Localization and Mapping for Autonomous Driving:

Fault Detection and Reliability Analysis

Shuran Zheng

A thesis in fulfillment of the requirements for the degree of
Doctor of Philosophy

School of Civil and Environmental Engineering
Faculty of Engineering

The University of New South Wales

March 2021

Thesis Title and Abstract | Declarations | Inclusion of Publications Statement | Corrected Thesis and Responses

Thesis Title

Localization and Mapping for Autonomous Driving: Fault Detection and Reliability Analysis

Thesis Abstract

Autonomous driving has advanced rapidly during the past decades.A key requirement to ensure the safety of automated driving is the ability of reliable localization and navigation. A map with rich environment information is essential to support autonomous driving system to meet these high requirements. Therefore, multi-sensor-based localization and mapping methods are studied in this Thesis.

A full quality control scheme to guarantee the reliability and to detect outliers in localization and mapping solutions is still lacking in this area.In this research, an extended Kalman filter and smoother based quality control scheme is investigated and has been successfully applied for different localization and mapping scenarios. The major contributions of this research are:

1. The equivalence between least squares and smoothing is discussed, and an extended Kalman filter-smoother quality control method is developed, which can be used to deal with system model outlier with detection, and identification, and to analyse the system quality. Relevant mathematical models of this quality control method have been developed to deal with issues such as singular system.
2. Quality control analysis for the integration of GNSS and IMU system is conducted. The results indicate the smoothing based positioning methods are more accurate, and reliable. Post Processing Kinematic (PPK) method can provide more stable positioning results for mapping.
3. Mathematical modelling and quality control aspects for online simultaneous localization and navigation (SLAM) are examined. A smoother based offline SLAM method is investigated with quality control. Geometry analysis for the SLAM system has been done according to the quality control results for both outdoor and indoor datasets.
4. A least squares based localization method is proposed that treats the High-Definition map as a sensor source. Geometry analysis is undertaken with the quality measures to analyse the influence of the geometry upon the estimation solution and the system quality.
5. A GNSS/INS aided LiDAR mapping and localization procedure is developed. A high-density map is generated offline, then, LiDAR-based localization can be undertaken online with this pre-generated map. Quality control is conducted for this system. The results demonstrate that the LiDAR based localization within map can effectively improve the accuracy and reliability compared to GNSS/INS only system, especially during the period that GNSS signal is lost.

Thesis Title and Abstract | Declarations | Inclusion of Publications Statement | Corrected Thesis and Responses

ORIGINALITY STATEMENT

☑ I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.

COPYRIGHT STATEMENT

☑ I hereby grant the University of New South Wales or its agents a non-exclusive licence to archive and to make available (including to members of the public) my thesis or dissertation in whole or part in the University libraries in all forms of media, now or here after known. I acknowledge that I retain all intellectual property rights which subsist in my thesis or dissertation, such as copyright and patent rights, subject to applicable law. I also retain the right to use all or part of my thesis or dissertation in future works (such as articles or books).

For any substantial portions of copyright material used in this thesis, written permission for use has been obtained, or the copyright material is removed from the final public version of the thesis.

AUTHENTICITY STATEMENT

☑ I certify that the Library deposit digital copy is a direct equivalent of the final officially approved version of my thesis.

UNSW is supportive of candidates publishing their research results during their candidature as detailed in the UNSW Thesis Examination Procedure.

Publications can be used in the candidate's thesis in lieu of a Chapter provided:

- The candidate contributed **greater than 50%** of the content in the publication and are the "primary author", i.e. they were responsible primarily for the planning, execution and preparation of the work for publication.
- The candidate has obtained approval to include the publication in their thesis in lieu of a Chapter from their Supervisor and Postgraduate Coordinator.
- The publication is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in the thesis.

☑ The candidate has declared that **some of the work described in their thesis has been published and has been documented in the relevant Chapters with acknowledgement**.

A short statement on where this work appears in the thesis and how this work is acknowledged within chapter/s:

> my results from paper 'High definition map-based vehicle localization for highly automated driving: Geometric analysis' published in 'International Conference on Localization and GNSS (ICL-GNSS)' is contained in Chapter 6. I'm the primary author for this paper. I plan and prepared the work for this publication. I wrote the code and conducting testing and analysis for this publication. The paper was written by me. Acknowledgement has been made for my supervisor who is another author.
> My literature review is partially comprised of a review paper that I wrote and to be submitted to the Journal of Applied Geodesy. Part of results in Chapter 7 is also comprised of this review paper. I'm the primary author for this paper and have done the planning, coding, experimenting, analysing, writing of this paper. I have acknowledge other authors who did revision and gave me comments for this paper in my Acknowledgements section.

Candidate's Declaration

I declare that I have complied with the Thesis Examination Procedure.

Abstract

Autonomous driving has advanced rapidly during the past decades and has expanded its application for multiple fields, both indoor and outdoor. One of the significant issues associated with a highly automated vehicle (HAV) is how to increase the safety level. A key requirement to ensure the safety of automated driving is the ability of reliable localization and navigation, with which intelligent vehicle/robot systems could successfully make reliable decisions for the driving path or react to the sudden events occurring within the path. A map with rich environment information is essential to support autonomous driving system to meet these high requirements. Therefore, multi-sensor-based localization and mapping methods are studied in this Thesis.

Although some studies have been conducted in this area, a full quality control scheme to guarantee the reliability and to detect outliers in localization and mapping systems is still lacking. The quality of the integration system has not been sufficiently evaluated. In this research, an extended Kalman filter and smoother based quality control (EKF/KS QC) scheme is investigated and has been successfully applied for different localization and mapping scenarios. An EKF/KS QC toolbox is developed in MATLAB, which can be easily embedded and applied into different localization and mapping scenarios. The major contributions of this research are:

a) The equivalence between least squares and smoothing is discussed, and an extended Kalman filter-smoother quality control method is developed according to this equivalence, which can not only be used to deal with system model outlier with detection, and identification, can also be used to analyse, control and improve the system quality. Relevant mathematical models of this quality control method have been developed to deal with issues such as singular measurement covariance matrices, and numerical instability of smoothing.
b) Quality control analysis is conducted for different positioning system, including Global Navigation Satellite System (GNSS) multi constellation integration for both Real Time Kinematic (RTK) and Post Processing Kinematic (PPK), and the integration of GNSS and Inertial Navigation System (INS). The results indicate PPK method can provide more reliable positioning results than RTK. With the

proposed quality control method, the influence of the detected outlier can be mitigated by directly correcting the input measurement with the estimated outlier value, or by adapting the final estimation results with the estimated outlier's influence value.

c) Mathematical modelling and quality control aspects for online simultaneous localization and mapping (SLAM) are examined. A smoother based offline SLAM method is investigated with quality control. Both outdoor and indoor datasets have been tested with these SLAM methods. Geometry analysis for the SLAM system has been done according to the quality control results. The system reliability analysis is essential for the SLAM designer as it can be conducted at the early stage without real-world measurement.

d) A least squares based localization method is proposed that treats the High-Definition (HD) map as a sensor source. This map-based sensor information is integrated with other perception sensors, which significantly improves localization efficiency and accuracy. Geometry analysis is undertaken with the quality measures to analyse the influence of the geometry upon the estimation solution and the system quality, which can be hints for future design of the localization system.

e) A GNSS/INS aided LiDAR mapping and localization procedure is developed. A high-density map is generated offline, then, LiDAR-based localization can be undertaken online with this pre-generated map. Quality control is conducted for this system. The results demonstrate that the LiDAR based localization within map can effectively improve the accuracy and reliability compared to the GNSS/INS only system, especially during the period that GNSS signal is lost.

Acknowledgments

Firstly, I would like to express my deepest appreciation to my supervisor A. Prof. Jinling Wang, for his advice, guidance and encouragement throughout my entire Ph.D. study. My sincere thanks also go to my co-supervisor A.Prof. Samsung Lim, who gives me some guidance on how to start research studies and how to search and treat with the research literature. Spatial thanks go to E.Prof. Chris Rizos, A. Prof. Ahmend El-Mowafy and Dr. Weidong Ding for their invaluable advices and support during a joint research activity sponsored by an Australian Research Council.

I would like to thank my panel of progress review: Prof. Arnaud Castel; Prof. Lucy Marshall; Prof. Linlin Ge; Dr. Johnson Xuesong Shen; Dr. Divya Jayakumar Nair; Dr. Bruce Harvey; Dr. Craig Roberts; Dr. Yincai Zhou for their insightful comments and encouragement. Besides, I need to thank Dr. Yincai Zhou and Mr. Peter Mumford for providing experimental equipment and operation assistance.

I am grateful to the UNSW Graduate Research School for awarding me the Australian Postgraduate Award, and thanks the Faculty of Engineering and the School of Civil and Environment Engineering for their carious support for my Ph.D. studies at UNSW.

I wish to thank other members of the research team: Mr. Zeyu Li, Mr. Yu Sun, Mr. Wenhao Zhang, Mr. Kye Chen, Mrs. Jingyuan Diao for helpful research discussions and other helps and encouragements making my four years of research student life at UNSW joyful.

I also would like to show my appreciation to my parents Mr: Yuanhui Zheng and Mrs. Jinyi Zheng, my husband: Mr. Letian Xu and my father-in-law Mr. Jiangtao Xu and my mother-in-law Mrs. Xiuyun Lv for their support in my life and in my research. Finally, I wish to thank my dear pet kittens Yabbi and MiuMiu, I really appreciate their accompany during my hard period.

Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Background

Autonomous (also called self-driving, driverless, or robotic) vehicle is now at the heart of academic and industrial research. It has been predicted that the fully autonomous vehicles will be available for sale and become the major part of total vehicle sales in the next decades (Litman, 2015). The promotion of autonomous vehicles can bring many advantages, for instance, it can service the injured people, reduce driver's stress and costs, increase road safety, increase road capacity by encouraging sharing cars, enhance fuel efficiency, and eliminate the need for conventional public transit services (Litman, 2015; Katrakazas et al., 2015). As human error is the primary reason for road traffic accidents, one main role of the advanced assistance system (ADAS) for autonomous vehicles is to replace humans to ensure vehicle and pedestrian safety.

A typical autonomous vehicle system contains four key parts: localization, perception, planning, controlling (Figure 1.1). Positioning is a process that obtaining an object (e.g. moving or static objects)'s coordinates with respect to a given coordinate system. The coordinate system can be a local coordinate system or a geodetic datum such as WGS84. Localization is a process of estimating the object's pose (position and attitude) related to a reference frame or a specific map while the perception system detects road environment around the host vehicle and identifies interested objects such as pedestrians, other vehicles, traffic light and sign, etc.

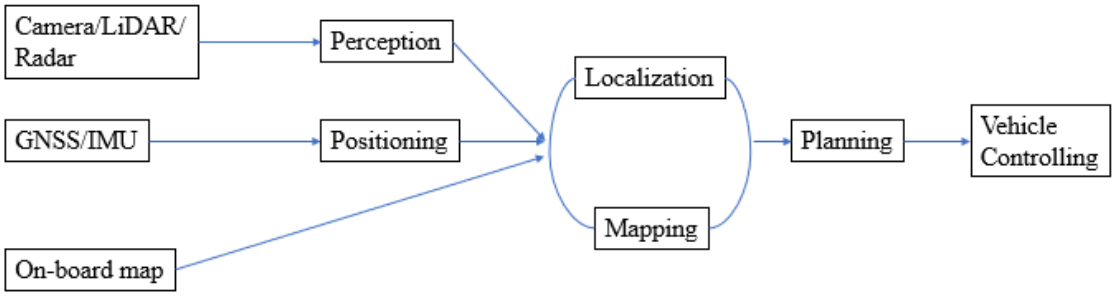


Figure 1.1 Functional components of an autonomous driving system.

By achieving the coordinates of the surrounding environment and associating the location information of the surrounded entities, a map can be generated, this process is named as Mapping.

Path planning is a step that utilizing the localization, mapping, and perception information to calculate and determine the optimal path in following driving epochs, guiding the moving platform (e.g. automated vehicle) from one location to another location. This plan is then converted into action with controlling part, such as brake control before the detected traffic light or intersection line. All these mentioned parts are closely related. Accurate localization and perception information is essential for obtaining correct planning and controlling.

1.2 Localization and Mapping for Autonomous Driving

1.2.1 Sensors for Localization and Mapping

Sensors, such as GPS receiver and Inertial Navigation System (INS), laser scanner, camera, odometer, and sonar, can be ‘eyes’ and ‘ears’ of a car. They will help build road 3D map, detect obstacles, and support advanced driver assistance systems (ADAS). However, no sensor is perfect to suit all the applications. They have their own characteristics and application range.

GPS or GNSS and INS are the traditional positioning methods and have been thoroughly studied. Nowadays, GNSS can provide very accurate absolute position information and millimeter to centimeter level relative ranging information (Kaplan and Hegarty, 2006).

However, GNSS alone is not enough for the autonomous driving application. GNSS error occurs when the GNSS signal is interrupted by heavy vegetation, urban canyon, tunnel, etc. Therefore, it is of great importance to locate the autonomous vehicles accurately when there is an outage of the GNSS signal. INS system can be used to estimate position information when the GNSS signal is lost, and it can provide velocity and altitude information. Inertial measurement bias needs frequent corrections, which can be done when the GNSS signal comes back. Therefore, the combination of GNSS and INS systems can complement each other and improve their performances on vehicle localization.

However, GNSS/INS systems cannot percept the environment around the vehicle. As highly automated driving requires both the positioning information of the host vehicle and that of the surrounding environment entities, the perceptive sensors are often used as the main sensors for the purpose of mapping and navigation. LiDAR, as a perceptive sensor, which has been widely studied currently, can achieve the 3D point cloud directly and map the environment precisely with the aid of GPS and INS. For instance, Suganuma et al. (2015) use LiDAR and dead-reckoning position information to successfully localise the vehicle on urban road up to 13km with an accuracy of centimetre level. However, this system's application is limited for commercial vehicles due to its high cost and large size.

Another promising perceptive sensor that can be used on an autonomous vehicle is camera, which, compared to LiDAR system, shows lower accuracy and has numerous error sources (Fuentes-Pacheco et al., 2015), but is much cheaper, with smaller size, requires less maintenance and less energy. Vision-based system can provide abundant environment information which is in vision form and is similar to what human eyes can percept, and can be fused with other sensors to give indications of the detected features.

The mentioned onboard perceptive sensor technologies, such as camera, LiDAR, Radar, and ultra-sonic, and the infusion technology, have all been deeply studied. Some of them can achieve promising results for precise positioning and navigation. However, one big problem of these sensors is their limited sensing range. The autonomous vehicles need to recognise and react to the events that happen in a range of 200 meters around it or further (Seif and Hu, 2016), which is far beyond the range of currently used sensors. LiDAR can

capture features with a reliable resolution at the distance of up to100 meter (Seif and Hu, 2016), camera is usable to detect closer environment about 30 meters and it needs a combination of multi-camera to acquire 360 degrees field of view, Radar has large sensor distance (200 m), however, it provides less information about the features and will be interrupt by the weather such as rain. Therefore, these perceptive sensors cannot satisfy the requirement of sensor range for autonomous driving. It is necessary to integrate different kinds of sensor to counteract the problems of each other.

1.2.2 High-definition Map

A map with rich road environment information is essential for the sensors mentioned above to provide accurate and robust localization and perception. Pre-stored road information makes autonomous driving robust to the changing environment and road dynamics. The recognition range requirement can be satisfied since an onboard map can provide timely information of the road network. Map-based localization and navigation have been studied using different types of map information. Google Map is one example as it offers worldwide map information, including images, topographic details and satellite images (He and Lai, 2010), and it is available via mobile phone and vehicle apps. However, the use of maps will be limited by the maps' accuracy, and in some selected areas the map's resolution may be restricted. Hosseinyalamdary et al. (2015) considered low accuracy maps for navigation by combining data from other sensors. They detected moving objects using LiDAR data, and utilized a GNSS/IMU system with a coarse open source GIS map. Their results show their fusion technique can successfully detect and track moving objects. A precise curb-map based localization method that uses a 3D-LiDAR sensor and a high precision map is proposed by Wang et al. (2017). However, this method will fail when curb information is lacking or obstructed.

Recently, so-called "high definition" (HD) maps have received considerable interest in the context of autonomous driving since they contain very accurate, and large volumes of road network information (Liu et al., 2020). According to some major players in the commercial HD map market, 10-20 cm accuracy has been achieved (HERE, 2017; TomTom, 2016), and it is predicted that in the next generation HD maps a few centimeter accuracy will be reached. Such maps contain considerable information on road

environment, not only the static road entities, such as road geometry (curvature, grades, etc), but also some traffic management information, such as traffic signs, traffic lights, speed limits, road markings, and so on. The autonomous car can use the HD map to precisely locate the host-car within the road lane and estimate the relative location of the car with respect to road objects by matching the landmarks, which are recognized by onboard sensors (such as cameras), with the pre-stored information within the HD map.

Therefore, maps, especially HD maps, play several roles in support of autonomous driving, are able to meet the stringent requirements of accuracy, precision, recognition ranging, robust and information richness. The “map” application for autonomous driving is facilitated by techniques such as Simultaneous Localization and Mapping (SLAM).

1.2.3 Integration of multi-sensor for localization and mapping

There are many studies that focus on the integration of the multi-sensor for the dynamic system. Kalman filter (KF) and smoothing can be used to combine GNSS/INS. This initial solution of trajectory can then be introduced to bundle-adjustment to estimated camera sensor position and attitude, this process has no feedback to the navigation solution (Rouzaud and Skaloud, 2011). SLAM is another approach that can integrate the navigation solution (inertial odometry or GNSS/INS) with the optical observation from LiDAR or camera, which may feed back the trajectory estimation to the navigation solution. Some traditional SLAM approaches are also KF based. For most localization and mapping problem, the platform will move, and the perceptive sensors will match among same features observed at different times, which can be used as constraint to the platform trajectory. However, this kind of spatial constraint between the observed features at different epochs is indirectly observed at KF updating step (Rouzaud and Skaloud, 2011). Some studies proposed dynamic networks to deal with this integration system with time dependent and spatial dependent parameters. According to Colomina and Blazquez (2005), the static and dynamic observation equations were combined to build up a time dependent network or a dynamic network. Optimal estimation is performed to solve this network. Its work shows that a particular type of dynamic network can be solved with Kalman filtering and smoothing algorithm (Colomina and Blazquez, 2005). In the work of Rouzaud and Skaloud (2011), the performance of KF, KS, dynamic

network is compared for a GNSS/INS/ LiDAR integration system. The dynamic network is solved by least squares optimization. Results show that the dynamic network's solution will be corresponded to the smoother's for case of GNS/INS integration or inertial navigation only. However, if LiDAR observation is integrated and an additional cross-time constraint with tie-features is added to the dynamic network, the network solution will be improved, especially when GNSS is denied. For the field of SLAM, network-based approach is also well established (Strasdat et al., 2010) and more interest is towards the inertial SLAM. Another study (Angelats and Colomina, 2014) solves the orientation and calibration of camera, laser, and camera-laser in a single, combined network adjustment. The traditional bundle adjustment is replaced by dynamic network (Cucci et al., 2017). Cucci et al (2017) combined raw inertial observation directly to the bundle-adjustment by driving a dynamic network in a form of factor-graph, and has demonstrated the superior of the proposed method, especially in challenging scenarios.

1.3 Simultaneous Localization and Mapping

SLAM is a process by which a moving platform builds a map of the environment and uses it to deduce its location simultaneously. SLAM, which is widely used in the robotic field, has been demonstrated (Bresson et al., 2017, Kuutti et al., 2018) as being applicable for autonomous vehicle driving since it can support not only accurate map generation but also online localization within a previously generated map.

Since initially introduced in 1986 (Smith and Cheeseman, 1986), a variety of SLAM techniques have been developed. SLAM has its conceptual roots in geodesy and geospatial mapping (Agarwal et al., 2014).

In general, there are two types of approaches to SLAM estimation: filter-based and optimization-based. Both approaches estimate the vehicle pose states and map states at the same time. The vehicle pose includes 3D or 2D vehicle position, but sometimes also velocity, orientation or attitude, depending on the sensor(s) used and on the application(s). In this study X_k represents the vehicle pose at time k. m is the map that consists of stored landmarks (*f1-f4*) with their position states. u_k is the control inputs representing the

vehicle motion information between time epochs k-1 and k (which can be acquired from vehicle motion sensors such as wheel encoders or an inertial sensor). At some time epochs such as epoch k, the onboard sensors (such as camera, LiDAR, radar) will perceive the environment and detect one or more landmarks. The relative observations between the vehicle and all the observed landmarks are denoted as Z_k. With this information, the vehicle pose and the map states can be estimated.

Figures 1.2 and 1.3 illustrates two general SLAM implementations: online SLAM and offline SLAM. The rectangle with blue background represents the state variables that are estimated in these two implementations. It can be seen that for online SLAM, the current vehicle pose X_{k+2} is estimated and the map is generated and updated with the most recent measurements (u_{k+2} and Z_{k+2}). In the case of the offline SLAM implementation, the vehicle's trajectory is updated together with the whole map. All the available control and observation measurements will be utilized for the offline SLAM estimation. Many filter-based and optimization-based SLAM methods have been developed.

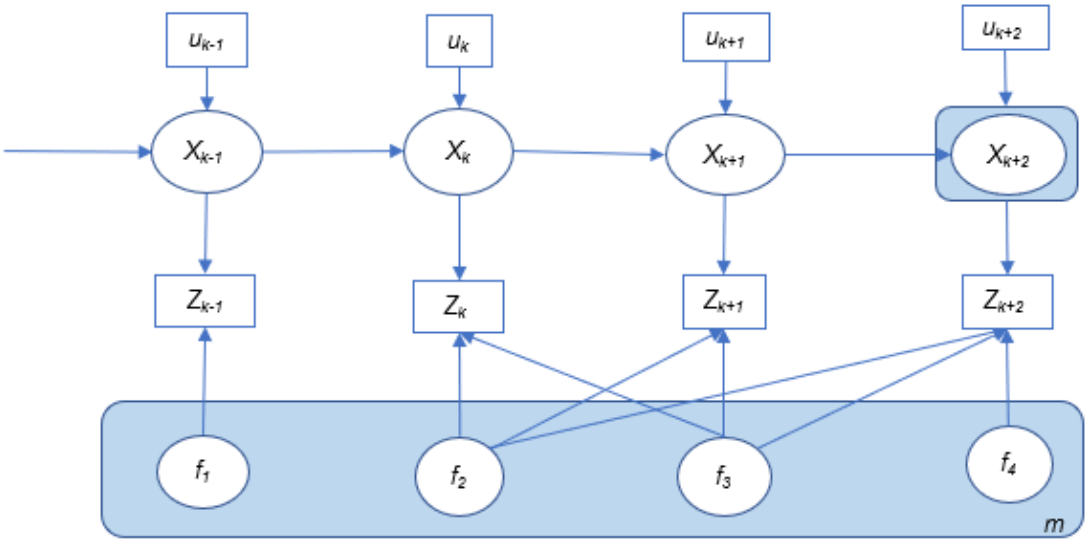


Figure 1.2 Description of online SLAM

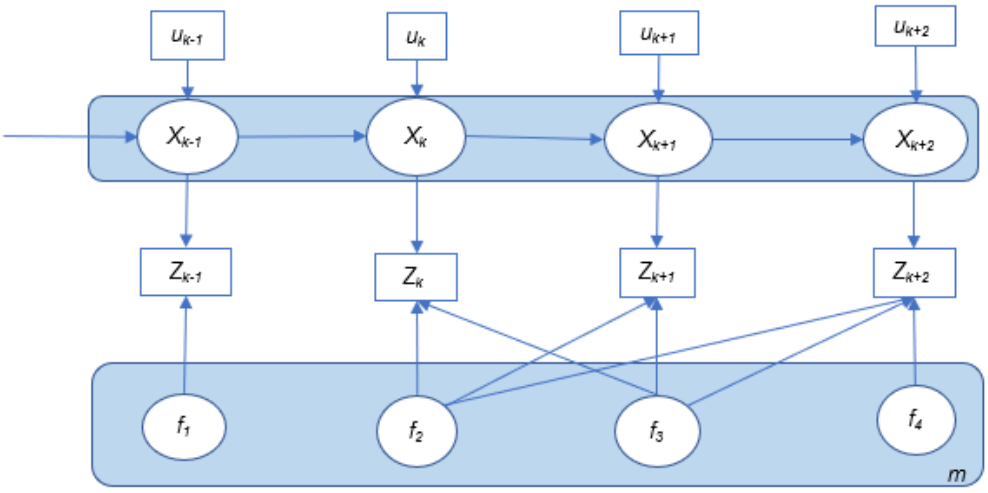


Figure 1.3 Description of offline SLAM

1.3.1 Online SLAM

Filter-based SLAM recursively solves the SLAM problem in two steps. Firstly, the vehicle and map states are predicted with processing models and control inputs. In the next step, a correction of the predicted state is done with the current sensor observations. Therefore, the filter-based SLAM is suitable for online SLAM.

Extended Kalman Filter based SLAM (EKF-SLAM) is a standard solution for the SLAM problem. It is derived from Bayesian filtering, in which all variables are treated as Gaussian random variables. It consists of two steps: time update (prediction) and measurement update (filtering). At each time epoch the measurement and motion models are linearized (using the current state with the first-order Taylor expansion). However, since the linearization is not made around the true value of the state vector, but around the estimated value (Dissanayake, 2011), the linearization error will accumulate and cause divergence of estimation. Therefore, inconsistency can occur.

Another issue related to EKF-SLAM is the continuous expansion of map size, making the quadratic calculation process of large-scale SLAM impractical. For autonomous driving, the complex road environment and long driving period will introduce a large number of features, which makes real-time computation unavailable. Many algorithms have been developed to improve computational efficiency. For example, the Compressed Extended

Kalman Filter (CEKF) (Guivant and Nebot, 2001) algorithm can significantly reduce computations by focusing on local areas and then extending the filtered information to the global map. Algorithms with sub-maps have also been used to address the computation issues (Williams, 2001; Bailey, 2002; Paz et al., 2008, Chli and Davison, 2009). New blank map is used to replace the old map when the old one reaches a predefined map size. A higher-level map is maintained to track the link between each sub-map.

There are some other filter-based SLAM approaches, such as some variants of Kalman filter. One of them, the Information Filter (IF), is propagated with the inverse form of the state error covariance matrix, which makes this method more stable (Thrun and Liu, 2005). This method is more popular in multi-vehicle SLAM than in single-vehicle systems.

Another class of filter-based SLAM techniques are the Particle Filters (PF) which have become popular in recent years. PF execute Sequential Monte-Carlo (SMC) estimation by a set of random point clusters (or particles) representing the Bayesian posterior. The Rao-Blackwellized Particle Filter was proposed by Grisetti (2007). Fast SLAM is a widespread implementation which treats the robot position distribution as a set of Rao-Blackwellized particles, and uses an EKF to maintain local maps. In this way, the computational complexity of SLAM is significantly reduced. Real-time application is possible with Fast SLAM (Eade and Drummond, 2006), making online SLAM possible for autonomous driving. Another advantage of FastSLAM over EKF is that the particle filters can cope with non-linear motion models (Thrun et al., 2004). However, according to Durrant-Whyte and Bailey (2006) and Bailey et al., (2006), FastSLAM suffers degeneration since it cannot forget the past. If marginalizing the map and when resampling is performed, statistical accuracy is lost.

1.3.2 Offline SLAM

Full offline SLAM estimates all the vehicle pose and map states using the entire sensor data. Similar to filter-based SLAM, a so-called “full SLAM” system consists of two main parts: the front-end and the back-end. In the front-end step, the SLAM system extracts the constraints of the problem with the sensor data, for example, by performing feature

detection and matching, motion estimation, loop closure detection, etc. Nonlinear optimization is then applied to acquire the maximum likelihood estimation at the back-end.

Graph SLAM is one of the main classes of full SLAM which uses graphical structure to represent the Bayesian SLAM. All the platform poses along the whole trajectory and all the detected features are treated as nodes. Spatial constraints between poses are encoded in the edges between the nodes. These constraints result from observations, odometry measurements and loop closure constraint. After the graph construction, graph optimization is applied to optimize the graph model of the whole trajectory and map. To solve the full optimization and calculate the Gaussian approximation of the posterior, many methods can be used, such as Gauss-Newton or Levenberg-Marquardt (Press, et al., 1992).

One advantage of graph-based SLAM over EKF SLAM is that its covariance matrix size and update time are constant after generating the graph. In contrast, for EKF SLAM these will be quadratic in relation to the number of map states and can grow dramatically. Therefore, graph SLAM has become popular for building large-scale maps. Reducing the optimization step's computational complexity has become one of the leading research topics for practical implementations of the high-dimensional SLAM problem. The key to solving the optimization step efficiently is the sparsity of the normal matrix. The fact that each measurement is only associated with a very limited number of variables makes the matrix very sparse. With Cholesky factorization and QR factorization methods, the information matrix and measurement Jacobian matrix can be factorized efficiently, and hence the computational cost can be significantly reduced. Several algorithms have been proposed, such as TORO and g2o. The sub-map method is also a popular strategy to solving the large-scale problem (Wagner, et al., 2014; Ni et al., 2007; Huang, et al., 2009; Pinies, et al., 2009; Ho, et al., 2018). The sub-maps can be optimized independently and be relative to a local coordinate frame. The sub-map coordinates can be treated as pose nodes, linked with motion constraints or loop closure constraints. Thus, a global pose graph is generated. In this way, the computational complexity and update time will be improved.

Smoothing and Mapping (SAM), another typical optimization-based SLAM algorithm, is a type of nonlinear least squares problem. Such a least squares problem can be solved incrementally by Incremental Smoothing and Mapping (iSAM) (Kaess, et al., 2008) and iSAM2 (Kaess, et al., 2012). Online SLAM can be obtained with incremental SAMs as they avoid unnecessary calculations with the entire covariance matrix. iSAM2 is more efficient as it uses a Bayes tree to get incremental variable re-ordering and fluid re-linearization.

SLAM++ is another incremental solution for nonlinear least squares optimization-based SLAM which is very efficient. Moreover, for online SLAM implementations, fast state covariance recovery is essential for data association, obtaining reduced state representations, active decision-making and next best-view (Ila et al., 2015; Ila et al., 2017). SLAM++ has an advantage as it allows for incremental covariance calculation which is faster than other implementations (Ila et al., 2017).

Table 1 is a summary of the characteristics of some typical SLAM techniques. Note that Graph SLAM utilizes all available observations and control information and can achieve very accurate and robust estimation results. However, it can only be used for offline applications and its performance relies on a good initial guess. Filter-based SLAM is more suitable for small-scale environments when used for online estimation. The incremental optimization method that can do incrementally updating so as to provide an optimal estimation of a large-scale map with very high efficiency.

Table 1.1 Characteristics of some typical SLAM techniques

SLAM	Type	Advantages	Disadvantages
EKF SLAM	Bayesian filter	• Mature method, widely studied • Uncertainty is estimated	• Suffers from linearization errors • No re-linearization step • Needs huge memory and computational resources for large maps

IF SLAM	Bayesian filter	• Already inversed covariance matrix • Faster and more stable than EKF • Suitable for multi-vehicle systems	• Suffers from linearization errors • No re-linearization step
CEKF SLAM	Bayesian filter	• Cost-effective • Outliers/errors only affect local maps • Auxiliary coefficient matrix is used for inactive parts	• Needs correct link between local and global maps
Fast SLAM	Particle filter	• Capable of updating with unknown data association • Less computation and memory cost than EKF • Suitable for nonlinear cases • Robust in cases where motion noise is high relative to measurement noise	• Loses accuracy when marginalizing the map and resampling is performed
Graph SLAM	Batch least squares optimization	• Suitable for nonlinear cases • More accurate • Can handle a large number of features	• Not suitable for online applications • Relies on good initial values
iSAM2	Incremental optimization	• Very fast • Suitable for nonlinear case • Allows re-linearization and data association correction	• Complexity grows when graph become dense

SLAM ++	Incremental optimization	• Suitable for nonlinear case • Very fast estimation (faster than iSAM2) • Efficient uncertainty estimation • Suitable for large-scale mapping	• Complexity grows with increasing number of observations

1.3.3 Inertial SLAM

New SLAM methods have appeared thanks to the advances in sensor technology and digital technology. These methods are also optimization based or filtered based. Inertial SLAM has become hot research topic. The inertial measurement units (IMUs) can be fused with the camera or LiDAR to support pose (position, velocity, attitude) estimation, which largely improve the SLAM accuracy. With IMU, the attitudes, especially the heading, are observable (Qin et al., 2017). The integration of IMU measurement can also improve the motion tracking performance during gap of observations. For instance, for Visual SLAM, illumination change, texture-less area, or motion blur will cause losses of visual tracks (Qin et al., 2017).

Fusion can be loosely coupled (Falquez et al., 2016, Fang, et al., 2017) or tightly coupled (Lynen et al., 2015, Qin et al., 2017, Zhang et al., 2017, Chen and Zhu, 2018). For loosely coupled SLAM system, IMU is mainly used to provide orientation information, while for tightly coupled SLAM, inertial navigation data is fused with camera/LiDAR states to build up measurement models, and then performs state estimation (Chen et al., 2018). Therefore, for loosely coupled inertial SLAM system, the IMU is used as prior for the whole system, thus it could not use IMU measurements for further optimization. The loosely coupled method is normally more efficiently than the tightly coupled methods, however, it is less accurate science it takes the odometry part as a black box (Ye, et al., 2019). For tightly coupled methods, a Kalman filter can be used to correct the IMU states,

which shows ability of accurate navigation during long-range GPS denied period (Hemann et al., 2016).

1.4 Application of SLAM in Autonomous Driving

Depending on the different characteristics of SLAM techniques, there could be different applications for autonomous driving. One classification of the applications is: offline or online. A map satisfying a high-quality requirement is typically generated offline, such as the High Definition (HD) map (Liu et al., 2020). For this kind of high density 3D point cloud map, an offline map generation process ensures the map's accuracy and robustness. Such maps can be pre-generated to support real-time operations of autonomous vehicles (Figure 1.4).

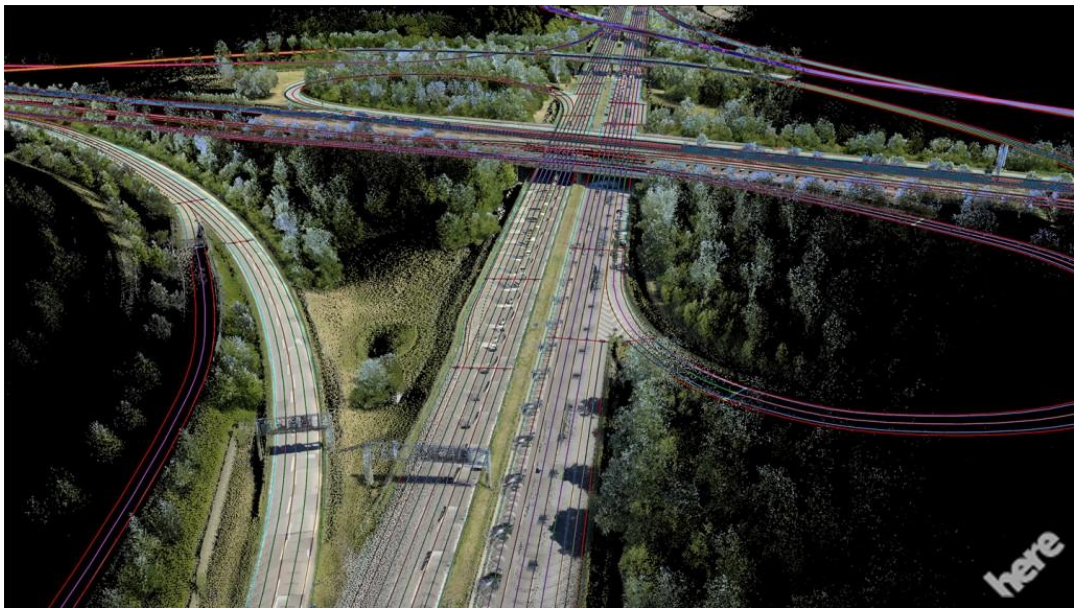


Figure 1.4 An image from a high-definition map (HERE, 2017)

1.4.1 High-definition Map Generation and Updating

As stated earlier, SLAM can be used to generate digital maps used for autonomous driving, such as the HD map (Liu et al., 2020). Due to the stringent requirements, high quality sensors are used. LiDAR is one of the core sensors for automated cars as it can generate high density 3D point clouds. High-end GNSS and IMU sensors are also used to provide

accurate position information. Cameras can provide information that is similar to the information detected by human eyes. The fusion of sensor data and analysis of road information to generate HD maps needs considerable computational power, which is not feasible in current onboard vehicle systems. Therefore, the HD map is built-up offline, using techniques such as optimization-based SLAM. The offline map creation can be done by driving the road network several times to collect information, and then process together all the collected perceptive sensor information and position information to improve the accuracy and quality of the final map.

The road environment and road rules may change, for instance, the speed limit may be reduced due to road work, road infrastructure may be changed due to building development, and so on. Therefore, the HD map needs frequent updates. Such updates can utilize the online data collecting from any autonomous cars. The data is transmitted to central (cloud) computers where the update computations are done. Other vehicles can receive such cloud-based updates and make a timely adjustment to driving plans. Jo et al. (2018) proposed a SLAM change update (SLAMCU) algorithm, utilizing a Rao-blackwellized particle filter approach for online vehicle position and (new) map state estimation. In the work of Kim et al. (2018), a new feature layer of HD maps can be generated using Graph SLAM when a vehicle is temporally stopped or in a parking lot. The new feature layer from one vehicle can then be uploaded to the map cloud and integrated with that from other vehicles into a new feature layer in the map cloud, thus enabling more precise and robust vehicle localization.

1.4.2 Small Local Map Generation

SLAM can also be used for small local areas. One example is within parking areas. The driving speed in a parking lot is low, therefore the vision technique will be more robust than other high-speed driving scenarios. The parking area could be unknown (public parking lot or garage), or known (home zone) – both cases can benefit from SLAM. Since SLAM can be used without GNSS signals, it is suitable for vehicles in indoor or underground parking areas, using just the perceptive sensor and odometry measurements (velocity, turn angle) or IMU measurements. For unknown public parking areas, the position of the car and the obstacles, such as pillars, side-walls, etc, can be estimated at

the same time, guiding the parking system. For home zone parking, the pre-generated map and a frequent parking trajectory can be stored within the automated vehicle system. Each time the car returns home, re-localization using the stored map can be done by matching detected features with the map. The frequent trajectory could be used for the planning and controlling steps.

An approach that utilizes multi-level surface (MLS) maps to locate the vehicle, and calculate and plan the vehicle path within indoor parking areas was proposed by Kummerle et al. (2009a). In this study, graph-based SLAM was used for mapping, and the MLS map is then used to plan a global path from the start to the destination, and robustly localize the vehicle with laser range measurements. In the work of Lee et al. (2018), a grid map and an EKF SLAM algorithm were used with W-band radar for autonomous back-in parking. In this work an efficient EKF SLAM algorithm was proposed to enable real-time processing. Im et al. (2019) proposed an around view monitor (AVM)/ LiDAR sensor fusion method to recognize the parking lane and provide rapid loop closing performance. The above studies have demonstrated that both filter-based SLAM and optimization-based SLAM can be used to support efficient and accurate vehicle parking assistance (local area mapping and localization), even without GNSS.

1.4.3 Localization within the Existing Map

In map-based localization, a matching method is used to match “live” data with map information, using methods such as Iterative Closest Point (ICP), Normal Distribution Transform (NDT), etc (Liu et al., 2020; Zheng and Wang, 2017). These algorithms can be linked to the SLAM problem since SLAM executes loop closing and re-localization using similar methods. For a SLAM problem, the ability to recognize a previously mapped object or feature and to relocate the vehicle within the environment is essential for correcting the maps (Bresson et al., 2017). Therefore, the reuse of a pre-generated map to localize the vehicle can be considered as an extension of a SLAM algorithm. In other words, the pre-generated and stored map can be treated as a type of “sensor” to support localization.

However, matching live data with a large-scale pre-prepared map requires substantial computational time. Hence some methods have been proposed to increase computational efficiency. One method is to firstly narrow down the possible matching area from the map with position estimated from GNSS or GNSS/INS, and then do detailed matching of the detected features with the map (Li and Nashashibi, 2012).

Due to the limited installation of LiDAR systems in commercial vehicles (high price of sensor and high power consumption), localization of a vehicle with a low-cost sensor (e.g. vision sensor) in a pre-generated HD map is of considerable practical interest. For instance, Wolcott and Eustice (2014) located a vehicle within a dense LiDAR-generated map using vision data, and demonstrated that a similar order of magnitude error rate can be achieved to traditional LiDAR localization, but with several orders of magnitude cheaper sensor technology.

In addition, moving objects within the road environment will cause a drift of perception, localization and mapping for autonomous driving. SLAM can be used to address the problem of DATMO (detection and tracking of moving objects) (Wang et al., 2007), because one of the SLAM's assumptions is that the detected features are stationary. As the static parts of the environment are localized and mapped by SLAM, the dynamic parts can be concurrently detected and tracked. Some approaches have dealt with dynamic obstacles (e.g., Vu, 2009; Wang et al., 2003; Fei, 2012).

1.4.4 Challenge of Applying SLAM in Autonomous Driving

1.4.4.1 Ensuring High Accuracy and High Efficiency

Localization and mapping for automated vehicles need to be accurate and robust to any environment changes, and be executed with high efficiency. With rapidly developing sensor technology, the combination of different sensors can compensate for the limitations of a particular sensor. Examples include GNSS/IMU+LiDAR/camera SLAM, Radar SLAM, and so others. There is considerable research and developments associated with low-cost and/or miniturized LiDAR sensors. New LiDAR sensor concepts promise significant reduction in LiDAR systems' cost, with the potential for real-time implementation in future autonomous vehicles. For instance, RoboSense has unveiled a

new $200 LiDAR sensor combining MEMS sensors and an AI-based deep-learning algorithm to support high-performance autonomous driving applications (Miller, 2018).

Choosing a SLAM approach should take into consideration of the different application scenarios. Optimization-based SLAM can provide more accurate and robust estimation. However, it is more suitable for offline estimation. EKF SLAM suffers from the quadratic increasing of the number of state variables, which restricts its online application in large-scale environments. Although high-resolution map generation can be offline, real-time, or near-real-time, solutions are essential for map updating and map-based localization applications.

Any change of road environment should be quickly updated in the map and transmitted to other road users. The emerging of 5G wireless technology can make the communication between vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-cloud more reliable and with higher throughput (Kuutti et al., 2018).

1.4.4.2 Expressing the Environment

There are different types of maps that can be used to represent the road environment. The road environments are always very complex, and hence have large scale, resulting in high computational burden and storage requirements. For instance, the high-density point cloud maps generated using LiDAR and/or vision sensors can provide abundant features and 3D structure surrounding the vehicle. However, transmission, updating and processing this volume of data is burdensome. Some researchers have proposed the concept of "Road DNA" to represent the road environment and to deal with the Big Data problem (Li and Yang, 2016; TomTom, 2017). Road DNA technology converts a 3D point cloud road pattern into a compressed, 2D view of the roadway without losing details (TomTom, 2017), with the objective to reduce processing requirements.

1.4.4.3 Issue of Estimation Drifts

SLAM estimation drifts may be caused by accumulated linearization error, the presence of dynamic obstacles, noisy sensor data, wrong data association, etc. In most SLAM algorithms, nonlinear models are used to represent the vehicle motion pattern and the environment. EKF SLAM suffers from a divergence problem due to the accumulation of

linearization error. Biases may occur when linearization is performed using values of state variables that are far from their true values. For optimization-based SLAM, poor initial guess of variables will lead to poor convergence performance. Rotation may be the cause of nonlinearity and has a strong impact on the divergence of estimation (Huang et al., 2010; Bresson et al., 2015). Thus, the accumulated vehicle orientation error will cause the inconsistency of the SLAM problem. One solution to the linearization challenge is the Linear SLAM algorithm proposed by Huang et al. (2010), which modifies the relative state vector and carries out "map joining". Sub-map joining, which involves solving a linear least squares problem and performing nonlinear coordinate transformations, does not require an initial guess or iteration. Other methods have been proposed, such as using a local sub-map (re-initialization of local map), an alternative linearization method (Martinea-Cantin et al., 2005), and a nonparametric method (Thrun et al., 2004).

Dynamic objects such as pedestrian, bicycles, other vehicles, may cause estimation drifts since the system may wrongly identify them as static road entities. There are some methods to avoid this. Probabilistic Maps that use probabilistic infrared intensity values has been proposed by Levinson and Thrun (2010). In this study, GNSS/INS and a 64-beam LiDAR sensor were combined to achieve robust position RMS errors of 9cm in dynamic environments. However, this system suffers from high cost and high processing requirement. The 3D Object Tracker (Wangsiripitak and Murray, 2009) can be used to track moving objects in visual SLAM methods. Another algorithm proposed by Asmar (2006) uses Canny's edge detector to find dominant edges in the vertical direction, of a tree trunk, and select these tree trunks as typical salient features.

Another source of drifts are outliers within the sensor observations. Each sensor has its own error sources. For example, in the case of a camera, the fuzzy image due to high speed and poor light conditions may cause wrong identification of landmarks. LiDAR sensors are sensitive to weather conditions (such as rainfall), and large changes of the road environment. GNSS may suffer from signal blockage. FDI (Fault Detection and Isolation system) techniques can be used to detect measurement outlies and reject these outliers' influence on positioning and localization.

The aforementioned SLAM error sources may also result in wrong data association, which is an important process to associate measurement(s) to a specific landmark. Wrong data association may happen due to not only the noisy sensor data, inconsistency, wrong detection of dynamic objects, etc, but also to some specific road environments. For instance, the highway environment is sometimes visually repetitive and contains many similar features, making it difficult to recognize a previously explored area.

Some researchers avoid the challenge of wrong data association directly at the front-end step of SLAM by using RANSAC (Kitt et al., 2010), which is commonly used in vision SLAM to reject outliers. Xie et al. (2017) proposed a middle layer, referred to as Graph-Tinker (GTK), that can detect and remove false-positive loop closures. Artificial loop closures are then injected into the pose graph when using an Extended Rauch-Tung-Striebel smoother framework.

The data association challenge can also be addressed at the back-end step since there is still a chance that outliers are not totally eliminated. Early work by Huber (1981) has recognized the lack of robusteness in least squares estimator for outlier. Malis and Marchand (2006) using an academic example to demonstrate that the LS approach is not robust to outliers. The development of robust alternatives is promoted in the object of reducing the sensitivity of estimator to outliers. For instance, Least Trimmed Squares estimator, Least Median of Squares (LMS) (Rousseeuw and Leroy, 1987), M-estimators (generalizations of a Maximum Likelihood estimator), L-estimators (linear combinations of order statistics of the observations), R-estimators (estimator based on waste ranking) (Huber, 1981). Among them, M-estimators has played a central role in modern robust statistics (Bosse et al., 2013) with numerous approaches based on it. Unlike the aforementioned algorithm, RANSAC, which discards observations that are considered as outliers and thus suffered from information loss due to reduced sample size, robust statistical estimators, such as the M-estimators, can provide automatically way to deal with outliers without discarding them (Bosse et al., 2013).

Agarwal (2015) proposed a Dynamic Covariance Scaling (DCS) method using principles of M-estimators. This method is robust to large front-end outliers while it also mitigates the effects of poor initializations (Agarwal, 2015). Olson and Agarwal (2012) proposed a

max mixture method which is robust to data-association outliers by allowing richer error models which allow the probability of a failure to be explicitly modelled. This mothed can be used to automatically identified loop closing errors and avoids exponential memory complexity. The concept of Switchable Constraints (SC) was introduced by Sünderhauf and Protzel (2012), that a switchable variable is introduced to each loop closure constraint. Once a constraint is considered as an outlier, it can be turned off during optimization. Latif et al. (2013) introduced an algorithm known as Realizing, Reversing, and Recovering (RRR), which is a consistency-based loop closure verification method. More recently, Carlone et al. (2014) used ℓ1_relaxation to select "reliable" measurements and Carlone and Calafiore (2018) used convex relaxations to solve the nonconvex problem without the need for an initial guess of unknown poses. The potential causes of SLAM drifts and the corresponding suggested solutions are summarized in Table 2.

Table 1.2 Potential causes of SLAM drifts and solutions

SLAM drift	Solutions
Linearization error	• Alternative linearization method (Martinea-Cantin et al., 2005) • Nonparametric (Thrun et al., 2004) • Sub-map joining (Huang, et al., 2010) • DCS (Agarwal, 2015)
Sensor outliers	• Fault Detection and Isolation (Morales et al., 2008) • Sensor fusion to compensate for different sensor errors (Wei et al., 2013; Zhang and Singh, 2015)
Dynamic objects	• Probabilistic maps (Levinson and Thrun, 2010) • 3D Object Tracker (Wangsiripitak and Murray, 2009) • Salient feature detection (Asmar, 2006)
Wrong data association	• RANSAC (Kitt et al., 2010) • Graph-Tinker (Xie, et al., 2017) • DCS (Agarwal, 2015) • Max mixture (Olson and Agarwal, 2012) • Switchable Constraints (Sünderhauf and Protzel,

	2012) • RRR (Latif et al., 2013) • ℓ 1_relaxation (Carlone et al. 2014), convex relaxations (Carlone and Calafiore, 2018)

1.4.4.4 Lack of Quality Control

The quantitative evaluation of the SLAM algorithms is another important challenge. There are some criteria to evaluate SLAM algorithms, such as their accuracy, scalability, availability, recovery (which is the ability to localize the vehicle inside a large-scale map), and updatability. Quantitative analysis of SLAM algorithms' performance is essential since they can provide numerical evaluation and provide a basis for comparison of different SLAM algorithms.

Estimation accuracy is a widely used quality analysis metric, but it can be difficult in practice for autonomous driving. Most approaches evaluate SLAM algorithms' performance by comparing the results to “ground truth” using, for example, an accurate ground truth map. However, a suitable ground truth map is seldom available, and furthermore the precise location of the vehicle trajectory on the actual road surface may not be always available. Vehicle pose Root Mean Square Error (RMSE) is normally used to indicate the SLAM trajectory estimation result's accuracy. Kummerle et al. (2009b) proposed a framework for analyzing the accuracy of SLAM by measuring the error of the corrected trajectory. Another widely used quality analysis method is the so-called Chi-squared ($\chi 2$) test. According to Kurlbaum and Frese (2009), the Chi-squared test is a statistic test to quantify the quality of the provided covariance matrices for landmark measurements and odometry error. When the minimum $\chi 2$ error is nearly equal to the difference of the dimension of the measurement vector and the size of the state vector, the measure would be considered as being of good quality.

Some researchers (Li et al., 2015; Cadena and Neira, 2009; Huang and Dissanayake, 2007) have considered the consistency of their SLAM algorithms. When the estimation error is beyond the uncertainty, it can be assumed that the estimation results are inconsistent. EKF-based SLAM suffers from such an inconsistency problem unless the Jacobians are

evaluated around the true system state. Whether inconsistency can be tolerated ultimately depends on the application of the SLAM results (Dissanayake, et al., 2011).

