of the Mahalanobis distance terms in Eqn. (4.24) can be rewritten as

\[\| e \|_{\Sigma}^{2} = e^{T} \Sigma^{-1} e = (\Sigma^{-T/2} e)^{T} (\Sigma^{-T/2} e) = \|\Sigma^{-T/2} e\|_{2}^{2} \]

(4.26)

where, \( \Sigma^{-1/2} \) is the matrix square root of \( \Sigma^{-1} \). Hereafter it is assumed that all the Mahalanobis distance terms have been pre-multiplied by the square root of the corresponding.
covariance matrix and the Mahalanobis notation is removed. Finally after collecting all
the Jacobian matrices into a single matrix $A$, and the residuals $\delta m_{jk}$, $\delta l_k$ and $\delta f_k$ into a
right-hand side (RHS) vector $b$, we obtain the following standard least-squares problem,

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} ||A\mathcal{X} - b||_2$$  \hspace{1cm} (4.27)

4.3 Sparse Linear Least Squares:

The structure of the linear system being solved for in Eqn. (4.27) is

$$\begin{bmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{bmatrix}
\begin{bmatrix}
\delta s_{1:T} \\
\delta \xi_g
\end{bmatrix} =
\begin{bmatrix}
b_1 \\
b_2
\end{bmatrix}$$ \hspace{1cm} (4.28)

where,

$$A_{11} =
\begin{bmatrix}
\phi_0 & 0 & \ldots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
\phi_{1k} & \cdots & H_{s1k} & \cdots \\
\vdots & \ddots & \vdots & \ddots \\
\phi_{Dk} & \cdots & H_{sDk} & \cdots \\
\phi_k & \cdots & G_{sk} & \cdots \\
\phi_k & \cdots & \Phi_{sk} - I & \cdots \\
\vdots & \ddots & \ddots & \ddots
\end{bmatrix}$$ \hspace{1cm} (4.29)
4.3. Sparse Linear Least Squares:

The sub-matrix $A_{12}$ is a matrix of zeros of the appropriate dimension. The RHS vector has the following terms

$$A_{22} = \text{diag} \left( \sigma_{g1}^{-1}, \sigma_{g2}^{-1}, \ldots, \sigma_{gG}^{-1} \right) \quad (4.31)$$

The sub-matrix $A_{21}$ is a matrix of zeros of the appropriate dimension. The RHS vector has the following terms

$$b_1 = \left[ \cdots \delta m_{1k} \cdots \delta m_{Dk} \delta l_k \delta f_k \cdots \right]_c \quad (4.32a)$$

$$b_2 = 0_{G \times 1} \quad (4.32b)$$

It can be seen that the matrix $A$ has relatively few non-zero entries and thus is a sparse matrix. Due to the sparse nature of the system, computationally efficient sparse linear solvers can be utilized to solve the least-squares problem in Eqn. (4.28). Two common techniques used to solve such systems are: Cholesky factorization and QR factorization.
4.3.1 Cholesky Factorization

For a full-rank $m \times n$ matrix $A$, with $m \geq n$, the unique least-squares solution to the problem in Eqn. (4.27) can be found by solving the normal equations

$$\left( A^T A \right) \delta \mathbf{x}^* = A^T \mathbf{b} \quad (4.33)$$

This is normally done by factoring the information matrix, $\mathcal{I}$, defined and factored as follows:

$$\mathcal{I} = A^T A = R^T R \quad (4.34)$$

where, $R$ is the Cholesky factor of $\mathcal{I}$ and is an upper triangular $n \times n$ matrix. The solution to the system is obtained by first solving

$$R^T y = A^T b \quad (4.35)$$

via forward substitution. Next the final solution $\delta \mathbf{x}^*$ is obtained by solving

$$R \delta \mathbf{x}^* = y \quad (4.36)$$

via backward substitution.

4.3.2 QR Factorization

The QR factorization can be used to compute the solution to (4.27) without having to compute the information matrix, $\mathcal{I}$. Instead, the QR-factorization of $A$ itself is computed
4.4. Simulation results

along with the corresponding RHS:

\[ A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}; \quad Q^Tb = \begin{bmatrix} d \\ e \end{bmatrix} \] (4.37)

where, \( Q \) is an \( m \times m \) orthogonal matrix, \( d \in \mathbb{R}^n, e \in \mathbb{R}^{m-n} \), and \( R \) is an upper-triangular matrix. Pre-multiplying the matrix \( A \) by the orthogonal matrix \( Q \) gives

\[ Q^T A = \begin{bmatrix} R \\ 0 \end{bmatrix} \] (4.38)

Thus, pre-multiplying the least squares equation by \( Q \) gives

\[ ||A\mathbf{x} - b||_2^2 = ||Q^T A\mathbf{x} - Q^T b||_2^2 = ||R\mathbf{x} - d||_2^2 + ||e||_2^2 \] (4.39)

It can be seen that \( ||e||_2^2 \) is the least-squares sum of squared residuals, and the least-squares solution \( \mathbf{x}^* \) can be obtained by solving the triangular system

\[ RX^* = d \] (4.40)

4.4 Simulation results

The developed method using factor graph Smoothing and Mapping (SAM) is evaluated using the simulation environment developed in Chapter 3. From here on the method developed in this chapter is referred to as SAM. The SAM method is compared against both the current State-of-the-Art (SOA) method as well as the EKFSLAM method from Chapter 3 using the RMSE and MAE error metrics along with the required computation time. For the baseline
simulation scenario (single lap), a plot of the localization and mapping errors in shown in Fig. 4.2 and the error metrics and computation time are summarized in Table 4.1. The error plots for the double lap simulation scenario are shown in Fig. 4.3 and the corresponding error metrics are summarized in Table 4.2. An analysis of the computation time taken by the SAM method for the single lap and double lap scenarios is presented in Table.