The reliability of output of the localization, mapping and navigation system should also be checked. However, few studies have been made on the quantitative analysis of the reliability of SLAM. Some reliability studies for other localization systems (such as GNSS, GNSS/IMU) can be used as a reference to guide the SLAM community. The system reliability can be considered as two components: internal reliability and external reliability. The former identifies the ability of the system to detect faults, which is quantified by the Minimal Detectable Bias (MDB), and is indicated by the lower bound for detectable faults. The latter estimates the influence of undetected faults on the final solution (Wang and Knight, 2012; Yang et al., 2013; Zheng and Wang, 2017; Baarda 1968; Li et al., 2016). When the MDB value is lower, the system is more reliable. Similarly, the reliability of the feature observation model and vehicle motion model can also be evaluated with these approaches.

Integrity is critical, as it is an indicator of the "trustworthiness" of the information supplied by the localization system, and can provide timely warning of the risks caused by inaccuracy (El-Mowafy et al., 2019). Integrity measures are used to quantify localisation safety requirements and its concept was firstly established in aviation. The integrity assessment is undertaken with 4 attributes: Protection Level (PL), Alert limit (AR), Time-to-alert, and Integrity risk (Wörner et al., 2016; International Civil Aviation Organization, 2006; Tossaint et al., 2007, Speidel et al., 2013). PL is the estimated upper bound on the true position error. An upper threshold (alert limit) for PL should be set and the error beyond the level should cause concern. Time-to-alert is the time between PL surpassing the AR and issuing an alert. Integrity risk is the probability that the true position error is beyond the AR without an alert being issued (Wörner et al., 2016). A graphical tool developed by Tossaint et al (2007), the Stanford-ESA Integrity Diagram, can be used to validate the system's integrity level.

Fault detection and isolation (FDI) is one of the most popular alert generation approaches for GNSS-based localization (Morales et al., 2008; Sundvall and Jensfelt, 2006; Hewitson and Wang, 2010). Receiver Autonomous Integrity Monitoring (RAIM) is based on FDI

and is the traditional integrity assessment technology developed for GPS signals in an aviation system.

Integrity monitoring is mostly studied in the GNSS system, and the integrity navigation study has been expanded in application such as military aerospace operations, civilian airplane (Larson, 2010), autonomous underwater vehicles (Fauske et al., 2008), autonomous outdoor vehicle (Nebot and Durrant-Whyte, 1999), etc. Larson (2010) studied the integrity quantification of an image-based navigation system and inferred more adding measurements may have potential to improvement the image-based integrity. The concept of integrity is therefore applicable to land vehicle localization (Wang and Ober, 2009).

Due to the strict safety requirement of autonomous driving, there is an increasing attention on integrity by autonomous driving researchers. The localization and navigation of a self-driving car is based on the use of multiple sensors, therefore the traditional integrity analysis methods for GNSS should be extended. Molina et al (2011) proposed a project that investigates the concept, characteristic, method of Autonomous Integrity Monitoring (AIM) for a multi-sensor UAV system, which is an extension of RAIM for multi-sensor application.

1.5 Challenging Issues in Localization and Mapping for Autonomous Driving

1.5.1 High Requirements for Autonomous Driving

To achieve full automated driving, there are some key criteria that need to be taken into consideration for localization and perception steps and there are stringent requirements according to these criteria. The first criterion is accuracy. For autonomous driving, the information about where the road is and where the vehicle is within the lane can give the following systems the cue of where to go next. To realize these applications and ensure vehicle safety, there is some stringent requirements for position estimation at lane level (1-3 m) or even where-in-lane level, also called sub-lane level (<0.5 m) (Farrell et al., 2016). Recognition range is also important since the planning and controlling steps need

enough processing time for the vehicle to react to any detected information, which should be up to 200 meter or farther (Seif and Hu, 2016) for the autonomous vehicle. Robustness means the localization and perception should be robust to any changes while driving, such as driving site (urban, high way, tunnel, far from city), light condition, weather or season, etc. Another requirement of the autonomous vehicle is to sensor the 3D environment other than 2D since the height of the objects and road surface will be useful for navigation.

Therefore, according to the characteristics of aforementioned localization and mapping techniques, any standalone technique cannot meet all the strict requirements. A combination of multi-sensor and different techniques is significant for autonomous driving.

1.5.2 Lack of Quality Control for Localization and Mapping System

For most of the practical application of autonomous driving, ground truth is hard to achieve, resulting in a lack of Quantitative evaluation for estimation accuracy. Therefore, other criteria should be introduced to evaluate the system solutions quality in order to ensure integrity when ground truth is unavailable. Since the outlier is unavoidable for any sensor-based systems, the method to mitigate the influence of the outliers should be further studies without losing the redundancy of the whole system.

1.5.3 Lack of Geometry Analysis for Localization and Mapping System

Geometry can significantly influence the estimation and quality evaluation of any localization and mapping system. This has been extensively researched by GNSS community. However, in localization and mapping by other sensor systems, it has not been fully studied. The influence of geometry on Minimum Detectable Bias (MDB), Minimum Separable Bias (MSB) and External Reliability (ER) still needs further comprehensive analysis with respect to different systems.

1.5.4 Efficiency of Applying High Definition Map for Autonomous Driving

High definition (HD) map is a new technique which is designed to support highly automated driving. However, its application for autonomous driving still faces some practical limitations. Such as the high burden of calculation, data transmission and real-time updating. An efficient and accurate procedure of applying HD map for autonomous driving is needed for investigating.

1.6 Contributions of this Study

To solve the issues in localization and mapping for autonomous driving, the research in this thesis acts as the major contributions, which can be summarized as the following aspects:

a) The equivalence between least squares and smoothing for dynamic system is discussed. A new Kalman smoother based quality control method is developed according to this equivalence, which can be used to deal with system model outlier with detection, and identification, also can be used to analyse, control and improve the system quality. Some variant mathematical models of this quality control method have been developed to deal with issues such as singular covariance system, numerical unstable of smoothing step.
b) Mathematical modelling aspects for GNSS multi constellation integration for both Real Time Kinematic (RTK) and Post Processing Kinematic (PPK) are investigated. The integration of GNSS and IMU system is developed. Quality control analysis is conducted for these positioning systems. PPK can provide more reliable positioning results for mapping since it can avoid re-convergence when geometry changes, and satellite is newly viewed and introduced to the state parameter vector.
c) Mathematical modelling and quality control aspects for online simultaneous localization and mapping (SLAM) are examined. A smoother based offline SLAM method is firstly introduced for SLAM problem with quality control. Both outdoor and indoor dataset have been tested with these SLAM methods. Geometry analysis for the SLAM system has been done according to the quality control

results. The system reliability analysis is essential for the SLAM designer as it can be conducted at the early stage without real measurement.

d) A localization method is proposed that utilizes the High Definition map as a sensor source to integrate with other perception sensors, which significantly improves localization efficiency and accuracy. Geometry analysis is undertaken with the MDB, MSB and ER value to analyse the influence of the geometry upon the estimation solution and the system quality, which can be hints for future design of the localization system.

e) A GNSS/INS aided LiDAR mapping and localization procedure is developed. High-density map is generated offline, then, LiDAR based localization can be undertaken online with this pre-generated map. Quality control is conducted for this system. The results demonstrate that the LiDAR based Localization within map can effectively improve the positioning accuracy compared to GNSS/INS, especially during period that GNSS signal is lost.

1.7 Thesis Outline

In summary, this thesis consists of eight chapters and the specific contents of each chapter are outlined as follows:

Chapter 1 gives a brief introduction to localization and mapping for autonomous driving and background of SLAM, including its principles, classification, current algorithms, and its applications. Furthermore, the critical issues and the main contributions of this thesis are stated.

Chapter 2 describes the optimal estimation models and corresponding quality control algorithms. The relationship between filtering, smoothing and least squares is discussed. The quality control procedures for outlier detection, identification, adaption and system reliability analysis are presented for all the three optimal estimation approaches. Numerical analysis is presented to illustrate the procedures and performance of the quality control method.

Chapter 3 analyses the performance of the proposed Extended Kalman filter/smoother quality control for multiple difference positioning systems (PPK, RTK, GNSS/INS integration). Quality control test is conducted with a real-world urban driving dataset.

Chapter 4 investigates the online EKF based SLAM algorithm with quality control. Numerical tests are presented with two real-world datasets, one indoor, and another outdoor. The influence of geometry upon the estimation reliability is investigated.

Chapter 5 proposed a new smoothing based SLAM algorithm with quality control. The relationship between the EKF SLAM, KS SLAM and graph optimization SLAM is demonstrated. Performance and quality of this new algorithm is presented with two real-world datasets. The results show the proposed KS SLAM is more reliable then filter based SLAM, and can achieve efficient quality control results for large scale map.

Chapter 6 investigates the capabilities of pre-generated map as a sensor source to support efficient localization. MSB and correlation coefficients are utilized to explain the performance of outlier detection, identification, and correction methods in a case study. The influence of system geometry is investigated with considering the Internal Reliability (MDB) and External Reliability (ER).

Chapter 7 designs an efficient multi-sensor mapping and localization framework for autonomous driving. On-board LiDAR/GNSS/IMU sensor is integrated to generate georeferenced road map offline. Online localization is then undertaken using the offline map as a sensor source. Road driving test is undertaken to examine the mapping and localization system. The proposed map-based localization method shows the advantage in supporting accurate localization, especially during period of GNSS blockage. Quality control is undertaken to monitor the quality of this proposed system.

Chapter 8 summarizes the research findings, and make recommendations for future research.

Chapter 2 Statistical Quality Control Framework for Optimal Filtering and Smoothing

2.1 Introduction

Filtering, prediction and smoothing are approaches used for finding the optimal estimation of unknown parameters, such as velocity and position, et al., under the noisy dynamic system. Prediction expects the parameter value according to the previous observations, while filtering is performed to obtain estimates with both current and previous observations. Smoothing is done if reanalysis of the state is needed, by which all the observations obtained during the full-time interval are utilized (Sarkka 2013; Cross 1994). Although filtering has already made optimal estimation, smoothing has been demonstrated that outperforms filtering (Einicke 2012; Sarkka 2013; Kaniewski et al. 2017; Wang 2012) as it uses all measurements for estimation. For some applications that require higher accuracy and precision, such as GPS positioning for automated vehicle or UAV, adding smoothing process will make the estimated solution of position and velocity much better and more reliable.

Kalman filter is the mostly popular algorithm that contains time update and measurement update steps, which represent the prediction and filtering processes, respectively. By using a Kalman filter, the estimation can be done recursively in real-time. Most of the smoothing algorithms, the fixed-interval sequential smoother, the fixed-lag smoother and the ensemble smoother, are based on the Kalman's hypotheses (Cosme et al. 2012). The most commonly used smoother, Rauch-Tung-Striebel smoother (RTS smoother, which is also called Kalman smoother) (Sarkka 2013), is a fixed-interval smoother constituted by forward and backward passes. The forward pass is a Kalman filter step, while the backward pass also depends on the predicted and estimated error covariance and transition matrix stored in the Kalman filter pass. Therefore, RTS smoother is simpler in

structure and easier to be used then other forms of smoothing. In this Chapter, all the derivation and analysis are based on Kalman filter and Kalman smoother.

Since the initial development of Kalman filtering and smoothing algorithms, many extensions have been studied, for instance, the online smoothing that making estimation in real time or with acceptable delay (Pillonetto et al. 2010; Chiang et al. 2012; Kaniewski et al. 2017), the extended Kalman filter and smoother (EKF and ERTSS) that dealing with non-linear measurement model or dynamic model by linearization (Sarkka 2013; Einicke 2012), the unscented Kalman filter and smoother (UKF and URTSS) that using the unscented transform (Sarkka 2013). These extensions have been widely employed in various fields (Aravkin et al 2013; Sarkka 2013), such as navigation, tracking, aerospace engineering, space exploration, geophysical study and weather prediction. Many research studies focused on the algorithms' implementation and performance, such as improving the stability, reducing the computation time, and enhancing the robustness against bad measurements (Cipra and Romera 1997; Aravkin et al. 2011; Aravkin et al. 2013; Auger et al. 2013). However, the study of quality control of Kalman filter and smoother is less found.

The occurring of error, or outlier, is somewhat not inhibited because the Kalman filter and smoother approaches are mainly used to treat the noisy measurement and dynamic system. The error may exist in the observations as a sudden outlier or systematic error. It can also be due to the improper system model or false modelled process noise and measurement noise (Wang 2008). The unpredictable model errors will eventually result in a degradation of the final estimation of state parameters' accuracy and reliability. Therefore, controlling the system quality of Kalman filter-smoother process is imperative for the purpose of reducing the influence of system errors.

One main part of Quality Control (QC) is the DIA method that dealing with the possible model errors with detection, identification and adaption. DIA method is a three steps procedure that firstly looks for the unspecified model errors, and then finds out the cause of the error and position of the error, therefore this model can recovery from these errors by the Adaption step (Teunissen 1990; Salzmanm 1993; Wang and Knight 2012). The evaluation of system performance is also part of quality control, including analysing the

reliability and separability. Reliability is the capability that a system can detect outliers and then evaluate the influence of the outlier on the estimation. Reliability analysis includes two parts. The Internal Reliability refers to the model property of the lower bounds of outliers that can just be detected with the chosen test statistic for a given set of Type I (α) error and Type II (β) error (Knight, et al., 2010). Thus, Minimal Detectable Biases (MDB) is applied to represent the detectability of outliers within the measurement and system. Another is the External Reliability which evaluates how the nondetectable outlier at the low bound will affect the estimation. Separability analysis evaluates the correlation coefficients between the fault detection statistics for each observation (Hewitson and Wang 2006).

These QC methods have been well tested in Global Navigation Satellite System (GNSS) community for GNSS positioning or GNSS/IMU integration (Alquarashi and Wang 2015; Wang and Knight 2012; Teunissen 2017). Teunissen (2017) explained how receiver-satellite geometry affects the MDBs in Single Point Positioning (SPP), and how DIA working in Differential Global Positioning System (DGPS). Alquarashi and Wang (2015) tested four positioning strategies (GPS, GNSS, GPS/INS and GNSS/INS integrations) with multiple faults detection and isolation algorithm. The application of these QC methods can also be found in other positioning and navigation system. For instance, Li (2016) and Li (2017) demonstrated quality analysis for vision-based navigation, Zheng and Wang (2017) illustrated the control of a High Definition map based vehicle localization system. Most of the mentioned applications of the QC methods are utilized under least squares framework while else are under Kalman filter framework. The quality of Kalman smoothing step should also be analysed but seldom be studied.

Therefore, this Chapter aims to investigate a new extended quality control method that monitors the Kalman filter-smoother procedure. The performance of the QC method is illustrated and explained by applying to selected examples. This Chapter's structure is as follows: Section 2.2 formulates the Kalman smoother as a least squares problem. Section 2.3 briefly demonstrates the previous quality control algorithms under Kalman filter framework. Then, the proposed quality control method under the Kalman smoother framework is derived, including outlier detection, identification statistic, adaption

procedure, MDB and External Reliability analysis. Section 2.4 conducts numerical examples to demonstrate the function of the proposed the proposed quality control method. Section 2.5 proposed a Unified Least Square based quality control method that is suitable to deal with problems with ill-conditioned matrix. Section 2.6 summarizes the QC methods' performance and points out the issues that need to be explored further.

2.2 Kalman Filter, Smoother and Least Squares

2.2.1 Kalman Filter

According to Cross (1994), the Kalman filtering, prediction, and smoothing acquire estimates by the least squares process. In this Chapter, the dynamical system is described by the following dynamic model:

$$x_k = f_k(x_{k-1}) + \tau_k \quad k = 1,2,\dots,T \tag{2.1}$$

and measurement model:

$$y_k = h_k(x_k) + \varepsilon_k \quad k = 1,2,\dots,T \tag{2.2}$$

where x_k is the $n_k \times 1$ state parameter vector at epoch k, T represents the last epoch, $f_k()$ is the function for state transition, y_k is the $m_k \times 1$ measurement vector at epoch k, $h_k()$ is the function for measurement with its Jacobian matrix H_k. τ_k and ε_k are the unknown errors in the dynamic and measurement models, respectively, with covariance matrices Q_k and R_k . x_0 is the initial state with uncertainty Q_{x_0}.

During the Kalman Prediction process, the predicted values of the state parameters $\bar{x}_k$ will be obtained by:

$$\bar{x}_k = f_k(\hat{x}_{k-1}) \tag{2.3}$$

with $\Phi_k = \left.\frac{df_k}{dx_k}\right|_{x_k=\hat{x}_{k-1}}$ is the Jacobian of the process model from epoch k-1 to k; here the hat ‘ ¯ ’over x_k represents the predicted value of state, ‘ ^ ’ over x_k represents the

estimated (filtered) value of state. Therefore, $\hat{x}_{k-1}$ is the optimal estimator of the state parameters acquired by Kalman filtering process at previous epoch *k*-1. The cofactor matrix of $\bar{x}_k$ is calculated as:

$$Q_{\bar{x}_k} = \Phi_k Q_{\hat{x}_{k-1}} \Phi_k^{\mathrm{T}} + Q_k \tag{2.4}$$

2.2.2 Solve Kalman Filter Estimation with Least Squares Theory

Each specific epoch in KF process can be formed as a least square model as $l_k = M_k x_k + \epsilon_k$, where $l_k = \begin{bmatrix} z_k \\ \bar{x}_k \end{bmatrix}$, $M_k = \begin{bmatrix} H_k \\ E \end{bmatrix}$, $\epsilon_k = \begin{bmatrix} \epsilon_{y_k} \\ \epsilon_{\bar{x}_k} \end{bmatrix}$, ϵ_k is the residual vector; E is the $m_k \times m_k$ identity matrix. The corresponding stochastic model is described as:$C_{l_k} = \begin{bmatrix} R_k & 0 \\ 0 & Q_{\bar{x}_k} \end{bmatrix}$. In the l_k vector, the measurement model part z_k is calculated by linearized the measurement model as:

$$z_k = H_k x_k + \varepsilon_{yk} = y_k - h_k(\bar{x}_k) + H_k \bar{x}_k \tag{2.5}$$

when the measurement model is linear, z_k equals to y_k.

Therefore, the Kalman filter estimation and the associated covariance matrix will be:

$$\begin{aligned} \hat{x}_k &= \left(M_k{}^{\mathrm{T}} C_{l_k}{}^{-1} M_k\right)^{-1} M_k{}^{\mathrm{T}} C_{l_k}{}^{-1} l_k \\ &= \bar{x}_k + G_k(z_k - H_k \bar{x}_k) = \bar{x}_k + G_k dy_k \end{aligned} \tag{2.6}$$

$$Q_{\hat{x}_k}{}^{f} = \left(M_k{}^{\mathrm{T}} C_{l_k}{}^{-1} M_k\right)^{-1} = Q_{\bar{x}_k} - G_k Q_{dy_k} G_k{}^{\mathrm{T}} \tag{2.7}$$

$$Q_{dy_k} = R_k + H_k Q_{\bar{x}_k} H_k{}^{\mathrm{T}}, \;\; G_k = Q_{\bar{x}_k} H_k{}^{T} Q_{dy_k}^{-1} \tag{2.8}$$

where G_k is the Kalman gain matrix, dy_k is the innovation, and Q_{dy_k} is its cofactor matrix.

2.2.3 Kalman Smoother

The smoothed state vector at time *k* and its covariance matrix are derived as:

$$x_k{}^s = \hat{x}_k + J_k(x_{k+1}{}^s - \bar{x}_{k+1}) \tag{2.9}$$

$$Q_{x_k{}^s} = Q_{\hat{x}_k} + J_k(Q_{x_{k+1}{}^s} - Q_{\bar{x}_{k+1}})J_k{}^T \tag{2.10}$$

$$J_k = Q_{\hat{x}_k}\Phi_{k+1}{}^{\mathrm{T}}Q_{\bar{x}_{k+1}}{}^{-1} \tag{2.11}$$

here the superscript '^{s}' represents the smoothing estimation, J_k is the Kalman smoother gain at epoch *k*.

2.2.4 Full Least Squares Structure

According to Cross (1994), the prediction, filtering, smoothing problems are to find the least squares estimates for x_1 to x_T. Therefore, when doing the Full Least Squares (FLS) that utilize the whole available measurement and dynamic information, the solution is obtained by minimizing $\sum_{k=1}^{T}(\varepsilon_k{}^{\mathrm{T}}R_k^{-1}\varepsilon_k + \tau_k{}^{\mathrm{T}}Q_k^{-1}\tau_k)$.

To solve this, the simplified data structures are introduced that containing the entire measurement and dynamic information. As all the measurements and dynamic models are taken into consideration together within the FLS, the total measurement model number is $m_{all} = \sum_{k=1}^{T} m_k$, and the total dynamic model number is $\sum_{k=1}^{T} n_k$. Therefore, the overall unknown parameter number is $n_{all} = \sum_{k=1}^{T} n_k$ and the overall function model number for FLS is $F_{all} = \sum_{k=1}^{T} m_k + \sum_{k=1}^{T} n_k$.

The state parameters and measurements from all the epochs are arranged in order respectively as vector X and vector Y, here:

$X = [x_1, x_2, \dots, x_T]^{\mathrm{T}}$

and

$Y = [y_1, y_2, \dots, y_T]^{\mathrm{T}}$

For full least squares, an approximate value of the unknown X_{app} is set and:

$$X = X_{app} + \Delta X \tag{2.12}$$

the X_{app} can be constituted by the predicted state parameters $\bar{x}_1$ to $\bar{x}_T$, therefore the design matrix ($A = \begin{bmatrix} A_{meas} \\ A_{dync} \end{bmatrix}$), vector ($L = \begin{bmatrix} L_{meas} \\ L_{dync} \end{bmatrix}$) can be obtained by the approximate value. A_{meas} and L_{meas} represent the measurement model part of the design matrix and L vector, while A_{dync} and L_{dync} represent the dynamic model part of the design matrix and L vector, respectively. They can be obtained by the following equations:

$$A_{meas} = \begin{bmatrix} A_1 & 0 & \cdots & 0 \\ 0 & A_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & A_T \end{bmatrix} \tag{2.13}$$

A_k is the design matrix of measurement model at epoch *k* and it equals to H_k if using the KF predicted value of state parameters as the approximate value.

$$A_{dync} = \begin{bmatrix} I & 0 & \cdots & 0 \\ -\Phi_2 & I & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & -\Phi_T & I \end{bmatrix} \tag{2.14}$$

here *I* is the $n_k \times n_k$ Identity matrix.

$$L_{meas} = \begin{bmatrix} L_{meas_1} \\ L_{meas_2} \\ \vdots \\ L_{meas_T} \end{bmatrix} = \begin{bmatrix} y_1 - A_1\bar{x}_1 \\ y_2 - A_2\bar{x}_2 \\ \vdots \\ y_T - A_T\bar{x}_T \end{bmatrix} \tag{2.15}$$

$$L_{dync} = \begin{bmatrix} L_{dync_1} \\ L_{dync_2} \\ \vdots \\ L_{dync_T} \end{bmatrix} = \begin{bmatrix} \Phi_1\bar{x}_0 - \bar{x}_1 \\ \Phi_2\bar{x}_1 - \bar{x}_2 \\ \vdots \\ \Phi_{T-1}\bar{x}_T - \bar{x}_T \end{bmatrix} \tag{2.16}$$

Since the measurement and the dynamic process are independent, the stochastic model, e.g., the covariance matrix, can be obtained from the measurement noise covariance matrix R and the process uncertainty Q ($D = {\sigma_0}^2 P^{-1} = {\sigma_0}^2 \begin{bmatrix} R & 0 \\ 0 & Q \end{bmatrix}$), here P is the weight matrix, $R = diag(\{R_k\}), Q = diag(\{Q_k\})$ and ${\sigma_0}^2$ is the a priori variance factor.

If the initial value is unknown, the covariance matrix for the first dynamic model (Q_1) should take into consideration of the uncertainty for the initial state parameters (Q_{x_0}), and is equal to the covariance matrix of predicted state parameter at epoch 1 ($Q_{\bar{x}_1}$).

2.2.5 Characteristic Least Squares Solution and Relationship to Kalman Smoothing

Hence, to solve the full least squares problem, the solution of the least squares normal equation could be found:

$$(A^{\mathrm{T}}PA)\Delta\hat{X} = A^{\mathrm{T}}PL \tag{2.17}$$

in here, for a dynamic system,

$$A^{\mathrm{T}}PL = {A_{meas}}^{\mathrm{T}}R^{-1}L_{meas} + {A_{dync}}^{\mathrm{T}}Q^{-1}L_{dync} \tag{2.18}$$

$$\begin{aligned} A^{\mathrm{T}}PA &= {A_{meas}}^{\mathrm{T}}R^{-1}A_{meas} + {A_{dync}}^{\mathrm{T}}Q^{-1}A_{dync} \\ &= \begin{bmatrix} E_1 & F_2^{\mathrm{T}} & 0 & \\ F_2 & E_2 & \ddots & 0 \\ 0 & \ddots & \ddots & F_T^{\mathrm{T}} \\ & 0 & F_T & E_T \end{bmatrix} \end{aligned} \tag{2.19}$$

with E_k and F_k defined as:

$$\begin{aligned} E_k &= {\Phi_{k+1}}^{\mathrm{T}}{Q_{k+1}}^{-1}\Phi_{k+1} + {Q_k}^{-1} + {A_k}^{\mathrm{T}}{R_k}^{-1}A_k \\ F_k &= -Q_k^{-1}\Phi_k \end{aligned} \tag{2.20}$$

when reach to the last epoch T, ${\Phi_{T+1}}^{\mathrm{T}}{Q_{T+1}}^{-1}\Phi_{T+1}$ is a $n \times n$ null matrix. A similar structure can be found in some early research (Aravkin, et al, 2013; Fahrmeir and Kaufmann, 1991).

By solving Equation 2.17 with the introduction structure in Equation 2.19 and 2.20, the Full Least Squares (FLS) estimation of correction to approximate state can be obtained:

$$\Delta\hat{X} = (A^{\mathrm{T}}PA)^{-1}A^{\mathrm{T}}PL \tag{2.21}$$

with $(A^{\mathrm{T}}PA)^{-1}$ being solved as:

$$(A^{\mathrm{T}}PA)^{-1}$$

$$= \begin{bmatrix} Q_{\hat{x}_1} + J_1(Q_{x_2}{}^s - Q_{\bar{x}_2})J_1{}^{\mathrm{T}} & J_1Q_{x_2}{}^s & J_1J_2Q_{x_3}{}^s & \cdots & J_1 \dots J_{T-1}Q_{x_T}{}^s \\ Q_{x_2}{}^sJ_1{}^{\mathrm{T}} & Q_{\hat{x}_2} + J_2(Q_{x_3}{}^s - Q_{\bar{x}_3})J_2^{\mathrm{T}} & J_2Q_{x_3}{}^s & \cdots & J_2 \dots J_{T-1}Q_{x_T}{}^s \\ Q_{x_3}{}^sJ_2{}^{\mathrm{T}}J_1{}^{\mathrm{T}} & Q_{x_3}{}^sJ_2{}^{\mathrm{T}} & Q_{\hat{x}_3} + J_3(Q_{x_4}{}^s - Q_{\bar{x}_4})J_3^{\mathrm{T}} & \cdots & J_3 \dots J_{T-1}Q_{x_T}{}^s \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Q_{x_T}{}^sJ_{T-1}{}^{\mathrm{T}} \dots J_1{}^{\mathrm{T}} & Q_{x_T}{}^sJ_{T-1}{}^{\mathrm{T}} \dots J_2{}^{\mathrm{T}} & Q_{x_T}{}^sJ_{T-1}{}^{\mathrm{T}} \dots J_3{}^{\mathrm{T}} & \cdots & Q_{\hat{x}_T} \end{bmatrix} \tag{2.22}$$

$$= \begin{bmatrix} Q_{x_1}{}^s & J_1Q_{x_2}{}^s & J_1J_2Q_{x_3}{}^s & \cdots & J_1 \dots J_{T-1}Q_{x_T}{}^s \\ Q_{x_2}{}^sJ_1{}^{\mathrm{T}} & Q_{x_2}{}^s & J_2Q_{x_3}{}^s & \cdots & J_2 \dots J_{T-1}Q_{x_T}{}^s \\ Q_{x_3}{}^sJ_2{}^{\mathrm{T}}J_1{}^{\mathrm{T}} & Q_{x_3}{}^sJ_2{}^{\mathrm{T}} & Q_{x_3}{}^s & \cdots & J_3 \dots J_{T-1}Q_{x_T}{}^s \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Q_{x_T}{}^sJ_{T-1}{}^{\mathrm{T}} \dots J_1{}^{\mathrm{T}} & Q_{x_T}{}^sJ_{T-1}{}^{\mathrm{T}} \dots J_2{}^{\mathrm{T}} & Q_{x_T}{}^sJ_{T-1}{}^{\mathrm{T}} \dots J_3{}^{\mathrm{T}} & \cdots & Q_{x_T}{}^s \end{bmatrix}$$

here, $J_k = Q_{\hat{x}_k}\Phi_{k+1}{}^{\mathrm{T}}Q_{\bar{x}_{k+1}}{}^{-1}$ is the Kalman smoother gain at epoch *k*, and $Q_{x_k}{}^s$ represented the covariance matrix of the smoothed state parameters (see Equation 2.10, 2.11). $(A^{\mathrm{T}}PA)^{-1}$ is also the covariance matrix of the FLS estimated parameters as: $Q_{\widehat{\Delta X}} = (A^{\mathrm{T}}PA)^{-1}$, therefore it can be clearly seen that the diagonal element of $Q_{\widehat{\Delta X}}$ from FLS estimation are equal to the covariance matrix of state after the smoothing step.

$A^{\mathrm{T}}PL$ is solved as:

$$A^{\mathrm{T}}PL = A_{meas}{}^{\mathrm{T}}R^{-1}L_{meas} + A_{dync}{}^{\mathrm{T}}Q^{-1}L_{dync}$$

$$= \begin{bmatrix} A_1{}^{\mathrm{T}}R_1{}^{-1}L_{meas_1} + Q_1{}^{-1}L_{dync_1} - \Phi_2{}^{\mathrm{T}}Q_2{}^{-1}L_{dync_2} \\ A_2{}^{\mathrm{T}}R_2{}^{-1}L_{meas_2} + Q_2{}^{-1}L_{dync_2} - \Phi_3{}^{\mathrm{T}}Q_3{}^{-1}L_{dync_3} \\ \vdots \\ A_{T-1}{}^{\mathrm{T}}R_{T-1}{}^{-1}L_{meas_{T-1}} + Q_{T-1}{}^{-1}L_{dync_{T-1}} - \Phi_T{}^{\mathrm{T}}Q_T{}^{-1}L_{dync_T} \\ A_T{}^{\mathrm{T}}R_T{}^{-1}L_{meas_T} + Q_T{}^{-1}L_{dync_T} \end{bmatrix} \tag{2.23}$$

now the FLS correction can be obtained as:

$$\Delta\hat{X} = (A^{\mathrm{T}}PA)^{-1}A^{\mathrm{T}}PL$$

$$= \begin{bmatrix} {dx_1}^f + J_1({x_2}^s - \bar{x}_2) \\ {dx_2}^f + J_2({x_3}^s - \bar{x}_3) \\ \vdots \\ {dx_{T-1}}^f + J_{T-1}({x_T}^s - \bar{x}_T) \\ {dx_T}^f \end{bmatrix} = \begin{bmatrix} {dx_1}^f + J_1{dx_2}^s \\ {dx_2}^f + J_2{dx_3}^s \\ \vdots \\ {dx_{T-1}}^f + J_{T-1}{dx_T}^s \\ {dx_T}^f \end{bmatrix} \tag{2.24}$$

within Equation 2.24 ${dx_k}^f$ is defined as the correction to the predicted states during the Kalman filtering step that: ${dx_k}^f = \hat{x}_k - \bar{x}_k$, while ${dx_{k+1}}^s$ is defined as the correction to the predicted states during the Kalman smoothing step at next epoch, that: ${dx_{k+1}}^s = {x_{k+1}}^s - \bar{x}_{k+1}$.

If using the Kalman predicted state as the assigned approximate value of unknown in FLS, the final estimation of unknown will be:

$$\hat{X} = X_{app} + \Delta\hat{X} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \vdots \\ \bar{x}_{T-1} \\ \bar{x}_T \end{bmatrix} + \begin{bmatrix} \hat{x}_1 - \bar{x}_1 + J_1({x_2}^s - \bar{x}_2) \\ \hat{x}_2 - \bar{x}_2 + J_2({x_3}^s - \bar{x}_3) \\ \vdots \\ \hat{x}_{T-1} - \bar{x}_{T-1} + J_{T-1}({x_T}^s - \bar{x}_T) \\ \hat{x}_T - \bar{x}_T \end{bmatrix}$$

$$= \begin{bmatrix} \hat{x}_1 + J_1({x_2}^s - \bar{x}_2) \\ \hat{x}_2 + J_2({x_3}^s - \bar{x}_3) \\ \vdots \\ \hat{x}_{T-1} + J_{T-1}({x_T}^s - \bar{x}_T) \\ \hat{x}_T \end{bmatrix} \tag{2.25}$$

$\hat{x}_k + J_k({x_{k+1}}^s - \bar{x}_{k+1})$ is the formula of Kalman smoother's estimated solution (Cross 1994) as shown in Equation 2.9.

therefore, a conclusion is coming out that the estimation solution and its covariance matrix by full least squares method, which deals with all the available measurements and dynamic information, is same to that calculated recursively by Kalman smoother if the FLS system uses the KF predicted state value as the approximately initial value.

2.3 Quality Control of Kalman Filtering, Smoothing, FLS

2.3.1 Residual and its Cofactor Matrix

The quality control methods contain two parts, first one deal with the outlier within the measurements or models by detecting and identifying the outlier. After that, the identified outlier can be removed from the measurement or its influence upon the final estimation can be mitigated. The second part of quality control method is reliability analysis, including the Internal Reliability (minimum detectable bias) and External Reliability analysis (influence of undetected bias).

The proposed Quality Control (QC) methods are residual-based. After calculating the estimated states/variables and their covariance within the estimation process, each model's residuals and their covariance matrix can be obtained.

For the filtering, residual of models can be acquired at every measurement update step. At the time k in the KF, the residual v_{k^f} and its covariance Q_{vk^f} will be:

$$v_k{}^f = \begin{bmatrix} v_{y_k}{}^f \\ v_{x_k}{}^f \end{bmatrix} = \begin{bmatrix} -R_k Q_{dy_k}^{-1} dy_k \\ G_k dy_k \end{bmatrix} \tag{2.26}$$

$$Q_{vk^f} = \begin{bmatrix} R_k Q_{dy_k}^{-1} R_k & -R_k G_k^T \\ -G_k R_k & G_k Q_{dyk} G_k \end{bmatrix} \tag{2.27}$$

$v_{y_k{}^f}$ and $v_{x_k{}^f}$ represent the residual vector of the measurement y and predicted sate parameters $\bar{x}$ estimated by Kalman Filter (KF) step in epoch k .

For the FLS the residual $\hat{v}$ for the models and its covariance $Q_{\hat{v}}$ can be acquired by:

$$\hat{v} = \begin{bmatrix} \hat{v}_{meas} \\ \hat{v}_{dync} \end{bmatrix} = A\Delta\hat{X} - L = \begin{bmatrix} A_{meas}\Delta\hat{X} - L_{meas} \\ A_{dync}\Delta\hat{X} - L_{dync} \end{bmatrix} \tag{2.28}$$

$$Q_{\hat{v}} = \begin{bmatrix} Q_{\hat{v}_{meas}} & Q_{\hat{v}_{md}} \\ Q_{\hat{v}_{dm}} & Q_{\hat{v}_{dync}} \end{bmatrix} = D - AQ_{\widehat{\Delta X}}A^{\mathrm{T}} \tag{2.29}$$

these estimated residuals for measurement model and dynamic model can be derived to:

$$\hat{v}_{meas} = \begin{bmatrix} -R_1 Q_{dy_1}^{-1} dy_1 + A_1 J_1 (x_2{}^s - \bar{x}_2) \\ -R_2 Q_{dy_2}^{-1} dy_2 + A_2 J_2 (x_3{}^s - \bar{x}_3) \\ \vdots \\ -R_{T-1} Q_{dy_{T-1}}^{-1} dy_{T-1} + A_{T-1} J_{T-1} (x_T{}^s - \bar{x}_T) \\ -R_T Q_{dy_T}^{-1} dy_T \end{bmatrix}$$

$$= \begin{bmatrix} v_{y_1}{}^f + A_1 J_1 (x_2{}^s - \bar{x}_2) \\ v_{y_2}{}^f + A_2 J_2 (x_3{}^s - \bar{x}_3) \\ \vdots \\ v_{y_{T-1}}{}^f + A_{T-1} J_{T-1} (x_T{}^s - \bar{x}_T) \\ v_{y_T}{}^f \end{bmatrix} = \begin{bmatrix} v_{y_1}{}^s \\ v_{y_2}{}^s \\ \vdots \\ v_{y_{T-1}}{}^s \\ v_{y_T}{}^s \end{bmatrix} \tag{2.30}$$

$$\hat{v}_{dync} = \begin{bmatrix} G_1 dy_1 + J_1 (x_2{}^s - \bar{x}_2) \\ Q_2 Q_{\bar{x}_2}^{-1} (x_2{}^s - \bar{x}_2) \\ Q_3 Q_{\bar{x}_3}^{-1} (x_3{}^s - \bar{x}_3) \\ \vdots \\ Q_T Q_{\bar{x}_T}^{-1} (x_T{}^s - \bar{x}_T) \end{bmatrix} = \begin{bmatrix} v_{x_1}{}^f + J_1 (x_2{}^s - \bar{x}_2) \\ B_2 (x_2{}^s - \bar{x}_2) \\ B_3 (x_3{}^s - \bar{x}_3) \\ \vdots \\ B_T (x_T{}^s - \bar{x}_T) \end{bmatrix}$$

$$= \begin{bmatrix} v_{dync_1}{}^s \\ v_{dync_2}{}^s \\ \vdots \\ v_{dync_{T-1}}{}^s \\ v_{dync_T}{}^s \end{bmatrix} \tag{2.31}$$

in Equation 2.31, $B_k = Q_k Q_{\bar{x}_k}^{-1}$;

As FLS has the same solution of state parameters to Kalman smoothing process, their residuals are also the same, hence $v_{y_1}{}^s$ and $v_{dync_1}{}^s$ are the residual vector of the measurement and the dynamic model estimated by Kalman Smoother (KS) at epoch *k*. At the last epoch, the smoothed states are equal to the filtered states. Hence they have the same residual of the measurements y_T .

For a specific epoch during Kalman smoothing, use epoch *k* as an example, the residual will be:

$$v_k{}^s = \begin{bmatrix} v_{y_k}{}^s \\ v_{dync_k}{}^s \end{bmatrix} \tag{2.32}$$

with the residuals for measurement and dynamic model be formulated as:

$$v_{y_k}{}^s = \begin{cases} v_{y_k}{}^f + A_k J_k (x_{k+1}{}^s - \bar{x}_{k+1}) = v_{y_k}{}^f + A_k J_k dx_{k+1}{}^s, & k < T \\ v_{y_k}{}^f, & k = T \end{cases} \tag{2.33}$$

$$v_{dync_k}{}^s = \begin{cases} v_{x_k}{}^f + J_k (x_{k+1}{}^s - \bar{x}_{k+1}) = v_{x_k}{}^f + J_k dx_{k+1}{}^s, & k = 1 \\ B_k (x_k{}^s - \bar{x}_k) = B_k dx_k{}^s, & k > 1 \end{cases} \tag{2.34}$$

therefore, these two equations also show the relationship between the residuals estimated by the three methods (FLS, KF and KS).

$Q_{\hat{v}}$ is consisted by 4 parts, the diagonal parts are $Q_{\hat{v}_{meas}}$ and $Q_{\hat{v}_{dync}}$ that are the cofactor matrix associated to $\hat{v}_{meas}$ and $\hat{v}_{dync}$ individually. $Q_{\hat{v}_{meas}}$ and $Q_{\hat{v}_{dync}}$ are formulated as:

$$Q_{\hat{v}_{meas}} = \begin{bmatrix} R_1 - A_1 Q_{x_1}{}^s A_1{}^{\mathrm{T}} & -A_1 J_1 Q_{x_2}{}^s A_2{}^{\mathrm{T}} & -A_1 J_1 J_2 Q_{x_3}{}^s A_2{}^{\mathrm{T}} & \cdots & -A_1 J_1 \dots J_{T-1} Q_{x_T}{}^s A_T{}^{\mathrm{T}} \\ -A_2 Q_{x_2}{}^s J_1{}^{\mathrm{T}} A_1{}^{\mathrm{T}} & R_2 - A_2 Q_{x_2}{}^s A_2{}^{\mathrm{T}} & -A_2 J_2 Q_{x_3}{}^s A_3{}^{\mathrm{T}} & \cdots & -A_2 J_2 \dots J_{T-1} Q_{x_T}{}^s A_T{}^{\mathrm{T}} \\ -A_3 Q_{x_3}{}^s J_2{}^{\mathrm{T}} J_1{}^{\mathrm{T}} A_1{}^{\mathrm{T}} & -A_3 Q_{x_3}{}^s J_2{}^{\mathrm{T}} A_2{}^{\mathrm{T}} & R_3 - A_3 Q_{x_3}{}^s A_3{}^{\mathrm{T}} & \cdots & -A_3 J_3 \dots J_{T-1} Q_{x_T}{}^s A_T{}^{\mathrm{T}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -A_T Q_{x_T}{}^s J_{T-1}{}^{\mathrm{T}} \dots J_1{}^{\mathrm{T}} A_1{}^{\mathrm{T}} & -A_T Q_{x_T}{}^s J_{T-1}{}^{\mathrm{T}} \dots J_2{}^{\mathrm{T}} A_2{}^{\mathrm{T}} & -A_T Q_{x_T}{}^s J_{T-1}{}^{\mathrm{T}} \dots J_3{}^{\mathrm{T}} A_3{}^{\mathrm{T}} & \cdots & R_T - A_T Q_{x_T}{}^s A_T{}^{\mathrm{T}} \end{bmatrix} \tag{2.35}$$

$$Q_{\hat{v}_{dync}} = \begin{bmatrix} B_1 dQ_{x_1} B_1{}^{\mathrm{T}} & B_1 J_1 dQ_{x_2} B_2{}^{\mathrm{T}} & B_1 J_1 J_2 dQ_{x_3} B_3{}^{\mathrm{T}} & \cdots & B_1 J_1 \dots J_{T-1} dQ_{x_T} B_T{}^{\mathrm{T}} \\ B_2 dQ_{x_2} J_1{}^{\mathrm{T}} B_1{}^{\mathrm{T}} & B_2 dQ_{x_2} B_2{}^{\mathrm{T}} & B_2 J_2 dQ_{x_3} B_3{}^{\mathrm{T}} & \cdots & B_2 J_2 \dots J_{T-1} dQ_{x_T} B_T{}^{\mathrm{T}} \\ B_3 dQ_{x_3} J_2{}^{\mathrm{T}} J_1{}^{\mathrm{T}} B_1{}^{\mathrm{T}} & B_3 dQ_{x_3} J_2{}^{\mathrm{T}} B_2{}^{\mathrm{T}} & B_3 dQ_{x_3} B_3{}^{\mathrm{T}} & \cdots & B_3 J_3 \dots J_{T-1} dQ_{x_T} B_T{}^{\mathrm{T}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ B_T dQ_{x_T} J_{T-1}{}^{\mathrm{T}} \dots J_1{}^{\mathrm{T}} B_1{}^{\mathrm{T}} & B_T dQ_{x_T} J_{T-1}{}^{\mathrm{T}} \dots J_2{}^{\mathrm{T}} B_2{}^{\mathrm{T}} & B_T dQ_{x_T} J_{T-1}{}^{\mathrm{T}} \dots J_3{}^{\mathrm{T}} B_3{}^{\mathrm{T}} & \cdots & B_T dQ_{x_T} B_T{}^{\mathrm{T}} \end{bmatrix} \tag{2.36}$$

here $B_k = Q_k Q_{\bar{x}_k}^{-1}$; $dQ_{x_k} = Q_{\bar{x}_k} - Q_{x_k}{}^s$ represents the difference between the smoothed state covariance and predicted state covariance at epoch *k*.

Because $\hat{v}_{meas}$ and $\hat{v}_{dync}$ are correlated to each other, according to the law of error propagation, the off-diagonal part of $Q_{\hat{v}}$ will be $Q_{\hat{v}_{md}}$ and $Q_{\hat{v}_{dm}}$, here $Q_{\hat{v}_{dm}} = Q_{\hat{v}_{md}}{}^{\mathrm{T}}$.

$$Q_{\hat{v}_{dm}} = \begin{bmatrix} -A_1 Q_{x_1}{}^s B_1{}^{\mathrm{T}} & A_1 J_1 dQ_{x_2} B_2{}^{\mathrm{T}} & A_1 J_1 J_2 dQ_{x_3} B_3{}^{\mathrm{T}} & \cdots & A_1 J_1 \dots J_{T-1} dQ_{x_T} B_T{}^{\mathrm{T}} \\ -A_2 Q_{x_1}{}^s J_1{}^{\mathrm{T}} B_1{}^{\mathrm{T}} & -A_2 Q_{x_2}{}^s B_2{}^{\mathrm{T}} & A_2 J_2 dQ_{x_3} B_3{}^{\mathrm{T}} & \cdots & A_2 J_2 \dots J_{T-1} dQ_{x_T} B_T{}^{\mathrm{T}} \\ -A_3 Q_{x_1}{}^s J_2{}^{\mathrm{T}} J_1{}^{\mathrm{T}} B_1{}^{\mathrm{T}} & -A_3 Q_{x_1}{}^s J_2{}^{\mathrm{T}} B_2{}^{\mathrm{T}} & -A_3 Q_{x_3}{}^s B_3{}^{\mathrm{T}} & \cdots & A_3 J_3 \dots J_{T-1} dQ_{x_T} B_T{}^{\mathrm{T}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -A_T Q_{x_T}{}^s J_{T-1}{}^{\mathrm{T}} \dots J_1{}^{\mathrm{T}} B_1{}^{\mathrm{T}} & -A_T Q_{x_T}{}^s J_{T-1}{}^{\mathrm{T}} \dots J_2{}^{\mathrm{T}} B_2{}^{\mathrm{T}} & -A_1 Q_{x_T}{}^s J_{T-1}{}^{\mathrm{T}} \dots J_3{}^{\mathrm{T}} B_3{}^{\mathrm{T}} & \cdots & -A_T Q_{x_T}{}^s B_T{}^{\mathrm{T}} \end{bmatrix} \quad (2.37)$$

the diagonal matrix of $Q_{\hat{v}_{meas}}$ and $Q_{\hat{v}_{dync}}$ from FLS is equal to the covariance matrix of residual estimated from Kalman smoothing for each epoch, therefore, when smoothing back to an epoch (epoch *k* as example), the residual covariance for measurement and process model will be:

$$Q_{\hat{v}_{y_k}}{}^s = R_k - A_k Q_{x_k}{}^s A_k{}^{\mathrm{T}} \quad (2.38)$$

$$Q_{\hat{v}_{dync_k}}{}^s = B_k dQ_{x_k} B_k{}^{\mathrm{T}} \quad (2.39)$$

2.3.2 Outlier Detection

Detection is a model test that is used to diagnose whether an unspecified model error occurred. In order to test if there is a model error (∇), the corresponding null and alternative hypotheses are formed as:

$$H_0\colon\ E(l) = A\Delta X \quad (2.40)$$

$$H_a\colon\ E(l) = A\Delta X + \nabla \quad (2.41)$$

for the full least squares, the T_{test} test statistic can be expressed with the FLS residual and be formulated as (Knight et al. 2010):

$$T_{test} = \frac{\hat{v}^{\mathrm{T}} P \hat{v}}{r {\sigma_0}^2} \quad (2.42)$$

here *r* is the number of redundancy which equals to $\sum_{k=1}^{T} m_k$.

This expression however is not in a form that facilities recursive testing. Since the P matrix is a block diagonal matrix as $P = blockdiag(R_1^{-1}, \dots, R_T^{-1}, Q_1^{-1}, \dots, Q_T^{-1})$, T_{test} for KF and KS step at epoch *k* can be respectively expressed as:

$$T_{test_k}^f = \frac{v_{y_k}^{f^T} R_k^{-1} v_{y_k}^f + v_{x_k}^{f^T} Q_{\bar{x}_k}^{-1} v_{x_k}^f}{r_k \sigma_0^2} = \frac{dy_k^T Q_{dy_k}^{-1} dy_k}{r_k \sigma_0^2} \tag{2.43}$$

$$T_{test_k}^s = \frac{v_{y_k}^{s^T} R_k^{-1} v_{y_k}^s + v_{dync_k}^{s^T} Q_k^{-1} v_{dync_k}^s}{r_k \sigma_0^2} \tag{2.44}$$

During these recursive tests, the redundancy number r_k will equal to the local redundancy m_k at each epoch.

2.3.3 Identification

According to Hewitson and Wang (2010), a fault/outlier in the measurement or the predicted states in the KF process can be estimated as:

$$\hat{\nabla}_{\mathrm{k,i}}{}^f = \begin{cases} \left(e_{k,i}^{\mathrm{T}} Q_{dy_k}^{-1} e_{k,i}\right)^{-1} {e_{k,i}}^{\mathrm{T}} Q_{dy_k}^{-1} dy_k, & (1 \le i \le m_k) \\ \left(f_{k,i}^{\mathrm{T}} H_k^{\mathrm{T}} Q_{dy_k}^{-1} H_k f_{k,i}\right)^{-1} f_{k,i}^{\mathrm{T}} H_k^{\mathrm{T}} Q_{dy_k}^{-1} dy_k, & (1 \le i \le n_k) \end{cases} \tag{2.45}$$

with the variance of its estimated error in a measurement model as:

$$Q_{\hat{\nabla}_{\mathrm{k,i}}{}^f} = \begin{cases} \left(e_{k,i}^{\mathrm{T}} Q_{dy_k}^{-1} e_{k,i}\right)^{-1}, & (1 \le i \le m_k) \\ \left(f_{k,i}^{\mathrm{T}} H_k^{\mathrm{T}} Q_{dy_k}^{-1} H_k f_{k,i}\right)^{-1}, & (1 \le i \le n_k) \end{cases} \tag{2.46}$$

here $e_{k,i}$ and $f_{k,i}$ are the $m_k \times 1$ and $n_k \times 1$ vector with the *i*th element in them equals to one, and other elements equal to zero.

Then, the fault/outlier identification statistic (w test) can be formulated as follows:

$$w_{k,i}{}^{f} = \frac{\widehat{\nabla}_{\mathrm{k,i}}{}^{f}}{\sigma_{\widehat{\nabla}_{\mathrm{k,i}}{}^{f}}} = \begin{cases} \dfrac{e_{k,i}{}^{\mathrm{T}} Q_{dy_k}^{-1} dy_k}{\sigma_0 \sqrt{e_{k,i}^{\mathrm{T}} Q_{dy_k}^{-1} e_{k,i}}}, & (1 \le i \le m_k) \\ \dfrac{f_{k,i}^{\mathrm{T}} H_k^{\mathrm{T}} Q_{dy_k}^{-1} dy_k}{\sigma_0 \sqrt{f_{k,i}^{\mathrm{T}} H_k^{\mathrm{T}} Q_{dy_k}^{-1} H_k f_{k,i}}}, & (1 \le i \le n_k) \end{cases} \tag{2.47}$$

here, k represents the current time epoch, i represent the index of measurement or state parameter. m_k and n_k are the number of measurement model and state parameter, and σ_0 is the priori variance factor. H_k is the design matrix of Kalman filter that relating the measurements to the state parameters. If the largest $|w_{k,i}|$ is larger than 3.29, it indicates there may be an outlier in the ith measurement (Wang and Knight 2012).

Similarly, the possible error/outlier can also be estimated through the KS or FLS process. In the step of Identification of FLS, the alternative hypothesis in Equation 2.41 is decomposed into:

$$H_{a_i}\colon\ E(l) = A\Delta X + h_i \nabla_i \tag{2.48}$$

here, h_i is the $F_{all} \times 1$ vector with its ith element equal to one and all other elements equal to zero; ∇_i is the size of the model error in the functional model l_i, and $i = 1,2,\dots,F_{all}$. During the full least squares process, the estimated error and its cofactor will be:

$$\widehat{\nabla}_i = Q_{\widehat{\nabla}_i} h_i^T P Q_v P L \tag{2.49}$$

$$\mathrm{Q}_{\widehat{\nabla}_{\mathrm{i}}} = (h_i^T P Q_v P h_i)^{-1} \tag{2.50}$$

in case the alternative hypotheses are related only to the model errors in the measurements models, or in the dynamic models, the structure of the vector h_i is further expressed as:

$$h_i = \begin{cases} (h_{m_i}, 0), & (1 \le i \le m_{all}) \\ \left(0, h_{d_j}\right), & (j = i - m_{all}, m_{all} < i < F_{all}) \end{cases} \tag{2.51}$$

where h_{m_i} and h_{d_j} are the $m_{all} \times 1$ and $n_{all} \times 1$ vector with its ith element in h_{m_i} and jth element in h_{m_i} are equal to one and all other elements equal to zero. Equation 2.49 and 2.50 will be solved as:

$$\hat{\nabla}_i = \begin{cases} \left(h_{m_i}^{\mathrm{T}} R^{-1} Q_{\hat{v}_{meas}} R^{-1} h_{m_i}\right)^{-1} h_{m_i}^{\mathrm{T}} R^{-1} Q_{\hat{v}_{meas}} R^{-1} L_{meas}, & (1 \le i \le m_{all}) \\ \left(h_{d_j}^{\mathrm{T}} Q^{-1} Q_{\hat{v}_{dync}} Q^{-1} h_{d_j}\right)^{-1} h_{d_j}^{\mathrm{T}} Q^{-1} Q_{\hat{v}_{dync}} Q^{-1} L_{dync}, & (\mathrm{j} = i - m_{all}, m_{all} < i < F_{all}) \end{cases} \tag{2.52}$$

$$\mathrm{Q}_{\hat{\nabla}_i} = (h_i^T P Q_v P h_i)^{-1} = \begin{cases} \left(h_{m_i}^{\mathrm{T}} R^{-1} Q_{\hat{v}_{meas}} R^{-1} h_{m_i}\right)^{-1}, & (1 \le i \le m_{all}) \\ \left(h_{d_j}^{\mathrm{T}} Q^{-1} Q_{\hat{v}_{dync}} Q^{-1} h_{d_j}\right)^{-1}, & (\mathrm{j} = i - m_{all}, m_{all} < i < F_{all}) \end{cases} \tag{2.53}$$

The w test for FLS will be:

$$w_i = \frac{\hat{\nabla}_i}{\mathrm{Q}_{\hat{\nabla}_i}} = \begin{cases} \dfrac{h_{m_i}^{\mathrm{T}} R^{-1} Q_{\hat{v}_{meas}} R^{-1} L_{meas}}{\sigma_0 \sqrt{h_{m_i}^{\mathrm{T}} R^{-1} Q_{\hat{v}_{meas}} R^{-1} h_{m_i}}}, & (1 \le i \le m_{all}) \\ \dfrac{h_{d_j}^{\mathrm{T}} Q^{-1} Q_{\hat{v}_{dync}} Q^{-1} L_{dync}}{\sigma_0 \sqrt{h_{d_j}^{\mathrm{T}} Q^{-1} Q_{\hat{v}_{dync}} Q^{-1} h_{d_j}}}, & (\mathrm{j} = i - m_{all}, m_{all} < i < F_{all}) \end{cases} \tag{2.54}$$

these expressions will be the basis for recursive identification, e.g. conducting error identification procedure recursively under the KS process with introducing $e_{k,i}$ and $f_{k,i}$, similar to that in KF. Equation 2.52 and 2.53 will be further derived as followed:

$$\hat{\nabla}_{\mathrm{k,i}}^{\ s} = \begin{cases} \left(e_{k,i}^{\mathrm{T}} (Q_{dy_k}^{-1} + C_k dQ_{x_{k+1}} C_k^T) e_{k,i}\right)^{-1} e_{k,i}{}^{\mathrm{T}} (Q_{dy_k}^{-1} dy_k - C_k dx_{k+1}{}^{s}), (1 \le i \le m_k) \\ \left(f_{k,i}^{\mathrm{T}} Q_{\bar{x}_k}^{-1} dQ_{x_k} Q_{\bar{x}_k}^{-1} f_{k,i}\right)^{-1} f_{k,i}^{\mathrm{T}} Q_{\bar{x}_k}^{-1} (-dx_k{}^{s}), \ (1 \le i \le n_k), \end{cases} \quad \begin{matrix} \\ (1 \le i \le n_k) \end{matrix} \tag{2.55}$$

$$Q_{\hat{\nabla}_{\mathrm{k,i}}{}^{s}} = \begin{cases} \left(e_{k,i}^{\mathrm{T}} (Q_{dy_k}^{-1} + C_k dQ_{x_{k+1}} C_k^T) e_{k,i}\right)^{-1}, & (1 \le i \le m_k) \\ \left(f_{k,i}^{\mathrm{T}} Q_{\bar{x}_k}^{-1} dQ_{x_k} Q_{\bar{x}_k}^{-1} f_{k,i}\right)^{-1}, \ (1 \le i \le n_k), & (1 \le i \le n_k) \end{cases} \tag{2.56}$$

The C_k matrix is defined as: $C_k = R_k^{-1} H_k J_k$.

The Corresponding w test in KS will be:

$$w_{k,i}{}^{s} = \frac{\widehat{\nabla}_{\mathrm{k,i}}{}^{s}}{\sigma_{\widehat{\nabla}_{\mathrm{k,i}}{}^{s}}} = \begin{cases} \dfrac{e_{k,i}{}^{\mathrm{T}}(Q_{dy_k}^{-1} dy_k - C_k dx_{k+1}{}^{s})}{\sigma_0 \sqrt{e_{k,i}^{\mathrm{T}}(Q_{dy_k}^{-1} + C_k dQ_{x_{k+1}} C_k^{T}) e_{k,i}}}, & (1 \le i \le m_k) \\ \dfrac{-f_{k,i}^{\mathrm{T}} Q_{\bar{x}_k}^{-1} dx_k{}^{s}}{\sigma_0 \sqrt{f_{k,i}^{\mathrm{T}} Q_{\bar{x}_k}^{-1} dQ_{x_k} Q_{\bar{x}_k}^{-1} f_{k,i}}}, & (1 \le i \le n_k) \end{cases} \tag{2.57}$$

2.3.4 Adaption

Once the specific model error is identified, its influence on the state parameter estimation should be removed, which can be performed by estimating the state parameters based on Equation 2.41. Therefore, the full least squares estimators of the adapted state parameters can be derived as:

$$\Delta X_i{}^{inf} = Q_{\widehat{\Delta X}} A^{\mathrm{T}} P h_i \widehat{\nabla}_i, \;\; i = 1,2,\dots,F_{all} \tag{2.58}$$

here $\Delta X_i{}^{inf}$ represents the influence of the estimated error in the ith functional model ($i = 1,2,\dots,F_{all}$) on the estimated correction by FLS. Therefore, the adapted results will be:

$$\hat{X}^{a} = X_{app} + \Delta\hat{X} - \Delta X_i{}^{inf} \tag{2.59}$$

while the KF adapted state parameters can be derived as:

$$x_k{}^{fa} = \begin{cases} \hat{x}_k - G_k e_{k,i} \widehat{\nabla}_{k,i}{}^{f}, & (1 \le i \le m_k) \\ \hat{x}_k - (I - G_k H_k) f_{k,i} \widehat{\nabla}_{k,i}{}^{f}, & (1 \le i \le n_k) \end{cases} \tag{2.60}$$

here G_k is the Kalman gain matrix which is defined in Equation 2.8.

For the KS process, the adapted state parameters are:

$$x_k{}^{sa} = \begin{cases} x_k{}^{s} - Q_{x_k{}^{s}} H_k{}^{\mathrm{T}} R_k^{-1} H_k e_{k,i} \widehat{\nabla}_{k,i}{}^{s}, & (1 \le i \le m_k) \\ x_k{}^{s} - Q_{x_k{}^{s}} Q_{\bar{x}_k}{}^{-1} f_{k,i} \widehat{\nabla}_{k,i}{}^{s}, & (1 \le i \le n_k) \end{cases} \tag{2.61}$$

their cofactor matrix read as:

$$Q_{\widehat{\Delta X}}{}^{a} = Q_{\widehat{\Delta X}} + Q_{\widehat{\Delta X}} A^{\mathrm{T}} P h_i Q_{\widehat{\Delta X}} h_i{}^{\mathrm{T}} P A Q_{\widehat{\Delta X}}, i = 1,2, \dots, F_{all} \tag{2.62}$$

and for KF:

$$Q_{\hat{x}^f}{}^{a} = \begin{cases} Q_{x_k{}^f} + G_k e_{k,i} \mathrm{Q}_{\hat{\nabla}_{\mathrm{k,i}}}{}^{f} e_{k,i}{}^{\mathrm{T}} G_k{}^{\mathrm{T}}, & (1 \le i \le m_k) \\ Q_{x_k{}^f} + (I - G_k H_k) f_{k,i} \mathrm{Q}_{\hat{\nabla}_{\mathrm{k,i}}}{}^{f} f_{k,i}{}^{\mathrm{T}} (I - G_k H_k)^{\mathrm{T}}, & (1 \le i \le n_k) \end{cases} \tag{2.63}$$

And for KS:

$$Q_{\hat{x}^s}{}^{a} = \begin{cases} Q_{x_k{}^s} + Q_{x_k{}^s} H_k{}^{\mathrm{T}} R_k^{-1} H_k e_{k,i} Q_{\hat{\nabla}_{k,i}}{}^{s} e_{k,i}{}^{\mathrm{T}} H_k{}^{\mathrm{T}} R_k^{-1} H_k Q_{x_k{}^s}, & (1 \le i \le m_k) \\ Q_{x_k{}^s} + Q_{x_k{}^s} Q_{\bar{x}_k}{}^{-1} f_{k,i} Q_{\hat{\nabla}_{k,i}}{}^{s} f_{k,i}{}^{\mathrm{T}} Q_{\bar{x}_k}{}^{-1} Q_{x_k{}^s}, & (1 \le i \le n_k) \end{cases} \tag{2.64}$$

2.3.5 Reliability

Two reliability measures will be taken into consideration. MDB is the minimum detectable bias. The MDB in the i$^{\mathrm{th}}$ measurement or predicted states in KF step can be obtained as:

$$MDB_{k,i}{}^{f} = \begin{cases} \dfrac{\delta_d \sigma_0}{\sqrt{e_{k,i}{}^{T} Q_{dy_k}^{-1} e_{k,i}}}, & (1 \le i \le m_k) \\ \dfrac{\delta_d \sigma_0}{\sqrt{f_{k,i}^{T} H_k^{T} Q_{d_{yk}}^{-1} H_k f_{k,i}}}, & (1 \le i \le n_k) \end{cases} \tag{2.65}$$

here δ_{d} is the shift of the outlier statistic determined by Type I and Type II errors [Knight and Wang, 2010].

According to the MDB concept, there could be an outlier with the size very close to the MDB value and could not be detected. However, this outlier may still influence the final

estimation solution of position. This influence, so-called External Reliability is critically important in evaluating the reliability of the dynamic system, such as the localization system for autonomous driving. External Reliability can be determined by evaluating the effect of the MDB on the estimated parameters as:

$$ER_{k,i}{}^{f} = \begin{cases} G_k e_{k,i} MDB_{k,i}{}^{f}, & (1 \le i \le m_k) \\ (I - G_k H_k) f_{k,i} MDB_{k,i}{}^{f}, & (1 \le i \le n_k) \end{cases} \tag{2.66}$$

similarly, the MDB and ER in the FLS and KS step can be derived as:

for FLS, the MDB is derived as:

$$MDB_i = \frac{\delta_d \sigma_0}{\sqrt{h_i{}^{T} \mathrm{P} Q_v \mathrm{P} h_i}} = \begin{cases} \dfrac{\delta_d \sigma_0}{\sqrt{h_{m_i}^{\mathrm{T}} R^{-1} Q_{\hat{v}_{meas}} R^{-1} h_{m_i}}}, & (1 \le i \le m_{all}) \\ \dfrac{\delta_d \sigma_0}{\sqrt{h_{d_j}^{\mathrm{T}} Q^{-1} Q_{\hat{v}_{dync}} Q^{-1} h_{d_j}}}, & (j = i - m_{all}, m_{all} < i < F_{all}) \end{cases} \tag{2.67}$$

$$ER_i = Q_{\widehat{\Delta X}} A^{\mathrm{T}} P h_i MDB_i \tag{2.68}$$

thus for KS:

$$MDB_{k,i}{}^{s} = \begin{cases} \dfrac{\delta_d \sigma_0}{\sqrt{e_{k,i}{}^{T} (Q_{dy_k}^{-1} + C_k d Q_{x_k+1} C_k^{\mathrm{T}}) e_{k,i}}}, & (1 \le i \le m_k) \\ \dfrac{\delta_d \sigma_0}{\sqrt{f_{k,i}^{T} Q_{\bar{x}_k}{}^{-1} d Q_{x_k} Q_{\bar{x}_k}{}^{-1} f_{k,i}}}, & (1 \le i \le n_k) \end{cases} \tag{2.69}$$

$$ER_{k,i}{}^{s} = \begin{cases} Q_{x_k}{}^{s} H_k{}^{\mathrm{T}} R_k^{-1} e_{k,i} MDB_{k,i}{}^{s}, & (1 \le i \le m_k) \\ Q_{x_k}{}^{s} Q_{\bar{x}_k}{}^{-1} f_{k,i} MDB_{k,i}{}^{s}, & (1 \le i \le n_k) \end{cases} \tag{2.70}$$

2.4 Numerical Analysis for the proposed Quality Control Method

2.4.1 Linear Position Estimation Case

The first numerical test is taken with a simulated 2D positioning case with measured position and constant velocity dynamic model (case 1). The tracked object moves with a sensor on it to give measurements of the target's position in Cartesian coordinates *x* and *y*. The state variables contain the position (x_k, y_k), velocities (vx_k, vy_k) on the epoch *k*. The data used in this case is simulated. This is a linear KF/KS case and the measurement matrix is set to:

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.71}$$

and the state transition matrix is set as:

$$\Phi_k = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.72}$$

The measured positions have a standard deviation of 1 meter while the covariance of system driving noise is set as diag(0.1). In this simulation case, the time interval is 1s, thus $\Delta t = 1$. Kalman filtering and smoothing are processed recursively under the above assumption for 20 epochs, and full least squares is done with the measurement and dynamic information within all the 20 epochs. The following figures show the estimation results by the three methods (KF, KS and FLS).

2.4.1.1 KF, KS and FLS estimation results

The following 4 figures show that the FLS and KS method has exactly the same estimation results, indicating the equivalence of Kalman smoother and full least squares. According to the first two figures, Kalman smoother's result are smoother and more accurate than Kalman filter's. The root mean squared estimated errors (RMSE) between the estimated and referenced position are 0.764 and 0.545 meters by KF and KS, respectively. The results also prove the advantage of Kalman smoothing to filtering in the precision of estimation, which can be obviously seen from Figure 2.3 and Figure 2.4 that

KS has lower standard deviation value of the final estimation than KF. It should be noticed, when the measurements are more accurate (e.g. lower ε_k), the estimation results by KF and KS are getting closer to each other.

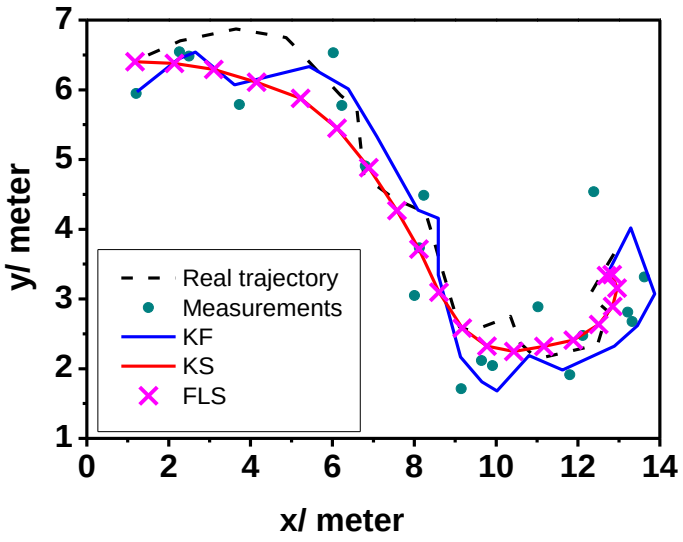


Figure 2.1 Measured position and estimated position states by KF, KS and FLS

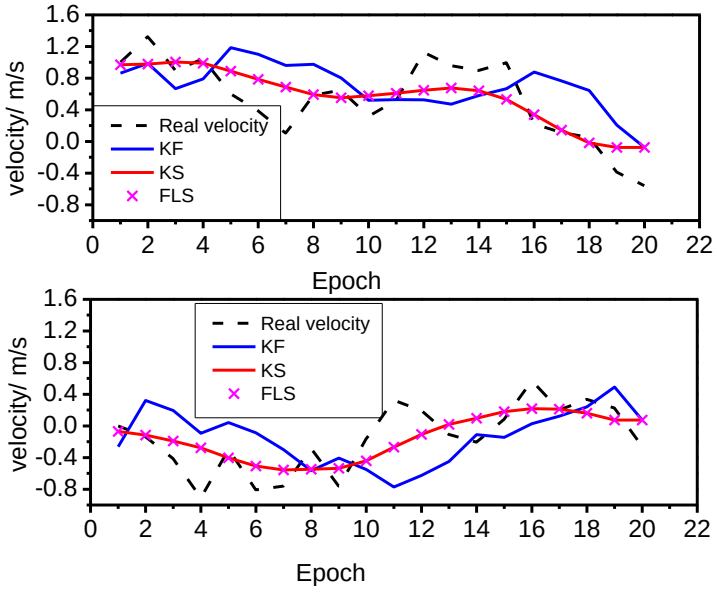


Figure 2.2 Estimated velocity states by KF, KS and FLS. [Top] velocity on *x* axis. [Bottom] velocity on *y* axis

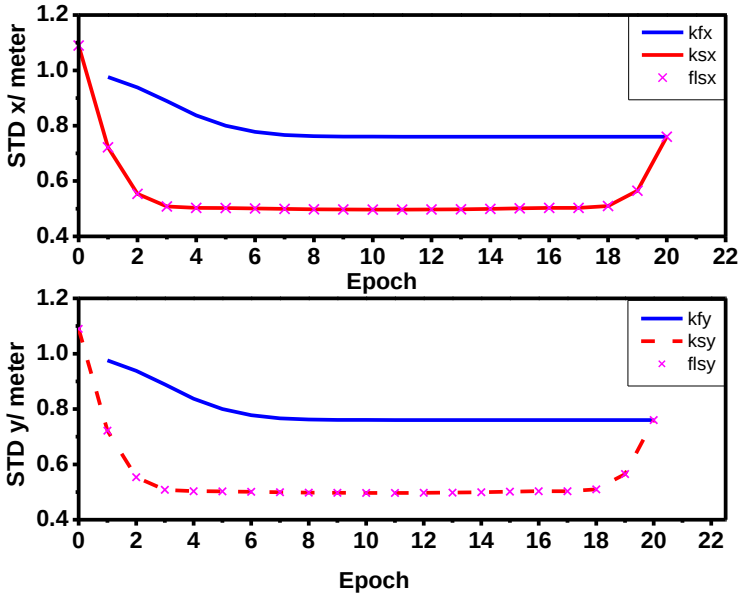


Figure 2.3 Standard deviation of estimated position [Top] *x* axis. [Bottom] *y* axis

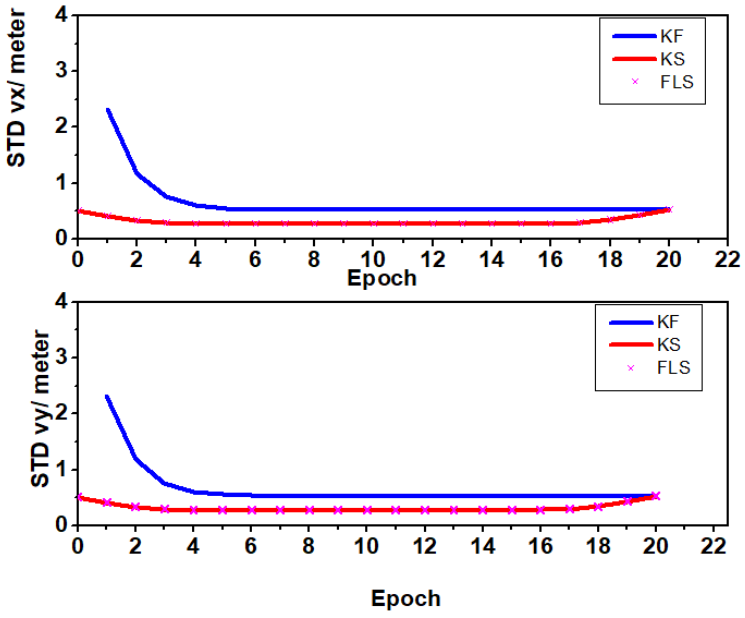


Figure 2.4 Standard deviation of estimated velocity [Top] *x* axis. [Bottom] *y* axis

2.4.1.2 DIA results with original setting

Figure 2.5 and Figure 2.6 shows the original simulated dataset does not contain unacceptable error/outliers with w test value all under 3.29.

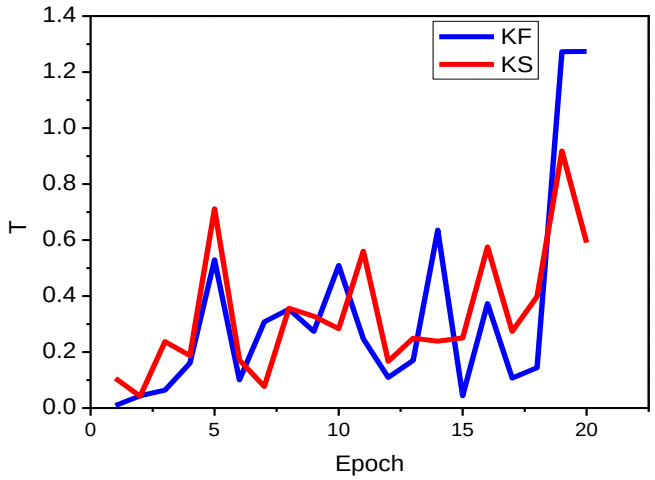


Figure 2.5 T test of simulated dataset by KF and KS

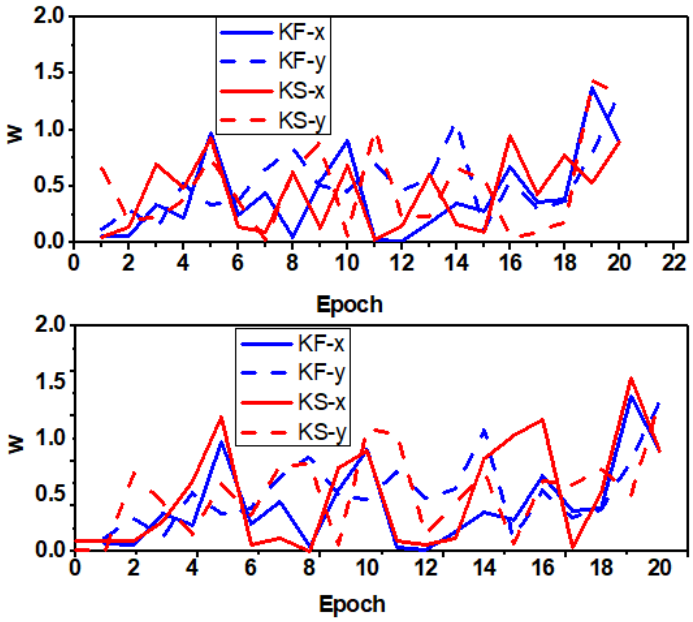


Figure 2.6 W test by KF and KS with simulated dataset. [Top] w statistic test of error in measurement model. [Bottom] w statistic test of error in dynamic state model

KS has better reliability. It has lower MDB value for both measurement and dynamic models, which means the KS method can detect error/outlier with lower value than the KF method. One thing needs to be noticed is that the w statistic test results for dynamic model part in Figure 2.6 only show the test of the position states related models in KF and KS step. The state model in KF step is as:

$$\begin{aligned} \bar{x}_k &= \bar{x}_k + \epsilon_{\bar{x}_k} \\ \bar{y}_k &= \bar{y}_k + \epsilon_{\bar{y}_k} \\ \overline{vx}_k &= \overline{vx}_k + \epsilon_{\overline{vx}_k} \\ \overline{vy}_k &= \overline{vy}_k + \epsilon_{\overline{vy}_k} \end{aligned} \tag{2.73}$$

equation (2.73) is obtained according to section 2.1 and it can be found that the predicted velocity states cannot be tested in KF process as they are not directly measured and there are also not correlated to the predicted position state within the specific epoch. Therefore, the w test results for these velocity predicted states are Nan in KF process. However, in KS process, the w statistic test for states is done with the dynamic process model, e.g.

$$\begin{aligned} x_k &= x_{k-1} + \Delta t v x_k + \tau_{x_k} \\ y_k &= y_{k-1} + \Delta t v y_k + \tau_{y_k} \\ vx_k &= vx_{k-1} + \tau_{vx_k} \\ vy_k &= vy_{k-1} + \tau_{vy_k} \end{aligned} \tag{2.74}$$

hence the velocity model can be tested as the velocity states are correlated to the position states. An example of the correlation coefficient value between the state model can be found in Table 2.1.

Table 2.1Correlation Coefficient value between states by KF method or KS method in epoch 4

	KF					**KS**			
States	x	y	vx	vy	**States**	x	y	vx	vy
x	1	0	Nan	Nan	x	1.0000	0	-0.5381	0
y	0	1	Nan	Nan	y	0	1.0000	0	-0.5381
vx	Nan	Nan	Nan	Nan	vx	-0.5381	0	1.0000	0
vy	Nan	Nan	Nan	Nan	vy	0	-0.5381	0	1.0000

2.4.1.3 DIA with simulated additional outlier

An additional outlier can be added to the measurement to illustrate our quality control (QC) method performance. After adding one outlier of 10 meters at measured position y in epoch 4, the detection and identification results are shown in the following figures.

It can be found from Figure 2.7 and Figure 2.8 that the T test value and W test value are all significantly getting higher at epoch 4 after adding the 10 meters outlier in measured position y. For KS method, the detected outlier only exists in epoch 4 with an estimated value of 9.57 meters. However, for KF method, outlier can also be detected and identified in epoch 5, which means the additional outlier in epoch 4 will also cause error in the following epochs. The estimated error value (9.49 meters) in epoch 5 is even higher than that in epoch 4 (9.05 meters) and its influence on the filtered state of position y is 6.34 meters and -6.09 meters in epoch 4 and 5 (Table 2.2). If the additional error value is larger, this undesirable influence will be higher and exist longer.

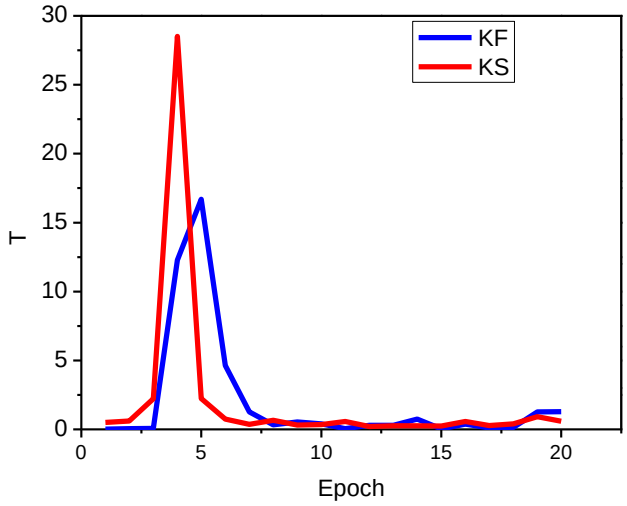


Figure 2.7 T test after adding a 10 meters outlier in measurement of position y at epoch 4

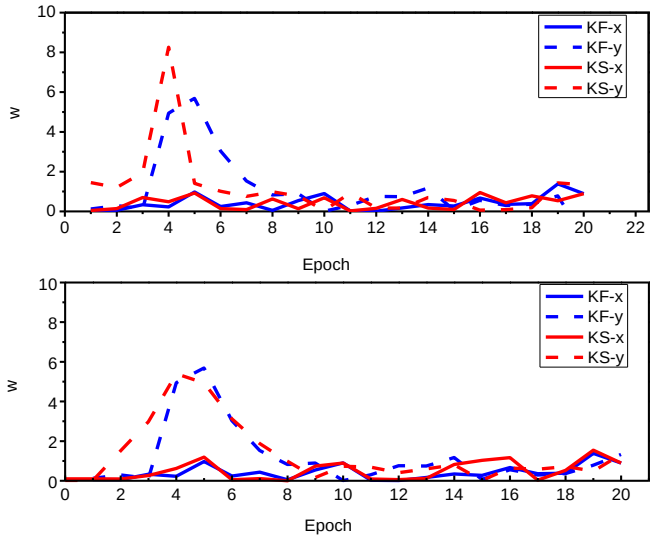


Figure 2.8 W test after adding a 10 meters outlier in measurement of position y at epoch 4. [Top] for measurement model. [Bottom] for dynamic state model

Table 2.2 Influence of the estimated error (exists in epoch 4 measured y) onto the KF and KS estimated state (Position y) result (unit: meter) before and after adding 10 meters outlier

Epoch		4	5	6	7
Before adding error	KF	-0.7232	0.1928	-0.2389	-0.4189
	KS	-0.1096	-0.3103	0.1609	0.3816
After adding error	KF	6.3409	-3.4187	-1.9062	-0.9795
	KS	2.4186	2.5176	1.5763	0.9470

Similarly, in the KF step, the w test has same results on measured position and predicted position due to the close to 1 correlation coefficient (CC) value between the measured and predicted position in this case, therefore, it cannot identify whether the estimated error is in the measurement or in the predicted states. Such as for the test results in epoch 5 by KF, the w value for both measured y and state y are 5.90, therefore it cannot exactly

determine whether the detected error is originally existed in the measurement or it is caused by the influence of the additional outlier in epoch 4.

For KS method, the results are better as the CC value between measurement and dynamic model is lower. Hence in epoch 4 the w test value for measurement (8.27) is larger than that for dynamic model (5.40). In contrast, in epoch 5, the w value for measurement (1.40) is lower than that for dynamic model (4.93), means the outlier is in the measurement in epoch 4. Suppose the outlier is big enough that its influence would be detected in the following epochs. In that case, the KS QC method can still identify it within a dynamic model rather than within the measurement model.

According to Table 2.2, it can also be found that before adding 10 meters outlier, the estimated error by KF and KS method are the random errors exist in the simulated dataset. Their influences on the final estimation of position y are about 0.1-0.7 meters. With the additional outlier of 10m, it caused about 6.34 meters bias of position estimation on *y* direction by the KF method, and its influence will store in the predicted states and will continue cause bias larger than 1 meter until epoch 7. To comparison, KS method is more robust to this additional error as its influence on position estimation are much lower than by KF step. It is because KS process has utilized the measurement and dynamic information after epoch 4 to estimate the states at epoch 4, thus the redundancy is higher, while in KF step, the estimation in epochs 4 and 5 cannot use the following more accurate measurements to correct the bias caused by the additional outlier.

If these influences caused by the additional outlier can be removed (with Adaptation), it can be found that KS has estimation of *y* closer to the reference trajectory. The estimated error can also be removed from the measurement. After removing the error estimated by KS method (9.57 meters in measured y), the T test and W test results (Figures 2.9 and 2.10) at epoch 4 and 5 are better than results before adding the outlier (Figures 2.5 and 2.6).

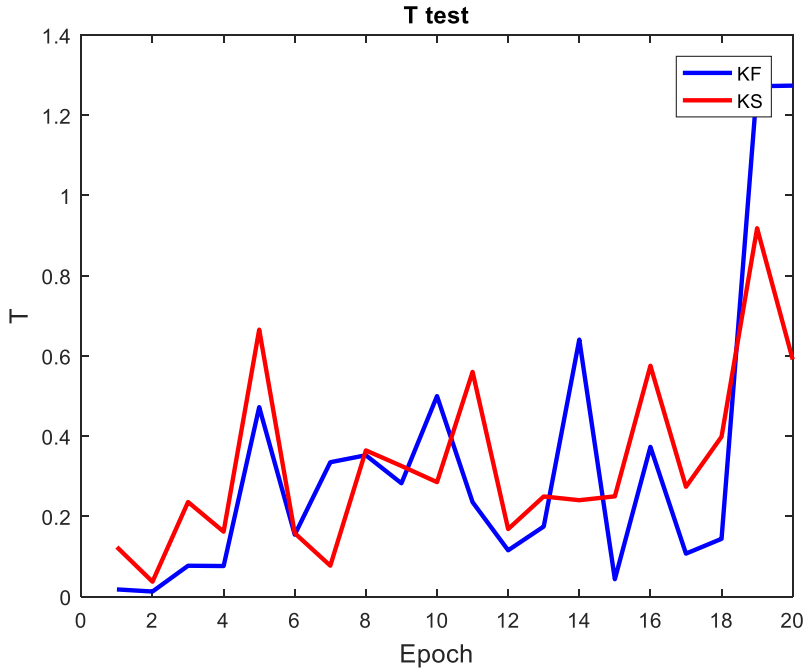


Figure 2.9 T test after removing the outlier estimated by KS

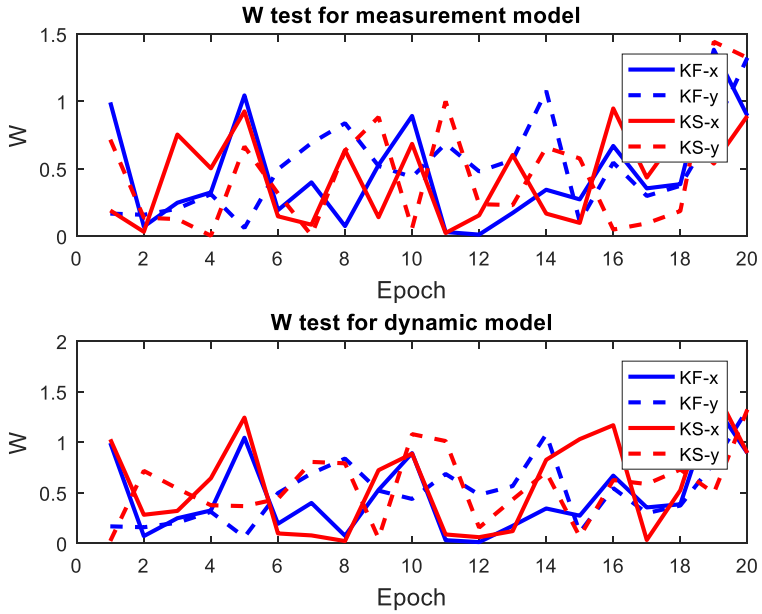


Figure 2.10 W test after removing the outlier estimated by KS. [Top] for measurement model. [Bottom] for dynamic state model

It should also be noticed that as the outlier is added to the measured position, it will also influence the estimation on velocity and cause bias on velocity estimation. By KF method the influenced or biased value is 2.74 m/s, about 5 times of the KF estimated STD value,

by KS, it is 0.26 m/s (Figure 2.11). As the state positions are observed directly while the velocities indirectly, the accuracy of velocity is depended on the accuracy of position measurement. Therefore, the outlier in position measurement will influence the velocity estimation accuracy. KS method is more robust against this influence (Figure 2.11) because the indirectly observed velocity states have more measurements and dynamic information to amend this influence.

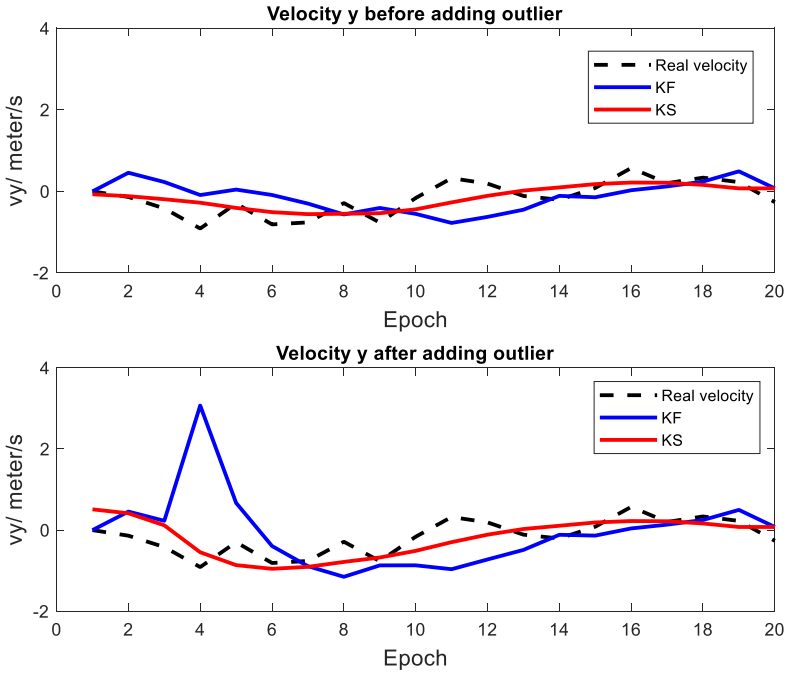


Figure 2.11 Estimated velocity y before and after adding additional outlier in measured y in epoch 4. [Top] for measurement model. [Bottom] for dynamic state model

2.4.1.4 Reliability analysis

Figure 2.12 shows the internal reliability (MDB) results of the original simulated dataset. According to this figure, it can be found the MDB values estimated by KS are always lower than that by KF until the last epoch. The MDB value by KF method reaches to about 6.4 meters for both measurement and state parameters after epoch 6, while for KS method, the MDB value converges quickly to around 5.6-6 meters at epoch 2. Therefore, the quality control method with KS step can successfully detect and identify error lower than with KF.

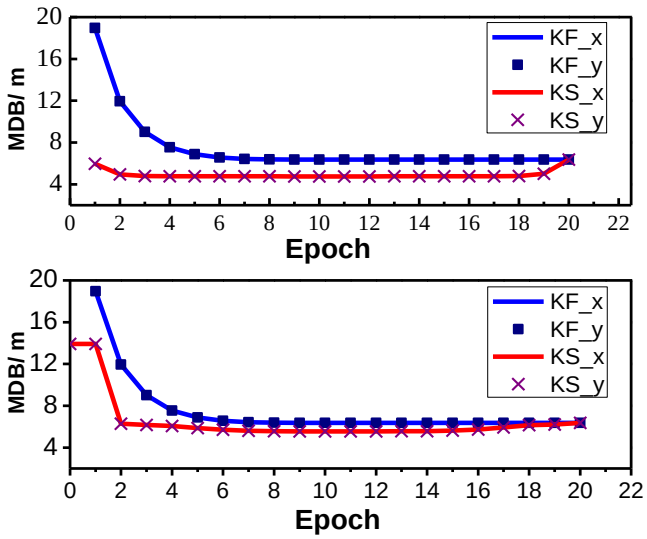


Figure 2.12 MDB by KF and KS with simulated dataset. [Top] for measurement model. [Bottom] for dynamic state model

The additional outlier in the previous section was 10 meters, larger than the MDBs in epoch 4 by both KF and KS. Therefore, the statistic tests in both steps have indicated the outliers (Figure 2.7 and 2.8). As KS has lower MDBs, if adding an outlier of 6 meters in the position measurement in epoch 4, the results will be shown in Figure 2.13 and Figure 2.14.

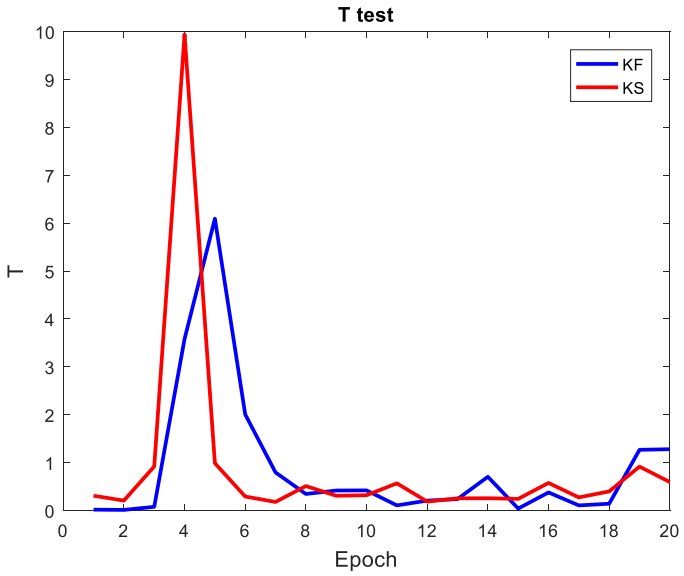


Figure 2.13 T test results after adding an outlier of 6 meters in the y measurement in epoch 4

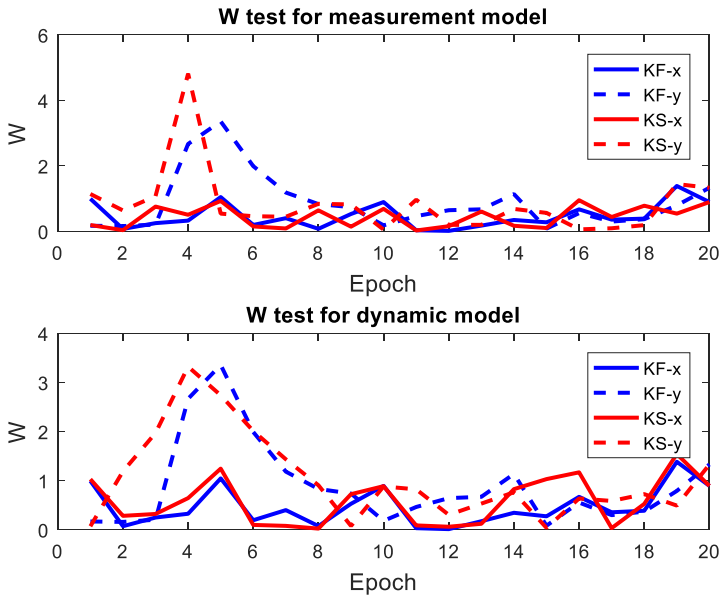


Figure 2.14 W test results after adding an outlier of 6 meters in the y measurement in epoch 4. [Top] for measurement model. [Bottom] for dynamic state model

The KF DIA method does not detect an outlier in epoch 4 but detects and identifies an -5.63 meters outlier with w test value 3.35 (>3.29) in epoch 5. Hence the additional outlier at this level (6 meters <MDB of KF) cannot be detected by the KF QC method. The successful DIA may be delayed as the outlier's influence will be stored in the filtered and predicted states, thus may cause mis-localization of the outlier. However, as discussed in the previous Section, the DIA results in epoch 5 by KF cannot precisely identify whether the detected outlier is in the measurement or is coming from the previous epoch. These problems can be solved with the KS method. As shown in Figure 2.13 and 2.14, the KS DIA method can successfully detect and identify the outlier in epoch 4.

Similar to the w test result, Figure 2.12 only shows the MDB results for the state of position as for the KF method, the predicted state of velocity cannot be tested and the MDB value for them are Infinite. But for the KS method, the dynamic models for velocity can be tested, the MDB of them are around 2.9-13 m/s except that in last epoch. For the last epoch, there are no further dynamic and measurement model to be used to examine the quality in the velocity state parameter at the last epoch. Hence the w test result of vy_k is close to zero and the MDB value are large to 10^8 meters at the last epoch.

Suppose there are errors with a value just lower than MDB value and cannot be detected. In that case, its influence on the final state estimation is called the External Reliability (ER). With the original simulated dataset, the ER values caused by MDB with KS step are also much lower than that with KF step, indicating if there are undetected errors, the KS process largely reduces their influence.

The MDB and ER values and trends do not change after adding or removing errors, since the system's geometry is not changed, indicating the error will not influence the Reliability of the system.

2.4.2 The Influence of Initial Value

In some other papers (Aravkin et al 2013), the initial values are assumed to be known and it is directly used in the first dynamic model as $x_1 = x_0 + \tau_1$, it does not consider the uncertainty and error in the initial values. Our method can be used to analyse and test the initial condition with a small modification, that considering an additional X_0 model $x_0 = x_0 + \tau_0$. Therefore, the x_0 and its uncertainty are further estimated in the KS method.

2.4.2.1 Difference between Known and Unknown Initial Values

As considering the x_0 with initial uncertainty Q_{x_0} =diag(10), the estimation of the state parameters is not change much. The estimation's precision changed slightly expect the initial one or two epochs by KF method for velocity states when compare Figure 2.15 to Figure 2.3 and 2.4. Table 2.3 shows the changing rate of standard deviation of estimated states before and after considering the x_0 uncertainty. In Table 2.3, only the first 10 epoch are shown because after 10 epochs, the influence of Q_{x_0} are nearly zero. It can be found in Table 2.3 that for KF step, whether to consider the x_0 with uncertainty will significantly influence the precision of velocity estimation in the initial 1 to 3 epochs, especially up to -26% changing rate in epoch 1. This influence will reduce with time. KS method seems less influenced by the initial value uncertainty, with only -2 to -2.5% change at epoch 1 for all states estimation precision. The precision of estimated velocity seems more sensitive than that of position estimation and becomes much better after considering the uncertainty of X_0 model, indicating this is a good way to increase the

estimation precision of indirectly observable state parameters for the initial several epochs. Although for the KS method, this kind of improvement is not very much.

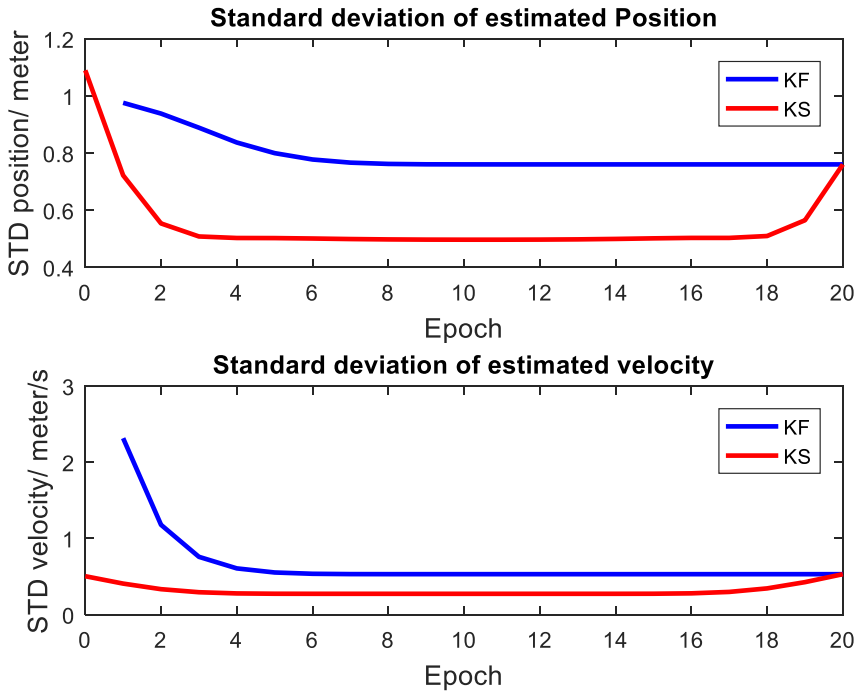


Figure 2.15 STD of estimated states after considering x_0 with uncertainty

Table 2.3 The changing rate (%) of STD between with and without considering the uncertainty of initial value x_0

Epoch		1	2	3	4	5	6	7	8	9	10
KS	x	-2.007	-0.321	0.187	0.0555	-0.0556	-0.0697	-0.0449	-0.0205	-0.00688	-0.00149
	y	-2.007	-0.321	0.187	0.0555	-0.0556	-0.0697	-0.0449	-0.0205	-0.00688	-0.00149
	vx	-2.535	-1.54	-0.581	-0.103	0.0131	0.0106	-0.0027	-0.0066	-0.00496	-0.00252
	vy	-2.535	-1.54	-0.581	-0.103	0.0131	0.0106	-0.0027	-0.0066	-0.00496	-0.00252
KF	x	2.365	-2.002	-1.638	-0.957	-0.495	-0.217	-0.0746	-0.0175	-0.00122	0.000911
	y	2.365	-2.002	-1.638	-0.957	-0.495	-0.217	-0.0746	-0.0175	-0.00122	0.000911
	vx	-26.78	-11.48	-4.328	-1.454	-0.38302	-0.061	0.00476	0.00529	-0.00089	-0.00294
	vy	-26.78	-11.48	-4.328	-1.454	-0.38302	-0.061	0.00476	0.00529	-0.00089	-0.00294

The quality control results for measurement model and dynamic model of position parameters are not much affected whether considering X_0 model or not, except the QC tests for the first epoch (Figure 2.16, 2.17, 2.18). For KF method, its MDBs for measurement and predicted states will increase about 40% from 13.70 meters to 18.97 meters at epoch 1, and get reduced at the following epochs. Therefore, the uncertainty within the initial value will make the estimation of directly observed states of epoch 1 less reliable, but for the following 2 to 4 epochs, it will become more reliable. For the indirectly observed variable, such as velocity, its estimation will be more reliable within all the epoch with the X_0 model. The MDBs for the velocity dynamic model between epoch 1 and 0 will be reduced by 25.5% from 13.18 m/s to 9.81 m/s. Therefore, the consideration of X_0 model may improve the quality of the estimation of indirectly observed variables.