Table 4.1: SAM - Comparison of error metrics - Single lap

<table>
<thead>
<tr>
<th>Variable</th>
<th>SOA</th>
<th>EKFSLAM</th>
<th>SAM</th>
<th>% improvement SAM over SOA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>MAE</td>
<td>RMSE</td>
<td>MAE</td>
</tr>
<tr>
<td>$p_x$ (mm)</td>
<td>8.57</td>
<td>13.65</td>
<td>6.09</td>
<td>9.27</td>
</tr>
<tr>
<td>$p_z$ (mm)</td>
<td>4.47</td>
<td>7.24</td>
<td>2.93</td>
<td>5.83</td>
</tr>
<tr>
<td>$\xi_g$ (mm)</td>
<td>3.88</td>
<td>8.94</td>
<td>2.69</td>
<td>8.05</td>
</tr>
<tr>
<td>Time (secs)</td>
<td>9.15</td>
<td>100.13</td>
<td>94.35</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.2: SAM - Error plots - Single lap
4.5. Discussion, future work and conclusion:

In this chapter a framework for solving the smoothing problem for mobile road mapping has been developed using the principles of Smoothing and Mapping (SAM). The simulation results demonstrate that the developed SAM framework achieves a significant improvement in the accuracy of the localization and mapping solutions of the mobile road mapping problem when compared to the State-of-the-Art (SOA) method and the EKFSLAM method. The improvement over the SOA method can be attributed to the simultaneous use of the localization and mapping sensor measurements and also to the use of the dynamic model of the mapping system. Compared to the EKFSLAM method, the SAM method differs in two important ways: Firstly the SAM method utilizes the entire set of sensor measurements (past, present and future) to estimate the state of the system and the grid node elevations whereas the EKFSLAM method utilizes only the sensor measurements upto the current time step. Secondly the SAM method estimates the joint posterior over the entire system state.

Table 4.2: SAM - Comparison of error metrics - Double lap

<table>
<thead>
<tr>
<th>Variable</th>
<th>SOA</th>
<th>EKFS</th>
<th>SAM</th>
<th>% improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>MAE</td>
<td>RMSE</td>
<td>MAE</td>
</tr>
<tr>
<td>$p_x$ (mm)</td>
<td>6.25</td>
<td>10.15</td>
<td>5.06</td>
<td>9.89</td>
</tr>
<tr>
<td>$p_z$ (mm)</td>
<td>4.27</td>
<td>7.62</td>
<td>2.78</td>
<td>6.34</td>
</tr>
<tr>
<td>$\xi_g$ (mm)</td>
<td>3.09</td>
<td>8.23</td>
<td>3.43</td>
<td>8.15</td>
</tr>
<tr>
<td>Time (secs)</td>
<td>15.24</td>
<td>156.80</td>
<td>148.76</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4.3: SAM - Analysis of computation time

<table>
<thead>
<tr>
<th>Time to assemble</th>
<th>Time to solve the system</th>
<th>Total time</th>
</tr>
</thead>
<tbody>
<tr>
<td>sparse matrices</td>
<td>using Cholesky factorization</td>
<td></td>
</tr>
<tr>
<td>Single lap</td>
<td>90.34</td>
<td>4.01</td>
</tr>
<tr>
<td>Double lap</td>
<td>143.12</td>
<td>5.64</td>
</tr>
</tbody>
</table>

4.5 Discussion, future work and conclusion:

In this chapter a framework for solving the smoothing problem for mobile road mapping has been developed using the principles of Smoothing and Mapping (SAM). The simulation results demonstrate that the developed SAM framework achieves a significant improvement in the accuracy of the localization and mapping solutions of the mobile road mapping problem when compared to the State-of-the-Art (SOA) method and the EKFSLAM method. The improvement over the SOA method can be attributed to the simultaneous use of the localization and mapping sensor measurements and also to the use of the dynamic model of the mapping system. Compared to the EKFSLAM method, the SAM method differs in two important ways: Firstly the SAM method utilizes the entire set of sensor measurements (past, present and future) to estimate the state of the system and the grid node elevations whereas the EKFSLAM method utilizes only the sensor measurements upto the current time step. Secondly the SAM method estimates the joint posterior over the entire system state.
trajectory and all the grid node elevations while the filtering based EKFSLAM method only estimates the joint posterior over the current system state and the elevations of grid nodes within the bounding neighborhood around the mapping system. In the EKFSLAM method previous system states and elevations of grid nodes outside the bounding neighborhood are marginalized out for computational feasibility but as a result the correlations between the estimates is lost. By contrast in the SAM method, the correlations among the estimates of all variables are maintained and are encoded in the entries of the sparse Jacobian matrix (or alternately the information matrix). Hence, when the mapping system returns to a section of road that has already been mapped the loop is closed and this results in a drastic improve-
ment in the accuracy of the system as demonstrated in Figs. 4.2 and 4.3. The computation time of the SAM method is almost 10x the computation time taken by the SOA method. From Table 4.3 it can be seen that the bulk of the computation time is spent on assembling
the sparse matrices of the linear least square problem and a small fraction of the time is spent to solve the linear system using Cholesky factorization. From an implementation standpoint the process of assembling the sparse matrices can be sped up further by using a parallel programming model.

The SAM framework presented in this chapter is not robust to outliers in the mapping sensor measurements, since a Gaussian distribution is used to model the errors in the mapping sensor range measurements. In chapter 5 a method is developed to identify outliers and eliminate them from the measurement set before using them in the SAM framework. However, in the future it is proposed to use a non-Gaussian heavy tailed distribution to model the mapping sensor range measurement errors which will also require modifying the SAM framework to use a non-Gaussian error distribution. The SAM solution requires an initial estimate of the system state trajectory and grid node elevations as a linearization point. In this work the solution of the filtering based EKFSLAM method is used as the initial linearization point. In the future it is proposed to investigate the use of the incremental versions of the Smoothing and Mapping (SAM) [38, 39].