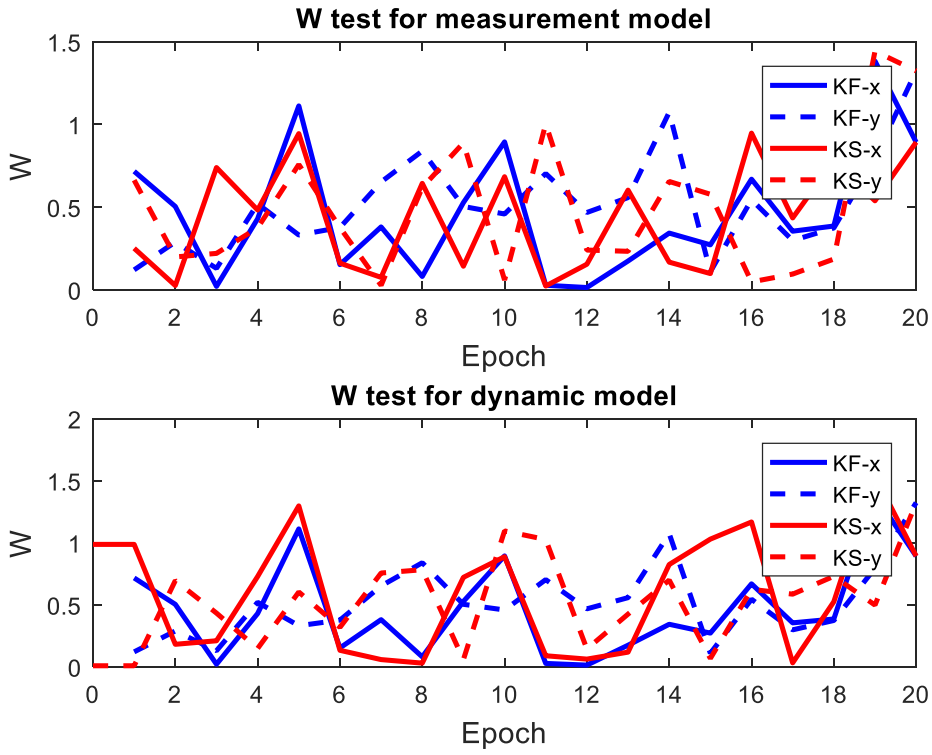


Figure 2.16 W test with considering X_0. [Top] for measurement model. [Bottom] for dynamic state model

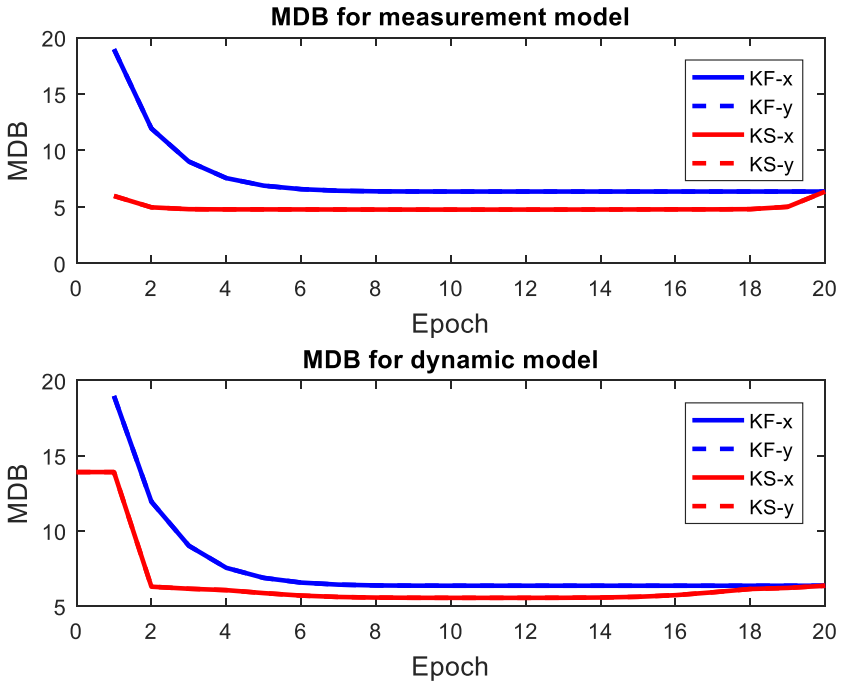


Figure 2.17 MDB with considering X_0. [Top] for measurement model. [Bottom] for dynamic state model

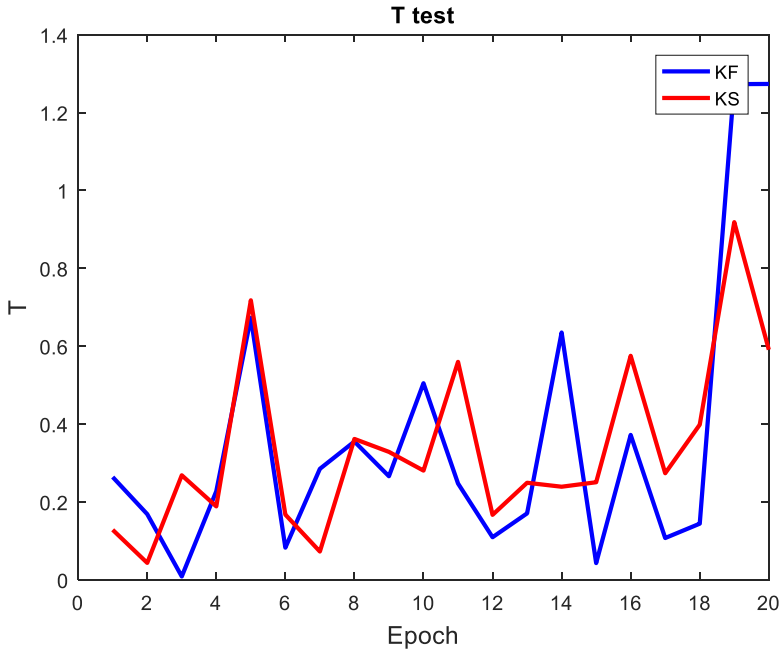


Figure 2.18 T test with considering X_0

2.4.2.2 Influence of fault in initial value

If there is an outlier in the initial value assumption, this outlier will also affect the estimation and DIA test of the following epochs. Table 2.4 and 2.5 show that if the additional outlier is in the state of position x in the assumed X_0 vector, it will not cause any change of the w test result of the measured position y and the states that related to the position y. But for the measured and predicted position x and velocity vx, an outlier is detected and identified at epoch 1. For KS method, the possible outlier is identified in the dynamic model between states parameter position x_0 and x_1 with the maximal w value of 4.549, and in the following epoch, the maximal w come from the state vx. The reason is that the error stored in the position state x_1 can be corrected with the measurement of position in epoch 1, whereas vx cannot be directly corrected and the influence of the additional error in the initial value of X_0 will keep longer.

For KF method, the w statistic test results for measurement and predicted states are the same. Therefore, it is hard to identify exactly where the outlier is. Also, KF cannot be used to test the indirectly observed state variable.

When the period is longer, the influence of the outlier in X_0 model will be negligible. Therefore, Table 2.4 and Table 2.5 only show the results of the initial 10 epochs. This length of the influence period depends on the initial outlier's size and the accuracy and redundancy of the following epochs.

Table 2.4 W test results for the measured position of KS and KF with/without additional 15 meters outlier in X_0 position x

Epoch		1	2	3	4	5	6	7	8	9	10
w results: X_0 without additional outlier											
KS	x	0.251	0.0261	0.739	0.483	0.944	0.161	0.075	0.644	0.143	0.684
	y	0.663	0.199	0.220	0.375	0.758	0.377	0.0273	0.619	0.888	0.0593
KF	x	0.716	0.504	0.021	0.430	1.112	0.153	0.380	0.0808	-0.527	0.894
	y	0.121	0.289	0.130	0.520	0.331	0.377	0.652	0.839	0.504	0.458
w results: X_0 with additional outlier of 15 meters in position x											
KS	x	1.434	0.447	0.895	0.479	1.014	0.244	0.004	0.696	0.176	0.665
	y	0.663	0.199	0.220	0.375	0.758	0.377	0.0273	0.619	0.888	0.0593
KF	x	3.329	2.274	1.228	1.247	1.661	0.210	0.156	0.206	0.47	0.875
	y	0.121	0.289	0.130	0.520	0.331	0.377	0.652	0.839	0.504	0.458

Table 2.5 W test results for the dynamic model of KS or predicted state of KF with/without additional 15 meters outlier in X_0 position x

Epoch		1	2	3	4	5	6	7	8	9	10
w results: X_0 without additional outlier											
KS	x	0.987	0.182	0.210	0.728	1.297	0.133	0.0586	0.0299	0.721	0.888
	y	0.00878	0.695	0.433	0.148	0.603	0.319	0.756	0.7826	0.0588	1.095
	vx	-0.658	0.509	0.497	0.0332	0.650	0.705	0.669	0.684	0.297	-0.180
	vy	0.0104	0.592	0.695	0.6778	0.953	0.762	0.354	0.0657	0.097	0.686

KF	x	0.716	0.504	0.021	0.430	1.112	0.153	0.380	0.0808	0.527	0.894
	y	0.121	0.289	0.130	0.520	0.331	0.377	0.652	0.839	0.503	0.459
	vx	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
	vy	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

w results: X_0 with additional outlier of 15 meters in position x

KS	x	4.549	0.546	0.021	1.1535	1.705	0.447	0.152	0.155	0.6575	0.86253
	y	0.00878	0.695	0.433	0.148	0.603	0.319	0.756	0.783	0.0588	1.09534
	vx	3.051	2.111	1.507	0.671	0.266	0.495	0.573	0.656	0.302	0.162
	vy	0.0104	0.592	0.695	0.678	0.953	0.762	0.354	0.0657	0.0972	0.686
KF	x	3.329	2.274	1.229	1.247	1.661	0.21	0.156	0.206	0.468	0.875
	y	0.121	0.289	0.130	0.520	0.331	0.377	0.652	0.839	0.504	0.459
	vx	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
	vy	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Estimation standard deviation and reliabilities, however, are not influenced by the outlier in the initial values.

2.5 Unified Least Squares based Quality Control for Ill Conditioned Covariance Problem

A unified least square based quality control method is investigated to deal with model that has covariance matrix which is ill-conditioned, such as being singular. This kind of matrix is invertible since its condition number is very large or equal to infinity. A singular

noise covariance will increase the possibility of numerical problems during optimization. The singular model will happen in the localization and navigation systems, such as for the IMU system. According to the error propagation model of the IMU system, the diagonal elements of Qk matrix for position states can be set as zero (Skog and Handel, 2005). For some other states such as gyro bias and acceleration bias can also be set as zero. Therefore, for any navigation systems that rely on IMU with least squares optimization, it will be a singular system. When there is a constraint on an observation, it will also cause the system to be singular, such as walking to a known point, there will be a constraint model with zero covariance value. Since the singularity will influence the least squares estimation and quality control results, another approach of LS estimation for the measurements with singular covariances is needed. Unified least squares method, which was suggested by Rao and Mitra (1971), is chosen as it can handle both conditions that the covariance is singular and non-singular. The corresponding quality control method is investigated under this unified least squares framework.

2.5.1 Unified Least Squares Estimation Model

The weight matrix P of the full least squares is replaced by another matrix T which can be used in least squares estimation to get best linear unbiased estimation (BLUE) of variable x in any case, whether singular or non-singular, regardless of the rank of matrix *D* (Rao and Mitra, 1971). The new weight *Pu* will be the general inverse of the matrix T where:

$$T = D + AUA^T \tag{2.75}$$

$$P_u = (T)^g = (D + AUA^T)^g, \;\; TP_uT = T \tag{2.76}$$

here superscript g represents g-inverse, U is a symmetric matrix, which can be simply chosen as an identity matrix. According to James (1978), to solve a solution of equation Ax=b with a generalised inverse of A matrix, the solution exists if and only if $AA^gb=b$. If A matrix has full column rank, the solution is unique.

Here, for our derivations below for a special case of ULS, we assume that

a) the T matrix in our study is squared and has full rank, thus $P_u T = I$,
b) A matrix has full rank;
c) $(A^{\mathrm{T}} P_u A)^g A^{\mathrm{T}} P_u A = I$;

where I is an identity matrix. Therefore, the solution of the unified least squares method could be found:

$$(A^{\mathrm{T}} P_u A)\Delta\hat{X} = A^{\mathrm{T}} P_u L \tag{2.77}$$

the solution for the ULS under the special conditions will be:

$$\Delta\hat{X} = (A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u L \tag{2.78}$$

$A^{\mathrm{T}} P_u A$ matrix will be squared and have full rank, it will have regular inverse. The corresponding covariance matrix of the estimated variables will be:

$$\begin{aligned}
&Q_{\widehat{\Delta X}} = (A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u D[(A^{\mathrm{T}} PA)^{-1} A^T P_u]^T \\
&= (A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u (T - AUA^T)[(A^{\mathrm{T}} PA)^{-1} A^T P_u]^T \\
&= [(A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u T - (A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u AUA^T] * (P_u A(A^{\mathrm{T}} PA)^{-1}) \\
&= (A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u T P_u A(A^{\mathrm{T}} P_u A)^{-1} - \\
&\quad (A^{\mathrm{T}} P_u A)^{-1} A^{\mathrm{T}} P_u AUA^T P_u A(A^{\mathrm{T}} PA)^{-1} \\
&= (A^{\mathrm{T}} P_u A)^{-1} - U
\end{aligned} \tag{2.79}$$

2.5.2 Quality Control for Unified Least Squares under Special Conditions

The quality control model can be derived under the Unified Least Square framework with a procedure similar to the normal Least Squares framework in Section 2.3.

2.5.2.1 ULS Residual and its Cofactor Matrix

The ULS residuals and the associated cofactor matrix can be written as:

$$\hat{v} = A\Delta\hat{X} - L = \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_uL - L = (\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_u - I)L \quad (2\text{-}80)$$

$$\begin{aligned}
Q_{\hat{v}} &= (\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_u - I)D(\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_u - I)^T \\
&= (\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_u - I)(T - AUA^T)(P_u\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - I) \\
&= (\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_uT - \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_uAUA^T - T + AUA^T) * \\
&\quad (P_u\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - I) \\
&= (\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - \mathrm{A}UA^T - T + AUA^T) * (P_u\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - I) \\
&= (\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - T) * (P_u\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - I) \\
&= \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}P_u\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} \\
&\quad -TP_u\mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} + T \\
&= \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} - \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}} + T \\
&= T - \mathrm{A}(A^{\mathrm{T}}P_uA)^{-1}A^{\mathrm{T}}
\end{aligned} \quad (2\text{-}81)$$

In order to test if there is a model error (∇), the corresponding null and alternative hypotheses are formed as:

$$H_0\colon\ E(l) = A\Delta X \quad (2.82)$$

$$H_a\colon\ E(l) = A\Delta X + \nabla \quad (2.83)$$

for the unified least squares, the T_{test} test statistic can be expressed with the ULS residual and be formulated as:

$$T_{test} = \frac{\hat{v}^{\mathrm{T}}P_u\hat{v}}{r\sigma_0{}^2} \quad (2\text{-}84)$$

here r is the redundancy number.

2.5.2.2 Outlier Identification

For the purpose of achieving a local test of outlier detection to locate the outlier within the measurements and to estimate the outlier size (also called Identification of outlier), the alternative hypothesis in Equation 2.83 will be decomposed into:

$$H_a\colon\ E(l) = A\Delta X + h_i\nabla_i \tag{2.85}$$

where h_i in (25) is the $m \times 1$ vector within which the i^{th} element equals to 1 and all other elements equal to zero. ∇_i is the model error in the i^{th} model. There the residual model will be:

$$v = A\Delta\mathrm{X} + h_i\nabla_i - L = [A\ h_i]\begin{bmatrix}\Delta\mathrm{X}\\ \nabla_i\end{bmatrix} - \mathrm{L} \tag{2.86}$$

Here let matrix $[A\ h_i]$ as *B*, under the unified LS structure, the unknown variable vector is: $X_o = \begin{bmatrix}\Delta\mathrm{X}\\ \nabla_i\end{bmatrix}$, a $U_{\_all} = diag(U,\ U_2)$,

the new matrix for g-inverse is:

$$T_2 = D + BU_{\_all}B^T = D + AUA^T + h_iU_2{h_i}^T \tag{2.87}$$

$$P_{u2} = (T_2)^g \tag{2.88}$$

U_2 is a $m\times m$ symmetric matrix, which can be simply chosen as an identity matrix. To achieve the minimal $\hat{v}^{\mathrm{T}}P_{u2}\hat{v}$, we will get solution as:

$$\begin{bmatrix}A^TP_{u2}A & A^TP_{u2}h_i\\ {h_i}^TP_{u2}A & {h_i}^TP_{u2}h_i\end{bmatrix}\begin{bmatrix}\Delta\mathrm{X}_o\\ \nabla_i\end{bmatrix} = \begin{bmatrix}A^TP_{u2}L\\ {h_i}^TP_{u2}L\end{bmatrix} = \begin{bmatrix}A^TP_{u2}\\ {h_i}^TP_{u2}\end{bmatrix}L \tag{2.89}$$

To solve this model, let a new *phi* matrix equals to:

$$phi = \begin{bmatrix}A^TP_{u2}A & A^TP_{u2}h_i\\ {h_i}^TP_{u2}A & {h_i}^TP_{u2}h_i\end{bmatrix} \tag{2.90}$$

Therefore, similar, the *phi* matrix is also a squared full rank and has regular inverse.

$$phi^{-1} \tag{2.90a}$$
$$= \begin{bmatrix} (A^{\mathrm{T}}P_{u2}A)^{-1} + (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}h_iQ_{n2}{h_i}^{T}P_{u2}A(A^{\mathrm{T}}P_{u2}A)^{-1} & -(A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}h_iQ_{n2} \\ -Q_{n2}{h_i}^{T}P_{u2}A(A^{\mathrm{T}}P_{u2}A)^{-1} & Q_{n2} \end{bmatrix}$$

Multiply the regular inverse of phi to the right part of Eq. 2.89:

$$\widehat{X_o} = phi^{-1} * \begin{bmatrix} A^{T}P_{u2} \\ {h_i}^{T}P_{u2} \end{bmatrix} * L$$

$$= \begin{bmatrix} (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2} + (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}h_iQ_{n2}{h_i}^{T}P_{u2}A(A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2} - (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}h_iQ_{n2}{h_i}^{T}P_{u2} \\ Q_{n2}(-{h_i}^{T}P_{u2}A(A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2} + {h_i}^{T}P_{u2}) \end{bmatrix} L$$

$$= \begin{bmatrix} (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}(I + h_iQ_{n2}h_i^{T}P_{u2}A(A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2} - h_iQ_{n2}{h_i}^{T}P_{u2}) \\ Q_{n2}h_i^{T}(P_{u2} - P_{u2}A(A^{\mathrm{T}}P_uA)^{-1}A^{T}P_{u2}) \end{bmatrix} L \tag{2.91}$$

$$= \begin{bmatrix} (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}(I + h_iQ_{n2}h_i^{T}(P_{u2}A(A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2} - P_{u2})) \\ Q_{n2}h_i^{T}Q_{T2} \end{bmatrix} L$$

$$= \begin{bmatrix} (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}(I - h_iQ_{n2}h_i^{T}Q_{T2}) \\ \widehat{\nabla}_i \end{bmatrix} L$$

$$= \begin{bmatrix} (A^{\mathrm{T}}P_{u2}A)^{-1}A^{T}P_{u2}(I - h_i^{T}\widehat{\nabla}_i) \\ \widehat{\nabla}_i \end{bmatrix} L$$

Here Q_{T2} and Q_{n2} are two new introduced matrixes for derivation:

$$Q_{T2} = P_{u2} - P_{u2}\mathrm{A}(A^{\mathrm{T}}P_{u2}A)^{-1}A^{\mathrm{T}}P_{u2}$$
$$Q_{n2} = (h_i^{T}Q_{T2}h_i)^{-1} \tag{2.92}$$

In the lower part of$\widehat{X_o}$, $\widehat{\nabla}_i$ is the estimated outlier, thus:

$$\widehat{\nabla}_i = (h_i^{T}Q_{T2}h_i)^{-1}h_i^{T}Q_{T2}L = Q_{n2}h_i^{T}Q_{T2}L \tag{2.93}$$

The corresponding variance value can be estimated as:

$$\mathrm{Q}_{\widehat{\nabla}_\mathrm{i}} = (h_i^{T}Q_{T2}h_i)^{-1} - U_{2(i,i)} \tag{2.94}$$

The w test for ULS will be:

$$w_i = \frac{\widehat{\nabla}_i}{\sigma_{\widehat{\nabla}_i}} = \frac{(h_i^T Q_{T2} h_i)^{-1} h_i^T Q_{T2} L}{\sigma \sqrt{(h_i^T Q_{T2} h_i)^{-1} - U_{2(i,i)}}} \tag{2.95}$$

2.5.2.3 Reliability for ULS under Special Conditions

the MDB and ER in the ULS can be derived as:

for the Unified LS:

$$MDB_i = \frac{\delta_d \sigma_0}{\sqrt{(h_i^T Q_{T2} h_i)^{-1} - U_{2(i,i)}}} \tag{2.96}$$

$$ER_i = (A^{\mathrm{T}} P_{u2} A)^{-1} A^T P_{u2} h_i^T MDB_i \tag{2.97}$$

2.5.3 Numerical Evaluation for the Unified Least Squares and its Corresponding Quality Control ULS under Special Conditions

A numerical case study (Figure 2.19) is conducted to test the proposed unified least squares quality control method. In this simulated localization case study, a platform is moving from initial point X0 ($x_0 = 0, y_0 = 0$) to other 3 points with measure the position displacement and turn angle (d_{ij}, w_{ij}). A known feature (f) is in the middle and is detected by the moving platform. Finally, the platform is moving back to the initial point. After moving back to the initial point, a constraint model can be used by insisting that the position of X4 is equal to X0, thus the unknown variable (x_4, y_4) satisfy a linear equation $d40$: $(x_0, y_0) = (x_4, y_4)$. Therefore, the measurement models between the moving points and between the points and the feature are combined together with the constraint model to build a full least square problem, which, can be solved by the unified least squares. Kalman smoothing is carried out to compare the solutions. Normal full least squares with Singular Value Decomposition (SVD) to solve the singular D matrix is also conducted to do comparison.

Firstly, it is numerically checked that this case study has a squared full rank T matrix, a full rank A matrix and $P_u T = I$.

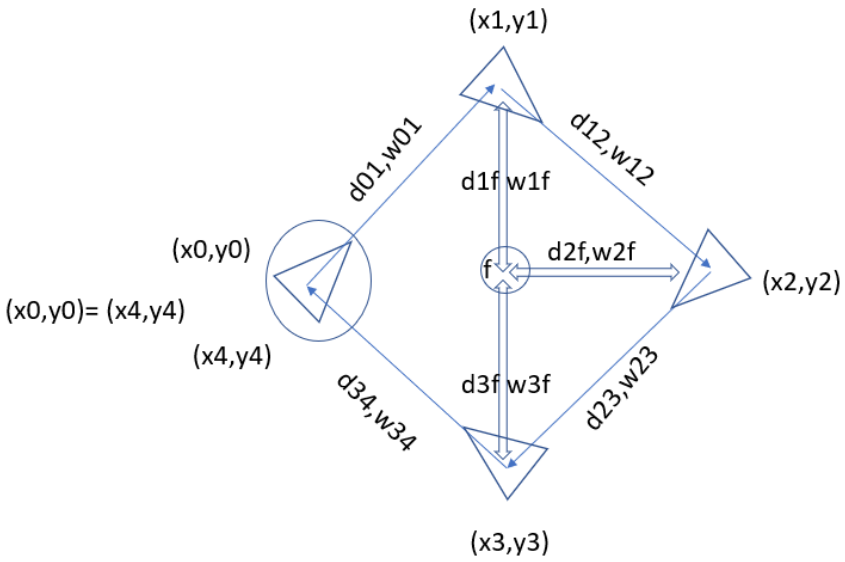


Figure 2.19 A simulated constrained localization case with constraint model: x4=x0, y4=y0

Table 2.6 shows the estimation solutions of the simulated singular constrained case with three mentioned methods. Although our KS equations are derived from the normal Full least squares framework, the KS and ULS have the same estimation results to this singular case, but different to the FLS/SDV results. Both KS and ULS can meet the constraint that X4 equals to the initial (0,0) point, whereas FLS' X4 still has a bias to the initial point. What's more, if the FLS results for this constrained case are compared with a non-constrained case (which means the constraint model is not added to the system), they will have almost same estimation and quality control results.

Table 2.6 Estimated position by ULS, KS, FLS for a singular constrained case

point		**X1**	**X2**	**X3**	**X4**
ULS	x/ meter	9.9074	19.9491	9.8749	0
	y/ meter	9.9378	0.03631	-9.9578	0
KS	x/ meter	9.9074	19.9491	9.8749	0
	y/ meter	9.9378	0.03631	-9.9578	0
FLS	x/ meter	9.9202	19.9597	9.9256	0.0261
	y/ meter	9.9249	0.01243	-10.0233	-0.1239

Similar consequence can be acquired from the estimation standard deviation (STD) results (Figures 2.20 and 2.21), for the normal FLS estimation with SVD, the STD value will increase with the moving steps. However, since we make a constraint to the last step that point X4 has the exact same position to X0, its estimated STD will be zero by the KS and ULS (not viewed in Figure 2.20, Figure 2.21 since their values of STD X4 are zero). Because the least squares optimization and the KS both take all the available observations and models into consideration, the zero covariance of the constraint model will also influence the estimated STD of other variables, such as STD for position of X1 to X3. Therefore, it can be found that the STD value are much lower than the normal FLS. STD of X2 is slightly higher than other points because it is less correlated to the constrained point X4, and is less influenced by the constraint model.

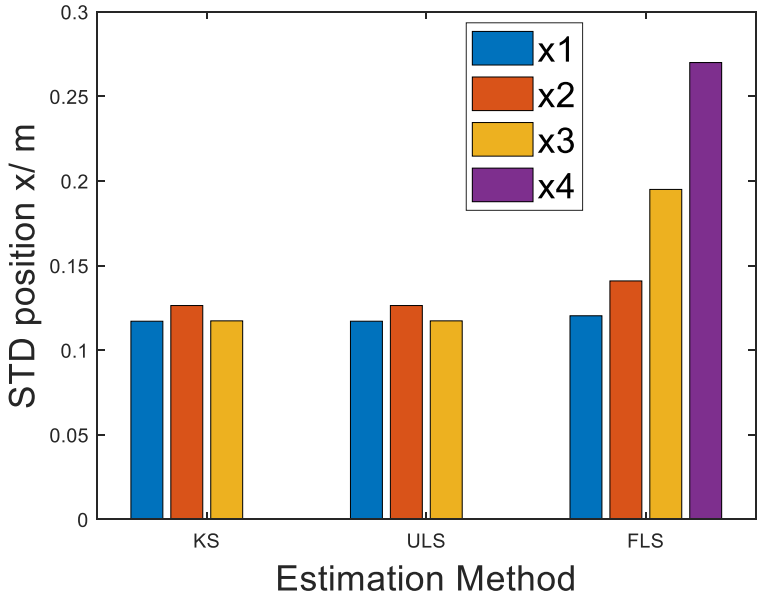


Figure 2.20 Standard deviation of estimated x by three methods

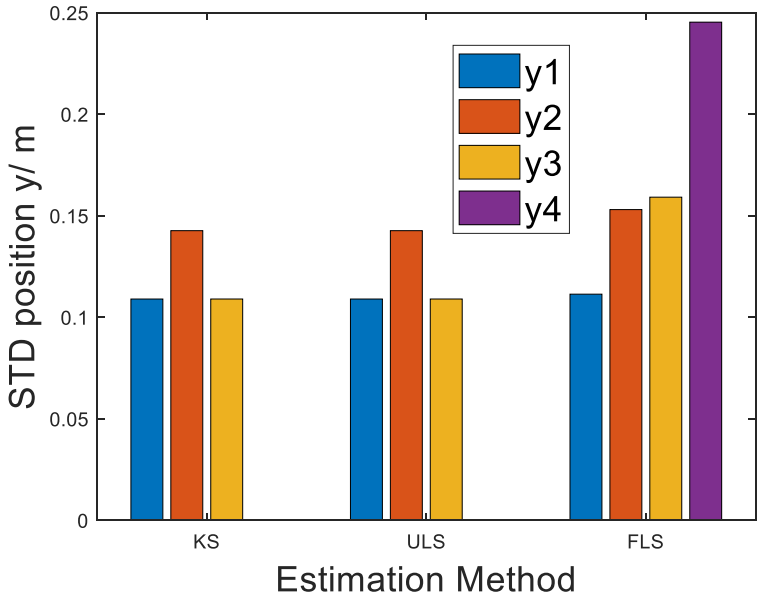


Figure 2.21 Standard deviation of estimated y by three methods

Quality control can be done under this unified least squares framework. Compared to the KS and FLS results, the ULS QC results are the same as the KS results. Since this simulated dataset only contains random noise, no outlier is detected by the three methods.

Figure 2.22 and Figure 2.23 show the MDB value of the three methods on the observation of the feature. It is found that by the KS and ULS for this singular case, the MDB values for the observation at the points near to the constrained point will be lower to that of the points that are far away, which indicating that the singular constrained model will affect nearby points more since they are more correlated. For the FLS estimation, since the constraint model is not usable, the ability of detecting outlier will be reduced along the moving, a loop closure or constraint may be needed to enhance this detection ability.

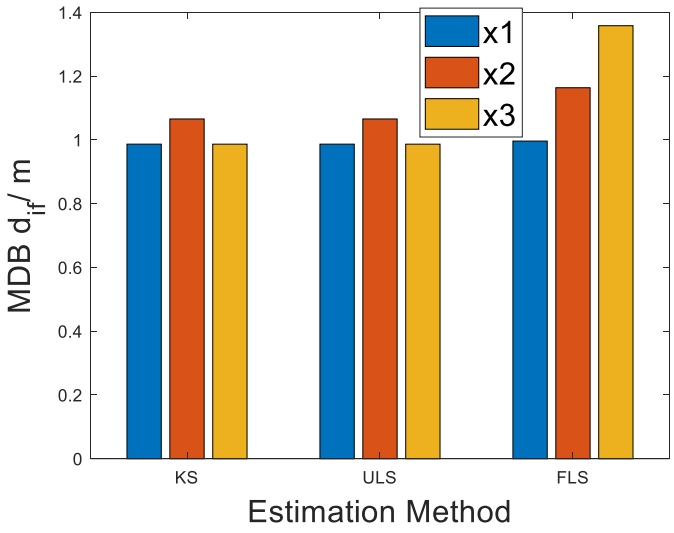


Figure 2.22 MDB of observation displacement model between moving point (i=1,2,3) and known feature (f) by three methods

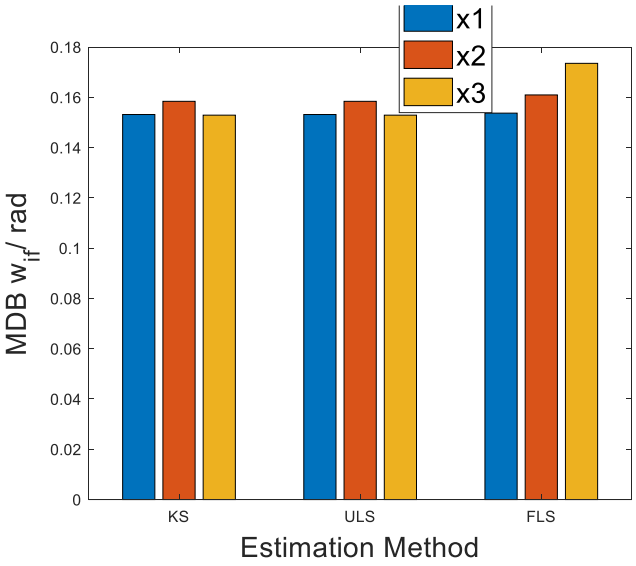


Figure 2.23 MDB of observation angle model between moving point (i=1,2,3) and known feature (f) by three methods

For the FLS results, since the constraint model is not numerically used for point X4, the MDB value for that constraint model by FLS will be huge or infinite (thus cannot be plotted in the Figure 2.24), which means if only deal the singular matrix with pseudo-inverse, this system cannot monitor the quality of the singular models, possible outliers within these models cannot be detected. Therefore, KS or ULS method is needed to test the singular models.

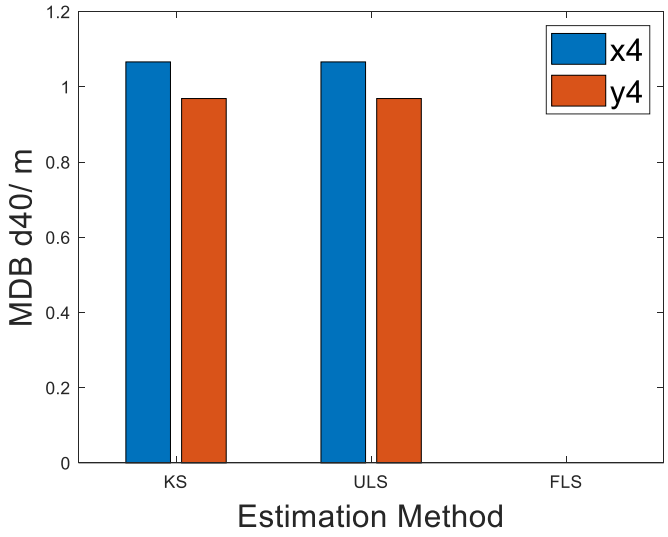


Figure 2.24 MDB of constraint model between moving point X4 and known initial point X0 by three methods

Figure 2.25 and Figure 2.26 shows the MDB results of the dynamic model between each moving steps. Similarly, the models near the constrained point X4 are more reliable. For the FLS results, since the constraint model is not taken effect, and there are no other observations at this point, the MDB value for the last dynamic step (X3 to X4) will be huge or infinite (thus not shows in the Figure 2.25 and Figure 2.26). This is a normal situation for cases that a tested epoch is only related to a dynamic model, but without any observation models, such as pose-graph SLAM. Without the examination of the observation model between the predicted position and other measurements (such as the GPS positioning or feature detection), the dynamic model's quality cannot be estimated by our quality control method.

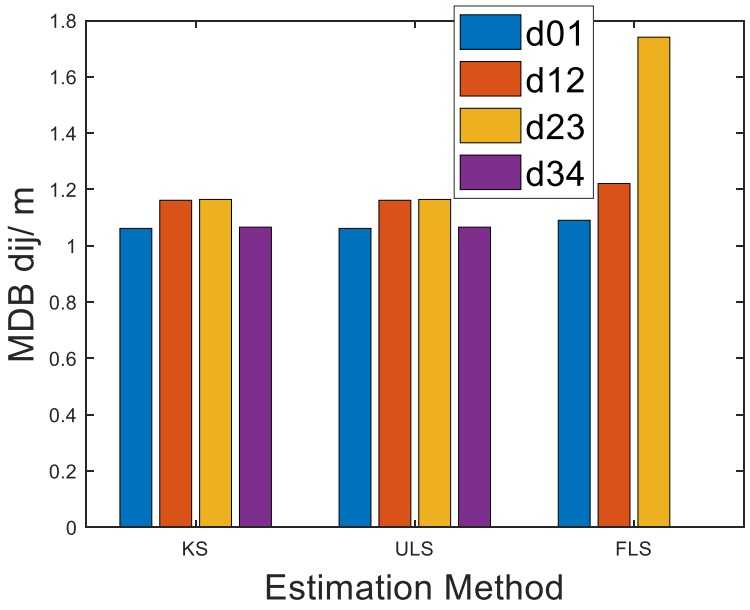


Figure 2.25 MDB of dynamic model of position displacement between moving points by three methods

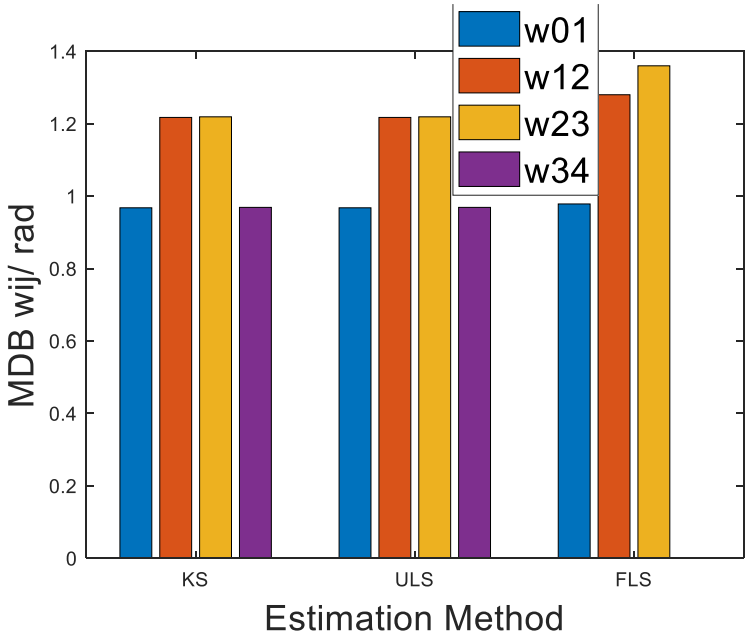


Figure 2.26 MDB of dynamic model of turn angle between moving points by three methods

2.6 Summary

In this Chapter, the equivalence of Kalman smoother and least squares for dynamic systems is discussed with a full least square structure. An extended Kalman filter-smoother quality control method is then developed according to this equivalence. It can be used to deal with outliers with detection, identification, adaptation (DIA) method, also can be used to analyse, control and improve the system quality. This quality control method is for the first time that applying the DIA method and Reliability analysis for the smoothing process.

With the tests of the proposed Extended Kalman filter-smoother quality control approach with different numerical examples containing Kalman filtering and smoothing steps (Section 2.4), our method shows the possibility of successfully analysing, controlling, and monitoring the noisy dynamical system. The quality control results demonstrate that the Kalman smoothing approach has some advantages over the Kalman Filtering. Firstly, the DIA process in KS has better performance as it can directly figure out where the fault happens. However, in the KF process, there are sometimes misidentification due to the high correlation coefficient between the outlier detection statistic, or delayed detection since the fault in one epoch may cause adverse influence on the estimation of the following epochs by the KF estimation. Secondly, the KS method can achieve more accurate and more reliable estimation for the states parameter that are indirectly observed. For instance, when the velocity parameter is not observed during the localization process, its quality will not be able to be tested by the KF method, but can be solved by the KS method.

The influence of the unknown initial values is also tested and discussed in this Chapter (see Section 2.4.2). Overall, the uncertainty of the initial value has been analysed. The Kalman smoother is less influenced by the uncertainty and outliers in the initial values than the filter.

The proposed ULS estimation and quality control method has also been demonstrated that it is suitable for a model with constraints which can be statistically monitored for their reliability. The state-constrained systems are often met when dealing with engineering

fields such as robotics and navigation. For example, when a robot travels and comes back to a known visited point, a known functional relation will exist among the unknown state parameters. Therefore, it can be used as constraints within the states. The test results have shown our ULS method can achieve same results with the proposed KS method for the constrained model. In contrast, the FLS method that can only treat the singular matrix with pseudo-inverse will make the constraint model ineffective.

Chapter 3 GNSS/INS Integration for Positioning and Georeferencing with Quality Control

3.1 Introduction

To achieve precise and reliable localization and navigation ability, on-board sensors system plays an important role and has caught attention of many research groups, such as GPS/INS system (Zhao, 2013; Lee, 2014), LiDAR system (Xiao et al., 2016), vision system (Paz et al., 2008; Klein and Murray, 2007), etc. GPS technology is traditionally applied on aviation positioning and has been fully studied for decades (Seo et al, 2011; Han et al., 2015). The accuracy of GPS has been increased to decimetre or centimetre level with the advanced GPS method such as double differencing GPS and Precise Point Positioning (Garcia et al, 2005; Cheng, 2017). GPS can provide accurate and continuous position information in open-sky conditions as it is easy to track more than 4 satellites. However, in land applications, especially for HAV positioning, GPS signals can be easily blocked or altered due to the high density of building and trees that may obstruct the sky or cause multi-path signal. In order to improve the satellite availability for accurate positioning, adding other GNSS satellites to the system is essential, such as the Chinese BeiDou Navigation Satellite System (Han, et al., 2015), Russian GLONASS system, the European global navigation satellite system (Galileo). However, in the complex road environment, there are still areas where GNSS signals are unavailable, such as in the tunnel. Therefore, other aiding methods are needed in this GNSS signal denied environment.

IMU/GNSS integration is a popular integrated localization and navigation method applied for aviation and vehicle application (Zhao, 2013; Lee, 2014; Farrell et al, 2000). Inertial measurement is attractive as it is self-contained, high sampling rate (100-200 Hz) and no external information is needed for position, therefore it can provide the localization and navigation solution during GNSS signal blockage periods. However, inertial navigation solutions may suffer time-dependent growth of sensor errors. Such sensor errors can be corrected when GNSS measurements are available, thus the integration of IMU and

GNSS can complement each other, thus improving the localization and navigation performance.

Therefore, in this study, the mathematical models for GNSS and INS integration is investigated. The comparison of Real Time Kinematic (RTK) and Post Processing Kinematic (PPK) positioning method is discussed. The quality control method, including outlier detection and identification, system reliability analysis, is introduced to the positioning methods. The structure of this Chapter is as follows: Section 3.2 presents the mathematical models for GNSS and GNSS/INS integration methods. Numerical analysis of this methods is undertaken in Section 3.3 with quality control. Section 3.4 summarizes the performance of the RTK, PPK, GNSS/INS integration methods.

3.2 Mathematical Models for GNSS and INS Integration

3.2.1 Basic Measurement Model of GNSS

GNSS receivers usually outputs raw measurements such as code pseudo-ranges, carrier-phase measurements, Doppler measurements. Pseudo-ranges and carrier-phase are basically used for positioning, while Doppler measurement can be used to determine the velocity.

Basic observation equations of pseudo-range, carrier-phase and Doppler are listed as follows [Hofmann-Wellenhof et al., 2001; Hauschild, 2017; Groves, 2013]:

$$p_{r,j}^{s} = \rho_r^s + c\delta t_r - c\delta t^s + I_{r,j}^{s} + T_r^s + \delta m + \varepsilon_p \tag{3.1}$$

$$\lambda_j \phi_{r,j}^{s} = \rho_r^s + c\delta t_r - c\delta t^s - I_{r,j}^{s} + T_r^s + \delta m + \lambda_j N_{r,j}^{s} + \varepsilon_\phi \tag{3.2}$$

$$\lambda_j D_{r,j}^{s} = \rho_r^s + c\delta t_r - c\delta t^s - I_{r,j}^{s} + T_r^s + \varepsilon_D \tag{3.3}$$

where:

- Subscript r denotes the receiver where the measurement generated;

- Superscript s denotes the satellite;
- Subscript j denotes the frequency band of the observation, here, $j = 1$ for L1, $j = 2$ for L2;
- $p_{r,j}^{s}$ is the code measurement from satellite *s* to receiver *r* on *jth* frequency;
- $\phi_{r,j}^{s}$ is the carrier phase measurement from satellite *s* to receiver *r* on *jth* frequency;
- $D_{r,j}^{s}$ is the Doppler measurement;
- ρ is the geometric distance between satellite and receiver;
- c is the speed of light;
- δt denotes the clock error for satellite or receiver;
- T denotes troposphere delay;
- I denotes ionosphere delay;
- λ_j is the wavelength on *jth* frequency;
- N integer ambiguity;
- δm denotes multipath error in measurement;
- ε_p is code measurement noise;
- ε_ϕ is carrier phase measurement noise;
- ε_D is Doppler measurement noise .

3.2.2 GNSS Double Differencing Positioning

Since there are many error sources in the GNSS measurements that will influence the accuracy of GNSS absolute positioning, relative positioning can achieve centimeter-level to millimiter-level accuracy by conducting differencing between measurements. Double differencing is conducted between two satellites and between two receivers to mitigate the impacts errors, such as the clock errors. The code measurement and phase measurement are mostly used for positioning with double differencing. The measurement model of the double differencing GNSS method can be described as:

$$\Delta\nabla p_{rb,j}^{sp} = \Delta\nabla\rho_{rb}^{sp} + \Delta\nabla I_{rb,j}^{sp} + \Delta\nabla T_{rb}^{sp} + \Delta\nabla\varepsilon_p \tag{3.3}$$

$$\lambda_j \Delta\nabla\phi_{rb,j}^{sp} = \Delta\nabla\rho_{rb}^{sp} - \Delta\nabla I_{rb,j}^{sp} + \Delta\nabla T_{rb}^{sp} + \lambda_j \Delta\nabla N_{r,j}^{s} + \Delta\nabla\varepsilon_{\phi} \quad (3.4)$$

here $\Delta\nabla$ refers to a double-differenting, the Subscript b denotes a stationary base station, whose coordinates are known. Superscript p denotes a chosen reference satellite. In the short baseline case, the effects of the ionospheric (I_{ion}) bias and tropospheric (T_{trop}) bias are significantly reduced. Therefor the above equation can be reduced to:

$$\Delta\nabla p_{rb,j}^{sp} = \Delta\nabla\rho_{rb}^{sp} + \Delta\nabla\varepsilon_{p} \quad (3.5)$$

$$\lambda_j \Delta\nabla\phi_{rb,j}^{sp} = \Delta\nabla\rho_{rb}^{sp} + \lambda_j \Delta\nabla N_{r,j}^{s} + \Delta\nabla\varepsilon_{\phi} \quad (3.6)$$

In real-time applications of kinematic positioning with double differencing, Real Time Kinematic (RTK) positioning is well known which can achieve centimeter-level accuracy with proper integer ambiguity resolutions.

3.2.3 IMU Sensor Error Model

Inertial navigation comprises accelerometers and gyroscopes to carry out navigation tasks. The output of an inertial measurement unit (IMU) are incremental angles and velocities. An inertial navigation system (INS) can be used to estimate the platform's pose: position, velocity and attitude. The error model for IMU measurements in this Thesis is the 15 states model, including nine navigation error states, three gyro sensor bias and three accelerometer sensor bias. The details of INS error states are given as follows:

- $\varphi = [\, \varphi_E \; \varphi_N \; \varphi_E]^T$: mathematical platform misalignment angles.
- $\delta v = [\, \delta v_E^n \; \delta v_N^n \; \delta v_U^n]^T$: velocity errors;
- $\delta pos = [\, \delta L \; \delta\lambda \; \delta h]^T$: latitude, longitude and height errors respectively;
- $\varepsilon^b = [\, \varepsilon_x^b \; \varepsilon_y^b \; \varepsilon_z^b]^T$: gyro drift errors expressed in body frame;
- $\nabla^b = [\, \nabla_x^b \; \nabla_y^b \; \nabla_z^b]^T$: accelerometer biases expressed in body frame;
- Superscript *n* represent2 the 'east-north-up' navigation n-frame;

- Superscript *b* represents the 'right-forward-up' body b-frame;
- Subscript [*E, N, U*] represents the 'east-north-up', respectively;
- Subscript [*x, y, z*] represents the axis '*x-y-z*' in body b-frame.

The IMU error update models are shown as (Baritzhack and Berman, 1988; Yan et al, 2015):

$$\dot{\varphi} = -\omega_{in}^{n} \times \varphi + \delta\omega_{in}^{n} - C_b^n \delta\omega_{ib}^{b} \tag{3.6}$$

$$\dot{v} = f_{sf}^{n} \times \varphi - (2\omega_{ie}^{n} + \omega_{en}^{n}) \times \delta v^{n} + v^{n} \times (2\delta\omega_{ie}^{n} + \delta\omega_{en}^{n}) + C_b^n \delta f_{sf}^{b} \tag{3.7}$$

$$\dot{\delta L} = \frac{1}{R_{Mh}} \delta v_N^n - \frac{v_N^n}{R_{Mh}^2} \delta h \tag{3.8}$$

$$\dot{\delta\lambda} = \frac{\sec L}{R_{Nh}} \delta v_E^n + \frac{v_E^n \sec L \tan L}{R_{Nh}} \delta L - \frac{v_E^n \sec L}{R_{Nh}^2} \delta h \tag{3.9}$$

$$\dot{\delta h} = \delta v_U^n \tag{3.10}$$

Here:

- ω_{ie} is the Earth's angular rate;
- ω_{in} is the angular rate vector of the true coordinate system with respect to the inertial frame;
- ω_{en} is the angular rate vector of the true coordinate system with respect to the Earth;
- ω_{ib} and f_{sf} are gyro sensed angular rate and accelerometer sensed specific force, respectively;
- R_{Mh} and R_{Nh} can be calculated according to the Earth's semi-major axis, the Earth's flattening and latitude at this position.

3.2.4 Procedure and Flowchart of Integration Methodology

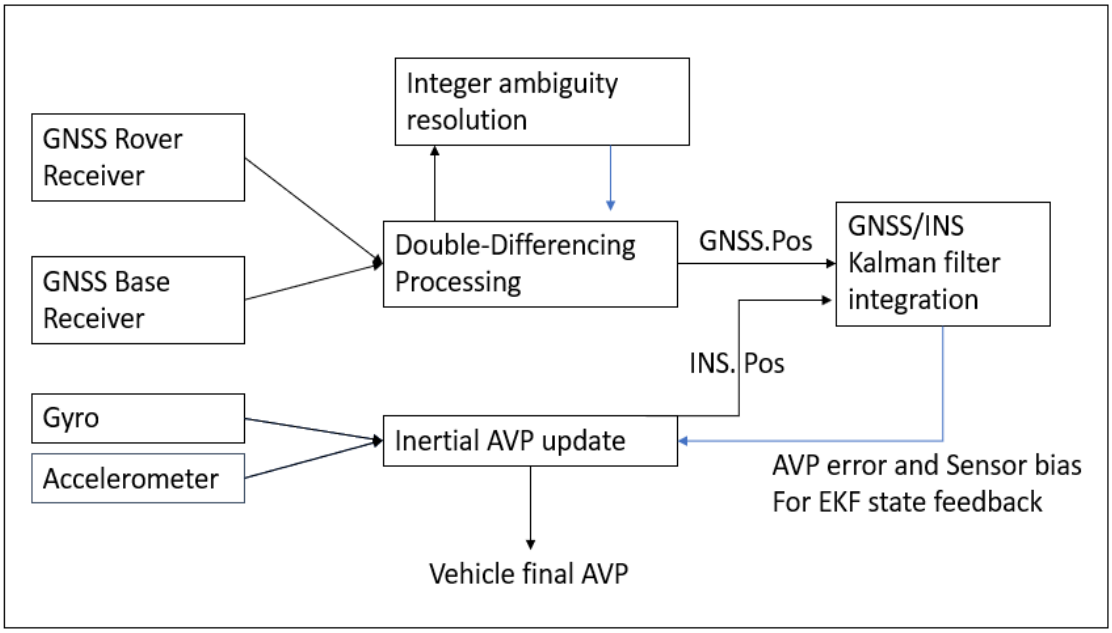


Figure 3.1 Flowchart of the loosely coupled GNSS/INS

Figure 3.1 shows the implementation of the loosely coupled GNSS/INS, while a similar model structure can be developed for tightly coupled GNSS/INS integration, which is not the focus of studies in this thesis. In this multi-sensor localization and navigation system, Extended Kalman Filter (EKF) is used to fuse measurement information from different sensor. Smoothing can also be used to improve the estimation precision and accuracy for integration. Strapdown inertial navigation system is used to update vehicle pose (Altitude, Velocity, Position - AVP**)** with high frequency. When GNSS signal is available, the GNSS observations of base station and rover station is used to do double-differencing. After the Kalman filter measurement update, the estimated states of AVP error, gyro bias and accelerometer bias will be used to amendment the INS estimated AVP pose and sensor parameters.

3.2.5 IMU Sensor Characteristics

The IMU sensor used in this study, 'Polynav 2000e', is a Tactical grade sensor. The characteristics of this sensor are shown in Table 3.1.

Table 3.1 IMU sensor characteristic

	Gyro	**Accelerometer**
Bias	20 deg/hr	100 ug
Random walk	0.15 deg/root-hr	0.5 m/s/root-hr

For this study, the initial attitude is achieved by doing initial alignment with the Novatel IE software. This attitude value is then be used as an initial attitude for the Extended Kalman filter based GNSS/INS integration system with quality control. The EKF uncertainty setting for the heading value is set as 0.3 degree.

3.3 Numerical Experiments and Analysis with Fault Detection and Reliability measures

An urban road testing was undertaking on 13/12/2020 in Sydney, Australia to test the RTK and fusion system with INS. Since the GNSS observation quality in urban area will be influenced due to the complex driving condition, smoothing method can be used to improve the GNSS positioning accuracy. Since smoothing is mostly done offline, this method can be called as Post processing kinematic (PPK), which can be more accurate than real-time kinematic methods. PPK results are commonly used for mapping.

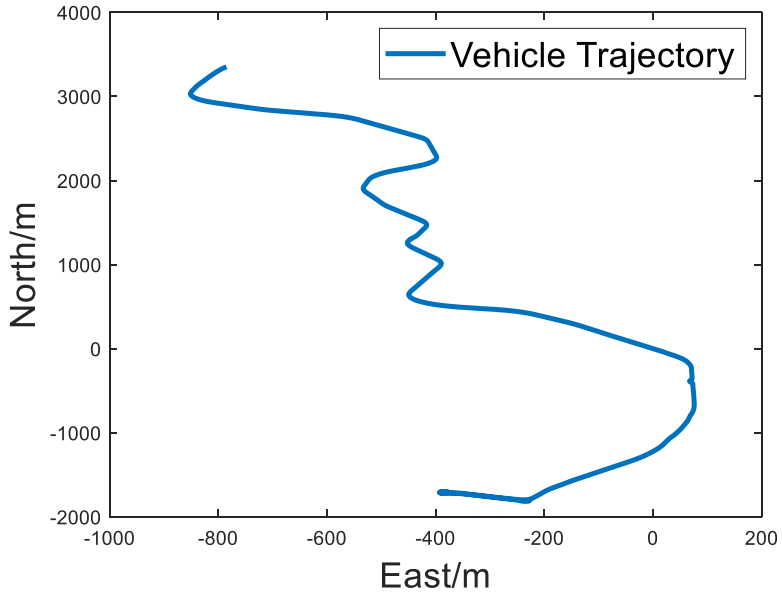


Figure 3.2 Vehicle Trajectory in local coordinate for testing GNSS, GNSS/INS

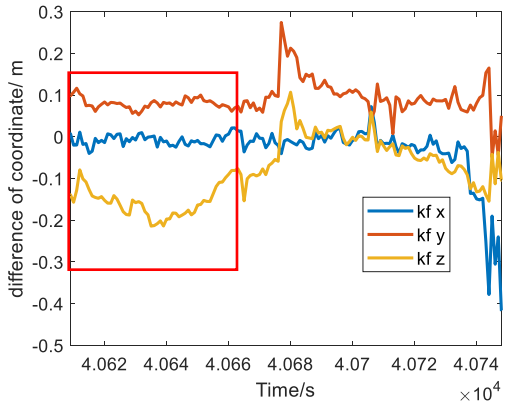
Because there is no ground truth for the driving test, an online navigation solution estimated by a commercial software (Polynav) is used as a reference to analyse our RTK, PPK and GNSS/INS solution. There are three sections of trajectory selected to do comparison to the reference solution, during which the online GNSS status is integer-ambiguity fixed. The whole trajectory is between GPS time 40550s to 41100s, about 9 minutes. The selected time periods are for coordinate comparison are GPS time:40609s-40748s, 40801s-40906s, 40908s-41100s. Figure 3.2 shows the whole driving trajectory. The multi-constellation dual-frequency RTK/PPK code is developed from an open-sourced single frequency GPS based RTK toolbox (Realini and Reguzzoni, 2013), then integration of GNSS/INS code is developed. EKFQC toolbox developed in this thesis is modified and embedded for quality control.

3.3.1 GNSS Double Differencing

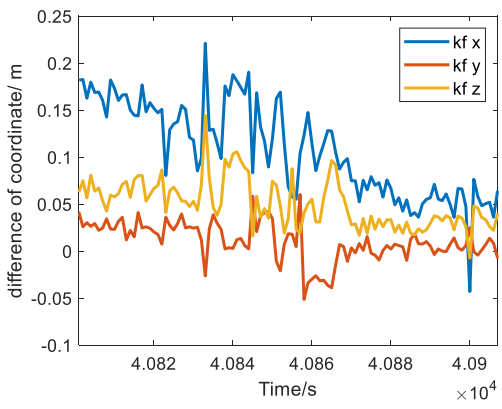
In this test both GPS and GLONASS signals were utilized for double difference GNSS positioning. Only one signal frequency for each navigation satellite system is taken into consideration in this study. The GNSS measurement frequency is 1 Hz. Both float and fixed solution for RTK and PPK is compared and analysed with quality control.

3.3.1.1 DD Ambiguity Float Results

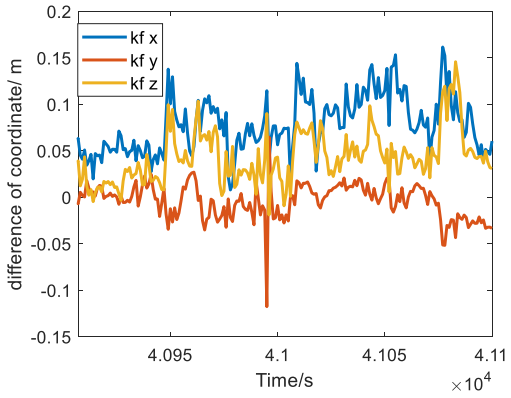
Figure 3.3 and Figure 3.4 show the coordinate difference to the reference trajectory of float RTK and PPK results for the three selected sections. It is found that overall, the float solution has a centimeter to decimeter differences from the reference solutions. Smoothing method (PPK) can improve the filter-based (RTK) estimation results, such as shown in the red box in Figure 3.3 and Figure 3.4.



Trajectory Section 1

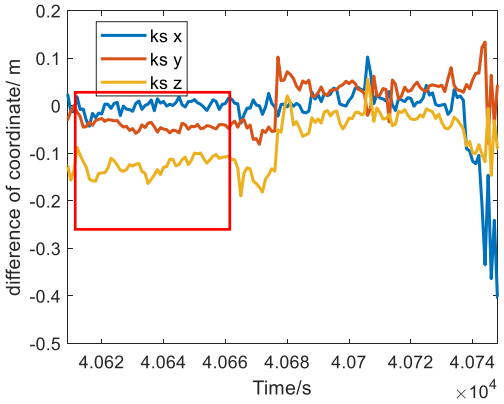


Trajectory Section 2

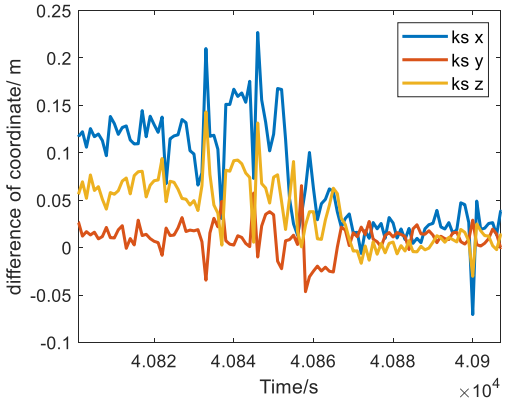


Trajectory Section 3

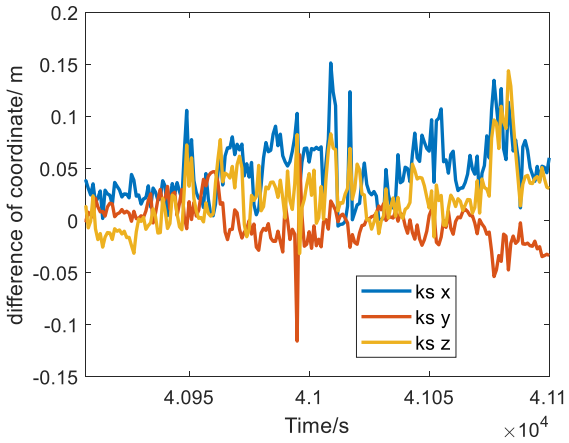
Figure 3.3 Coordinate differences between the GNSS KF float solution and the reference trajectory



Trajectory Section 1



Trajectory Section 2



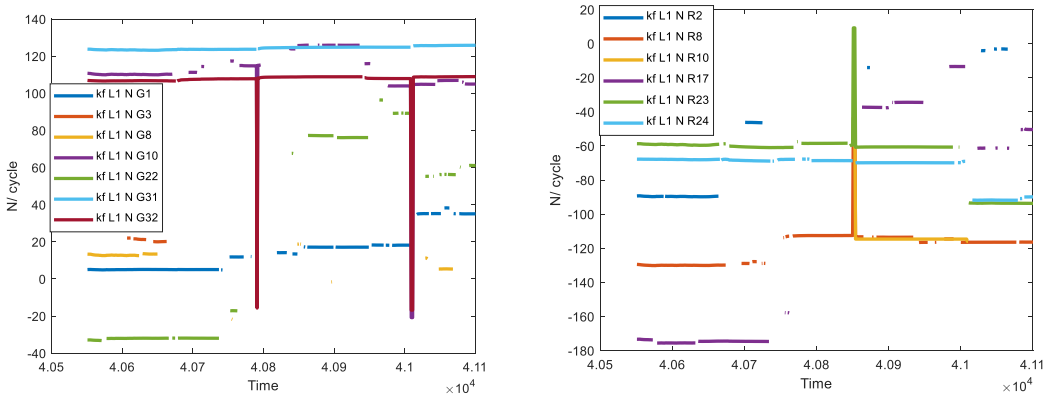
Trajectory Section 3

Figure 3.4 Coordinate differences between the GNSS KS float solution and the reference trajectory

One reason for this improvement of the post processing step is that the double differenced ambiguity (N) is more stable than the real-time N since the KS step utilize all

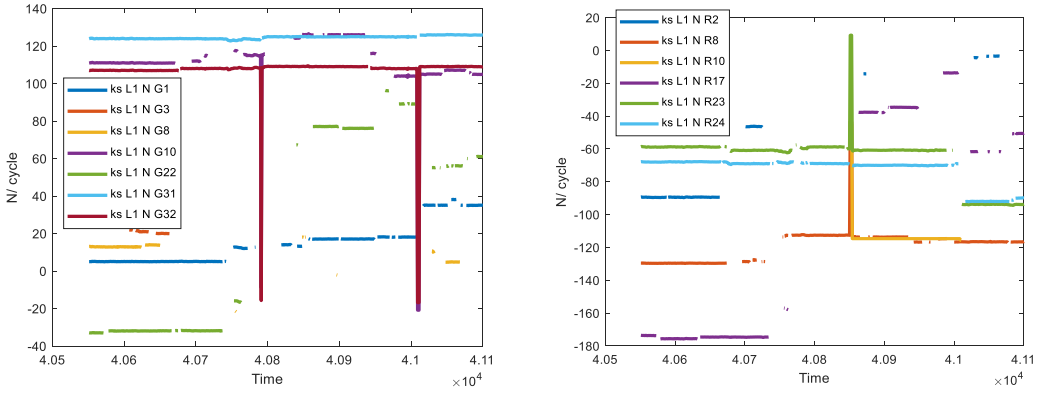
measurements to estimate the N. For the real-time case, at each time epoch when one satellite is initially viewed, its N value will be initialized, and it will take some time to get convergent. However, for the backward smoothing, it will use the ambiguity value which is very closed to the final converged N value at the last epoch of this viewed period, although this N may be still not converged to an integer. Figure 3.5 and Figure 3.6 show the double difference Ambiguity value by the RTK and PPK.

It is found there are some sudden jumps of N value in Figure 3.5 and Figure 3.6, this is due to the change of reference satellites. This may happen both for the GPS and the GLONASS systems.



(a) (b)

Figure 3.5 Double differenced Ambiguity in float RTK solution: (a) GPS; (b) GLONASS of the whole driving trajectory

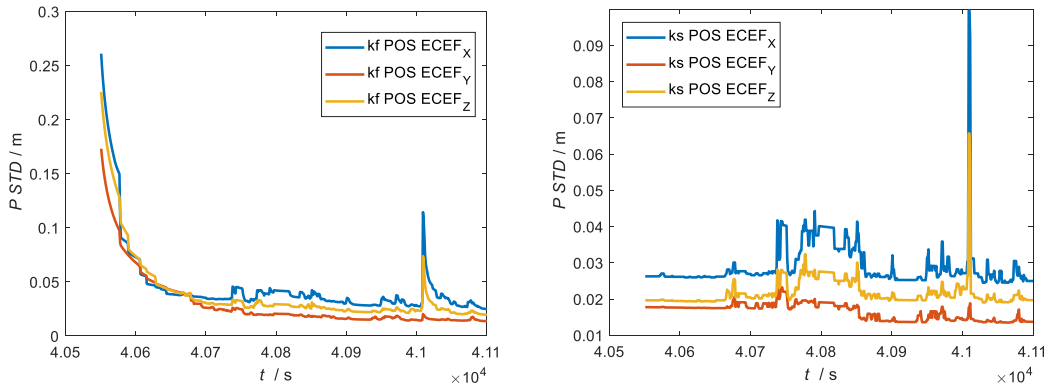


(a) (b)

Figure 3.6 Double differenced Ambiguity in float PPK solution: (a) GPS; (b) GLONASS of the whole driving trajectory

Figure 3.7 shows the Standard Deviation results of the position in this ambiguity float case. Overall, PPK has better accuracy than RTK. It is found the lost and newly viewed

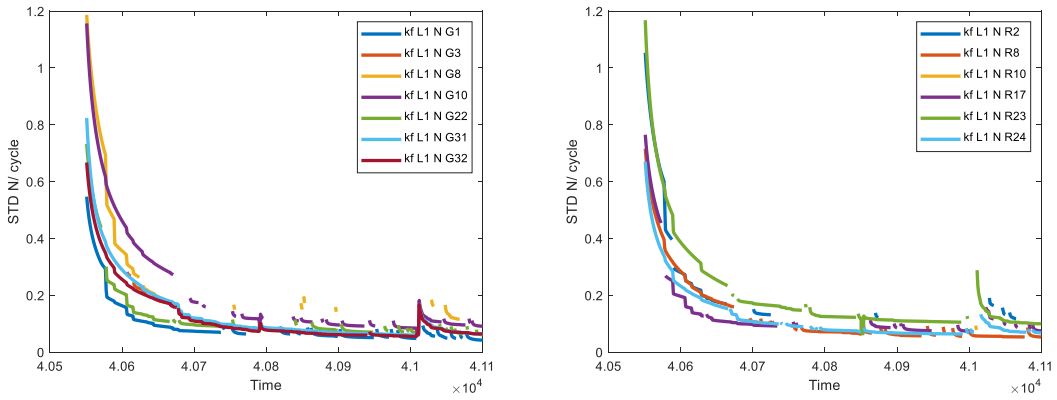
satellites will influence the accuracy of the position estimation, because the geometry is changed.



(a) (b)

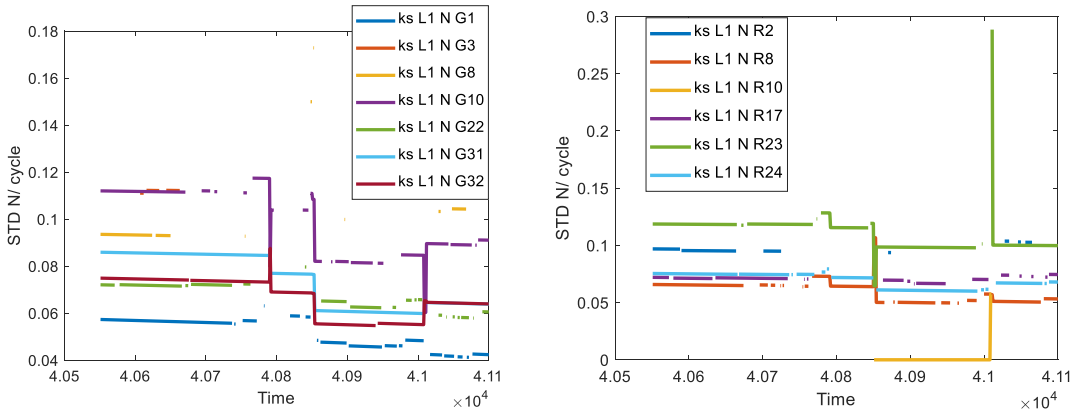
Figure 3.7 Standard deviations of the position states in ambiguity float solutions: (a) RTK solution; (b) PPK solution of the whole driving trajectory

Figure 3.8 and Figure 3.9 are the Standard Deviation results of the ambiguity in this ambiguity float RTK and PPK cases. It can be found that the satellite geometry change will cause reconvergence for any newly viewed satellites in RTK, but for PPK, this uncertainty value is stable as all the measurements are used together.



(a) (b)

Figure 3.8 Standard deviations of the Double differenced Ambiguities in float RTK solution: (a) GPS; (b) GLONASS of the whole driving trajectory

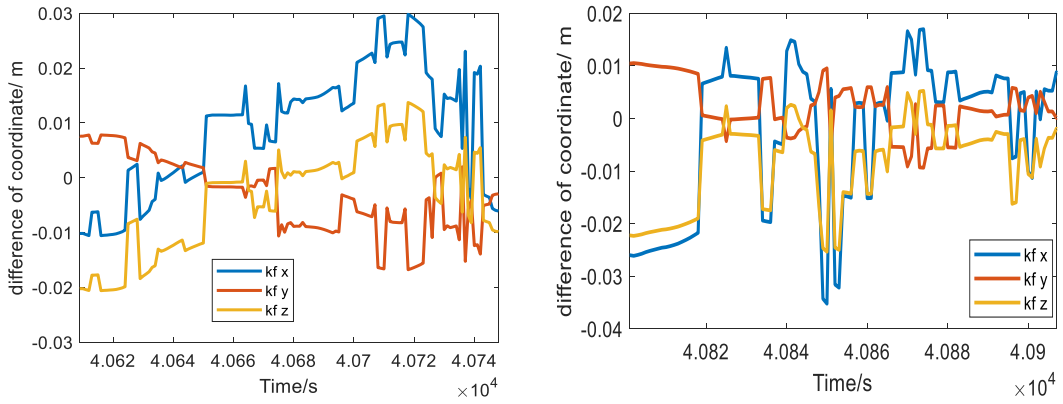


(a) (b)

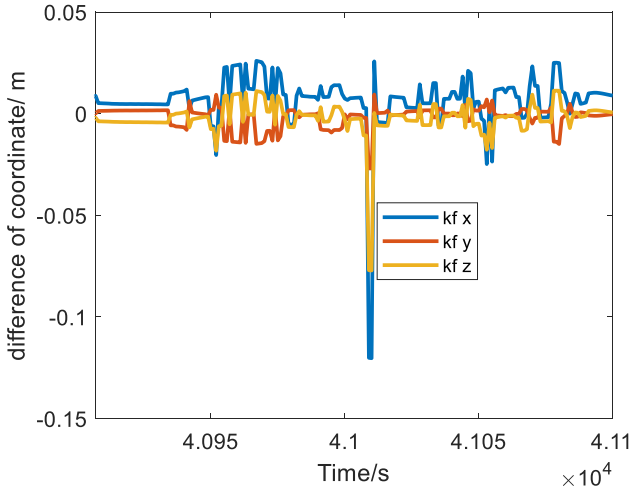
Figure 3.9 Standard deviations of the Double differenced Ambiguity in float PPK solutions: (a) GPS; (b) GLONASS of the whole driving trajectory

3.3.1.2 DD Ambiguity Fixed Results

Figure 3.10 and Figure 3.11 shows the ambiguity fixed solutions for the RTK and PPK, which are compared with the reference trajectory. It is found if with proper integer ambiguity resolutions, the accuracy will be improved to centimeter level. The fixed RTK and fixed PPK will get almost same solutions. This is due to the fact that, once DD ambiguities are fixed into their correct integers, RTK and PPK are essentially the individual epoch-based solutions. However, the epoch with very poor geometry strength in the GNSS only solutions has about 0.1 meters difference from the GNSS/IMU integration-based reference. Such a difference happened during the epochs in which less than 5 satellites being tracked.

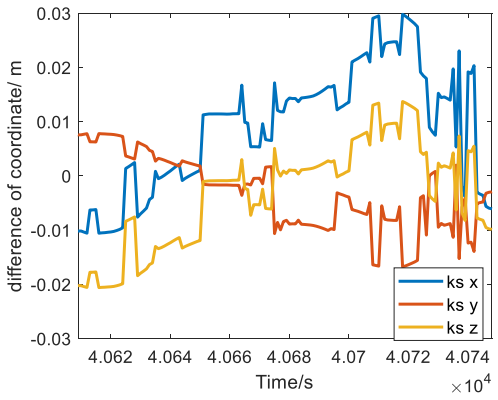


Trajectory Section 1 Trajectory Section 2

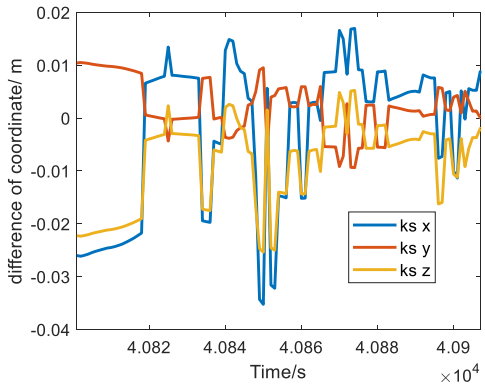


Trajectory Section 3

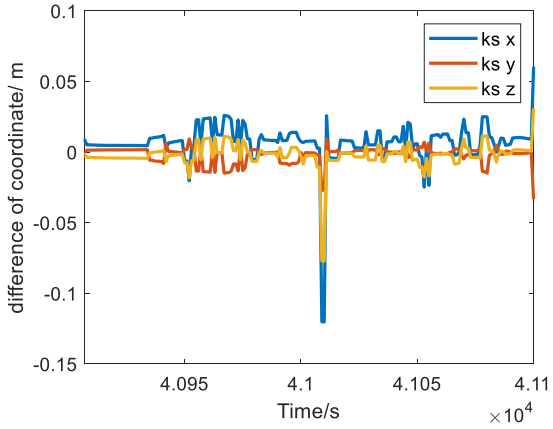
Figure 3.10 Coordinate differences between the GNSS KF fixed solution and the reference trajectory



Trajectory Section 1



Trajectory Section 2

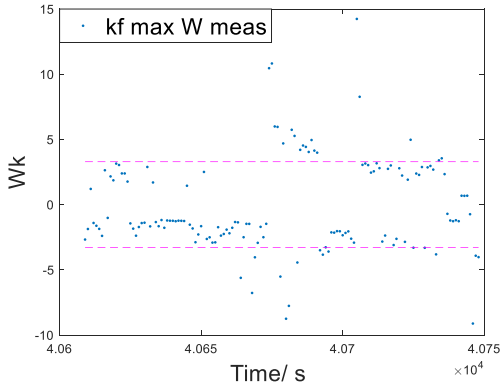


Trajectory Section 3

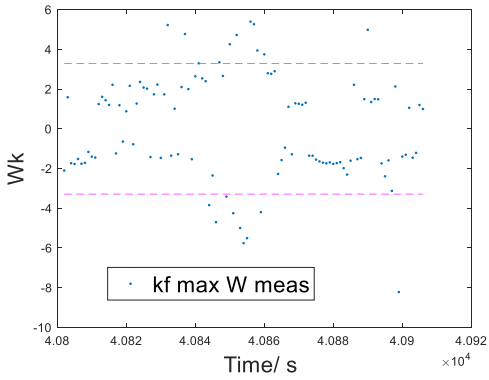
Figure 3.11 Coordinate differences between the GNSS KS fixed solution and the reference trajectory

3.3.1.3 Quality Control for the GNSS Double Differencing Method

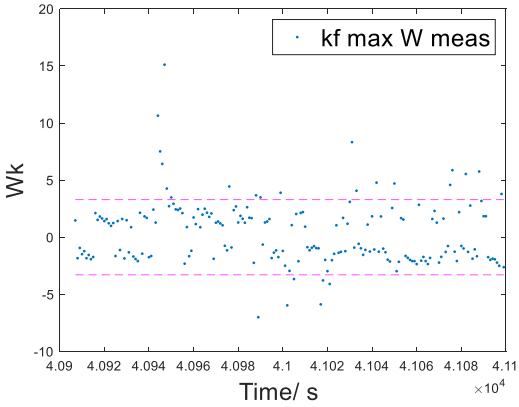
Quality control can be done for both RTK and PPK. Both float RTK and PPK has detected some outliers (Figure 3.12 and Figure 3.13).



Trajectory Section 1

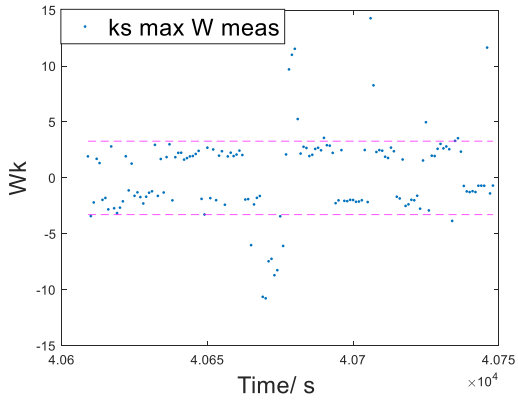


Trajectory Section 2

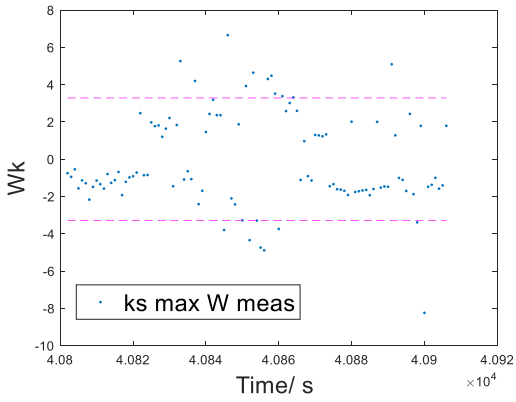


Trajectory Section 3

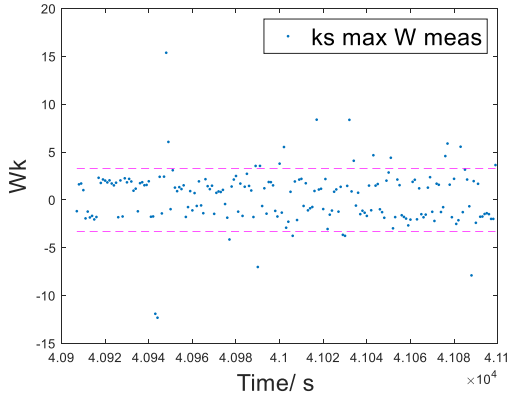
Figure 3.12 Statistic test results of the float RTK measurements



Trajectory Section 1



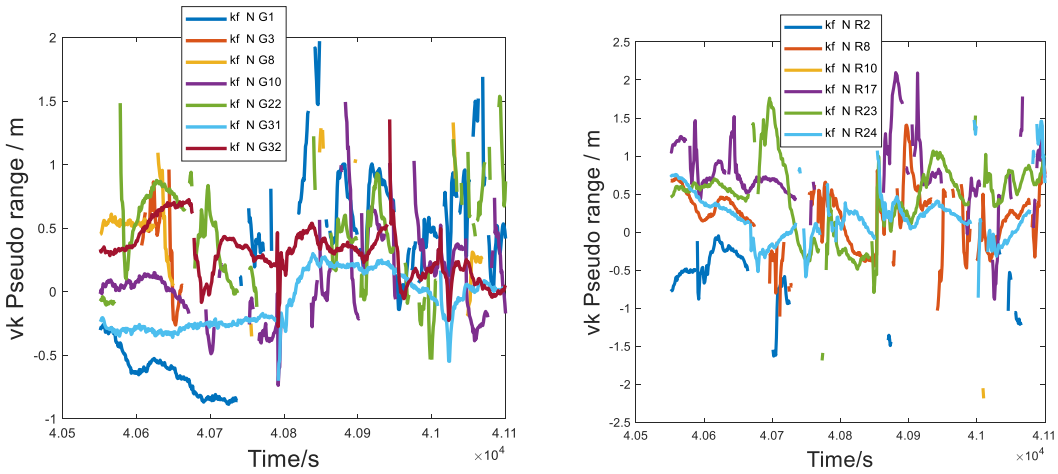
Trajectory Section 2



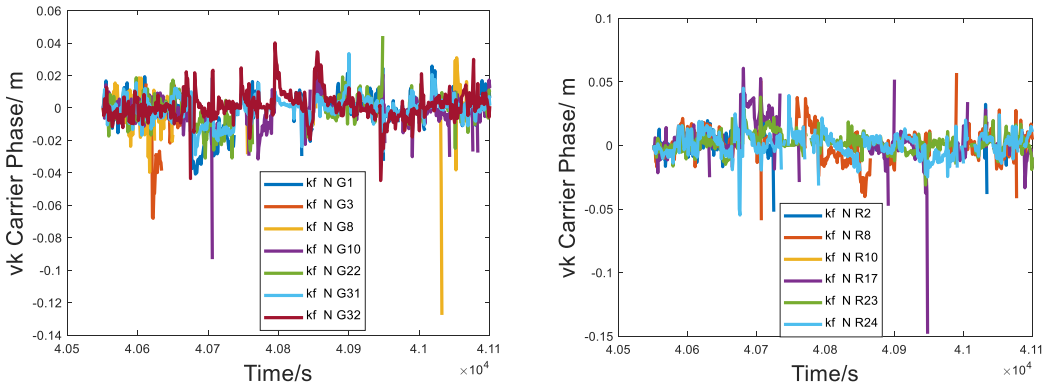
Trajectory Section 3

Figure 3.13 Statistic test results of the float PPK measurements

Since our quality control method is residual based, and after ambiguity resolution (AR), the residual can be re-calculated with the new fixed states, the residual results before and after ambiguity fixing can be used to show the influence of AR. Figure 3.14 to 3.17 shows the residual results of both Pseudo Range and Carrier Phase measurements. It is found for both float RTK and PPK, there are some big jumps of Carrier Phase, which is up to 15 cm, indicating that the float Ambiguity will influence the estimation. The PPK has slightly more stable residual value for some satellites, but still contains the jumps. If ambiguity integers are fixed, it can be found the residual values of Carrier Phase are much stabler than the float cases, however, due to the fact that less number of unknown parameters in ambiguity-fixed solutions, some remaining systematic errors, such as multipath, will show up in the residuals which appears to be less random, in comparison with the residuals from the ambiguity float solutions.

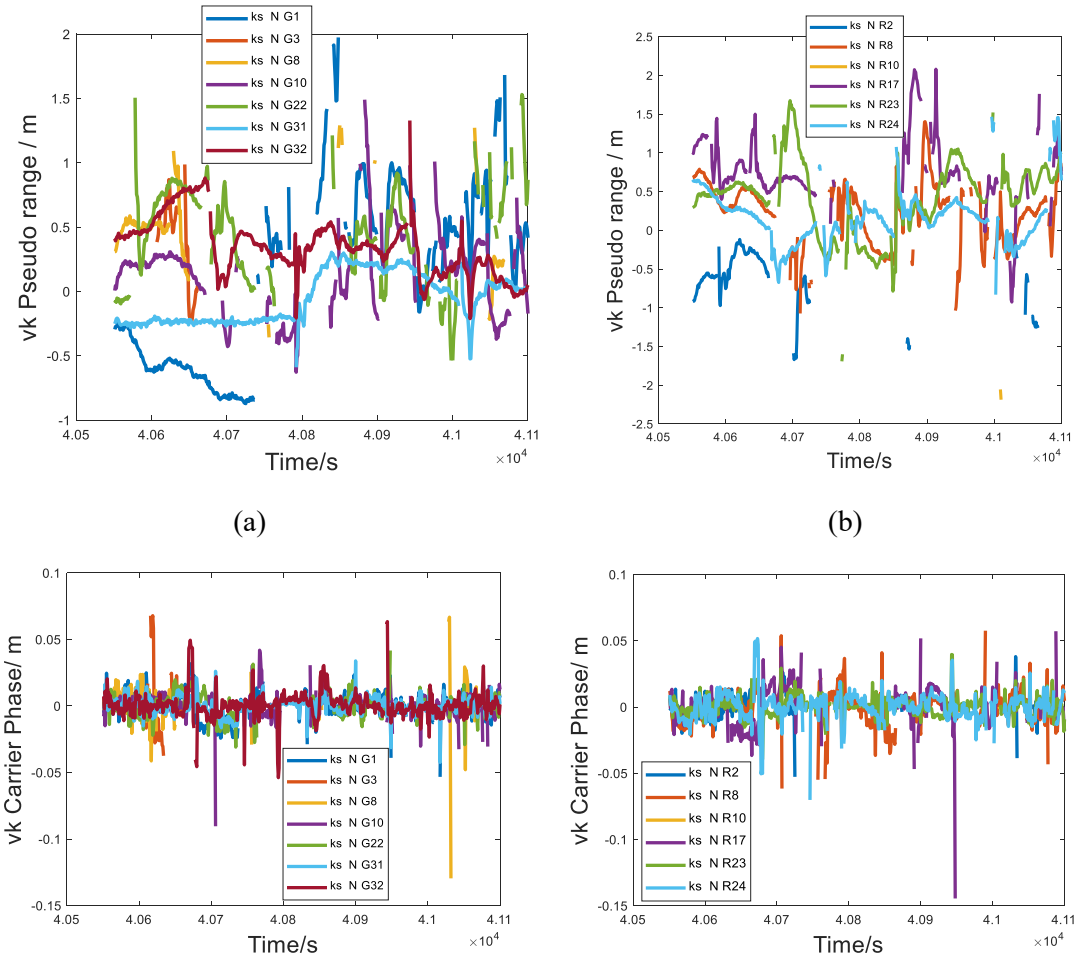


(a) (b)



(c) (d)

Figure 3.14 Residual results of the float RTK measurement: (a) Pseudo Range of GPS; (b) Pseudo Range of GLONASS; (c) Carrier Phase of GPS; (d) Carrier Phase of GLONASS of the whole driving trajectory



(a) (b)

(c) (d)

Figure 3.15 Residual results of the float PPK measurement: (a) Pseudo Range of GPS; (b) Pseudo Range of GLONASS; (c) Carrier Phase of GPS; (d) Carrier Phase of GLONASS of the whole driving trajectory

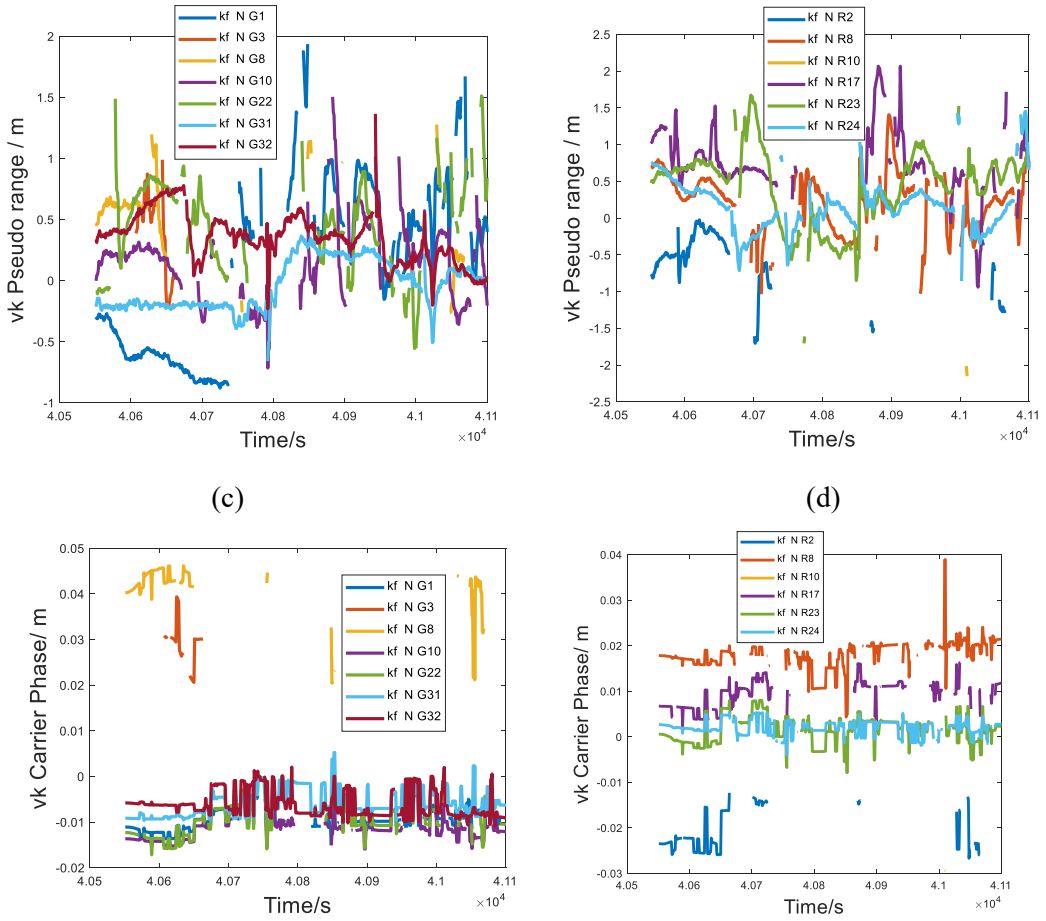
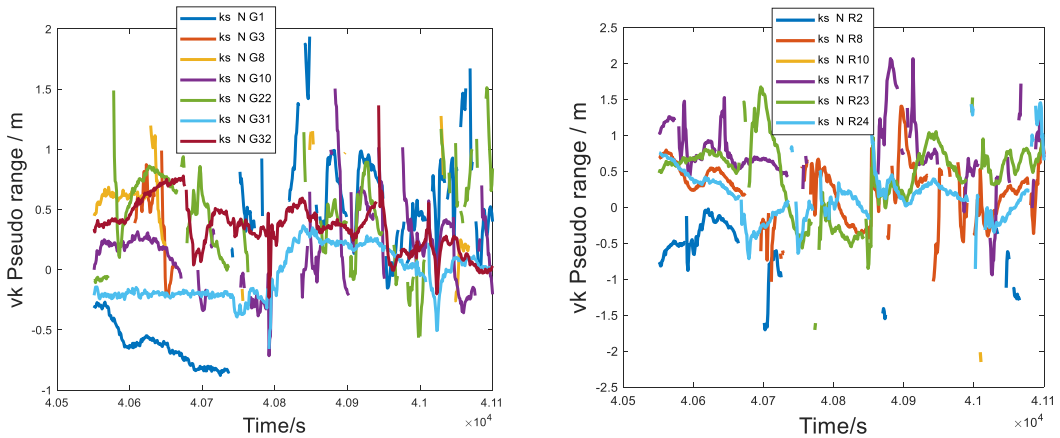
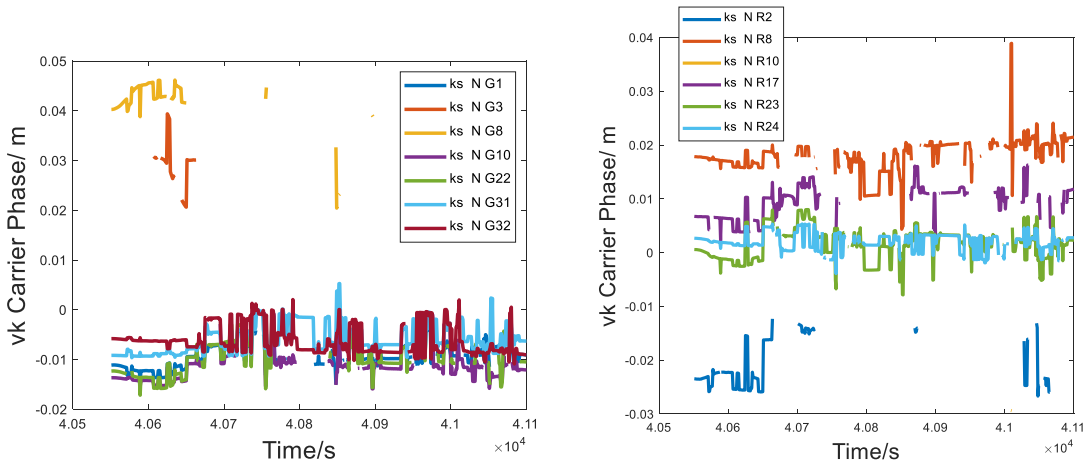


Figure 3.16 Residual results of the fixed RTK measurement: (a) Pseudo Range of GPS; (b) Pseudo Range of GLONASS; (c) Carrier Phase of GPS; (d) Carrier Phase of GLONASS of the whole driving trajectory

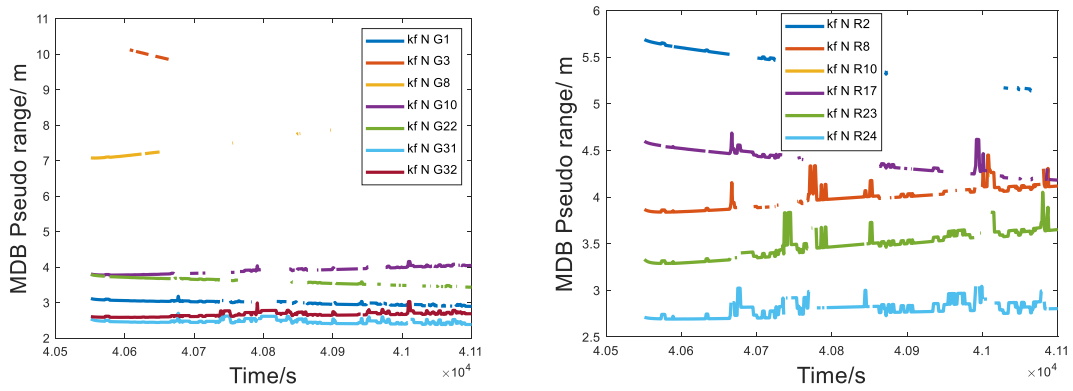




(c) (d)

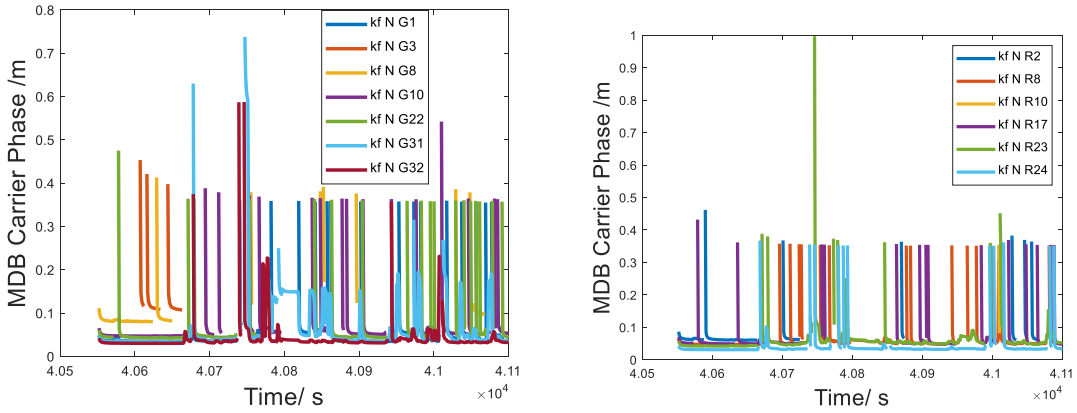
Figure 3.17 Residual results of the fixed PPK measurement: (a) Pseudo Range of GPS; (b) Pseudo Range of GLONASS; (c) Carrier Phase of GPS; (d) Carrier Phase of GLONASS of the whole driving trajectory

Reliability analysis has been done for the Double Differencing case. It can be found the change of geometry will influence the MDB value of both Pseudo Range (Figure 3.18) and Carrier Phase measurement (Figure 3.19). Each time one satellite is newly viewed, its MDB value will restart the converging. Geometry change will also influence the Reliability of predicted N. In poor geometry periods, there will be very high MDB for N value.



(a) (b)

Figure 3.18 MDB values of the Pseudo Range measurement for (a) GPS, and (b) GLONASS in RTK of the whole driving trajectory



(a) (b)

Figure 3.19 MDB values of the Carrier Phase measurement for (a) GPS, and (b) GLONASS in RTK of the whole driving trajectory

The predicted states of position and ambiguity N can also be tested with the Reliability (Figure 3.20 and Figure 3.21). It should be noticed that since in RTK and PPK, the position dynamic processing is treated as a random walk, therefore, the covariance of the dynamic model is set as very high. In this case, the MDB value of the predicted state of position is also high, up to 413 meters.

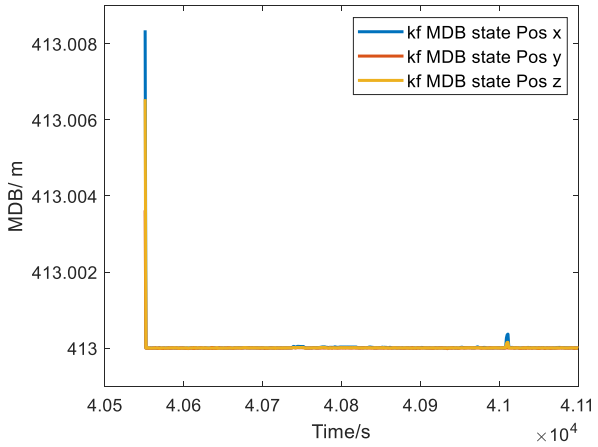
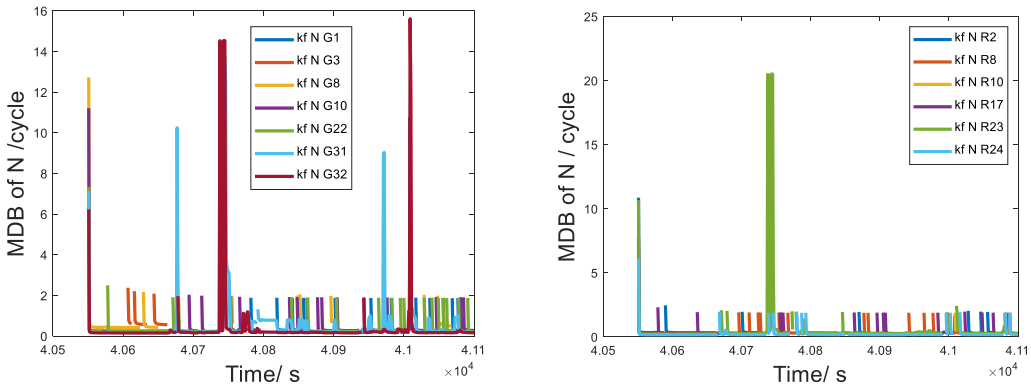


Figure 3.20 MDB values of the predicted position states in RTK of the whole driving trajectory

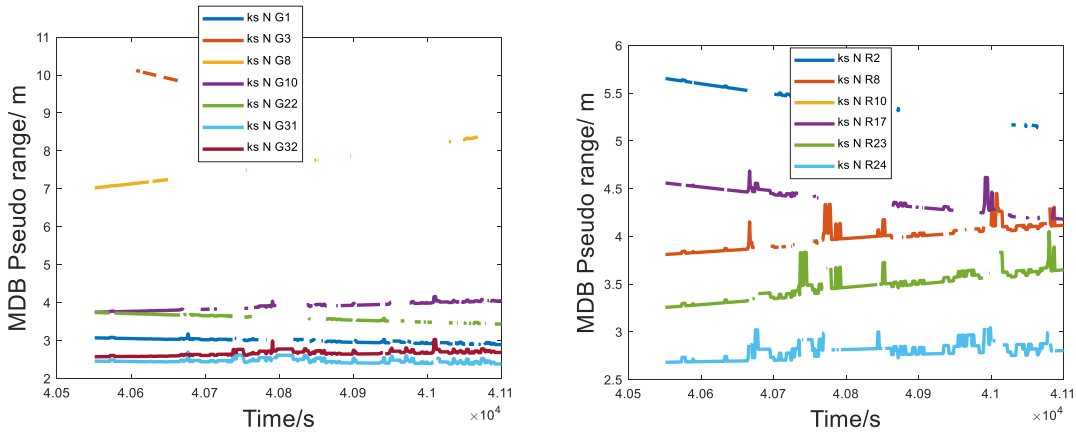
The MDB value of the predicted Ambiguity states can get converged very quickly down to 0.2-0.4 cycle, but still higher than the estimation uncertainty.



(a) (b)

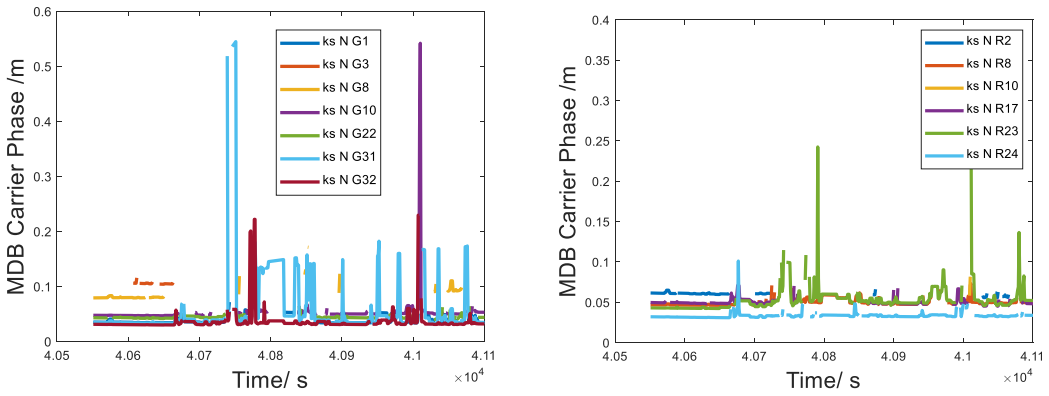
Figure 3.21 MDB values of the predicted double difference ambiguity states for (a) GPS, and (b) GLONASS in RTK of the whole driving trajectory

PPK can do Reliability analysis for the measurement model and the dynamic model. The MDB for Pseudo Range is closed to the RTK case (Figure 3.22), while for the Carrier Phase (Figure 3.23), the PPK MDB value are more stable and lower. The dynamic model is also less sensitive to the geometry change, especially for the case of low satellite number (Figure 3.25).