In conclusion, a framework for solving the smoothing version of the mobile road mapping problem has been developed. The framework is based on the seminal Smoothing and Mapping (SAM) [16] framework, in which the problem is formulated as a sparse linear least squares problem. Simulation results from a proof-of-concept model show that using the SAM framework can improve the accuracy of the MRMS by almost an order of magnitude at best over the current state-of-the-art methods.
Chapter 5

Identifying outliers in road surface datasets

5.1 Introduction

Laser measurement systems are common mapping sensors in Mobile Road Mapping Systems (MRMS) and they range from single point lasers to scanning lasers capable of measuring millions of data points per second. However, these laser measurement systems have several mechanisms through which outliers manifest in the datasets. When scanning real road surfaces, deformable debris (e.g., leaves, grass) are sampled in the same manner as non-deformable surface features (e.g., spalled concrete, loose asphalt aggregate). Similarly, a laser sensor may generate outliers due to inappropriate intensity levels that saturate the sensor (e.g., resulting from highly reflective surface features). In this work, the identification of statistical outliers is used a means to identify invalid measurements that originate from exogenous sources other than the road surface itself.

Regardless of the exogenous source from which statistical outliers manifest in road surface datasets, these invalid measurements must be removed before they are used for estimation or simulation purposes. Outliers in mapping sensor measurements, when not accounted (modeled) for, can lead to divergence of filtering and smoothing algorithms. Outliers in vehicle model excitations typically produce simulation results with significant artificial vehicle
responses, particularly suspension loads, requiring their removal whenever predicting ride quality and durability. Ride quality predictions are highly sensitive to events that cause sharp transient movements of the vehicle [24, 67]. Predicting limit handling capability is especially important in automated and autonomous vehicles, for which the vehicle response is predicted based on models of interactions between the vehicle, its tires and the road surface. The predicted vehicle response can be used to compute performance measures [52], adjust the dynamic properties of the vehicle [82] and also implement various control strategies [7]. Misinterpreting outliers as being valid road surface points can have substantial negative impact on the ability of a real-time control system to accurately predict the vehicle dynamics and safely navigate different maneuvers.

In this chapter, a novel, axiomatic, probabilistic process for simultaneously identifying outliers and estimating the road surface elevations at grid nodes is developed. The axioms developed in this work draw from concepts utilized in the seminal model fitting algorithm, Random Sampling and Consensus (RANSAC) [26]. Similar to RANSAC, randomly sampled combinations of points are used to generate candidate surfaces to represent the dataset. In the proposed method, it is presumed that the likelihood that a point is valid depends on the validity of the candidate surfaces in which it is a member, and that the likelihood that a candidate surface is valid depends on the validity of the points that are members of that surface. Thus, the objective is to estimate the likelihood that measured points and candidate surfaces are valid. In this work, likelihood estimates are developed as logistic functions for: closeness of grid nodes, membership of points in a candidate surface, validity of candidate surfaces, and validity of measured points in the road surface dataset. The approach to achieving this objective exploits the information available from all candidate surfaces. The road surface elevation at a grid node is estimated using information about the validity of points that lie within the neighborhood of that grid node.
The remainder of this work is developed as follows: A framework for identifying outliers and estimating grid node elevations from measured road surface datasets is developed in section 5.2. Also developed are methods to estimate the probabilities that: points are close to a grid node, points are members of a candidate surface, candidate surfaces are valid, and points are valid. In section 5.3 the developed concepts are evaluated using a simulated dataset with a significant percentage of outliers and finally section 5.4 contains a discussion followed by recommendations for future work and concluding remarks.

5.2 Identifying Outliers and Estimating Grid Node Elevations

The proposed method to identify outliers is established from axioms regarding the measured road surface points in a neighborhood, and the candidate surfaces used to represent these points.

1. The likelihood of a candidate surface being valid monotonically increases with the fraction of valid neighborhood points it has as members.

2. The likelihood of a point being valid monotonically increases with the fraction of valid surfaces in which it is a member.

These qualitative axioms drive the need to develop methods to estimate membership of points in candidate surfaces, and to estimate the validity of points and candidate surfaces. Once the validity of points has been established, the elevation of each grid node is estimated using the validity of neighborhood points and as well as their closeness to the grid node. This in turn requires that precise definitions of neighborhoods and a means to determine
closeness of points to the grid node are developed.

### 5.2.1 Estimating Grid Node elevations

Consider the measured road surface points as a set of 3D points \( \mathcal{R} \) and the grid nodes as a set of regularly spaced grid nodes, \( \mathcal{G} \) as defined in section 2.5. Consider a grid node, \( g_i \in \mathcal{G} \), and a measured road surface point \( r_j \in \mathcal{R} \). For this chapter, the neighborhood of a grid node, \( g_i \), is defined as the set of measured points whose horizontal euclidean distance from \( g_i \) is less than a certain threshold, \( d_{th} \). Thus,

\[
\mathbb{N}_{g_i} = \{ r_j : ||q_j - g_i|| \leq d_{th} \} \ \forall \ r_j \in \mathcal{R} \tag{5.1}
\]