(a) (b)

Figure 3.22 MDB values of the Pseudo Range measurement for (a) GPS, and (b) GLONASS in PPK of the whole driving trajectory



(a) (b)

Figure 3.23 MDB values of the Carrier Phase measurement for (a) GPS, and (b) GLONASS in PPK of the whole driving trajectory

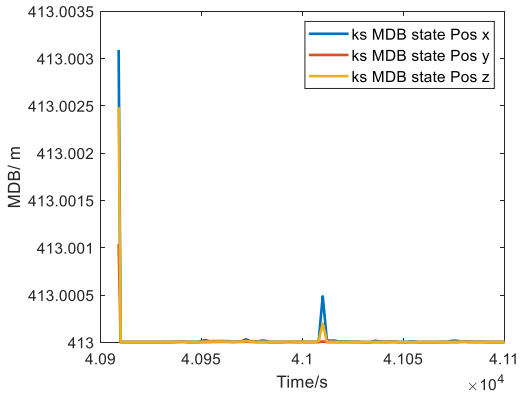
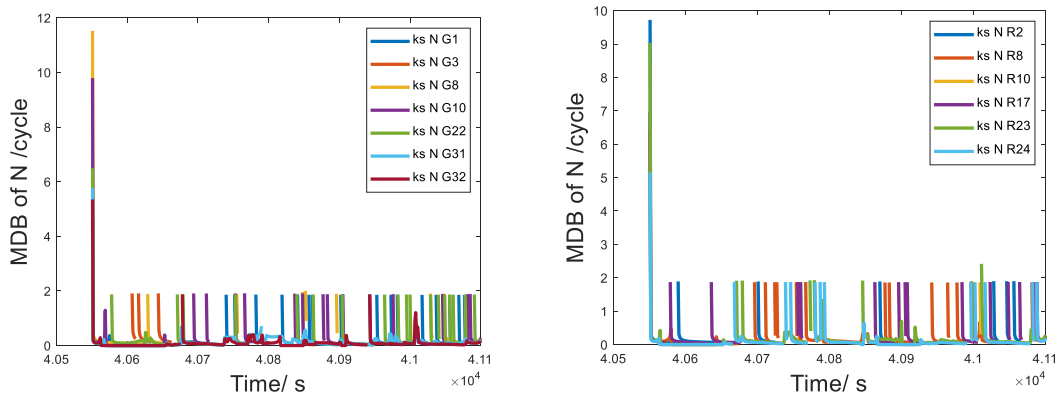


Figure 3.24 MDB values of the dynamic position model in PPK of the whole driving trajectory



(a) (b)

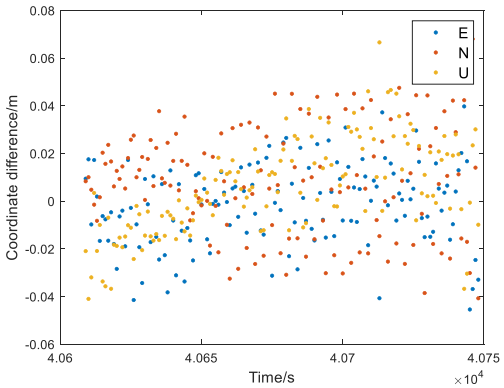
Figure 3.25 MDB values of the dynamic double difference ambiguity state model for (a) GPS, and (b) GLONASS in PPK of the whole driving trajectory

3.3.2 GNSS/INS Integration

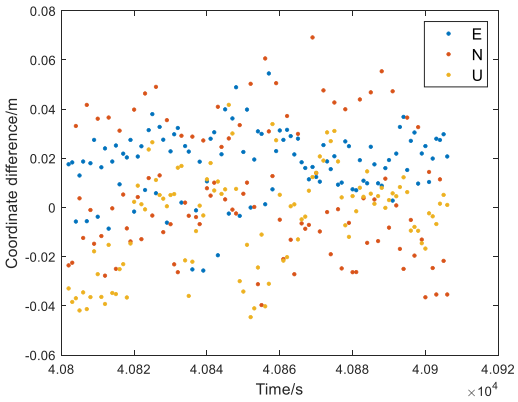
The fixed double differencing results are used to integrate with the IMU measurements.

3.3.2.1 GNSS/INS Estimation Results

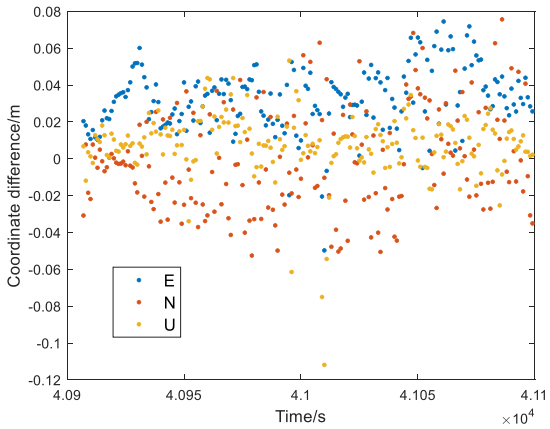
Figure 3.26 and Figure 3.27 shows the coordinate differences of the loose-coupled integration strategy by KF and KS with respect to the reference trajectory. The overall accuracy is at centimeter level while at some epoch, the accuracy is reduced to sub-meter level which may be due to low accuracy of input GNSS measurements. It is found that the backward smoothing method has slightly better accuracy, especially for the Trajectory Section 3.



Trajectory Section 1

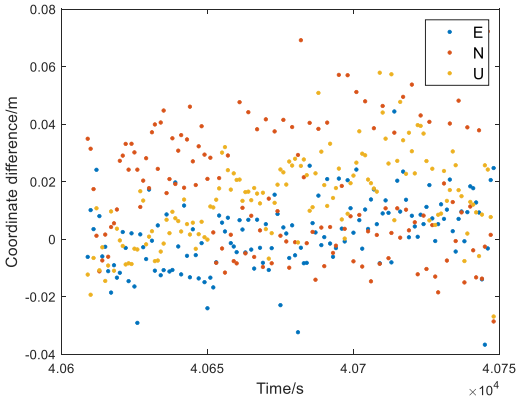


Trajectory Section 2

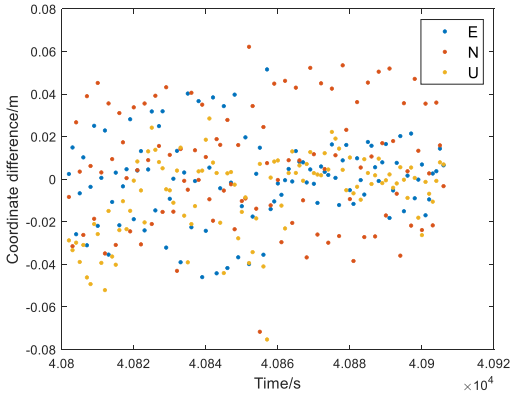


Trajectory Section 3

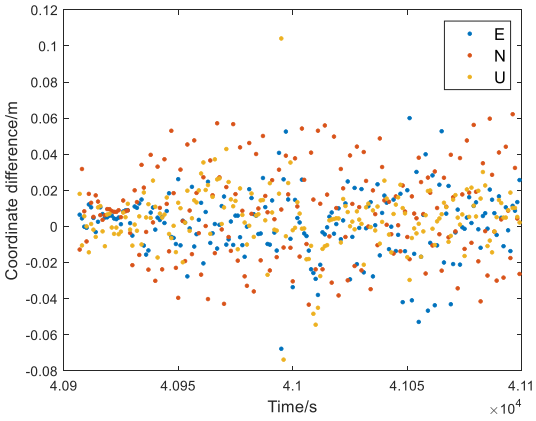
Figure 3.26 Coordinate differences between the GNSS/INS KF integration solution and the reference trajectory



Trajectory Section 1



Trajectory Section 2



Trajectory Section 3

Figure 3.27 Coordinate differences between the GNSS/INS KS integration solution and the reference trajectory

Figure 3.28 is the estimation Standard Deviation for the 9 navigation error sates and 6 IMU sensor bias states by KF. Quick convergence can be achieved for position, velocity and pitch and roll. However, the heading STD value has some increasing period, which may because the heading angle error is unobservable when the horizontal specific force components are zero (Farrell, and Wendel., 2017). The heading angle error cannot be affected by the velocity errors in such condition. As a consequence, the uncertainty of the heading error state will grow with time when the platform is static or driving in nonaccelerating motion (Farrell, and Wendel., 2017). Since the driving tests will have some period static, such as waiting for the traffic light, it will cause the increase of heading uncertainty.

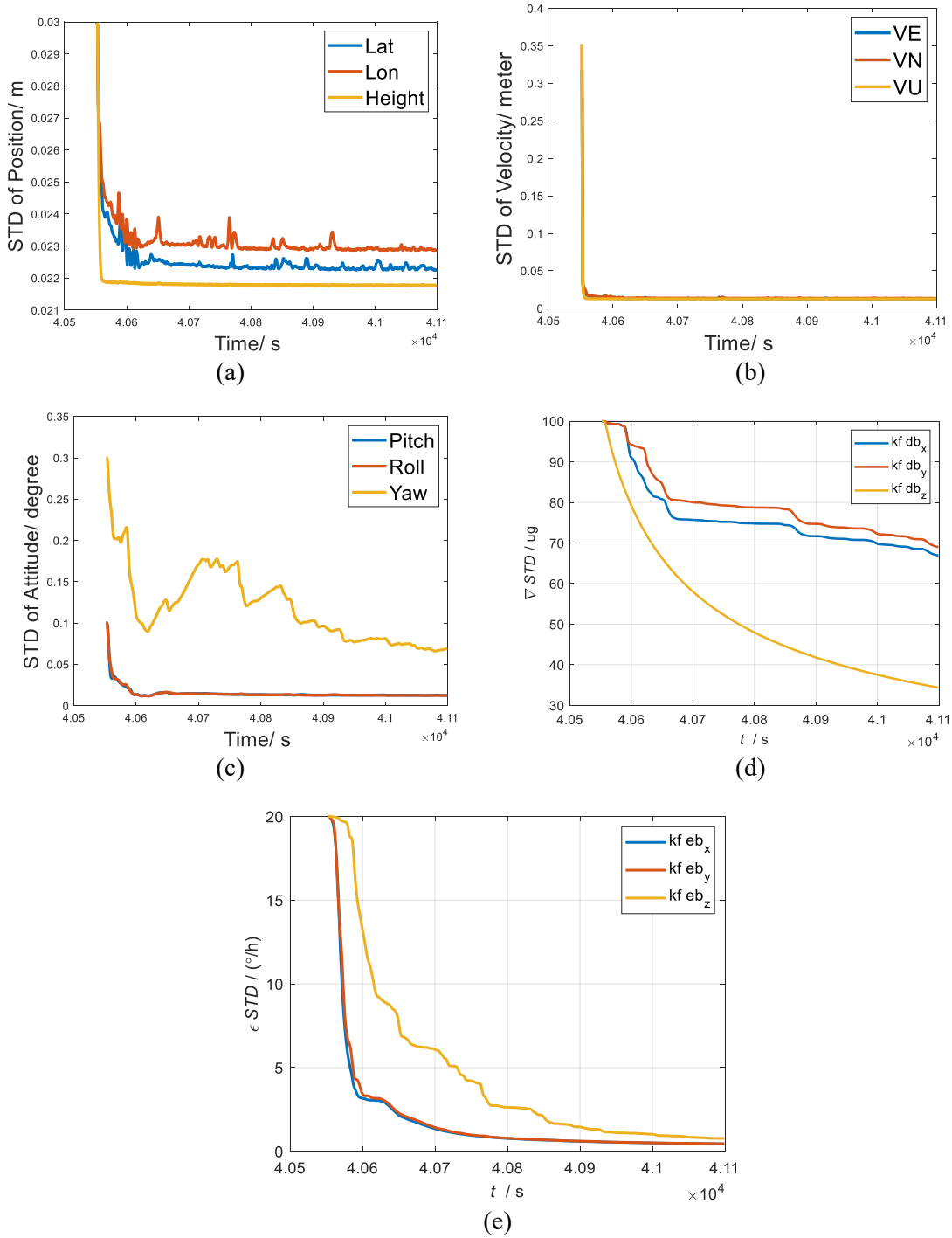


Figure 3.28 Standard deviations of the KF GNSS/INS integration solution for states: (a) position error; (b) velocity error; (c) attitude error; (d) accelerometer biases; (e) gyro drifts

Figure 3.29 is the estimation Standard Deviation for the 9 navigation error sates and 6 IMU sensor bias states by KS. It is found that much better estimation precision can be achieved by the backward smoothing methods. The STD values keep quict stable, especially for the accelemeter bias states and gyro drift states.

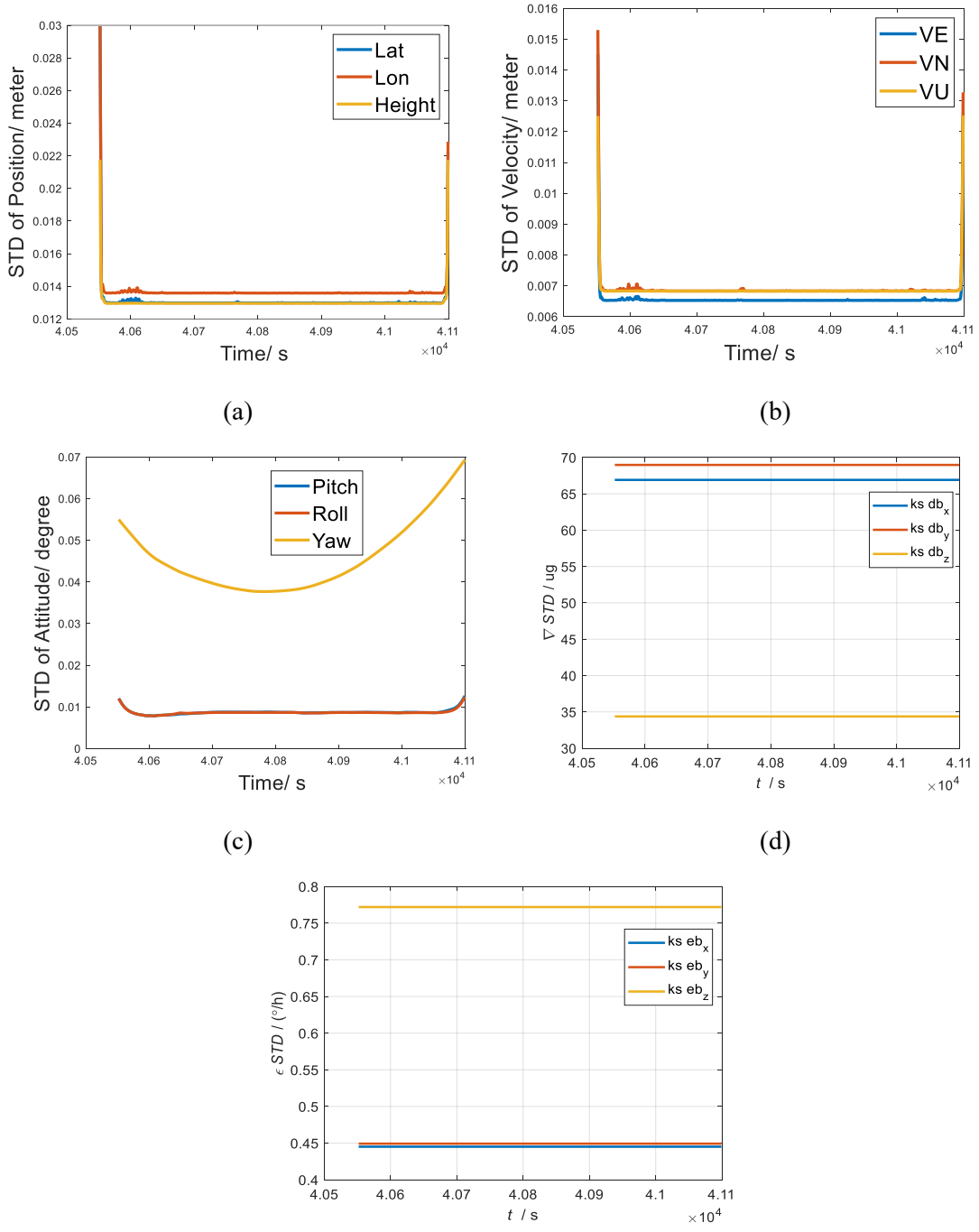
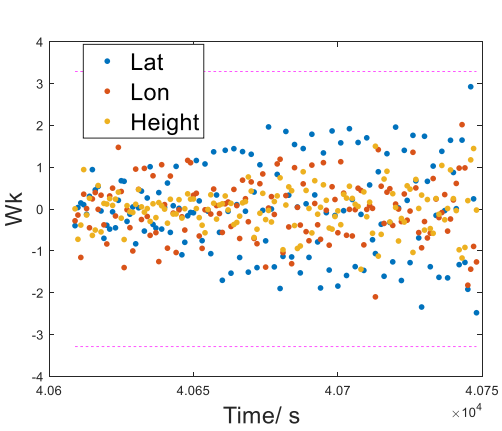


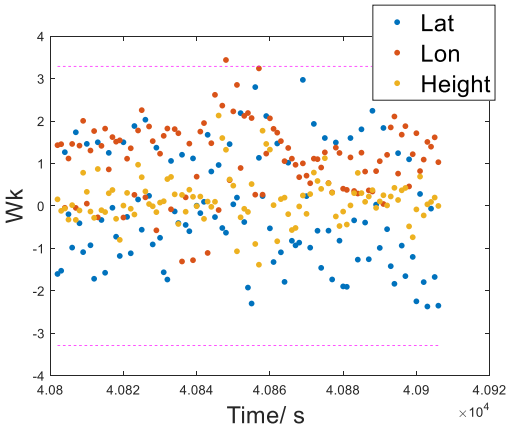
Figure 3.29 Standard deviations of the KS GNSS/INS integration solution for states: (a) position error; (b) velocity error; (c) attitude error; (d) accelerometer biases; (e) gyro drifts

3.3.2.2 Quality Control for GNSS/INS Integration

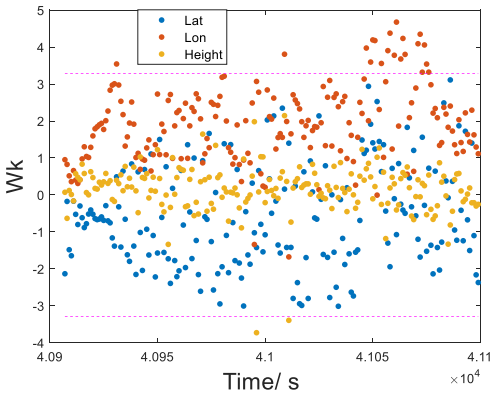
Figure 3.30 shows the *W* values for GNSS measurements are mostly less than 3.29, indicate no outlier within it for most of the time, except for some epochs, indicating some outlier detected in such epochs, especially in Trajectory Section 3 by KF.



Trajectory Section 1



Trajectory Section 2



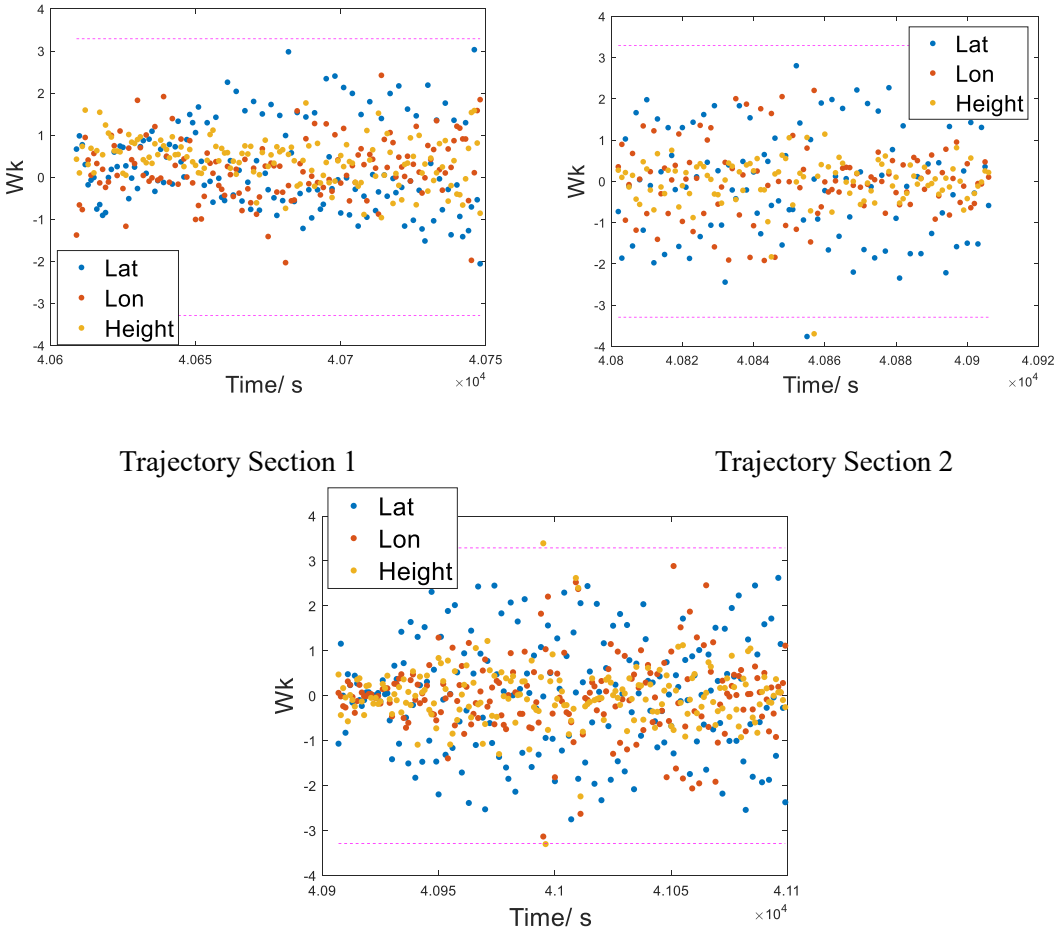
Trajectory Section 3

Figure 3.30 Statistic test results of the KF GNSS/INS measurement

Figure 3.31 is the statistic test results by KS. It is found that a smaller number of outliers are found by the KS method than the KF method, especially for the Trajectory Section 3 part. By checking the Coordinate difference results in Figure 3.26 and 3.27, it is also found that KS has better accuracy than KF at the period that KF has many outliers detected. Therefore, the possible outliers may not exist in the GNSS measurements, but are within the KF predicted states. These outliers may be due to the influence of some outliers in GNSS measurement before this period.

Since for this Loosely Coupled integration system, the GNSS measured and predicted position are highly corelated, the w test has almost same results on measurement and predicted position, therefore, it cannot exactly identify whether the outlier in within the measurement model or within the KF predicted states, which may cause misidentification of measurement outlier. For the KS method, it utilized the measurements after the epoch

with measurement outliers to do full optimization, therefore estimation for the period after that epoch will be less influenced by the outlier.

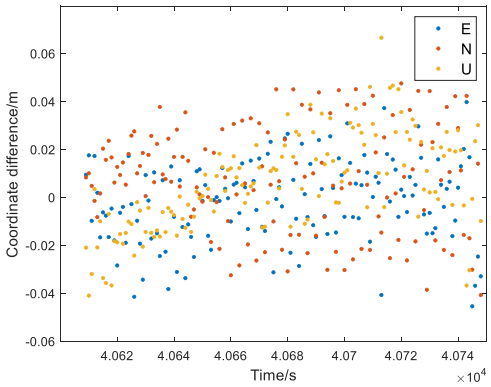


Trajectory Section 1 Trajectory Section 2

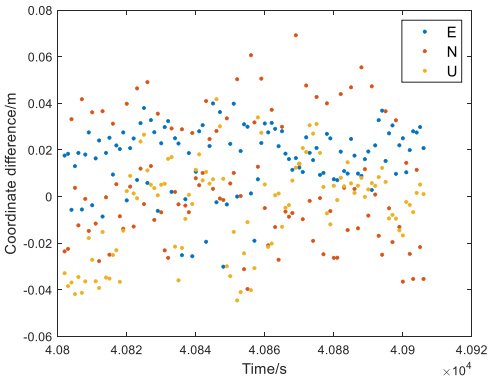
Trajectory Section 3

Figure 3.31 Statistic test results of the KS GNSS/INS measurements

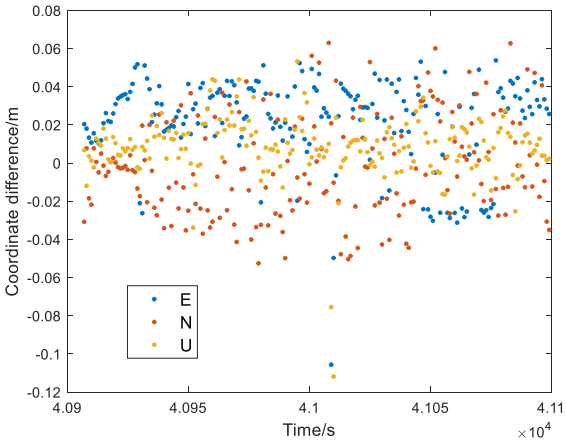
With the detection and identification results, the outlier influence can be mitigated with the quality control results. There are two ways to mitigate the influence of outliers. Firstly, the Adaption method can be used to achieve adapted state parameters. Secondly, the estimated outlier value is directly used to correct the measurements. Figure 3.32 and Figure 3.33 are the coordinate difference results with respect to the reference trajectory with the adapted KF and KS solution. It is found the KF results in Trajectory Section 3 is improved during period that outliers are detected.



Trajectory Section 1

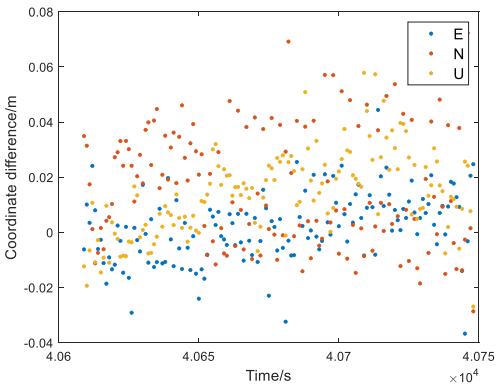


Trajectory Section 2

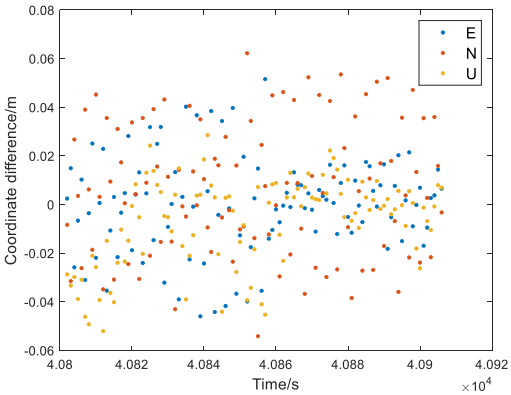


Trajectory Section 3

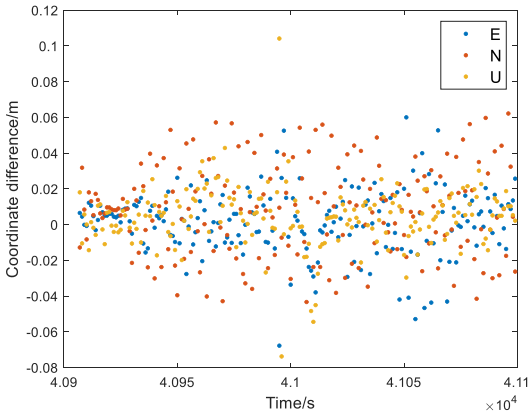
Figure 3.32 Coordinate differences between the Adapted GNSS/INS KF integration solutions and the reference trajectory



Trajectory Section 1



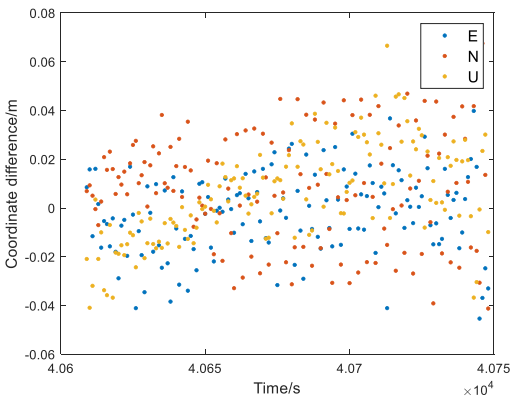
Trajectory Section 2



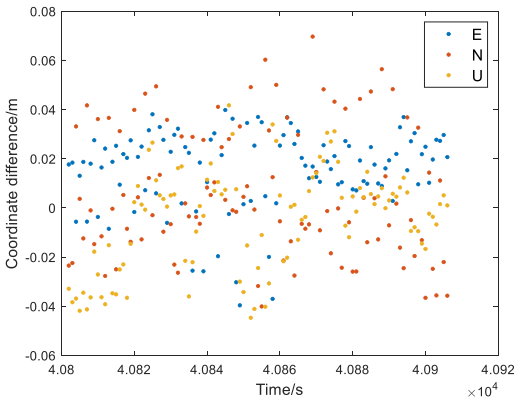
Trajectory Section 3

Figure 3.33 Coordinate differences between the Adapted GNSS/INS KS integration solutions and the reference trajectory

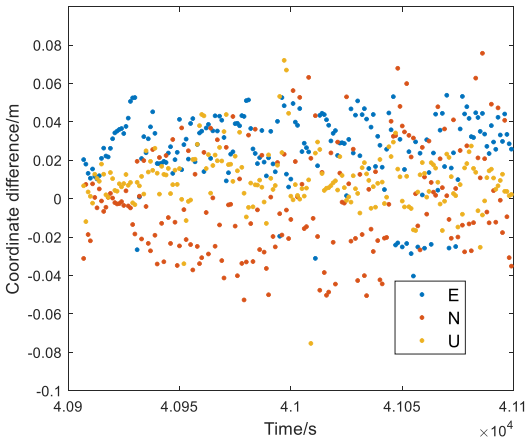
Figure 3.34 and Figure 3.35 is the coordinate difference results with respect to the reference trajectory by correcting the input measurement directly during the KF step with the KF estimated outlier value. It is found for the KF, the results in Trajectory Section 3 are improved during period that outliers are detected. However, for the KS, this period's results are getting worse, which may be due to the outlier value used for correction is estimated by KF. Hence for the offline KS method, it should use the KS estimated outlier value to correct the measurement and then do KF and KS again to achieve corrected KS solutions.



Trajectory Section 1

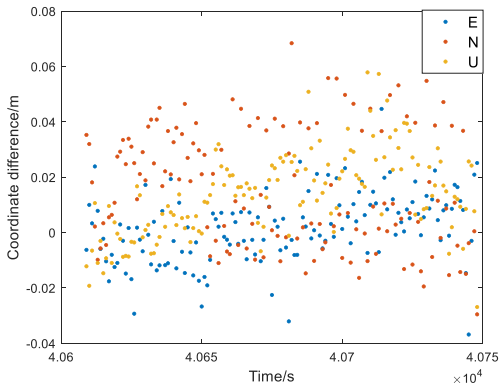


Trajectory Section 2

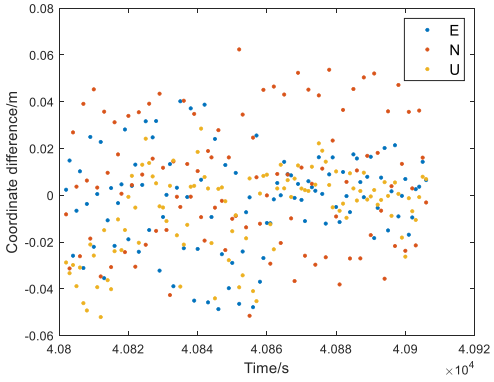


Trajectory Section 3

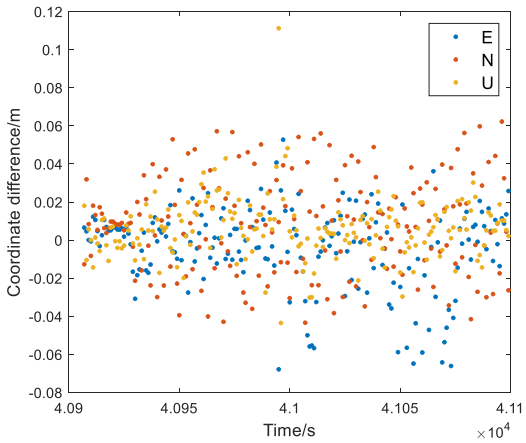
Figure 3.34 Coordinate differences between the Corrected GNSS/INS KF integration solutions and the reference trajectory



Trajectory Section 1



Trajectory Section 2



Trajectory Section 3

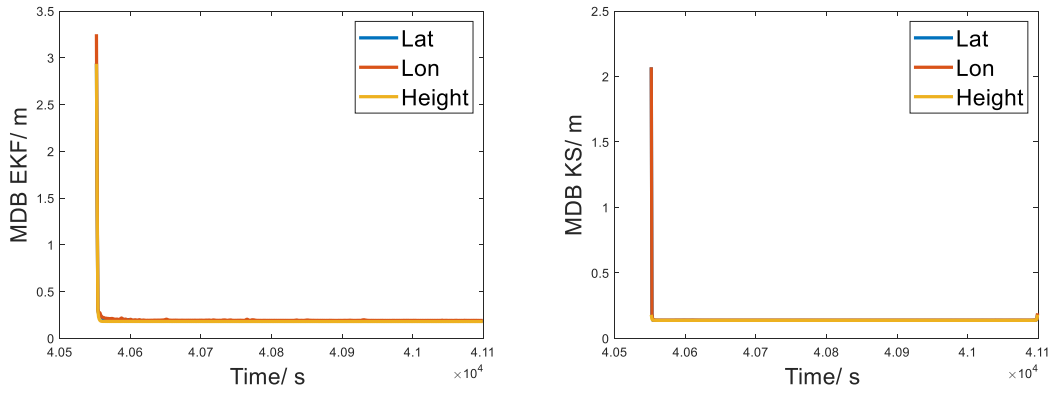
Figure 3.35 Coordinate differences between the Corrected GNSS/INS KS integration solutions and the reference trajectory

the mean value and std value of the coordinate difference of original solutions and solutions with Adaption and Correction of the estimated outliers is summarised in Table 3.2.

Table 3.2 Mean and standard deviation for the differences between the GNSS (KF and KS) results and the reference online GNSS/INS localization results before and after adaption or correction the influence of detected outliers

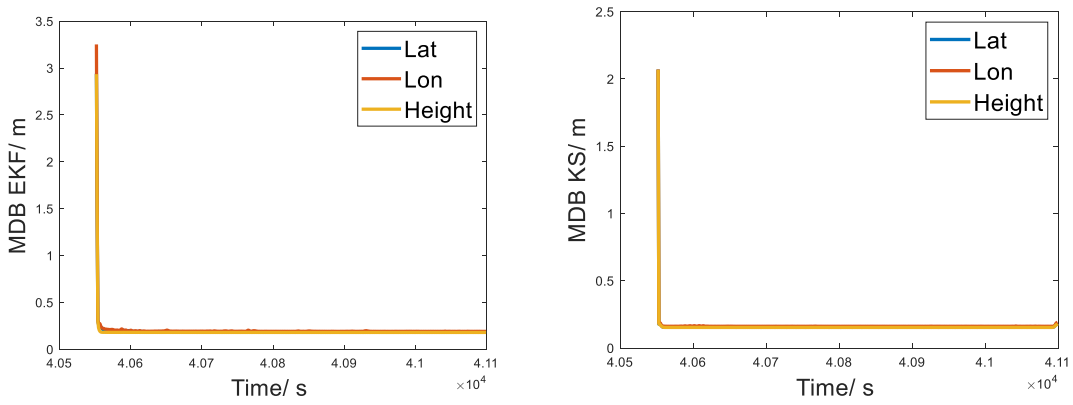
			Section 1			**Section 2**			**Section 3**		
			Before correction	After correction	With Outlier Adaption	Before correction	After correction	With Outlier Adaption	Before correction	After correction	With Outlier Adaption
KF	Mean (m)	E	-0.0020	-0.0020	-0.0023	0.0178	0.0164	0.0155	0.0322	0.0216	0.0246
		N	0.0082	0.0082	0.0079	0.0060	0.0061	0.0061	-9.22×10^{-4}	-0.0013	-9.066×10^{-4}
		U	0.0057	0.0057	0.0057	-0.0044	-0.0044	-0.0045	0.004275	0.0084	0.0081
	Stdev (m)	E	0.0170	0.0170	0.0169	0.0140	0.0139	0.0163	0.0172	0.0238	0.0230
		N	0.0226	0.0226	0.0226	0.0261	0.0261	0.0261	0.0275	0.0263	0.0275
		U	0.0199	0.0199	0.0199	0.0204	0.0204	0.0204	0.0177	0.0165	0.0175
KS	Mean (m)	E	0.0017	0.0017	0.0016	-1.387×10^{-4}	-1.387×10^{-4}	-0.0023	0.0011	0.0011	-0.0065
		N	0.0165	0.0165	0.0163	0.0063	0.0062	0.0066	0.0080	0.0080	0.0080
		U	0.0137	0.0137	0.0137	-0.0045	-0.0045	-0.0042	0.0042	0.0042	0.0057
	Stdev (m)	E	0.0125	0.0125	0.0126	0.0197	0.0197	0.0200	0.0180	0.0180	0.0253
		N	0.0215	0.0215	0.0216	0.0245	0.0241	0.0240	0.0252	0.0252	0.0252
		U	0.0152	0.0152	0.0152	0.0191	0.0183	0.0183	0.0169	0.0169	0.0158

Reliability analysis is also made for this integration system. Figure 3.36 and Figure 3.37 are the MDB results for the measurements and the predicted states or dynamic model of position. For the KF method, it will have almost same MDB values between the measurement model and the predicted position states. KS method has lower MDB values (around 0.16 meters) for the measurement models than the KF method (around 0.18 meters), indicating that KS step can detect smaller outliers and is more reliable.



(a) (b)

Figure 3.36 MDB values of the measurement for GNSS/INS integration: (a) KF, (b) KS



(a) (b)

Figure 3.37 MDB values of (a) the predicted position state KF, and (b) the position dynamic model KS for GNSS/INS integration

In addition, since the KF quality control method cannot test the states which are not directly observed by the measurement model, the other states in this system: velocity, attitude, accelemeter bias, gyro drifts, etc., will not be tested by the KF method. However, the KS method can directly test the dynamic model for both directly observed states, such as vehicle inertial position, and indirectly observed state parameters, such as vehicle attitude, velocity, acceleration bias, gyro bias.

The next 6 figures are the External Reliability results of the KF/KS method, which show the influence of the undetected outliers within different types of measurement model upon the position, velocity and attitude states estimation.

For KF method, the undetected outlier within its input latitude and longitude measurement will leads to around 0.1 meters' influence upon the final latitude and longitude estimation,

which is around 5 times higher than the estimation uncertainty. But for vertical estimation, the influence is only around 0.04 meters, around 2 times of its uncertainty. For velocity estimation, the ER value is around 0.03-0.04 m/s, around 3 times of the velocity uncertainty. For attitude estimation, it is found that the heading is more sensitive to the outliers than pitch and roll. But the outlier in the position measurement will cause very low influence on the final attitude estimation with respect to the estimated Standard Deviation value.

If more types of observations are utilized by the Loosely Coupled integration, such as the velocity estimated by GNSS and inertial navigation progress, the influence of outlier within them can also be analysed with their ER value.

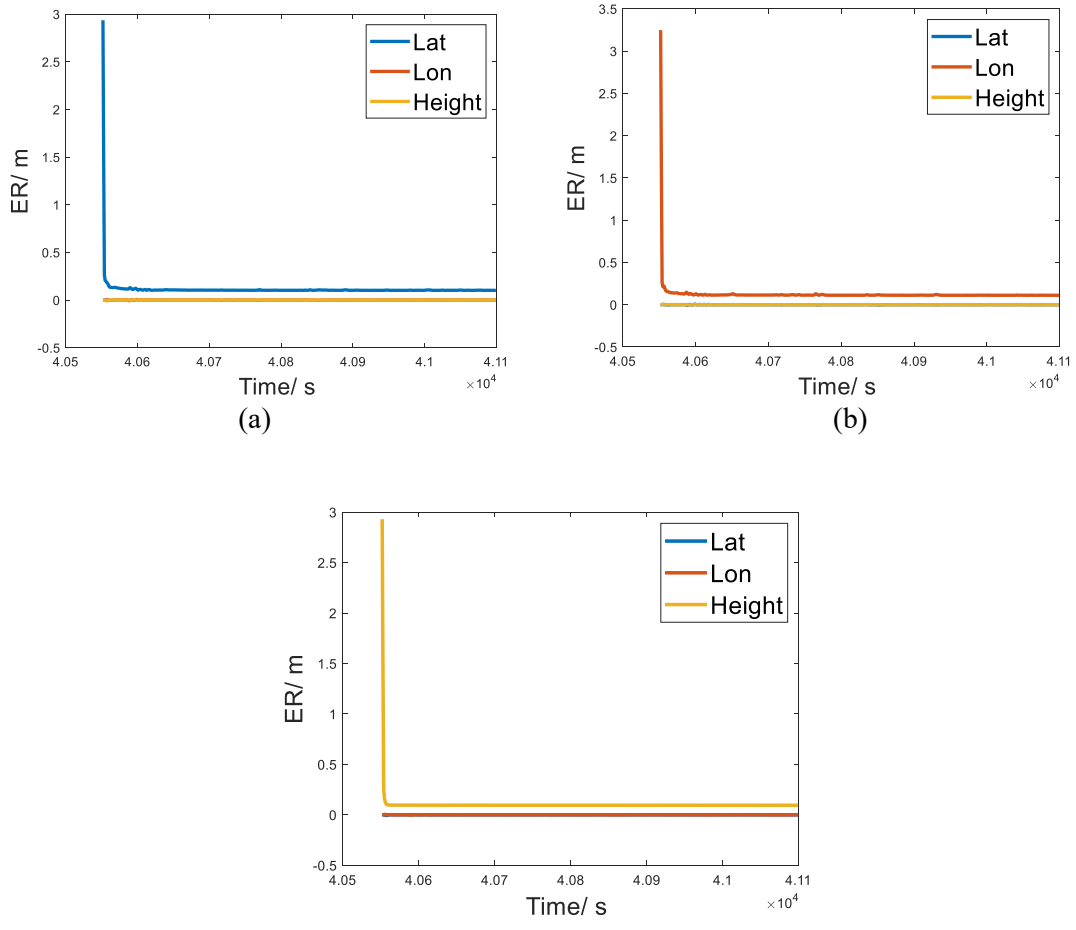
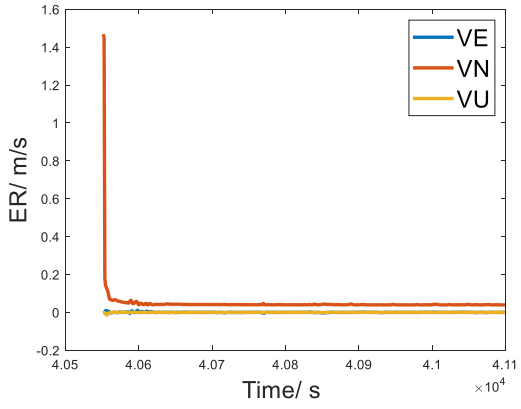
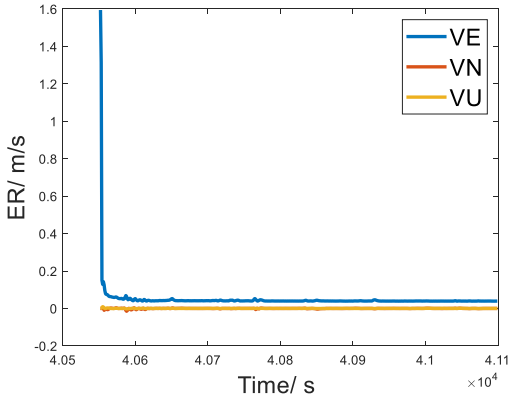


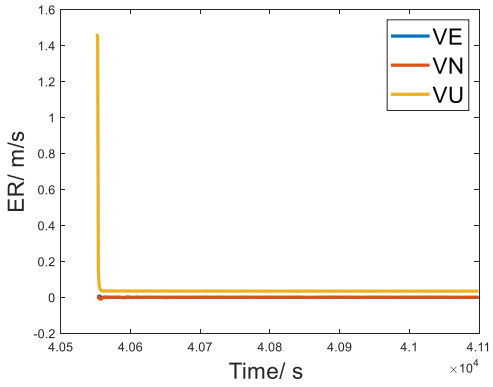
Figure 3.38 External Reliability values of the undetected outlier within KF GNSS/INS integration measurement models (a) Latitude differences, (b) longitude differences, (c) height differences toward position error states



(a)

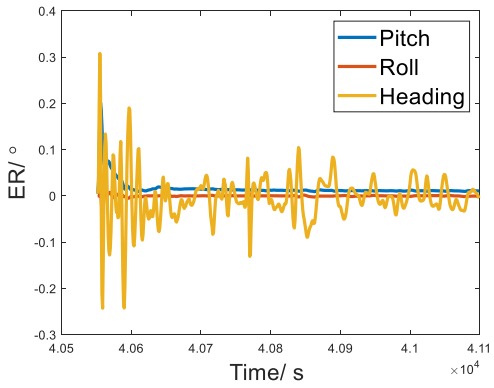


(b)

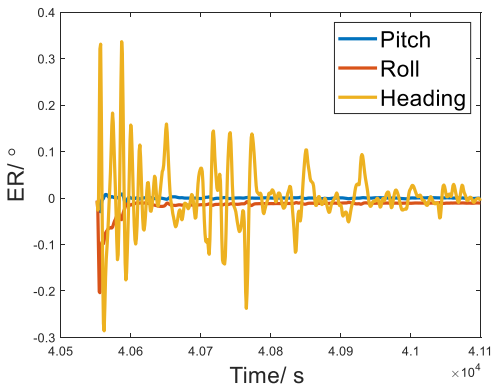


(c)

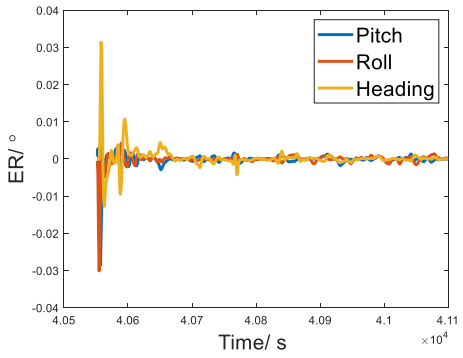
Figure 3.39 External Reliability values of the undetected outlier within KF GNSS/INS integration measurement models (a) Latitude differences, (b) longitude differences, (c) height differences toward velocity error states



(a)



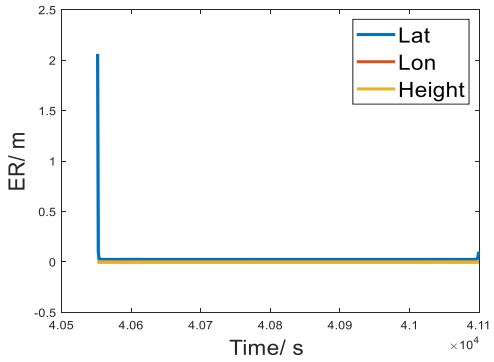
(b)



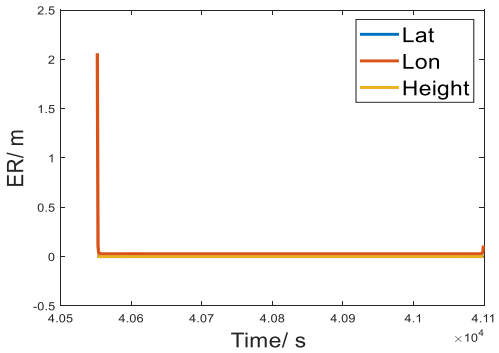
(c)

Figure 3.40 External Reliability values of the undetected outlier within KF GNSS/INS integration measurement models (a) Latitude differences, (b) longitude differences, (c) height differences toward attitude error states

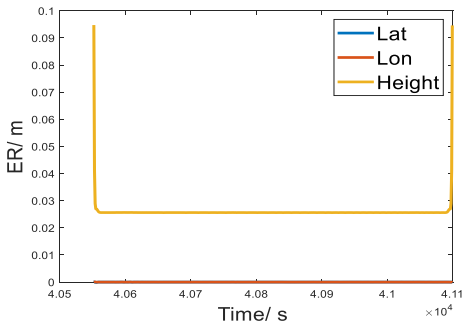
For the KS method, lower External Reliability results can be achieved than the KF method. The ER for position states is around 0.022-0.025 meters, around 2 times of the estimation uncertainty. The ER values for velocity and attitude are both lower than their corresponding STD values.



(a)

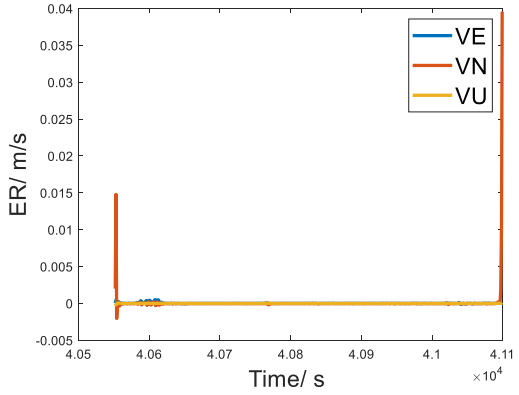


(b)

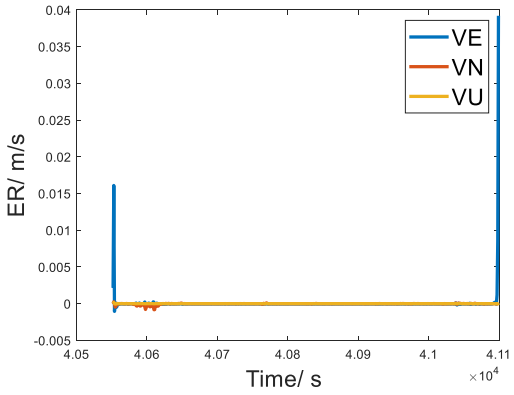


(c)

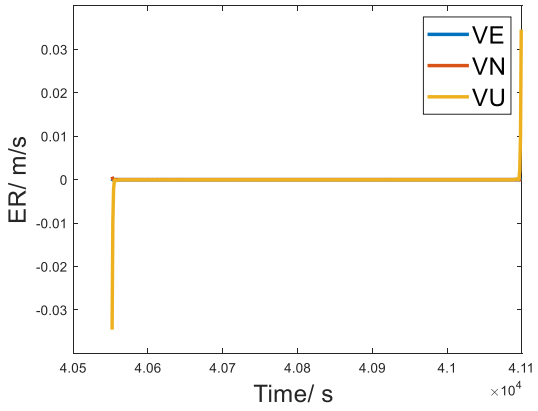
Figure 3.41 External Reliability values of the undetected outlier within KS GNSS/INS integration measurement models (a) Latitude differences, (b) longitude differences, (c) height differences toward position error states



(a)

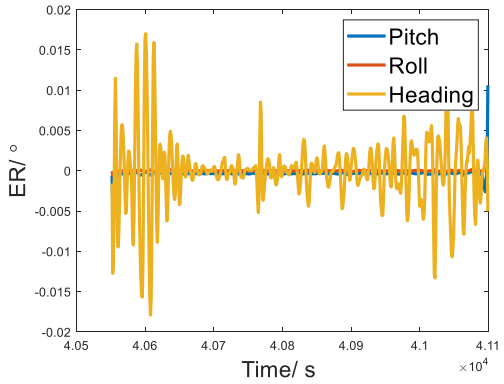


(b)

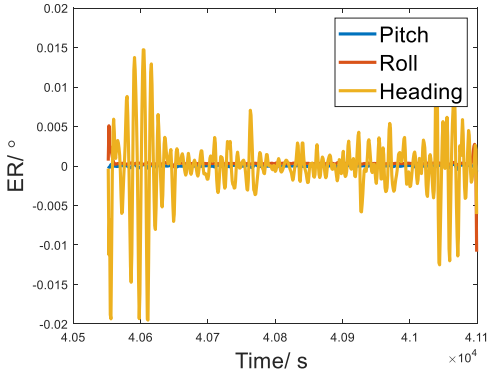


(c)

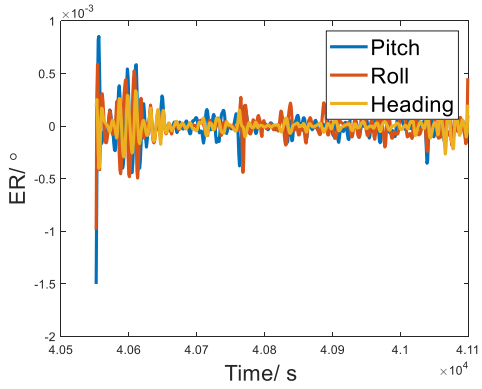
Figure 3.42 External Reliability values of the undetected outlier within KS GNSS/INS integration measurement models (a) Latitude differences, (b) longitude differences, (c) height differences toward velocity error states



(a)



(b)



(c)

Figure 3.43 External Reliability values of the undetected outlier within KS GNSS/INS integration measurement models (a) Latitude differences, (b) longitude differences, (c) height differences toward attitude error states

3.4 Summary

In this Chapter, a real-world urban driving dataset is used to test some positioning strategies, including RTK, PPK, and GNSS/INS. For the testing with RTK and PPK, centimeter to decimeter level accuracy can be obtained with the ambiguity float double differencing method. If proper integer ambiguity resolutions can be achieved, the fixed RTK and PPK accuracy will be improved to centimeter level. By integrating GNSS and inertial system, centimeter level accuracy can be achieved.

Reliability analysis has been done for the Double Differencing RTK/PPK cases. Geometry change will influence the MDB value of both Pseudo Range measurement and Carrier Phase measurement and also the dynamic models. Since in this urban driving testing, there is very frequently appearing and disappearing of satellites. Each time one satellite appears, its ambiguity is treated as a new state parameter, therefore, its value, uncertainty and reliability will converge again during RTK. The smoothing based PPK will have more stable reliability for the ambiguity state's model and is less sensitive to the geometry change.

Outlier can be detected by our quality control methods in this positioning testing. The influence of the outlier towards to the final estimation can be mitigated by directly correcting the input measurement with the estimated outlier value, or by adapting the final estimation results with the estimated outlier influence value. With these methods, the

accuracy of the estimation can be improved. However, for case like GNSS/INS loosely coupled system, the measurement model and the position states are highly correlated, which make the KF system has the potential of misidentification and mis-correction, which can be avoided by smoothing system since it deals with both measurement model and dynamic processing model, not the predicted state parameters.

Chapter 4 Quality Control for Online SLAM

4.1 Introduction

Simultaneous localization and mapping (SLAM) is an effective and well-known technique in the field of autonomous robotics. Localisation is a process of estimating an object's pose (position and attitude) related to a reference frame while mapping refers to the gathering of interested features, such as the landmarks and obstacles around the object. As a chicken-and-egg problem, SLAM is comprised of processes that a moving platform travels to an unknown environment, builds a map of the environment and uses the map to deduce its location in it simultaneously (Durrant-Whyte and Bailey, 2006; Cadena, et al., 2016). SLAM technique is of great importance for the navigation purpose. According to the vehicle/robot pose information and its location information within the environment, a planning step for the following path and a controlling step of the next action can be done. For instance, the offline SLAM can be used to generate high precision and high definition map of the road environment with the multiple sensor information acquired on a vehicle. This map will be a powerful tool for achieving navigation of autonomous driving by supporting online localization. SLAM can also be used to detect obstacles to avoid accident.

Over the last decades, numerous works have been done to solve the SLAM problem. Mapping the environment requires the use of perceptive sensors which can provide the relative location information between the platform and the environments, while localization, also sensory based, is to provide the position and orientation of the platform within the built map. Increasing types of sensors (camera, laser, radar, IMU, car sensors etc.) and platforms (robot, UAV, automated car) are utilized to provide data with different characteristics and to create new types of maps (Bresson, et al., 2017). The application of SLAM has expanded to both indoor and outdoor system, both online and offline.

The various SLAM estimation algorithms can be divided into two mainstreams: Filter-based and Optimization based approaches. Extended Kalman Filter SLAM (EKF SLAM) is the first solution for the SLAM problem and has been widely used and studied. It can achieve online estimation with focusing on the most recent robot pose based on the last sensor information (Bresson, et al., 2017). Therefore, in this Chapter which focusing on online estimation of SLAM, EKF SLAM is studied.

Despite the huge developments made by the SLAM research communities, there are still some open problems that adversely affect the behaviour of the SLAM approaches. The presence of outliers is one of the unsolved problems that limit robust operation of SLAM approaches, especially in practical applications due to the unavoidable presence of outliers (Bresson, et al., 2017). The occurring of outliers is not inhibitable as SLAM system is sensor based. It may exist in the sensor observations (or called measurements) as a sudden jump, or due to the improper system model or falsely modelled process. Since the SLAM system usually uses the odometry measurements (such as those collected by odometers and inertial sensors) to describe the movement between the robot poses and uses the perception measurements (such as those collected by Laser, camera, radar) to represent the relationship between the robot and the environment, the erroneous or faulty measurements will severely corrupt the quality of the estimation. For instance, if there are outliers in the odometry measurements, the predicted position in filter SLAM may drift from the true trajectory. The outliers in the observations (perception measurements) will cause wrong correction of the predicted or initial guessed trajectory and landmark position. These faults in measurements may also result in wrong data association or loop closure, which in turn will cause introducing of spurious measurements, resulting in wrong final estimations.

Some studies have been devoted to improving the robustness of their SLAM systems to the outliers, by rejecting the outliers or mitigating the effect of wrong data association (Wangsiripitak and Murray, 2009; Xie, et al., 2017). RANSAC is commonly used in vision SLAM to reject the outliers (Kitt, et al., 2010). Although these studies focus on rejecting outlier and making robust loop closure detection at the front end in the graph SLAM, or before measurement update in the filtered SLAM, there are still chance that

spurious measurements are fed to the final estimation step. Even with all the correct loop closures and data association introduced, the outlier within the measurements or with the system models may still causes wrong final estimation, which drive the SLAM system to failure. Some studies introduce their robust SLAM algorithms that can recognize and reject outliers during the optimization (Sünderhauf and Protzel, 2012; Latif et al., 2013). However, they focus on the outliers caused by wrong data association or loop closure, the problem of sensor measurement outliers is still not solved. Fault Detection and Isolation (FDI) system has been introduced to the localization and navigation to deal with different kind of faults, such as from sensors, models, etc (Dedeoglu, et al., 2000). However, isolating or rejection measurement with outlier will reduce the system redundancy and affect the reliability of estimation. Therefore, our study here has proposed an outlier detection and identification method that can directly deal with the measurement outliers at the measurement update step for the Extended Kalman Filter (EKF) SLAM. The influence of the outlier can be corrected with the estimated outlier value rather than rejecting the faulty measurement.

Although most of the studies are devoted to achieving efficient and robust estimation of SLAM, the reliability of their estimation and system is seldom studied. Reliability analysis is essential for the SLAM designer as it can be conducted at the early stage without real measurement. Only with the given system structure is enough to acquire the knowledge of the impact of the possible errors. Therefore, an appropriate SLAM system can be designed based on the reliability analysis to fulfil its requirement and application, such as design the redundancy of sensor, the size of loop, the geometry of landmark, the precision of sensors.

Therefore, in this study, a quality control method, including outlier detection and identification, system reliability analysis, is introduced to the final estimation step of online SLAM. The structure of this Chapter is as follows: Section 4.2 presents the mathematical models for online SLAM methods. The proposed online SLAM quality control system is tested with two real-world SLAM datasets in Section 4.3. Section 4.4 summarizes the performance of the EKF online SLAM and its quality control system.

4.2 Mathematical Models for Online SLAM Methods

The Extended Kalman Filter (EKF) SLAM algorithm is a standard SLAM algorithm and has been widely studied and used for mapping and localization. The time update (prediction) and measurement update (filtering) steps used in this research follow the conventional EKF algorithm.

The state vector at time *k* is denoted as

$$X_{(k)} = (X_{r(k)}; X_{f(k)}) = (X_{r(k)}; X_{f_1(k)}; \ldots; X_{f_q(k)}) \tag{4.1}$$

where $X_{r(k)}$are the robot or vehicle states including robot position, and sometimes also robot velocity, orientation or attitude depends on the used sensor and on the applications, at the current time *k*, $X_{f(k)}$ are the gathering of landmark states with $X_{f(k)} = (X_{f_1(k)}; \ldots; X_{f_q(k)})$ the positions of the total q landmarks at the current map. The dimension and constitutes of $X_{r(k)}$ and $X_{f(k)}$ are dependent on the specific measurement types and map type in the SLAM problem, such as for the 2D laser SLAM, $X_{r(k)} = (x_{r(k)}; y_{r(k)}; \phi_{r(k)})$contains robot horizontal position and orientation (or heading), while $X_{f_q(k)} = (x_{f_q(k)}; y_{f_q(k)})$is the horizontal position of the q^{th} landmark.

The prediction step of the robot states from time *k* to *k+1* is described by the odometry model:

$$X_{r(k+1)} = f\left(X_{r(k)}, u_{(k)}\right) + \tau_{(k)} \tag{4.2}$$

here $f(\ )$ is the non-linear state transition function from time *k* to *k+1* with Jacobian matrix $\Phi_{r(k+1)}$ and $g_{(k+1)}$ w.r.t the robot state vector $x_{r(k)}$ and the control noise, respectively, $u_{(k)}$is the control input at time *k*, $\tau_{(k)}$is the control noise errors in the process model with covariance matrix $P_{\tau_{(k)}}$.

The Landmarks are assumed to be static, hence their process model is:

$$X_{f_i(k+1)} = X_{f_i(k)}, \quad i = 1,\ldots,q. \tag{4.3}$$

Therefore, after prediction, the predicted whole state vector will be:

$$\overline{X}_{(k+1)} = \begin{vmatrix} \overline{X}_{r(k+1)} \\ \overline{X}_{f(k+1)} \end{vmatrix} = F(X_{(k)}, u_{(k)}, 0) = \begin{vmatrix} f(X_{r(k)}, u_{(k)}) \\ X_{f(k)} \end{vmatrix} \tag{4.4}$$

here the hat '¯'over X represents the predicted value of state, 'ˆ' over X represents the estimated (filtered) value of state. The total Jacobians for $F(\quad)$ in Equation 4.4 will be:

$$\Phi_{(k+1)} = \begin{vmatrix} \Phi_{r(k+1)} & 0 \\ 0 & E \end{vmatrix}, \qquad G_{(k+1)} = \begin{vmatrix} g_{(k+1)} \\ 0 \end{vmatrix} \tag{4.5}$$

here E is the identity matrix. The covariance matrix of state will be given by Equation 4.6 after prediction:

$$P_{\overline{X}_{(k+1)}} = \Phi_{(k+1)} P_{X_{(k)}} \Phi_{(k+1)}{}^T + Q_{(k)}, \qquad \text{here } Q_{(k)} = G_{(k+1)} P_{\tau(k)} G_{(k+1)}{}^T \tag{4.6}$$

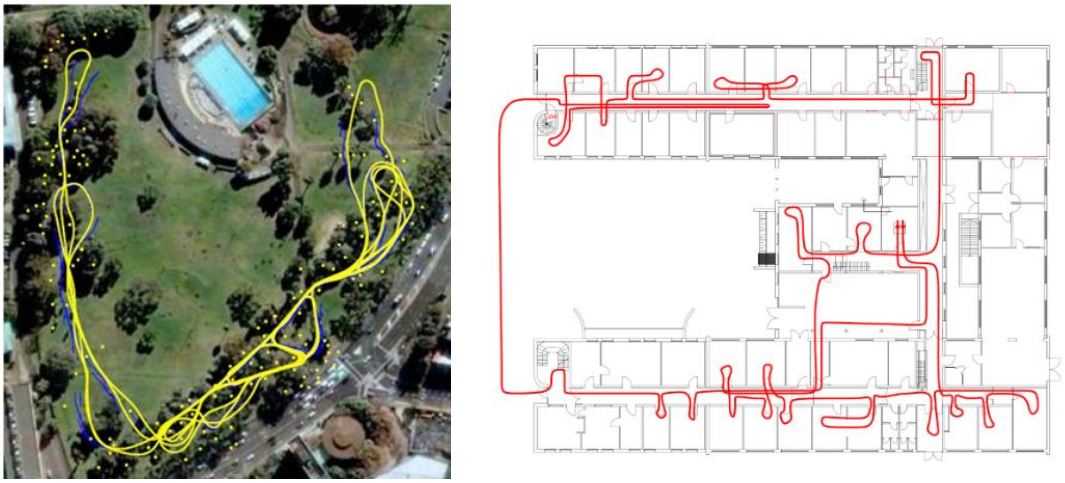
Then solve the SLAM with standard EKF measurement updating.

The EKF based quality control method demonstrated in Chapter 2 is used in this Chapter to monitor the quality of the online SLAM.

4.3 Experiments and Analysis

4.3.1 Two publicly available SLAM Datasets

Two real-world SLAM datasets, the Victoria Park dataset (Guivant et al., 2002), and the DLR dataset (Kurlbaum and Frese, 2009) are used to test the proposed quality control methods in different SLAM estimation frameworks. The overview of the two datasets can be found in Figure 4.1.



(a) (b)

Figure 4.1Overview of the real-world SLAM dataset, (a) the Victoria Park dataset map and trajectory (Kaess, et al., 2008) and (b) the DLR dataset trajectory (Kurlbaum and Frese, 2009)

The Victoria Park dataset contains an approximately 4 kilometers long trajectory within an outdoor park in Sydney, Australia. The detected landmarks are tree features with laser observation of ranging and bearing. The DLR dataset was recorded with a camera at the DLR building in Germany, which covers a region of 60m x 45m, and the robot path consists of three large loops within the building (plus a small outside path) with a total length of 505 meters. On the way the robot visits 29 rooms with artificial circle landmarks on the ground along the trajectory (Kurlbaum and Frese, 2009).

4.3.2 Kalman Filter Estimation Result

Extended Kalman Filter based SLAM can be conducted with these two real world datasets efficiently. Figure 4.2 shows the path and feature estimation results by Extended Kalman Filter based SLAM (EKF SLAM) for both two datasets, and the corresponding estimation uncertainty. Overall, the uncertainty results of the vehicle/robot will increase when keep moving into an unknown environment with new landmarks.

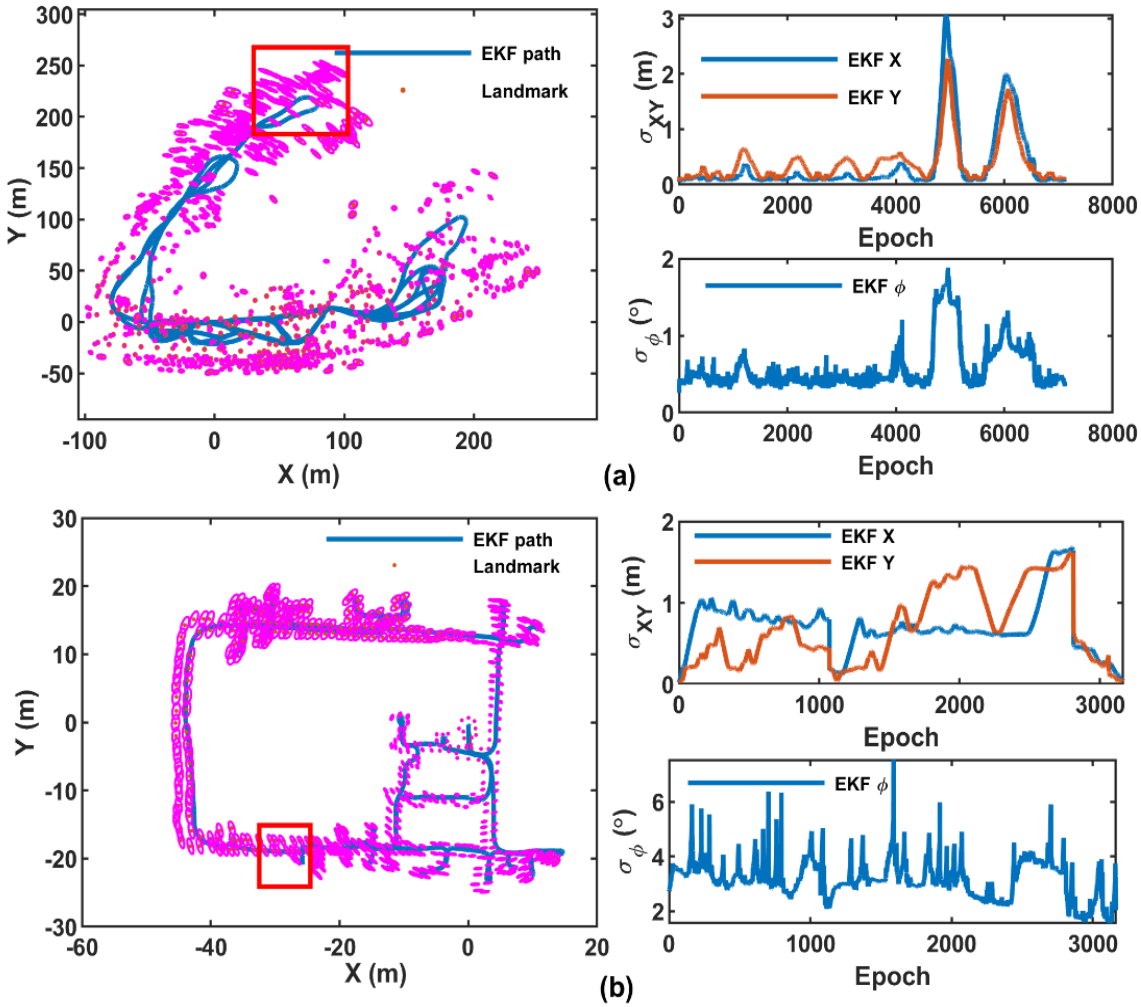


Figure 4.2 EKF estimated SLAM trajectory, feature position estimated at last epoch and their corresponding standard deviation values for two real-world datasets ((a)Victoria Park dataset, and (b)DLR dataset)

4.3.3 Kalman Filter Quality Control for Victoria Park Dataset

The outdoor Victoria Park dataset is tested with the proposed Extended Kalman Filter Quality Control (EKFQC) method. Firstly, possible outliers are identified, and their influences are corrected. Then, the Reliability of the EKF SLAM system for this dataset is analysed.

According to Figure 4.3, there are some possible outliers that can be detected during the whole trajectory. The outlier seems mostly exist within the observations (landmark detection measurements), not the predicted state model, since the W test (outlier statistic test) results of the predicted states (Figure 4.3b) are always below the threshold of 3.29.

Overall, with this EKF statistic test, there are 13 identified outliers and all of the outliers are in the bearing measurements. These bearing outliers will cause biases of the final estimation of parameters.

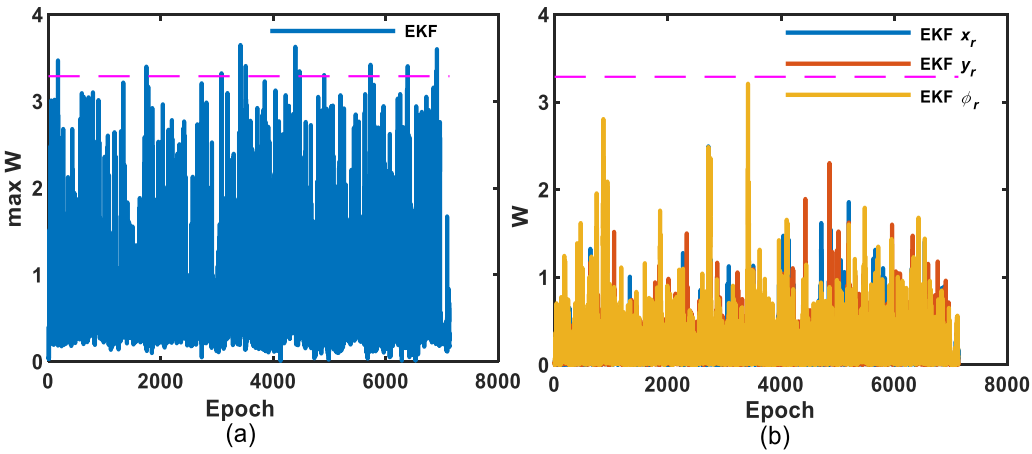


Figure 4.3 Outlier statistic test for outlier identification within (a) landmark detection model, and (b) for predicted vehicle state by EKF for the Victoria Park dataset

Figure 4.4 shows the influence of each outlier upon the vehicle pose (position and heading) estimation at the epoch that it is identified. The identified outliers within the observation of the Victoria Park dataset will cause relatively low influence upon the estimation with respect to the estimation uncertainty. These influence values upon position estimation are between -0.1 to 0.2 meters, around 0.5-57% of the position estimation uncertainty, while that upon heading are between -0.6 to 0.5 degree, around 16-92% of the heading estimation uncertainty.

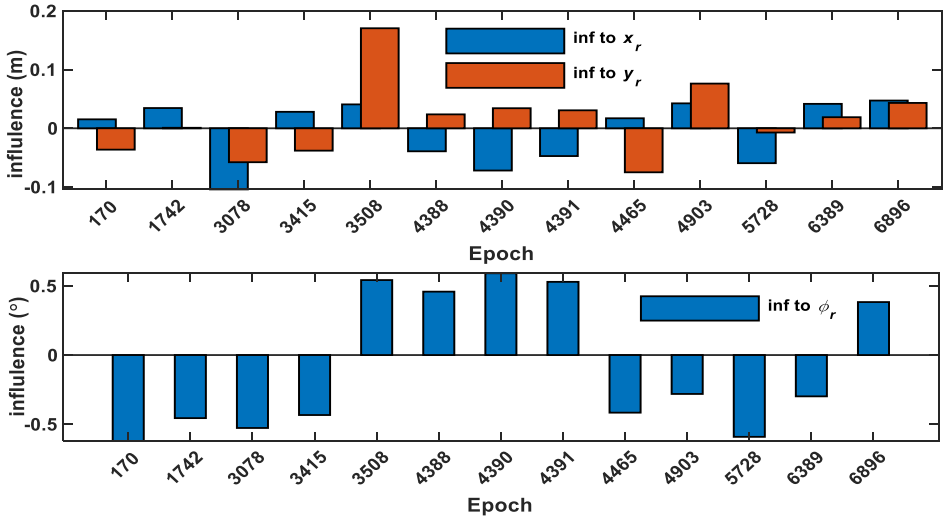


Figure 4.4 Influences by the EKF identified outliers upon the final vehicle pose (upper: position x_r and y_r; down: heading ϕ_r) estimation for the Victoria Park dataset

One detailed example is the first detected outlier, which is within the bearing measurement at epoch 170 with a W statistic test result of 3.47. This outlier can be directly located at the bearing measurement of the landmark lm34. The possible error size can also be calculated with the Extended Kalman Filter Quality Control (EKFQC) method, which is about 4 degrees. As according to Figure 4.4, this outlier will cause about 1.5 cm and -3.6 cm drifts onto the vehicle horizontal position estimations, and -0.62 degree onto the orientation estimation.

After the identification step, the influence of this outlier can be corrected and do filtering again. No outlier will be detected and identified at that epoch any more after correcting, the new W statistic test result (Figure 4.5) for this measurement of lm34 will be closed to zero. For other measurements at the same epoch, their W statistic test values are also reduced since the measurements are correlated to each other.

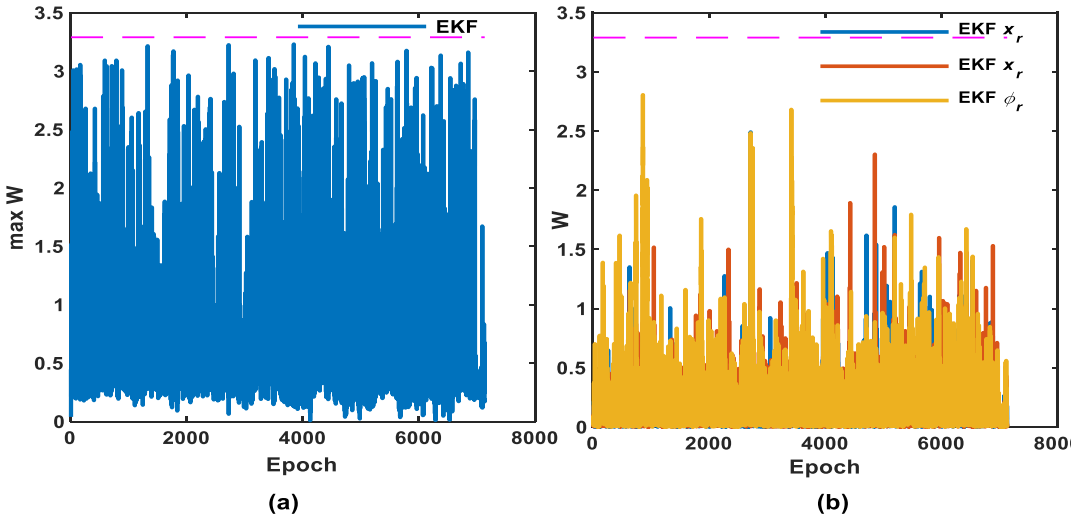


Figure 4.5 Outlier statistic test after correction of the influence of identified outliers for (a) landmark detection model, and (b) for predicted vehicle state by EKF for the Victoria Park dataset

Outliers at other epochs can also be detected, and identified with this EKFQC method, the process can be done automatedly during the extended Kalman filter online estimation, and the influence of the outlier can be corrected and mitigated in time, thus will not affect the following epochs of state prediction and estimation. Figure 4.5 shows that with the automatic correction step, no outlier will be detected and identified anymore after correction.

Figure 4.6 shows the difference of vehicle pose estimation with and without the outlier correction step. It should be noticed that although at this test, if without the proposed correction step, an identified outliers will cause relatively low influence upon the filtered results at the identified epoch (as shown in Figure 4.4), its influence will be accumulated and affect the following prediction and filtering epochs, especially for large loop case that trajectory drift cannot be corrected with the loop closure. It can be found, increasing position difference (up to 0.4 meter for each horizontal direction) between with/without outlier correction will be achieved between epoch 5728 and epoch 6050 (Figure 4.6), during this time period the vehicle is travelling into an unknown environment (see red box in Figure 4.2a), and then the difference reduced after the vehicle travelling back. Since the estimation uncertainty is also increased during the travelling in the unknown place, this accumulated influence of outlier is still lower than the uncertainty. Overall, our proposed EKFQC method will improve the estimation accuracy, especially for the large loop case.

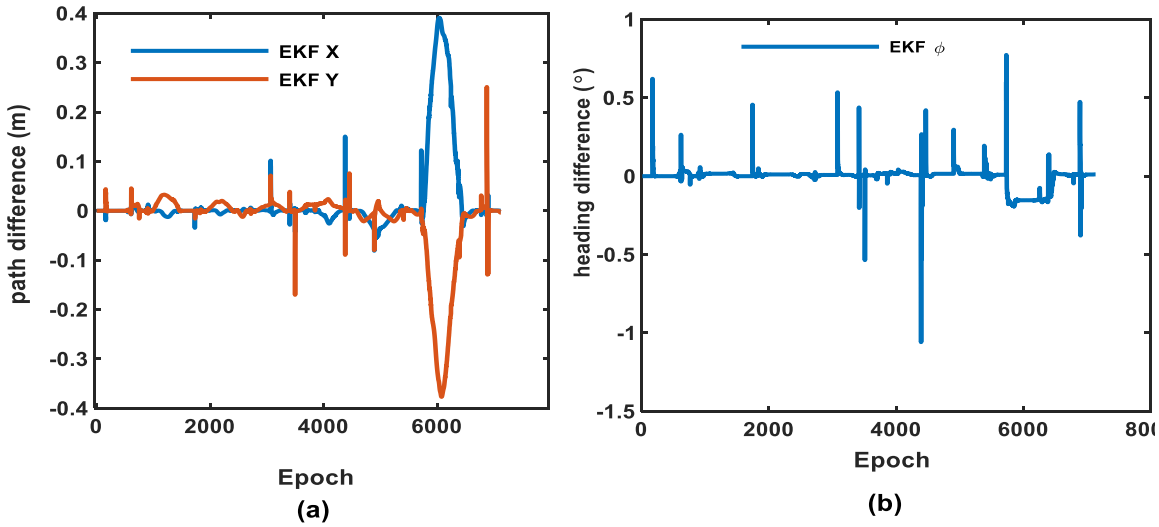


Figure 4.6 Differences in the estimated vehicle pose ((a) position x_r and y_r; (b) heading ϕ_r) with and without correcting the influence caused by EKF identified outliers for the Victoria Park dataset

The reliability of the EKF SLAM system for the Victoria Park dataset is also tested. Figure 4.7 and Figure 4.8 represent the change of system internal reliability (MDB) along the whole vehicle journey. In overall the MDB results are stable with some peaks. For the landmark detection measurement (Figure 4.7), the minimum bound that an outlier can be detected is fluctuating between 2-5 meters for ranging and 4-6 degrees for bearing. For

the predicted state model (Figure 4.8), the MDB value for vehicle position and heading states are around 0.5-1 meters and 2.5-4 degrees, respectively. Relatively higher MDB value of predicted states model will happen when the vehicle is travelling into unknown environments.

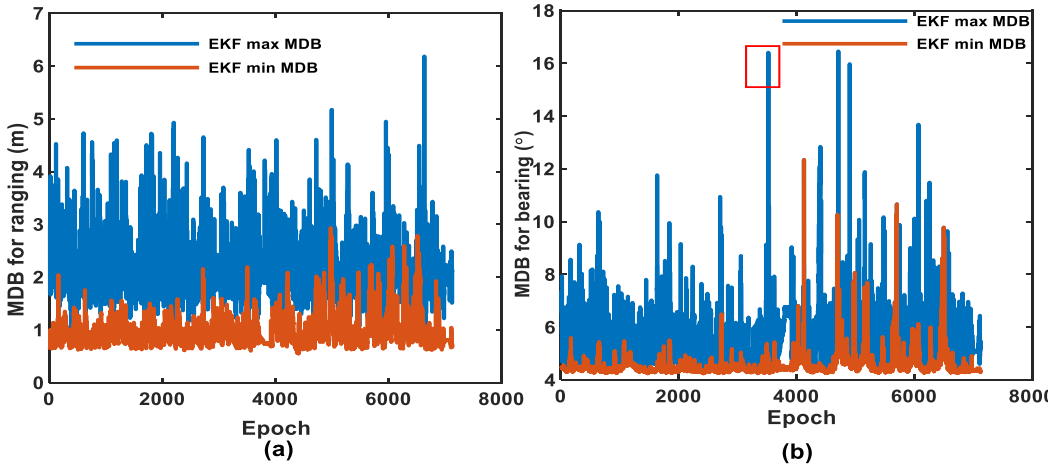


Figure 4.7 Minimum detectable bias (MDB) results for the landmark detection models of (a) ranging and (b) bearing measurement by method EKF for the Victoria Park dataset

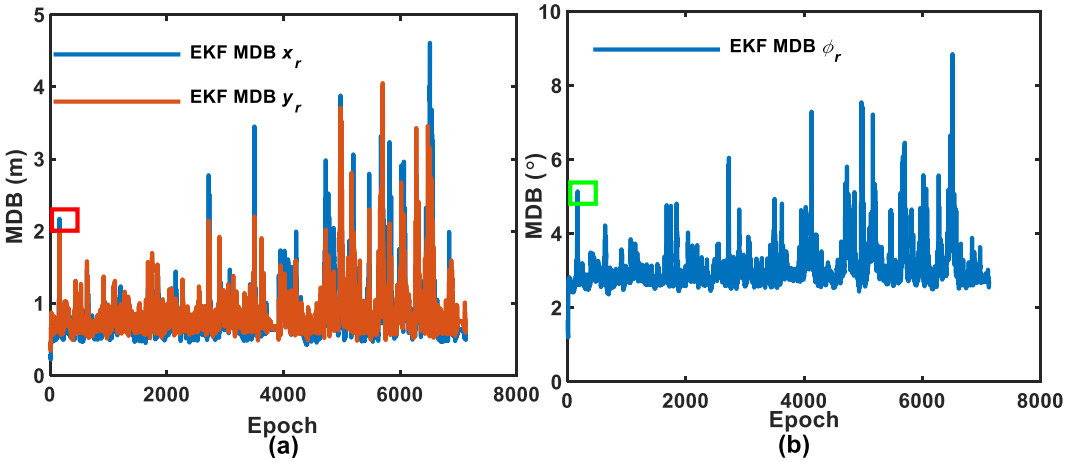


Figure 4.8 Minimum detectable bias (MDB) results for the vehicle predicted states of vehicle states (a) x_r and y_r and (b) heading ϕ_r by method EKF for the Victoria Park dataset

There are also some jumps of the MDB value, such as in Figure 4.7b, there is a very high MDB value up to 16.4 degrees at epoch 3522 (within red box in Figure 4.7b), which is around 16 times higher than the measurement noise at this epoch. The measurement that has huge MDB value is the bearing measurement of Landmark lm62. This means if there

is an outlier that is lower than 16.4 degrees within the bearing measurement of Landmark lm62, it will not be able to be detected with the statistic test during this extended Kalman filter SLAM process, this measurement is not reliable.

In order to find out the reason of this huge MDB value, the geometry of the measurement is checked with Figure 4.9, which shows the path of vehicle from the initial point to a specific time epoch, location of all the stored landmarks, including the previously detected, and the current detected landmarks.

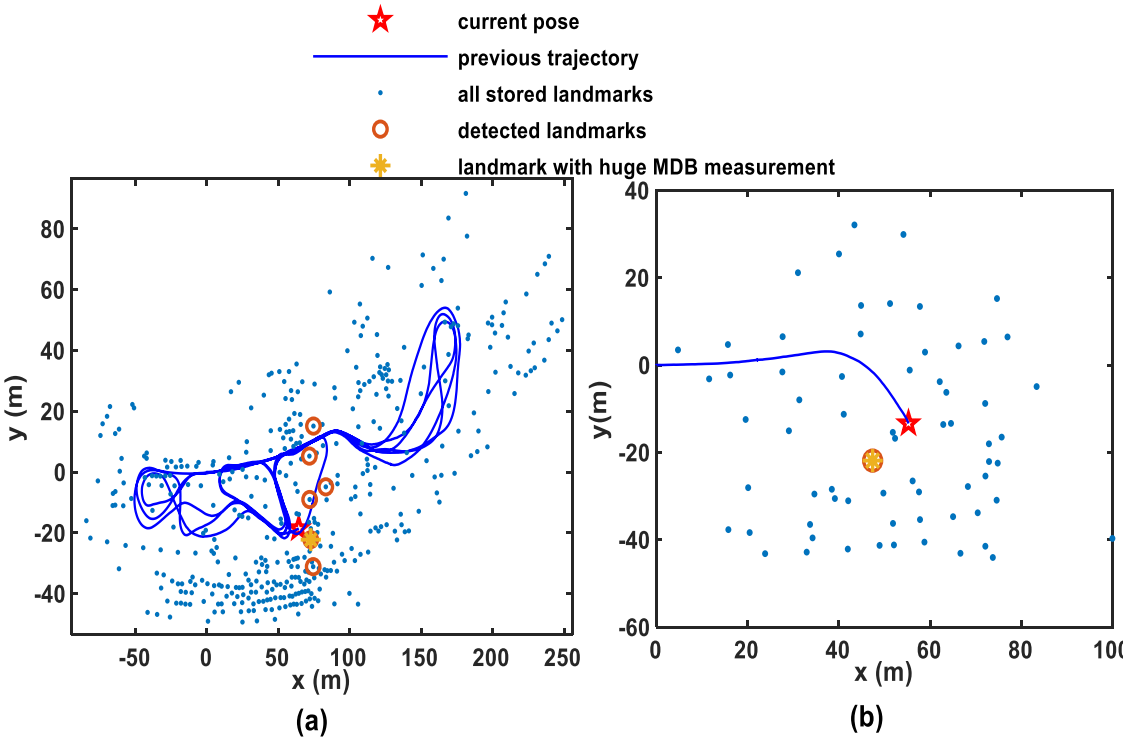


Figure 4.9 Geometry of measurements when a high MDB value or External Reliability value is detected during the EKF SLAM process at (a) epoch 3522, and at (b) epoch 169 for the Victoria Park dataset

As looking into Figure 4.9a, '*' represent the Landmark lm62, whose bearing measurement has the huge MDB value at epoch 3522. There are in total 6 landmarks being detected at this epoch. Thus, the low reliability at this epoch seems not related to the redundancy. When analysing the whole trajectory, it can be found the epoch 3522 is the only time that Landmark lm62 is recognised as an old feature after added to the state vector. Other landmarks around this landmark had been redetected many times after their first detections, their revisits happened very quickly just after being added to the state vector. Therefore, the lack of quickly redetection and correction after the initial visit and

the long blank period are the reasons why the measurement of Landmark lm62 is much less reliable.

This example indicates that if the SLAM designers want to achieve reliable estimation, their choice of revisited landmarks should be more careful. The measurements of landmarks that has been seldom revisited for a long period (such as Landmark lm62) maybe not be reliable and whether to use this measurement for estimation needs to be decided with considering the MDB value. However, the resulting influence on the final estimation can also affect the decision of whether to use this measurement. The influence caused by the outlier that has a value of the MDB is the External Reliability (ER) (Figure 4.10). For instance, although the MDB value of Landmark lm62 bearing measurement is 16.2 degrees high, its influence on the final estimation is not very large. Its ER values for this measurement are around 7 and 18 centimeters for the vehicle horizontal position estimations and 0.34 degree for the orientation estimation, which are no more than 2 times of their corresponding uncertainty. These ER values caused by the undetected bias within the observations could be acceptable, depending on the specific requirements and the applications of the SLAM approach.

There are also some other reasons that may cause low reliability of the measurement and system. Poor geometry strength and low redundancy may both result in nonreliable estimation. One example is at epoch 169 during which only 1 landmark is detected (Figure 4.9b), the MDB value of the predicted state at this epoch is up to 5.2 degrees (7 times higher than the noise, see green box in Fig.8b), which will cause 4.1 degrees bias of the final estimation of vehicle orientation (5 times higher than the uncertainty). Another example is at epoch159 (see red box in Figure 4.8a), during which not only a low number of landmarks are detected, the geometry is also very poor as one of the detected landmarks is far away from the vehicle and from other named features. The possible undetected bias within the vehicle position states may cause 2.0 meters drift of final estimation (see red box in Figure 4.10d), about 12 times higher than the position uncertainty. Furthermore, the continuing travelling into an unknown environment will also cause the EKF SLAM system unreliable, such as the increase MDB and ER values after epoch 4000 that the vehicle is travelling into a new part of the Victoria Park.

The estimation of landmark position is also influenced by the outliers within the observation and state models (Figure 4.10 c and f). It can be found, for landmark position estimation, low reliability will happen when it is an initial detection or revisit. When a landmark is invisible, its ER value caused by undetected outlier within current measurements and models will be close to zero, indicating no influence from the current measurement outliers to its estimation.

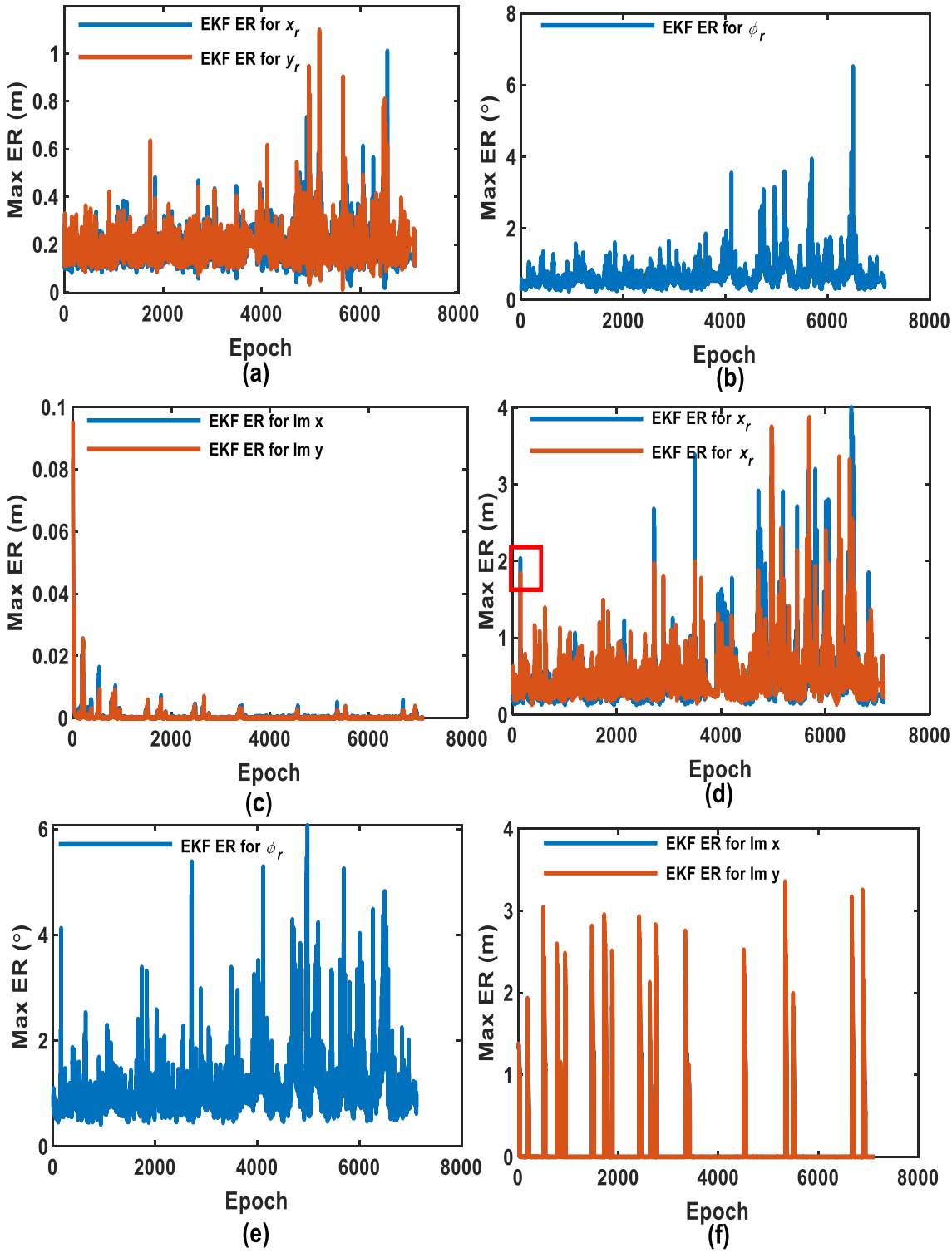


Figure 4.10 The External Reliability results: influence of outlier with MDB value within landmark detection measurement upon the final vehicle (a) position, (b) heading and (c) landmark lm1 position estimation; Influence of outlier with MDB value within the predicted state model upon the final vehicle (d) position, (e) heading and (f) landmark lm1 position estimation of the Victoria Park dataset by EKF SLAM.

4.3.4 Kalman Filter Quality Control for DLR Dataset

The indoor DLR dataset's quality is also analysed under the EKF SLAM framework.

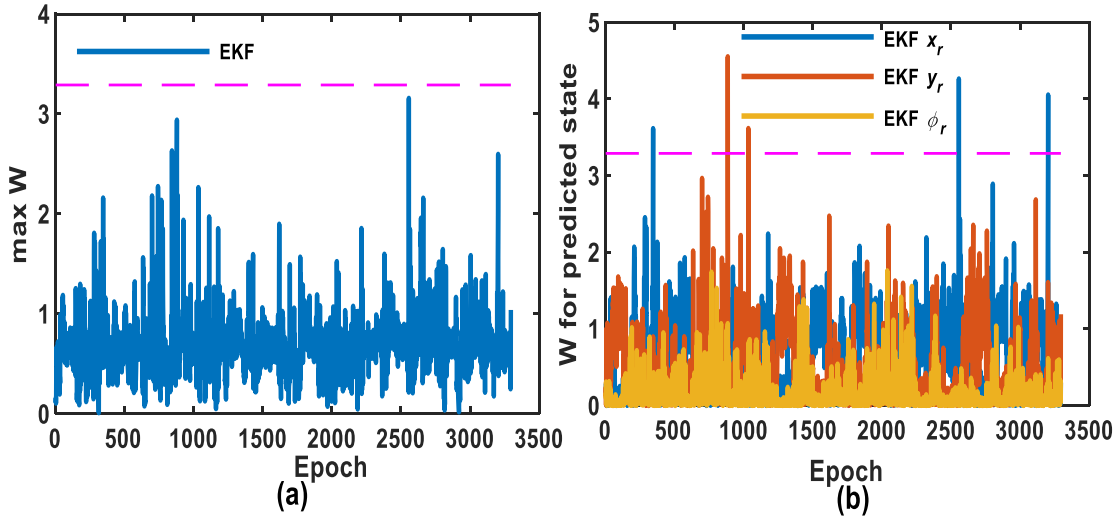


Figure 4.11 Outlier statistic test for outlier identification test for (a) the landmark detection model, and for (b) predicted vehicle state by EKF for the DLR dataset

The outliers within the measurements and predicted state models of the DLR dataset are also tested with the extended Kalman filter quality control method. It is clearly that according to Figure 4.11, no outlier is detected from the landmark detection models and there are 5 detected outliers within the predicted states along the whole trajectory period, among them, 3 outliers are within the predicted robot state x_r, and another 2 are within the predicted robot state y_r. These outliers within the predicted states will cause bias of the estimation, which can be mitigated with the proposed quality control method.

Figure 4.12 shows the influence of the identified outliers upon the estimation at the identified epochs. These influences are around 0.3-120% of the corresponding estimation uncertainty. It is also found from Figure 4.12 that a possible outlier within one type of predicted state models will have relatively higher influence upon the estimation of the same type of state parameters than upon other types of parameters. This is due to the low correlation between different state parameters.

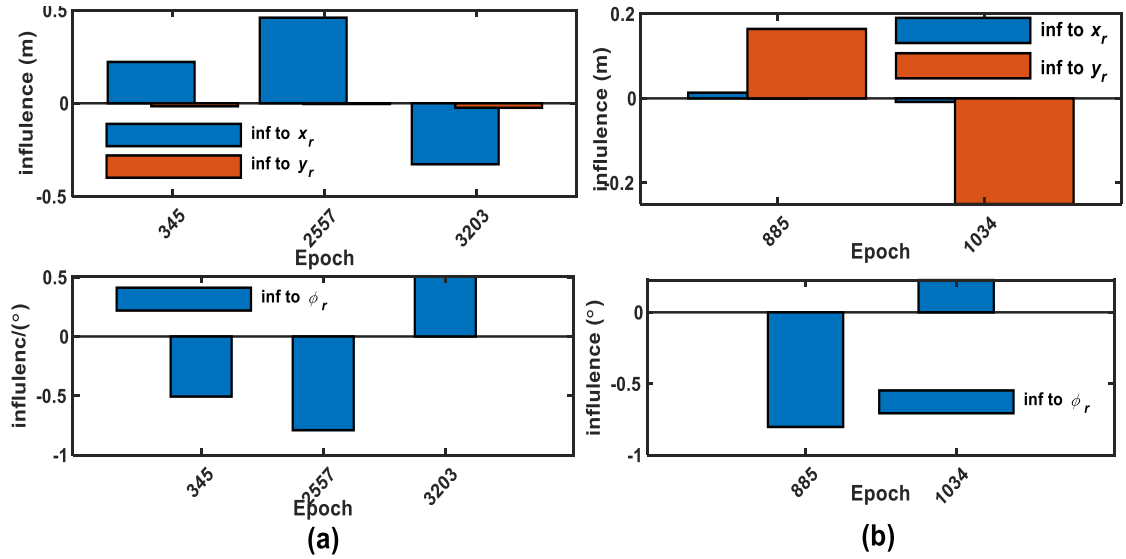


Figure 4.12 Influences by the identified outliers within the predicted robot state to the final estimation by EKF for the DLR dataset. (Influences caused by the outliers (a) within the predicted x_r, and (b) within the predicted y_r)

After corrections, no outlier is identified anymore during the EKF SLAM process (Figure 4.13). A change of estimation can be achieved after conducting the correction of the influence from the identified outliers (Figure 4.14).

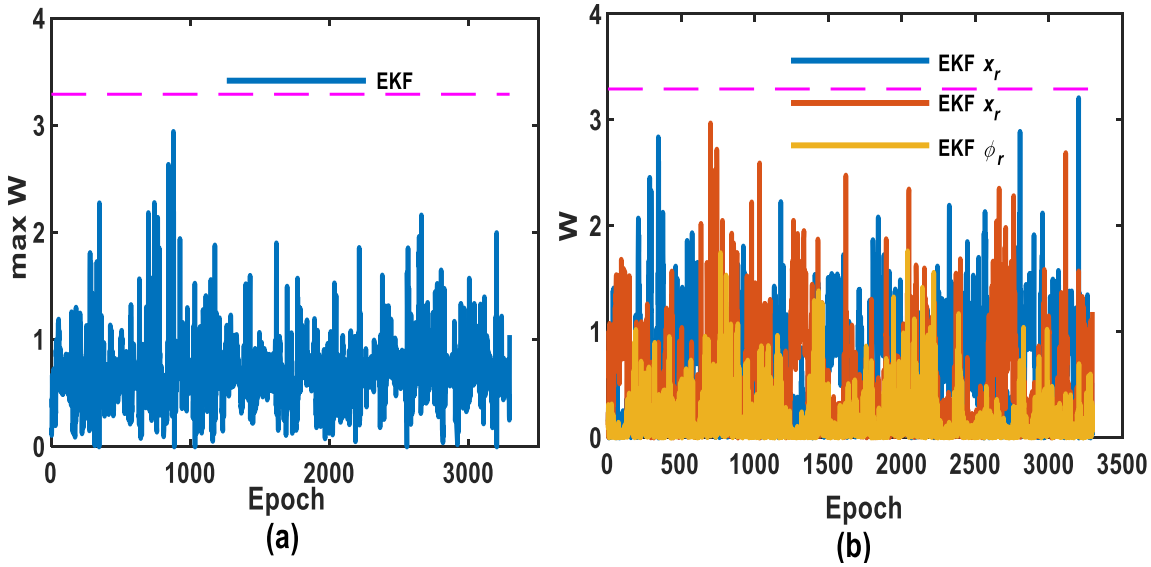


Figure 4.13 Outlier statistic test after correction of the influence of identified outliers for (a) landmark detection model, and (b) for predicted vehicle state by EKF for the DLR dataset

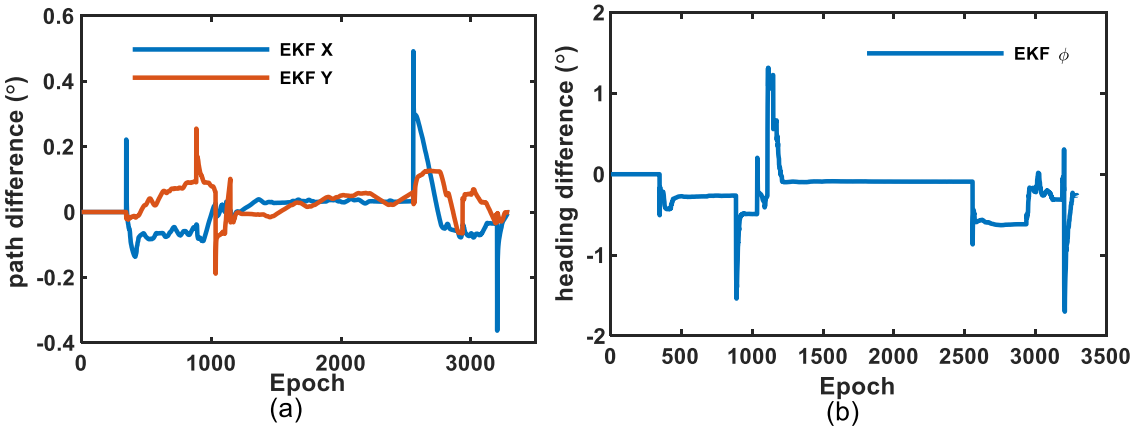


Figure 4.14 Differences in the estimated vehicle pose ((a) position x_r and y_r; (b) heading ϕ_r) with and without correcting the influence caused by identified outliers by EKF for the DLR dataset

However, referring to the estimated trajectory shown in Figure 4.2b, drifted trajectory and an unnormal displacement of vehicle trajectory (see red box in Figure 4.2b) can be found, and this divergence of trajectory is not solved with the proposed EKFQC method. This is because this divergence is mainly caused by accumulated large odometry error, especially being accumulated during the period the robot travels outside of the building with no loop closing and no overlapping landmark. But according to our statistic test results, only 5 outliers are identified, and the drift of trajectory still exists after correcting the influence of the identified outliers. For the EKF quality control method, the statistic test is done for the observation and the predicted state model at one specific epoch. The dynamic process from previous to current epoch is not directly tested. Therefore, if an outlier is within a robot dynamic (odometry) model, it may cause a fault of the prediction result, this faulted predicted result is then being tested at the measurement update step. If the fault caused by the outlier in odometry information is lower than the identify ability, it will not be identified. Such a problem may be addressed in Kalman Smoother SLAM.