Let \( \mathcal{M}_i \) denote the set of candidate surface models chosen to represent \( \mathbb{N}_{g_i} \). A minimum number of neighborhood points are required to construct a candidate surface model (e.g., three non-collinear points for a plane), denoted herein as \( n_{cp} \). The \( k^{th} \) candidate surface model, \( m_{ik} \in \mathcal{M}_i \), is constructed by fitting a surface to \( n_{cp} \) randomly selected construction points. The process by which the sets of construction points are randomly selected is outside the scope of this work; however, the process should result in points being selected about the same number of times (selected as members of approximately the same number of sets) and there should be significant mixing in the combinations of points. Recall from section 2.5 that each measured road surface point is given by

\[
r_j = \begin{bmatrix} q_j \\ \xi_j \end{bmatrix}^T \ \forall r_j \in \mathcal{R} \tag{5.2}
\]

where,
$q_j$ – Horizontal location of measured point $r_j$

$\xi_{q_j}$ – Elevation of measured point $r_j$

The elevation of grid node $g_i$ is denoted as $\xi_{g_i}$. In this chapter a method is developed to estimate a weighted empirical distribution function for the elevation at each grid node by using the elevations of the measured neighborhood points as samples from the true underlying distribution. The empirical distribution function is defined as

$$F_w [\xi_{g_i}] = \frac{1}{\sum_j w_{ij}} \sum_j w_{ij} \cdot 1 [\xi_{q_j} \leq \xi_{g_i}] \quad \forall r_j \in N_{g_i} \quad (5.3)$$

where,

1 $[A]$ – Indicator function for event $A$

$w_{ij}$ – Weights of the empirical distribution function

It is proposed that the weight of each measured point of the empirical distribution be proportional to the joint probability that: the measured point is valid and the measured point is close to the grid node. Consider the following events that have some likelihood of occurrence.

$$A_{ij} = \{ r_j \text{ is a valid point in } N_{g_i} \} \quad (5.4a)$$

$$B_{ij} = \{ r_j \text{ is close to } g_i \} \quad (5.4b)$$

Clearly these two events are independent and thus the weights, $w_{ij}$, can be computed upto proportion as

$$w_{ij} \propto Pr (A_{ij}) \cdot Pr (B_{ij}) \quad (5.5)$$

Furthermore, it is proposed that the final estimate of the elevation of each grid node be computed as the median over its estimated empirical distribution. The probability of event
5.2. Identifying Outliers and Estimating Grid Node elevations

\( B_{ij} \) depends on the horizontal distance from the point to the grid node relative to the accuracy of measurement in the horizontal plane. The probability of event \( A_{ij} \) is based on the validity of the candidate surfaces in the neighborhood and the membership of \( r_{ij} \) in the candidate surfaces. Methods to estimate the membership of points in surfaces, the probability of validity of points, and the probability of validity of surfaces are developed in the following sub-sections.

5.2.2 Closeness to a Grid Node

The probability that point, \( r_j \), is close to grid node \( g_i \) depends on the the horizontal distance which is computed as

\[
d_{ij} = \|q_j - g_i\|_2
\]  

(5.6)

It is proposed that the results of [51] be used, in modified form, to estimate \( Pr(B_{ij}) \).

\[
\hat{Pr}(B_{ij} | d_{ij}) = \frac{\tilde{d}^2}{d^2 + d_{ij}^2}
\]  

(5.7)

Note that the median horizontal distance, \( \tilde{d} \), is a parameter that needs to be determined; this can be accomplished using information from other regions in the dataset or adapted to meet the application requirements. As a guideline, it is proposed that an initial estimate of the median horizontal distance, \( \tilde{d} \), be set equal to the Circular Error Probability (CEP), which is readily available from navigation system manufacturers. As a guide, if the estimated median horizontal distance is \( \tilde{d} \) then the initial estimate of the threshold radius can be set to \( d_{th} \geq \tilde{d} \).
5.2.3 Membership of points in candidate surfaces

The membership of a point in a candidate surface is based on its normal distance from the surface. Consider a point $r_j$ and a candidate surface, $m_{ik} \in M$. Define the projection of $r_j$ on the candidate surface $m_{ik}$ as $s_{ijk}$. The normal distance error, $e_{ijk}$, is defined in Eqn. (5.8)

$$ e_{ijk} = \| r_j - s_{ijk} \|_2 $$

(5.8)

Next consider the following event which has some likelihood of occurrence.

$$ C_{ijk} = \{ r_j \text{ is a member of } m_{ik} \} $$

(5.9)

The probability of this event, $Pr(C_{ijk})$, given the normal distance error, $e_{ijk}$, would have some required properties and desired characteristics. Specifically, it would,

- Obey the axioms of probability.
- Be a monotonically decreasing function of all positive finite values of $e_{ijk}$
- Asymptote to one as the error magnitude approaches zero
- Asymptote to zero as the error magnitude approaches infinity
- Have a median magnitude of error, $\tilde{e}$, for which $Pr(C_{ijk} | e_{ijk} = \tilde{e}) = \frac{1}{2}$

It may not be possible or practical to identify the conditional probability function $Pr(C_{ijk} | e_{ijk})$, so admissible functions that satisfy the properties and characteristics of the probability function are proposed as surrogates. A simple, admissible, monotonically decreasing logistic
5.2. Identifying Outliers and Estimating Grid Node elevations

A function is used in this work.

\[
\hat{Pr}(C_{ijk} | e_{ijk}) = \frac{\hat{e}^2}{\hat{e}^2 + e_{ijk}^2}
\]  

(5.10)

The median normal distance error, \(\hat{e}\), is the parameter that controls the shape of the function and needs to be determined. It is related to the variation of the error present in the valid neighborhood points. Consider the case in which the normal distance error is a Gaussian distribution with standard deviation \(\sigma\); about half of the variation would be expected to occur within two-thirds of the standard deviation, \(Pr(C_{ijk} | e_{ijk} = \frac{2}{3}\sigma) \approx \frac{1}{2}\). As a guide, therefore, \(\hat{e}\) can be set to two-thirds of the expected standard deviation of error for the set of valid points as an initial estimate.

5.2.4 Validity of candidate surfaces

Let \(D_{ik}\) denote the event that \(m_{ik}\) is a valid surface.

\[
D_{ik} = \{m_{ik} \in M_i \text{ is a valid surface}\}
\]

(5.11)

The probability of the event, \(Pr(D_{ik})\), is derived based on Axiom 1. According to this axiom, the probability that a candidate surface, \(m_{ik}\), is valid depends on the fraction of valid points in the neighborhood that are its members. Let \(\gamma_{ik}\) denote the expected value of this fraction, estimated as

\[
\gamma_{ik} = \frac{\sum_j Pr(A_{ij})Pr(C_{ijk})}{\sum_j Pr(A_{ij})}
\]

(5.12)

A function is desired that can parameterize the probability of event \(Pr(D_{ik})\) conditioned on the expected fraction of valid points, i.e \(Pr(D_{ik} | \gamma_{ik})\). Since \(\gamma_{ik}\) is a fraction taking on
values in the interval [0, 1], it is desirable that

\[ Pr(D_{ik} \mid \gamma_{ik} = 0) = 0 \]  
\[ Pr(D_{ik} \mid \gamma_{ik} = 1) = 1 \]

(5.13a)  
(5.13b)

Based on axiom 1 in this work \( Pr(D_{ik} \mid \gamma_{ik}) \) should monotonically increase with \( \gamma_{ik} \). Thus,

\[ \frac{dPr(D_{ik} \mid \gamma_{ik})}{d\gamma_{ik}} > 0 \quad \forall \gamma_{ik} \in [0, 1] \]

(5.14)

It may not be possible or practical to identify the probability function \( Pr(D_{ik} \mid \gamma_{ik}) \), so an admissible function that satisfies the properties and characteristics of the conditional probability function is proposed as a surrogate for the true conditional probability function. In this work a simple variant of a monotonically increasing logistic function is chosen as the surrogate function for the conditional probability function.

\[ \hat{Pr}(D_{ik} \mid \gamma_{ik}) = (1 - \tilde{\gamma}^2) \cdot \frac{\gamma_{ik}^2}{\gamma_{ik}^2 - 2\gamma_{ik}\tilde{\gamma}^2 + \tilde{\gamma}^2} \]

(5.15)

The chosen function is parameterized by a single parameter \( \tilde{\gamma} \), which controls the shape of the function. The surrogate function and its parameter are chosen in such a way that when \( \gamma_{ik} = \tilde{\gamma} \) the probability of \( D_{ik} \) is 0.5 meaning that the likelihood of surface \( m_{ik} \) being valid is 50%. As a guide, an initial estimate of \( \tilde{\gamma} \) can be set to the expected fraction of valid points in the dataset. Let \( \alpha \) denote the expected fraction of valid points in the dataset. However, for larger neighborhoods that are likely to contain multiple underlying surfaces it is recommended to lower the value of \( \tilde{\gamma} \).
5.2. Identifying Outliers and Estimating Grid Node Elevations

5.2.5 Validity of Neighborhood Points

According to axiom 2, the probability of event \( Pr(A_{ij}) \) depends on the fraction of valid surfaces in which it is a member. Let \( \beta_{ij} \) denote the expected value of this fraction, estimated as

\[
\beta_{ij} = \frac{\sum_k Pr(C_{ijk}) Pr(D_{ik})}{\sum_k Pr(D_{ik})}
\] (5.16)

Similar to Eqn. (5.15) and based on axiom 2, a variant on a monotonically increasing logistic function is utilized as a surrogate for the true conditional probability function, \( Pr(A_{ij} | \beta_{ij}) \).

\[
\tilde{Pr}(A_{ij} | \beta_{ij}) = (1 - \tilde{\beta}^2) \cdot \frac{\beta_{ik}^2}{\beta_{ik}^2 - 2\beta_{ik}^2 \tilde{\beta}^2 + \tilde{\beta}^2}
\] (5.17)

In this case, the logistic function is monotonically increasing with respect to the fraction of valid surfaces in which the point is a member. The single parameter \( \tilde{\beta} \) controls the shape of the function and is the value at which the probability of event \( B_{ij} \) is 50%. The parameter \( \tilde{\beta} \) needs to be tuned based on the properties of the road surface dataset. Recall that \( n_{cp} \) denotes the minimum number of neighborhood points required to construct a candidate surface, \( n_i \) denotes the number of points in neighborhood \( N_i \), and \( \alpha \) is the expected fraction of valid points in the dataset. As a guide, the expected fraction of valid candidate surfaces, \( \tilde{\beta} \), can be initially estimated as

\[
\tilde{\beta} = \alpha^{n_{cp}}
\] (5.18)

Once again, for larger neighborhoods that are likely to contain multiple underlying surfaces it is recommended to use a lower value of \( \tilde{\beta} \). It should also be clear that \( n_i > n_{cp}/\alpha \) in order
to expect any valid surfaces to be generated.

### 5.2.6 Validity of Surfaces and Points

The validity of points and candidate surfaces is derived from the axioms proposed in this work. It is apparent that the likelihood that a point is valid depends on the fraction of valid surfaces of which it is a member and, similarly, the likelihood that a surface is valid depends on the fraction of valid points that are its members. As such, the following probabilities need to be estimated:

1. Probability of validity of points, $Pr(A_{ij})$, which depends on $Pr(C_{ijk})$ and $Pr(D_{ik})$

2. Probability of validity of surfaces, $Pr(D_{ik})$, which depends on $Pr(C_{ijk})$ and $Pr(A_{ij})$

A potential solution is to use an iterative approach for each neighborhood. For example, initially all measured points can be assumed to be valid with probability one (alternately, the probability can be set to the expected fraction of valid points). The estimated probability of validity of neighborhood points and their membership in the candidate surfaces is used to calculate the initial estimate of the validity of the candidate surfaces. The probability of validity of candidate surfaces is then in turn used to recompute the probability of validity of the neighborhood points.

### 5.2.7 Identifying outliers

For each grid node $g_i$, the probability of validity and points and candidate surfaces in the neighborhood $N_{g_i}$ is determined. It is possible for a single point, $r_j$ to belong to the neighborhoods of multiple grid nodes. The overall probability of validity of a point $r_j$ in the
dataset is estimated as the mean of the its probability of validity within each neighborhood it belongs to. Thus,

\[ Pr(A_j) = \frac{\sum_i Pr(A_{ij})}{N_j} \quad \forall \; i \; s.t \; r_j \in N_{g_i} \]  

(5.19)

Points for which the overall probability of validity is less than a threshold value are labeled as outliers.

5.3 Simulation and Results

5.3.1 Proof of concept

The simulation dataset being used to evaluate the performance of the proposed method is a two dimensional dataset called Stair-4 [85]. This dataset contains 4 surfaces (lines) and 60% outliers. Gaussian noise is added to the inliers. The parameters of the simulation dataset are given in Table 5.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of inliers</td>
<td>200</td>
</tr>
<tr>
<td>Number of outliers</td>
<td>300</td>
</tr>
<tr>
<td>Number of true surfaces(lines)</td>
<td>4</td>
</tr>
<tr>
<td>Standard deviation of inlier error ($\sigma_{\text{in}}$)</td>
<td>0.0075</td>
</tr>
</tbody>
</table>

Table 5.1: Parameters of the Simulation dataset

The baseline being used for comparison is the original RANSAC method followed by an Inverse Distance to a Power (IDP) gridding method in the neighborhood of each grid node. The original RANSAC algorithm is applied to each neighborhood and the candidate surface with the highest consensus is determined. All the points whose normal distance from this surface is less than the error threshold ($e_{th}$) are considered inliers and only these points are used in the IDP gridding process (the IDP power is chosen to be 2). The performance
metric is the Root Mean Square Error (RMSE) between the estimated grid node elevation and the true grid node elevation over all grid nodes. The parameters used in the RANSAC implementation are listed in Table 5.2. The error threshold parameter in this implementation is set to two times the standard deviation of the error in the inlier points as recommended in the original RANSAC formulation [26]. The proposed RSPC method is applied to the same dataset. The probability of validity of each point is initially assumed to be 1 from which the validity of each candidate surface is estimated; this process is iterated until the probability of validity of points converge to within a pre-set threshold ($\epsilon$). The RSPC method requires a greater number of parameters to be set. In this example, the parameters are set according to the guidelines provided in this work (i.e. there was no tuning to improve the results), which are listed in Table 5.2.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>RANSAC with IDP gridding</th>
<th>RSPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius of neighborhood ($r_{th}$)</td>
<td>0.1 to 0.4</td>
<td>0.1 to 0.4</td>
</tr>
<tr>
<td>Number of candidate surfaces ($n_s$)</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Normal distance error threshold ($\epsilon_{th}$)</td>
<td>0.015</td>
<td>NA</td>
</tr>
<tr>
<td>Median normal distance error ($\bar{\epsilon}$)</td>
<td>NA</td>
<td>0.005</td>
</tr>
<tr>
<td>Critical fraction of valid points ($\bar{\gamma}$)</td>
<td>NA</td>
<td>0.4</td>
</tr>
<tr>
<td>Critical fraction of valid surfaces ($\bar{\beta}$)</td>
<td>NA</td>
<td>0.15</td>
</tr>
<tr>
<td>Median horizontal distance ($\bar{d}$)</td>
<td>NA</td>
<td>0.015</td>
</tr>
</tbody>
</table>

Figure 5.1 illustrates a comparison of the RSPC method against the original RANSAC method with IDP gridding for a neighborhood radius of 0.2. The effect of the radius of the neighborhood ($d_{th}$) on the performance of both methods is investigated by varying it from 0.1 to 0.4 and the results are shown in Table 5.3. In all cases the RSPC method performs significantly better when compared to the original RANSAC with IDP gridding.
5.4 Discussion, Future work and Conclusions

As demonstrated in the proof of concept, there are significant improvements possible over the original RANSAC algorithm. For example, in a typical dataset if the radius of the neighborhood is increased, it is likely to contain multiple underlying true surfaces. Incorrect grid node estimates may result from applying the original RANSAC method since only the single best candidate surface in each neighborhood is considered. This is remedied in the proposed RSPC method in which all candidate surface information is used in the grid node
height estimation. However, this does require manual tuning of the parameters used in the algorithm based on a priori knowledge of the properties of the dataset. It should also be noted that since both the methods employ randomized algorithms, the results presented are the average values over a number of runs. The improvement in the RMSE is significant with the usage of the new RSPC method. However, due to the iterative implementation the RSPC method takes longer time to run than the original RANSAC method and the increase in run time is linear in the number of grid nodes, $O(n)$. The rate of convergence of the estimated probability of validity of points for the proof-of-concept example is illustrated in Fig. 5.2 for neighborhoods of different radii. It is observed that convergence is slower for neighborhoods of larger radii. Prior knowledge about the percentage of outliers in the dataset could be used to initialize the probability of validity of points and potentially reduce the convergence time.

![Convergence for Probability of Validity of Points](image)

Figure 5.2: Stair 4 Dataset - Convergence of probability of validity of points

From proof of concept example in this work, it can be observed that the performance of the RSPC method degrades when the radius of the grid node neighborhood is increased and this is due to the presence of multiple true surfaces in the neighborhood. The parameters $\gamma$ and
\( \tilde{\beta} \) can be tuned in such cases in order to account for the existence of multiple true surfaces in larger neighborhoods. Table 5.4 shows the improvement in performance obtained by tuning the parameters \( \tilde{\gamma} \) and \( \tilde{\beta} \) for larger neighborhoods by decrementing their initial estimated values and it can be seen that reducing the values of \( \tilde{\gamma} \) and \( \tilde{\beta} \) improves the performance of the RSPC method. Figure 5.3 illustrates a comparison of the RSPC method utilizing the final tuned parameters, against the original RANSAC method with IDP gridding for a neighborhood radius of 0.4.

<table>
<thead>
<tr>
<th>Parameter Value</th>
<th>Fraction of initial estimate</th>
<th>RSPC RMSE ( d_{th} = 0.3 )</th>
<th>RSPC RMSE ( d_{th} = 0.4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \gamma = 0.4 &amp; \beta = 0.15 )</td>
<td>1</td>
<td>0.0758</td>
<td>0.0831</td>
</tr>
<tr>
<td>( \gamma = 0.3 &amp; \beta = 0.112 )</td>
<td>0.75</td>
<td>0.0738</td>
<td>0.0641</td>
</tr>
<tr>
<td>( \gamma = 0.2 &amp; \beta = 0.075 )</td>
<td>0.5</td>
<td>0.0614</td>
<td>0.0468</td>
</tr>
<tr>
<td>( \gamma = 0.1 &amp; \beta = 0.037 )</td>
<td>0.25</td>
<td>0.0433</td>
<td>0.0420</td>
</tr>
</tbody>
</table>

Figure 5.3: Grid node height estimates for Stair-4 dataset \( r_{th} = 0.2 \). Tuned parameters \( \tilde{\gamma} = 0.1 \) and \( \tilde{\beta} = 0.0375 \)

In the future the proposed method will be evaluated on real world road surface datasets.
Areas for improvement of the method include a method for adaptively estimating the optimal values of the tuning parameters given approximate initial estimates for their values. Another potential area of investigation is an analytical solution for the probability of validity of points and surfaces in order to improve the run time of the algorithm.

In conclusion, it has been demonstrated that the application of the proposed axiomatic probabilistic approach produces accurate and robust estimates of grid node heights from a dataset that contains a significant percentage (60%) of outliers. In the proof-of-concept example, the RSPC method performs significantly better than the original RANSAC method with IDP gridding. The improvement in performance can be attributed to the use of soft switching logistic functions for classification of neighborhood points and also the retention of information regarding all candidate surfaces in the neighborhood. The RSPC method can be used to refine and grid measured road surface datasets which in turn can be used for modeling and simulation of vehicles.
Chapter 6

Summary, Conclusions and Future work

6.1 Summary

The main objective of this dissertation was to investigate the techniques for improving the accuracy of Mobile Road Mapping Systems (MRMS). Based on existing literature, theoretical concepts have been formulated specifically to address the mobile road mapping problem and the concepts have been evaluated in a simulated testing environment. The following categorical objectives have been achieved in this work:

1. Development of a theoretical framework which applies the principles of Simultaneous Localization and Mapping (SLAM) to the mobile road mapping problem in order to improve the accuracy of MRMS.

2. Utilization of a dynamic model of the mapping system within the developed framework to further improve the accuracy of the MRMS.

3. Development and simulation testing of a two stage solution to solve the mobile road mapping SLAM problem: A filtering solution that uses an Extended Kalman Filter (EKF) followed by a smoothing solution that uses a factor graph formulation of the problem.

6.2 Conclusions

The main conclusions from the work in this dissertation are:

- Using the principles of SLAM for the mobile road mapping problem along with knowledge of the dynamics of the host vehicle leads to an overall improvement in the accuracy of the system.

- From the simulation results in Chapters 3 and 4 it can be seen that the two stage solution (EKFSLAM followed by SAM) yields a significant improvement in the accuracy of the localization and mapping solutions when compared to current state-of-the-art methods.

- The improvement in accuracy is even greater in special cases when the MRMS is able to close the loop by returning to map a particular section of the road surface. For the loop closure case the improvement in the accuracy of the final mapping solution can reach up to 80%.

- The overall time required for computing the SLAM solution is however many times higher than the computation time taken by current state-of-the-art methods.

- In Chapter 5 a novel, axiomatic method for identifying outliers and estimating grid node heights in road surface datasets is developed. The simulation results demonstrate that the method is effective for datasets containing up to 60% outliers.
6.3 Future Work

The following research topics are proposed as areas of future work:

- Extension of the framework developed in this work to a prototype MRMS would require relaxing the simplifying assumptions which neglected the rotation of the earth and the curvature of the earth surface, and other assumptions about the sensor error models.

- The simulation testing in this work uses a simplified model of the mapping system with limited degrees of freedom. Future work will include testing with a more complex dynamic model with complete degrees of freedom. The accuracy of the SAM method depends on the fidelity of the dynamic model of the system and a priori knowledge of the parameters of the dynamic model. In this work a simplified dynamic model of the system whose parameters are known a priori has been used to demonstrate the effectiveness of the framework. In the future it is proposed to extend the framework using more complex system dynamic models and also to estimate the parameters of the dynamic model as part of the framework. The model used to predict the vertical tire-road interaction force in this work is a simple single point contact spring tire model. In the future it is proposed to use a more detailed tire force model that is more suitable for high frequency road inputs.

- The filtering based solution to the online SLAM problem uses an Extended Kalman Filter (EKF) which involves linearizing the sensor measurement models and system dynamic model. The Unscented Kalman Filter (UKF) does not linearize the system models and thus a logical recommendation is to study its application to the online road mapping SLAM problem.

- Future work will address the issue of the high computation time taken by the SLAM
methods. For the filtering based EKFSLAM method the computation time can be reduced by lowering the mapping sensor measurement frequency and by reducing the number of grid nodes in the augmented state vector. For the smoothing based SAM method the computation time can be reduced by using a parallel processing model. In the future it is also proposed to evaluate the use of Incremental Smoothing and Mapping (iSAM) [38] for the mobile road mapping problem, wherein the sensor measurements are processed in real-time as they become available.

- The solutions to the SLAM problem developed in this work are sensitive to the presence of outliers in the mapping sensor measurements and even a tiny fraction of outliers can have a detrimental effect on the accuracy of the solution. This is because mapping sensor measurement model assumes a Gaussian distribution over the range measurement errors whereas a heavy-tailed distribution is more suitable to model mapping sensor measurements with outliers. It is proposed in the future to use a non-Gaussian heavy tailed distribution to model the mapping sensor measurement errors.
Bibliography


on Fluid Power and Motion Control, pages 375–381. American Society of Mechanical Engineers, 2011.


[87] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application


Appendices
Appendix A

Simulated Mobile Road Mapping System (MRMS)

A.1 Proof-of-concept mobile road mapping system

A simulation based proof-of-concept Mobile Road Mapping System (MRMS) is used to evaluate the theoretical framework developed in this work. The simulated system resembles the MRMS at the Vehicle Terrain Performance Laboratory at Virginia Tech. The system comprises of road scanning lasers for mapping the road surface and a GNSS/INS system for localization. All of the sensors and the auxiliary equipment are mounted inside a trailer which acts as the host vehicle for the MRMS. The trailer in turn is towed by a Sports Utility Vehicle (SUV) through a hitch. The dynamics of the host vehicle (trailer) are modeled by a simple 4-Degree of Freedom (DOF) dynamic model, which are:

- Translation of the sprung mass along x and z axes of \textit{w-frame}
- Pitching of the sprung mass about y-axis of \textit{w-frame}
- Vertical motion of unsprung mass in \textit{w-frame}

A schematic of the proof-of-concept model is shown in Fig. A.1.

The parameters of the dynamic model are:
• $m_s$ - Sprung mass (kg)

• $m_u$ - Unsprung mass (kg)

• $\mathcal{I}_{3\times3}$ - Inertia of the sprung mass.

• $k_s$ - Suspension spring stiffness $N/m$

• $c_s$ - Suspension damping coefficient $N - s/m$

• $k_t$ - Tire spring stiffness $N/m$

Based on the geometry of the vehicle, two position vectors are defined in the $b$-frame:

1. $p_h^b$ - Position vector of the point of application of hitch forces relative to the C.M. of the sprung mass with components resolved in the $b$-frame.
2. \( p_{\text{bus}} \) - Position vector of the point of application of the suspension force relative to the C.M of sprung mass with components resolved in the \( b\)-frame

Inputs to the model:

The inputs to the simulated dynamic model are the vertical tire-road profile interaction force and the forces acting at the hitch of the trailer. The forces at the trailer hitch are modeled as unknown inputs and are indirectly estimated by modeling them as stochastic processes.

States of the model

The elements chosen to represent the state of the dynamic model are:

- Position vector of sprung mass C.M in the world frame denoted as \( p^w = \begin{bmatrix} p_x \\ p_z \end{bmatrix}^T \)
- Velocity vector of sprung mass in the world frame denoted as \( v^w = \begin{bmatrix} v_x \\ v_z \end{bmatrix}^T \)
- Rotation angle of the sprung mass body frame w.r.t world frame which is denoted as \( \theta \)
- Angular velocity of the sprung mass denoted as \( \omega \)
- Unsprung mass vertical position and velocity in the \( w\)-frame - \( z^w_u \) and \( \dot{z}^w_u \)

In addition, the forces at the trailer are also included in the state vector and the final augmented state vector is denoted as

\[
\mathbf{s} = \begin{bmatrix} p^w \\ v^w \\ \theta \\ \omega \\ z^w_u \\ \dot{z}^w_u \\ f^h \end{bmatrix}^T \quad (A.1)
\]
### A.2 Dynamic model equations of motion

The equations of motion of the dynamic model are derived using Newtonian mechanics.

- **Sprung mass dynamics:**
  
  The forces acting on the sprung mass are the suspension force and the forces at the hitch of the trailer. The suspension force acts in the vertical direction of the $w$-frame. The forces at the hitch act along the axes of the $b$-frame. Using Newton’s Second Law of motion, the forces acting on the sprung mass are used to compute the rate of change of linear momentum as

\[
m \cdot \ddot{v}^w = \sum F^w = F^w_s + C_b^w F_h^b \tag{A.2}
\]

where,

- $F^w_s$: Suspension force vector with components resolved in $w$-frame
- $C_b^w$: Direction Cosine Matrix (D.C.M) from $b$-frame to $w$-frame.

The D.C.M from $b$-frame to $w$-frame is given by

\[
C_b^w = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix} \tag{A.3}
\]

The rate of change of the position vector of the sprung mass C.M is determined from the velocity vector as:

\[
\dot{p}^w = v^w \tag{A.4}
\]

The rate of change of angular velocity of the sprung mass is computed as

\[
\mathcal{T} \dot{\omega} = M^b \tag{A.5}
\]
where,

\[ M^b = \sum M + \sum p \times F \]  \hspace{1cm} (A.6)

The rate of change of the pitch angle (θ) is then computed using the angular velocity as

\[ \dot{\theta} = \omega \]  \hspace{1cm} (A.7)

**Computing suspension forces:**

In this work a linear spring rate and damping coefficient are used to compute the magnitude of the suspension force as:

\[ F_s = -k_s(p_z - a\sin\theta - z_u - z_e) - c_s(v_z - a\omega - \dot{z}_u) \]  \hspace{1cm} (A.8)

where,

\[ z_e \] - Equilibrium length of suspension spring under the action of the weight of the vehicle.

**Unsprung mass dynamics:**

The forces acting on the unsprung mass are the suspension force and the vertical tire spring force. Applying Newton’s second law of motion

\[ m_u \ddot{z}_u = F_s - F_t \]  \hspace{1cm} (A.9)
where,

\[ F_t = k_t(z_u^w - \xi_t) \] - Vertical tire spring force