The MDB results are shown in Figure 4.15 and Figure 4.16. Overall, the MDB values of landmark observations (Figure 4.15) are around 0.4-0.8 meters, which are around 5-8 times higher than the corresponding observation uncertainty.

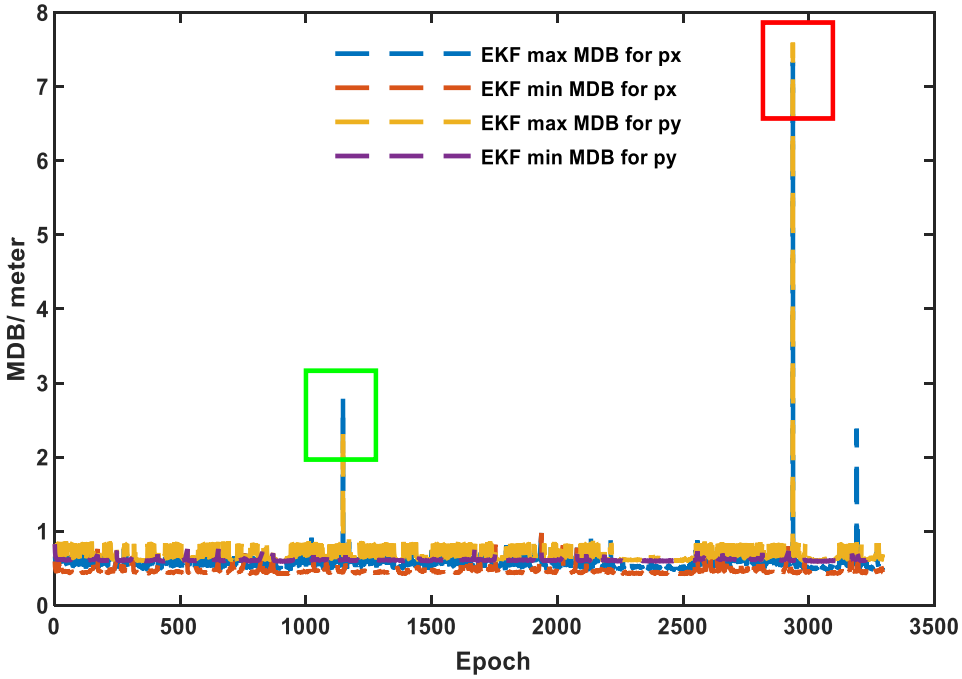


Figure 4.15 Minimum detectable bias (MDB) results for the landmark detection models for the DLR dataset by EKF

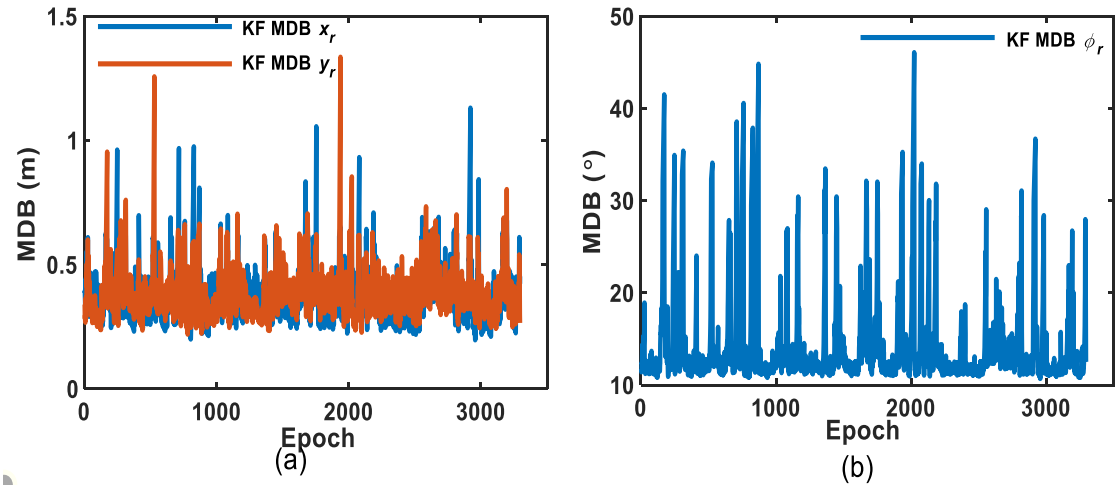


Figure 4.16 Minimum detectable bias (MDB) results for the predicted states of vehicle states (a) position x_r, y_r and (b) heading ϕ_r for the DLR dataset by EKF

There are some epochs which have sudden increase of observation's MDB and ER values, which indicating low reliability at these epochs. The reason of this low reliability may be due to the low geometry strength, such as low feature detected number, and poor distribution of landmark. According to the discussion of the Victoria Park dataset in

Section 4.3.3, high MDB value within the SLAM system may also due to the revisit of an old position after a long journey, or due to the detection of a long missing landmark. These reasons can also be demonstrated with this DLR dataset EKF results. One example of the sudden increased MDB happened at epoch 1148 (green box in Figure 4.15), when the MDB of landmark detection measurements are up to 2.8 meters with External Reliability value of 2.13 and 1.62 meters for robot x and y (green box in Figure 4.18a). Fig.17a shows the geometry at epoch 1148. At this epoch, landmark lm10 was revisited for the first time after epoch 16. Another similar high MDB and ER values occurred at epoch 2937 for landmark lm170 (geometry of this epoch is shown in Figure 4.17b). Similarly, this landmark was added to the map as a new landmark at epoch 806 and after this epoch, it has not been detected until epoch 2937 at when 7.34 meters MDB happens (red box in Figure 4.15) and will cause up to 5.75 meters bias (red box in Figure 4.18a) of the final vehicle position estimation. Hence, the tests of the two SLAM datasets both prove that revisiting an old environment after a long journey and redetecting a landmark that is invisible for a long time may result in low reliability of the system.

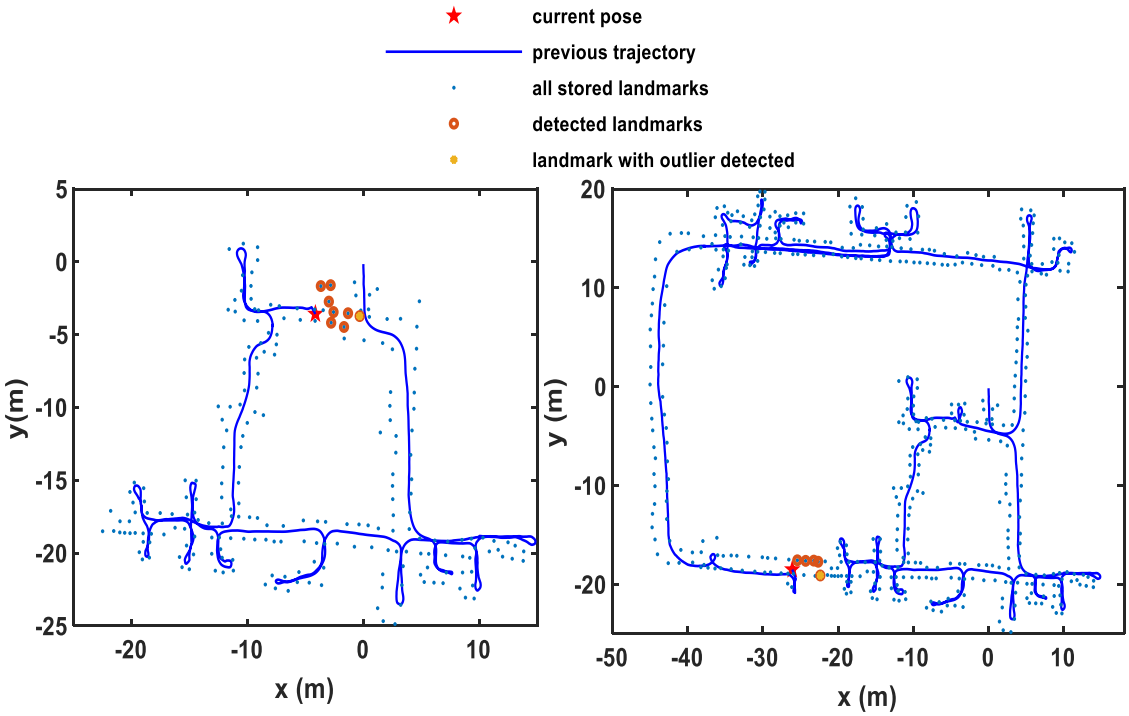


Figure 4.17 Geometry of measurements when a high MDB value or External Reliability value is detected during the EKF SLAM process at epoch (a)1148, and at epoch (b) 2937 for the DLR dataset

For the predicted states, the outliers within the orientation states are harder to be detected than the robot position states (Figure 4.16). The low bond of outlier value that can be successfully detected is about 12 to 50 times higher than the uncertainty. Since no angle related observation of the robot orientation is provided by the DLR dataset, the orientation states are indirectly observed and are corrected indirectly by the position related observation. Thus, they are less reliable than other states.

The external reliability results of the influence caused by outlier with MDB value within different kinds of measurements upon robot pose and landmark position estimation for the DLR dataset are shown in Figure 4.18. Overall, the landmark observation outlier with a MDB value will cause influence upon the position estimation of both vehicle and landmark mostly around mm level to cm level and are much lower than the estimation uncertainty. However, for the robot orientation (heading) estimation, the minimal detectable outlier within the camera measurement will cause 1-25 degrees bias. This relatively high ER value results of the orientation estimation may be due to the fact that the landmark detection measurement is range only measurement, no angle related measurement is provided in this DLR case, therefore, the orientation of the robot is indirectly observed by the landmark detection measurement which causes lower reliability than that of position estimation.

The estimation of landmark position will also be influenced by the outliers within the observations and the predicted states, using landmark lm1 as an example (Figure 4.18c and Figure 4.18f), when the lm1 is not visible during the trajectory, the outlier within the landmark detection measurement and predicted state of the EKF measurement update step will not influence the estimation of this invisible landmark. However, when the landmark is initially added to the state or revisited after a long journey for the first time, its estimation is largely adversely influenced by the measurements and the predicted states. This influence will be quickly reduced during the following detection. Therefore, for landmark estimation with the filtered based SLAM, the low reliability should be carefully considered when the landmark is initially detected or revisited.

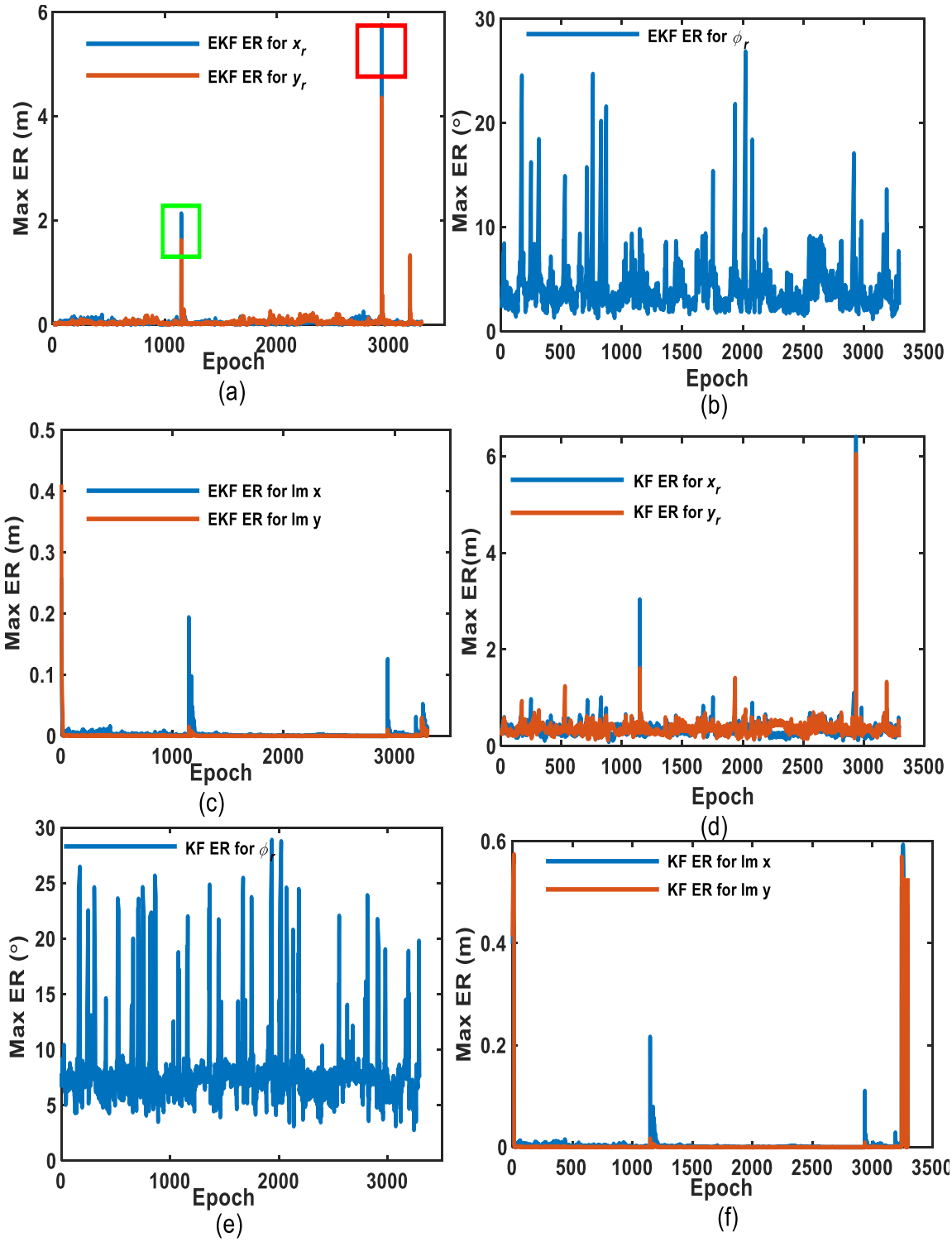


Figure 4.18 The External Reliability results: influence of outlier with MDB value within landmark detection measurement upon the final vehicle (a) position, (b) heading and (c) landmark lm1 position estimation; Influence of outlier with MDB value within the predicted state model upon the final vehicle (d) position, (e) heading and (f) landmark lm1 position estimation of the DLR dataset by EKF SLAM.

4.4 Summary

The EKF based only SLAM algorithms is tested with two real-world datasets. The precision of the estimation is getting worse when moving into new environment, and getting improved after travelling back to pre-visited area. Quality control has been conducted along with the estimation. Several outliers can be detected and identified with the quality control methods and their influence can be corrected immediately after identification, which indicating one of the advantages of our method, that it can mitigate the influence of the outliers without removing any measurements and without reducing the system redundancy.

Similar to the trend of the uncertainty, the Internal Reliability (MDB) value of the SLAM system will get higher when continues moving in new environment, which means the ability to detected outliers are reduced. Therefore, a frequent loop travelling is essential to maintain good reliability for online SLAM.

There are some other factors that may influence the reliability, especially cause some sudden jumps of MDB value. According to the numerical results and geometry analysis, it is found a redetection of a feature which is missing for a long period will cause the MDB of this measurement very high. The statistic test of this feature's measurement model will be unreliable. Other possible factors may be the redundancy of the systems, or the geometry strength.

Although MDB value will have some jumps, the undetected outlier may not always cause unacceptable results upon the final estimation. The External Reliability should also be taken into consideration. According to the numerical results, the undetected outlier in some high MDB periods may only cause less than 2 times of uncertainty value. Whether to accept these models with high MDB will depends on the requirement and applications of the SLAM system.

Chapter 5 Quality Control for Offline SLAM

5.1 Introduction

Offline SLAM, or full SLAM, which provides optimal estimation solutions with all available control and observation measurement, has become popular for building large-scale maps. The least squares based SLAM (such as graph-SLAM) treats all the robot pose along the whole trajectory and all the detected features as nodes. Spatial constraints between poses are encoded in the edges between the nodes. These constraints are resulting from observations or from odometry measurements (Grisetti, et al., 2010). The most likely trajectory and landmark position is then obtained by optimization.

Similar to the online SLAM case introduced in Chapter 4, offline SLAM also suffers from the unavoidable outliers within the measurements. It is essential to monitor outlier situation and reliability of the offline SLAM system.

Many studies for full SLAM are focusing on enhancing the efficiency and robustness of the optimization. However, due to the sparsity and singularity of the normal matrix in the optimization step, the uncertainty of the estimation is hard to achieve. For this reason, the property of precision is normally not evaluated in many full SLAM studies. As mentioned in Chapter 2, Unified LS (ULS) can be an alternative to dealing with ill conditioned matrix, however, the calculation of uncertainty and quality control criteria will be computation costly due to the introduce of the assisted non-singular matrix to replace the normal matrix, which makes the application of ULS impracticable for large scale map building. Therefore, a more efficient quality evaluation method is essential for offline SLAM.

Some researchers mentioned fix-lag smoothing method to achieve incremental smoothing with most recent poses and map (Dellaert and Kaess, 2017; Sünderhauf, 2012). When a lag of 1 is chosen, the robot state variables before the current pose are all marginalized out, that it corresponds a standard Kalman filter SLAM (Dellaert and Kaess, 2017). In

case of keeping smoothing back to the initial epoch, a full SLAM updating of the whole trajectory and the whole map will be achieved. Therefore, if quality control can be conducted recursively along with the Kalman Smoothing (KS) to the initial pose, the quality of the whole trajectory and map can be obtained simultaneously.

In this Chapter, a Kalman Smoothing based offline SLAM (KS SLAM) algorithm is proposed. The equivalence between the KS SLAM and a Graph SLAM is demonstrated. The recursively quality control for the whole offline system is tested based on this KS SLAM framework. The structure of this Chapter is as follows: Section 5.2 presents the mathematical models for offline SLAM methods. A simulated numerical testing is undertaken in Section 5.3 to show the relationship of EKF, KS, Graph SLAM. The proposed offline KS SLAM quality control system is tested with two real-world SLAM datasets in Section 5.4. Section 5.5 summarizes the performance of the KS online SLAM and its quality control system.

5.2 Mathematical Models for Offline SLAM Methods

5.2.1 Kalman Smoothing SLAM

The most commonly used smoother, Rauch-Tung-Striebel smoother (RTS smoother, which is also called Kalman smoother) (Sarkka, 2013), is a fixed-interval smoother that constituted by forward and backward passes. The forward pass is an Extended Kalman filter step, while the backward pass also depends on the predicted and estimated covariance and transition matrix stored in the Kalman filter pass. Therefore, RTS smoother is simpler in structure and easy to use. In this thesis, a Kalman Smoother (KS) based SLAM estimation framework is introduced to recursively solve the full SLAM problem offline.

As according to Chapter 2, the smoothed state vector at time *k* and its covariance matrix are derived as:

$$X^{s}_{(k)} = X_{(k)} + J_{(k)}(X^{s}_{(k+1)} - \overline{X}_{(k+1)}) \tag{5.1}$$

$$P_{X^s_{(k)}} = P_{X_{(k)}} + J_{(k)}(P_{X^s_{(k+1)}} - P_{\overline{X}_{(k+1)}})J^T_{(k)} \tag{5.2}$$

here the superscript ' ^{s} ' represents the smoothing estimation.

5.2.2 Full Least Squares SLAM

Another estimation framework used in this work is a graph-based least squares optimization SLAM, here the front-end part is not discussed as this work mainly focuses on the back-end optimization step. As this SLAM algorithm aims to solve the least squares problem with full update of all the robot poses along the whole trajectory and all the detected maps, in this work, 'Full Least Squares (FLS)' is used to refer to this SLAM framework.

The node of the FLS SLAM is $X = (X_r; X_f) = (X_{r(1)}; \dots X_{r(t)}; X_f; \dots X_{f(t)})$with last time *t* and total observed landmark number *p*, here we can see, $X_{f(k)} \in X_f$. There are two types of edges. The first edge represents the odometry displacements of robot pose, which corresponds the motion model of the robot or vehicle states in Chapter 4. Another edge represents the observation of the landmarks, which corresponds the measurement model in the EKF framework. The information matrix of the FLS is $P = blkdage(Q^{-1}, R^{-1}) = blkdage({Q_{(1)}}^{-1}, \dots, {Q_{(t)}}^{-1}, {R_{(1)}}^{-1}, .., {R_{(t)}}^{-1})$. After given a predicted value $\bar{\bar{X}}$ of the node variables (also called approximate value), the FLS SLAM is to solve Equation 5.3:

$$\left(A^T P A\right)\Delta X = A^T P l \tag{5.3}$$

$$
\begin{aligned}
A &= \begin{vmatrix} A_{dync} \\ A_{meas} \end{vmatrix} = \begin{vmatrix} A_{rr} & 0 \\ H_r & H_f \end{vmatrix}, \\
l &= \begin{vmatrix} l_{dync} \\ l_{meas} \end{vmatrix} = \begin{vmatrix} l_{dync(1)} \\ \vdots \\ l_{dync(t)} \\ l_{meas(1)} \\ \vdots \\ l_{meas(t)} \end{vmatrix} = \begin{vmatrix} f(\bar{\bar{X}}_{r(0)}, u_{(0)}) - \bar{\bar{X}}_{r(1)} \\ \vdots \\ f(\bar{\bar{X}}_{r(t-1)}, u_{(t-1)}) - \bar{\bar{X}}_{r(t)} \\ y_{(1)} - h\left(\bar{\bar{X}}_{(1)}\right) \\ \vdots \\ y_{(t)} - h\left(\bar{\bar{X}}_{(t)}\right) \end{vmatrix}
\end{aligned} \tag{5.4}
$$

$$
\begin{aligned}
A_{rr} &= \begin{vmatrix} I & & & & \\ -\Phi_{r(2)} & I & & & \\ & -\Phi_{r(3)} & \ddots & & \\ & & \ddots & I & \\ & & & -\Phi_{r(t)} & \end{vmatrix}, \\
H_r &= \begin{vmatrix} H_{r(1)} & & & & \\ & H_{r(2)} & & & \\ & & \ddots & & \\ & & & H_{r(t)} & \end{vmatrix}, H_f = \begin{vmatrix} H_{f(1)} \\ H_{f(2)} \\ \vdots \\ H_{f(t)} \end{vmatrix}
\end{aligned} \tag{5.5}
$$

here $H_{r(k)}$ and $H_{r(k)}, k = 1, \ldots, t$ is the Jacobians of the measurement model w.r.t the robot node at time k and the landmark nodes. The solution and its covariance matrix of FLS SLAM will be:

$$\hat{X} = \bar{\bar{X}} + \Delta\hat{X} \tag{5.6}$$

$$Q_{\hat{X}} = (A^T P A)^{-1} \tag{5.7}$$

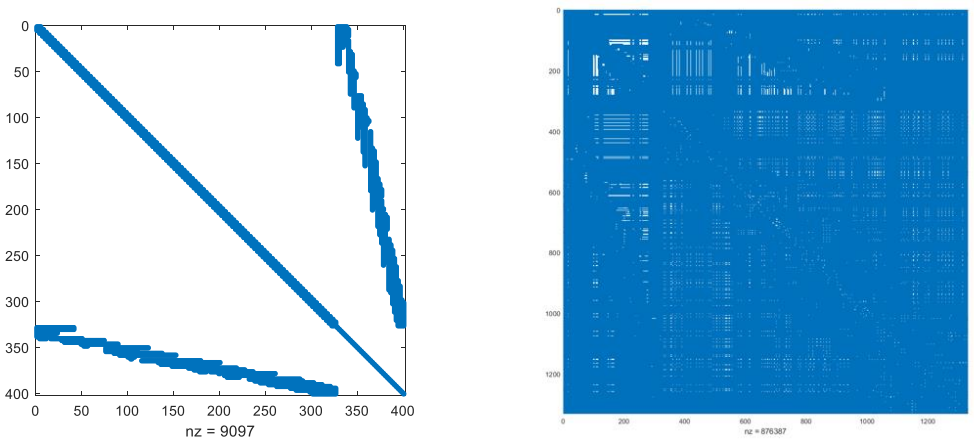
According to the formulation, the full least squares SLAM is equivalent to the recursively estimation of Kalman Smoother SLAM if it uses the KF prediction state value as the approximate value. Unified LS estimation can be applied in this FLS SLAM problem to deal with singular covariance matrix. Quality control method presented in Chapter 2 can be used to test the full SLAM problem.

5.3 Optimal Estimation and Quality Control Methods for Offline SLAM applications

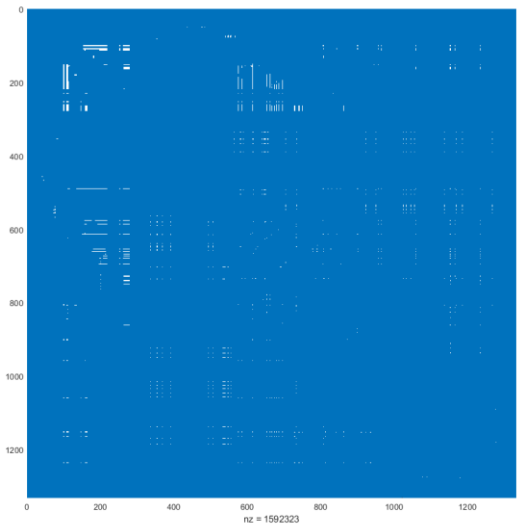
5.3.1 Importance of Incremental Update of Full LS SLAM

Solving variables with a sparse matrix is efficient in the FLS case. However, our quality control method is residual-based, the calculation of the covariance matrix of residual (Qv) is essential, which is calculated according to the estimated variable covariance $Q_{\hat{X}}$ and the observation Jacobian matrix A. For the large scale least squares optimization, the inverse step of the normal (A^TPA) matrix to achieve $Q_{\hat{X}}$ matrix is very time consuming and requires extensive memory. After achieving the full Qv matrix, the further calculation with the Qv matrix is also very costly due to the low number of zero elements in the Qv matrix.

Figure 5.1 shows the visualization of matrixes used during the calculation of quality control. For the A^TPA matrix, the nonzero elements' locations are concentrated along the diagonal and two sides, while for the Qv and PQvP matrix, they are full matrix with most elements nonzero. Therefore, the computation and memory requirement will be huge for the large-scale SLAM case.



(a) (b)



(c)

Figure 5.1 Visualization of matrixes during the calculation of quality control ((a) ATPA, (b) Qv, (c) PQvP)

Due to the low efficiency and huge memory cost of the calculation with full matrix $Q_{\hat{X}}$, Qv, incremental estimation of the system quality is preferred. To achieve this, incremental smoothing (Kalman smoothing) method and its quality control method are introduced to the full optimization SLAM.

5.3.2 Demonstrating Equivalence between KS and FLS SLAM

Some research mentioned fig lag smoothing method can be used to achieve incremental smoothing with most recent poses and map (Dellaert and Kaess, 2017; Sünderhauf, 2012). When a lag of 1 is chose, the robot state variables before the current pose are all marginalized out, that it corresponds a standard Kalman filter SLAM (Dellaert and Kaess, 2017). In case of smoothing back to the initial epoch, a full slam update of the whole trajectory and whole map will be achieved. Therefore, if quality control can be conducted recursively along with the Kalman Smoothing (KS) to the initial pose, the quality of the whole trajectory and map can be obtained simultaneously.

A small simulated numerical case is used to show the equivalence between the Kalman smoothing and a full least squares optimization based graph SLAM. This simulated case

study can also be used to show the performance on outlier detection, identification, and correction of the three systems, since these added outliers' value can be controlled and the ground truth is known. This numerical case has a similar mode of control and observation model to the one tested in Chapter 2, Section 2.5. In this simulation case, the initial point is a known point X0. By walking to point 1, a feature *f* is detected at the first time from the platform. Therefore, the feature state is added to the graph structure as a new node. In the following epochs, the feature is detected by 4 times. Finally, the platform walks back to the initial point. The Constraint between X0 and X8 can be added for the situation that the loop is to be closed.

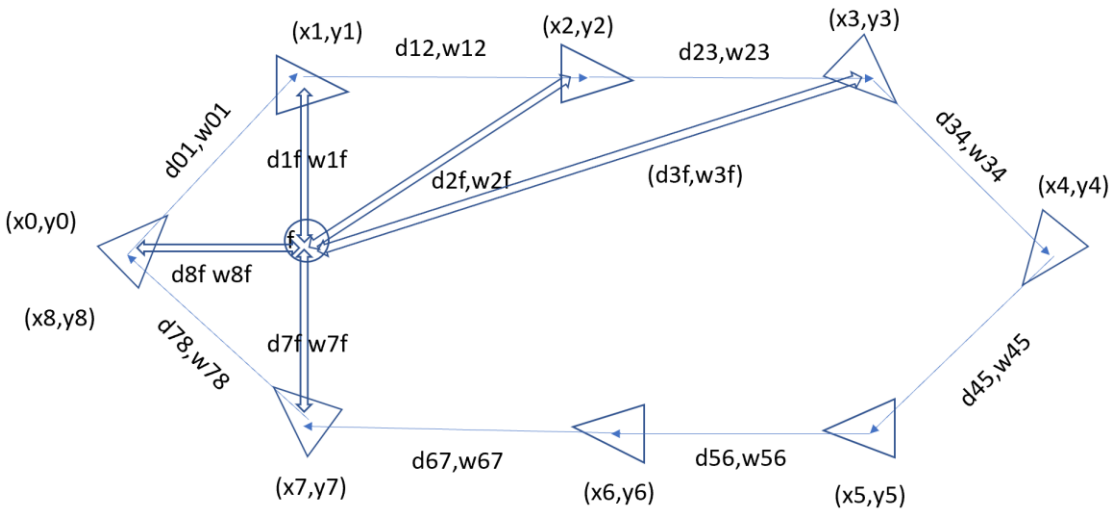


Figure 5.2 A simulated SLAM case

5.3.2.1 Estimation by EKF, KF, Graph SLAM for simulated SLAM Case

There will be two scenarios for the estimation, one is no Loop closure case, another is considering X8=X0, which makes the loop closed. The estimated pose results by EKF, KS, Graph SLAM and the ground truth is shown in Table 5.1. In this table, only the results of non-closure case are shown. It is found KS SLAM and Graph SLAM have the same estimation results.

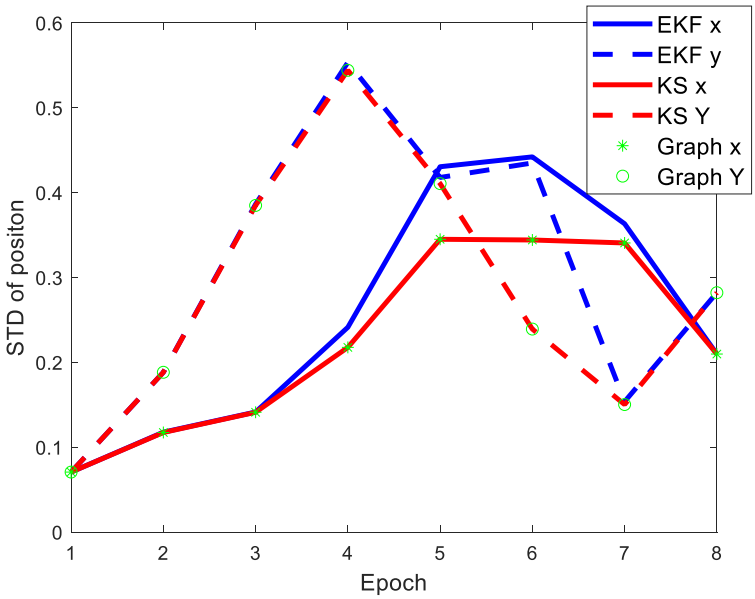
Table 5.1 Estimated pose (position and heading) by EKF, KS, Graph SLAM and the ground truth of the simulated SLAM case (no Loop Closed case)

point		**X1**	**X2**	**X3**	**X4**	**X5**	**X6**	**X7**	**X8**
EKF	x/ meter	7.071	17.038	27.018	34.093	27.025	17.025	6.993	0.0033
	y/ meter	.0391	7.071	7.056	-0.0117	-7.086	-7.091	-6.977	0.0487
	Heading/ degree	0	-0.052	-44.972	-134.972	-179.972	-179.972	134.698	44.191
KS	x/ meter	7.071	17.035	27.193	34.092	27.016	17.019	7.021	0.0033
	y/ meter	7.071	7.070	7.550	-0.0153	-7.078	-7.055	-7.023	0.0487
	Heading/ degree	-0.0046	-0.0873	-44.990	-135.054	-180.131	-180.186	134.782	44.191
Graph	x/ meter	7.071	17.035	27.193	34.092	27.016	17.019	7.021	0.0033
	y/ meter	7.071	7.070	7.550	-0.0153	-7.078	-7.055	-7.023	0.0487
	Heading/ degree	-0.0046	-0.0873	-44.990	-135.054	-180.131	-180.186	134.782	44.191
Ground Truth	x/ meter	7.071	17.071	27.071	34.142	27.071	17.071	7.071	0
	y/ meter	7.071	7.071	7.071	0	-7.071	-7.071	-7.071	0
	Heading/ degree	0	0	-45	-135	-180	-180	135	45

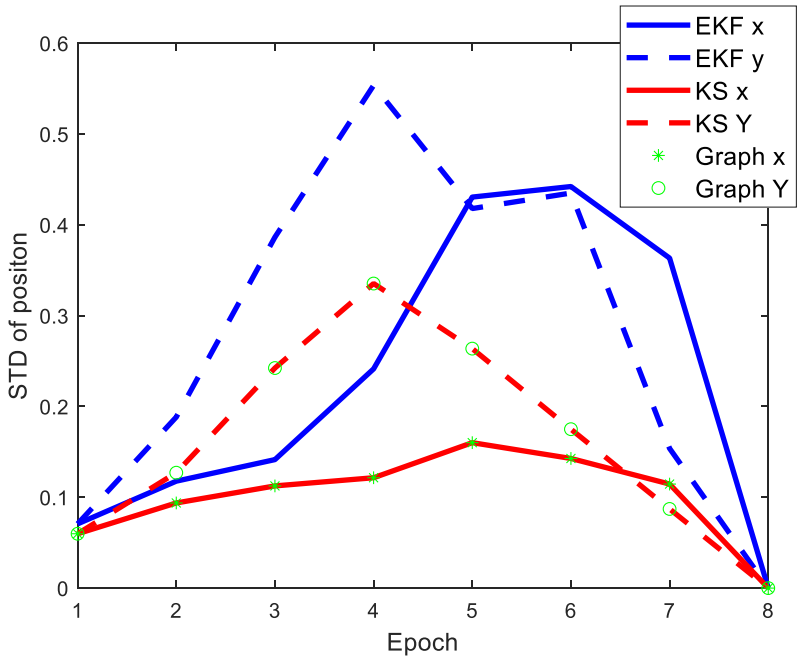
Table 5.2 RMSE of estimated pose (position and heading) by EKF, KS, Graph SLAM with respect to the ground truth of the simulated SLAM case

RMSE	EKF			KS			Graph		
	x/ meter	y/ meter	Heading/ degree	x/ meter	y/ meter	Heading/ degree	x/ meter	y/ meter	Heading/ degree
No Loop Closed	0.0457	0.0391	0.307	0.0429	0.0263	0.309	0.0429	0.0263	0.309
Loop Closed	0.0451	0.0351	0.110	0.0269	0.0572	0.165	0.0269	0.0572	0.165

The RMSE value with respect to the ground truth (Table 5.2) also shows that KS and Graph SLAM can achieve same estimation results, for both non-closure case and closed case. The smoothing and least squares optimization has better accuracy (Table 5.2) and better precision (Figure 5.3 and 5.4) for the estimation of platform position and heading.

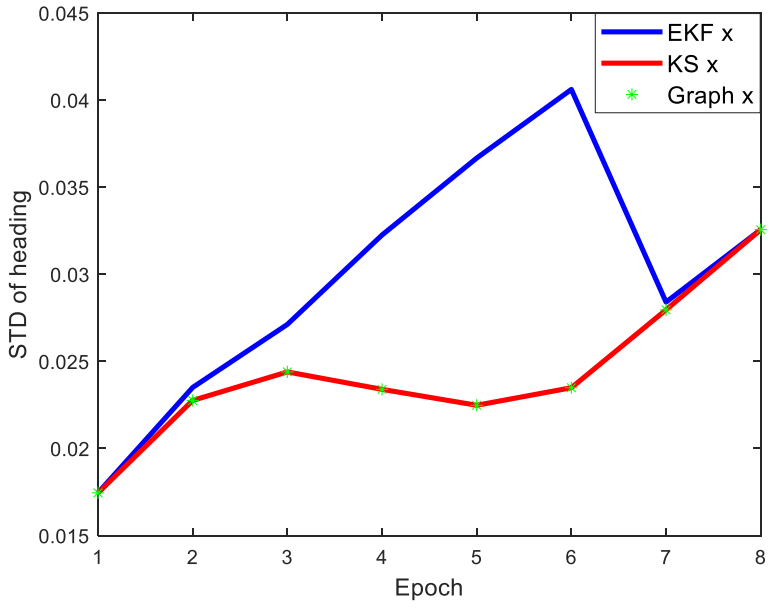


(a)

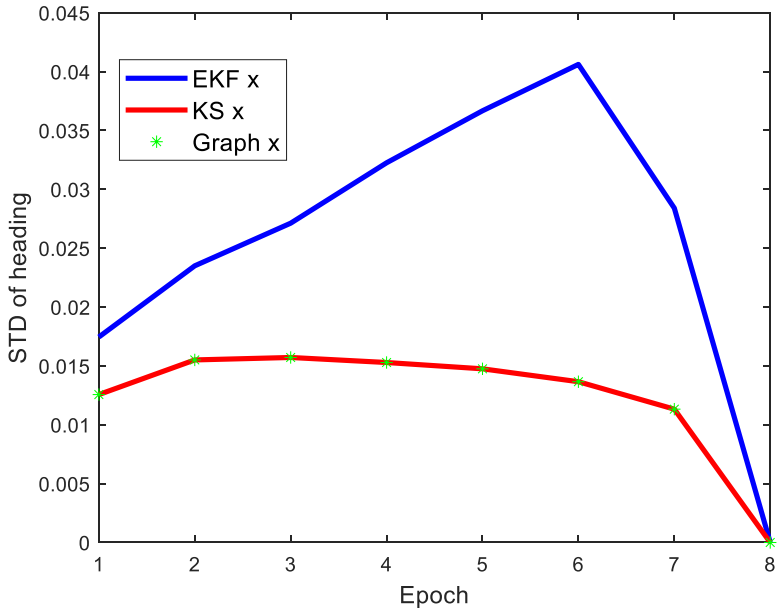


(b)

Figure 5.3 Standard deviations of estimated position by EKF, KS, graph SLAM of the simulated case ((a) not loop closed; (b) loop closed)



(a)



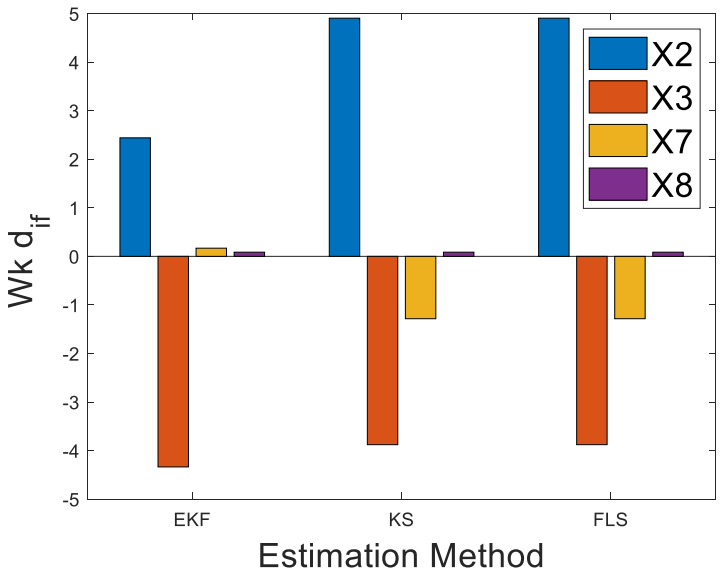
(b)

Figure 5.4 Standard deviations of estimated heading by EKF, KS, graph SLAM of the simulated SLAM case ((a) not loop closed; (b) loop closed)

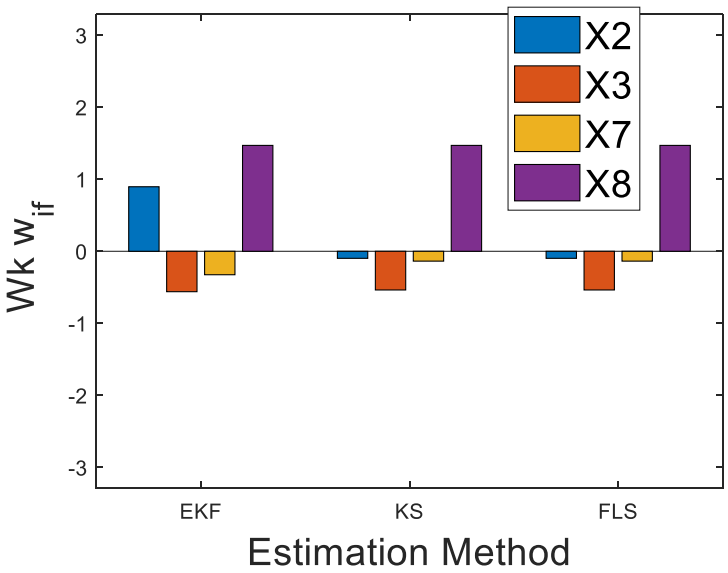
Here in the testing with the case of closed loop, the X0 in this simulated case is a known point with known position values, its uncertainty is zero. Therefore, uncertainty of the last point X8 will recovered to near to zero. Much better of estimation precision will be achieved by the smoothing or LS optimization under this condition (Figures 5.3 b and 5.4 b).

5.3.2.2 Quality Control by EKF, KF, Graph SLAM for simulated SLAM Case

Quality control procedure is also tested with this simulated SLAM case. Since in the original simulation, only random error is added to the measurement models, the outlier detection statistic test results of the models will be very small, no outlier is detected. In order to have a more clearly explanation of the quality control performance, a 0.8 meters additional outlier is added to the range measurement between point X2 and the feature. The outlier statistic testing results is shown in figure 5.5.



(a)



(b)

Figure 5.5 Outlier statistic test results for (a) range and (b) turn angle observation model of the simulated SLAM case

According to Figure 5.5, since an additional outlier is added to the range observation at point X2, all the three estimation methods has detected outliers in range measurement model (Figure 5.5a) with Wk value higher than 3.29. KS and Graph SLAM have same quality control results. However, for the EKF method, the detected outlier is not found in

point X2, but in X3. The reason for this delayed outlier detection may be due to the value of the additional outlier (0.8 meters) is lower than the minimal detectable outlier value for the EKF system at epoch 2, which is 0.983 meters (Table 5.3). Since the additional outlier cannot be successfully detected by EKF method at the position it is added, it will influence the following epochs. Misdetection happens for these simulated cases by EKF quality control method. However, for the KS and FLS method, since they have lower MDB value, the outlier can be successfully detected and identified in epoch 2. For the KS or FLS Wk results, they also show the influence of the addition outlier on the following epochs after epoch 2, which may because the exist of outlier will influence the prediction results of the following states.

Table 5.3 MDB results of observation model by EFK, KS, Graph SLAM of the simulated SLAM case

MDB	**Range/ meter**				**Turn angle /degree**			
	X2	X3	X7	X8	X2	X3	X7	X8
EKF	0.983	0.733	2.951	0.933	9.565	9.474	13.554	10.068
KS	0.628	0.678	1.163	0.933	9.139	9.903	9.139	10.068
Graph	0.628	0.678	1.163	0.933	9.139	9.903	9.139	10.068

According to the Outlier statistic test results, the corresponding estimated outlier values can be obtained for the models with Wk value higher than the threshold of 3.29. These estimated outlier values can be used to correct the measurements and mitigate the influence of the detected outlier. Figure 5.6 shows the estimation results with and without correcting for the estimated outlier. It is found that the EKF mis-detected outlier value can only be used to mitigate a few parts of the influence of the real added outlier (Figure 5.6c). With a more accurate detection and identification of the outlier by the KS/FLS method, more accurate outlier value can be estimated, and more accurate final solution will be acquire with the outlier correction method (Figure 5.6d).

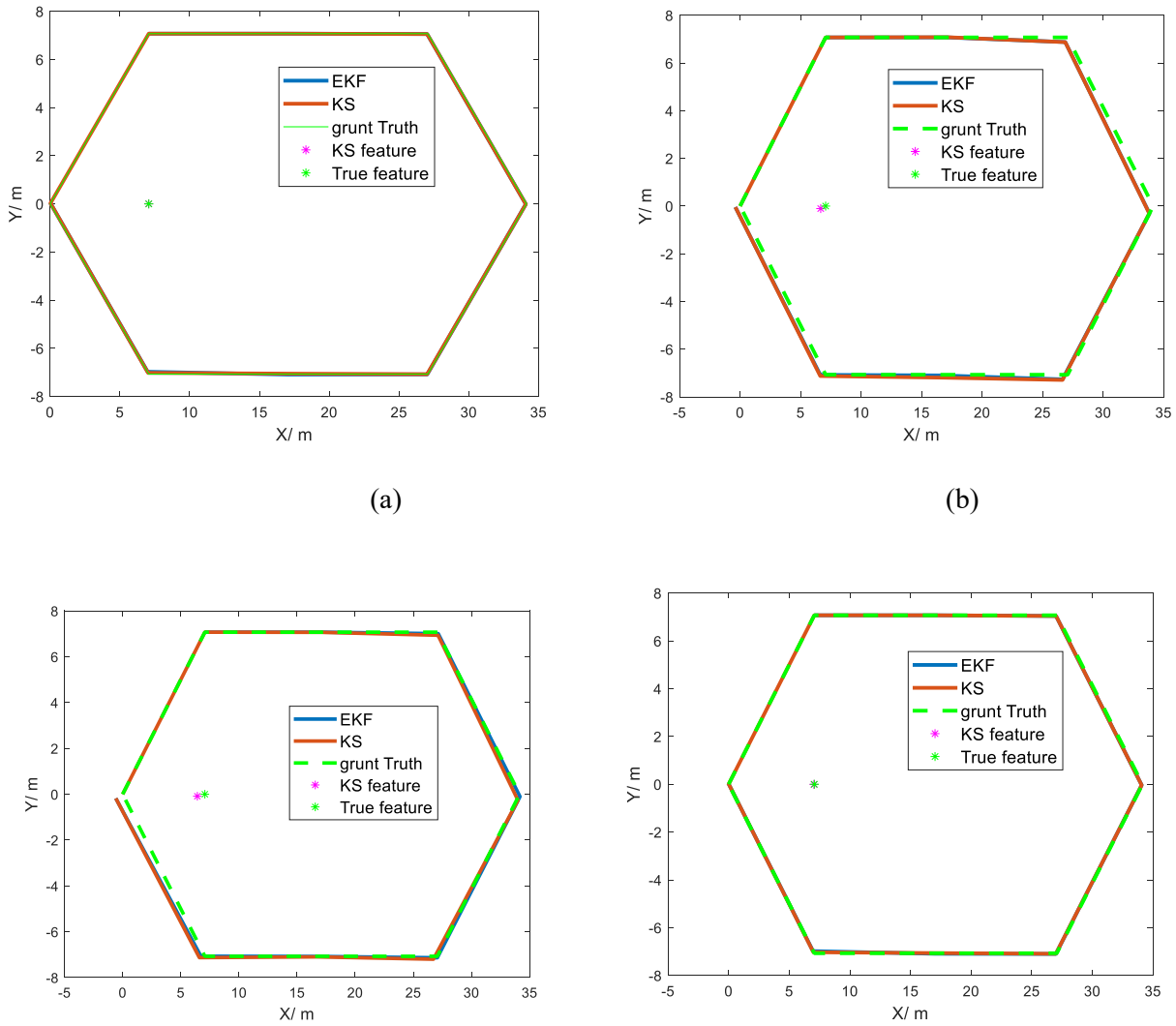


Figure 5.6 Estimated trajectory and feature position in case of (a) no additional outlier; (b) adding additional outlier in epoch 2; (c) correcting influence of outlier estimated by EFK at point X3; (d) correcting influence of outlier estimated by KS at point X2

Kalman smoothing based SLAM and full least squares based SLAM can acquire the same estimation and quality control results for the simulated SLAM case study. Because KS method is much more efficient than the full least squares one since it is doing full estimation recursively, the KS method is suitable for the estimation and quality controlling for the large-scale SLAM problem. The result of EKF step can be improved by the smoothing step on both accuracy and precision. Additional Outliers within the SLAM system can be successfully detected and their influences on the estimation are mitigated according to these numerical results.

For next section, two standard real-world SLAM datasets have been fully analysed with the proposed Kalman smoothing SLAM method. The tests and analysis with Victoria Park dataset and DLR dataset are used to demonstrate how the estimation is improved by the smoothing method and how the proposed quality control method will perform during the KS estimation.

5.4 Experiments and Analysis

The two publicly available real-world SLAM datasets, which were used in Chapter 4, will be analysed here with the proposed offline KS SLAM system.

5.4.1 Kalman Smoothing Quality Control for Victoria Park dataset

Victoria Park dataset is tested with the Kalman Smoothing SLAM (KS SLAM) method with quality control. The KS estimated trajectory has about up to 20-40 cm difference to the EKF results of estimated trajectory, and smoothed results has much lower and more smooth uncertainty results (Figure 5.7b), which indicating better estimation precision. Some increase of uncertainty can be found as a result of continuing travelling in unknown environment. The estimation precision will fall down when visit a pre-visited place.

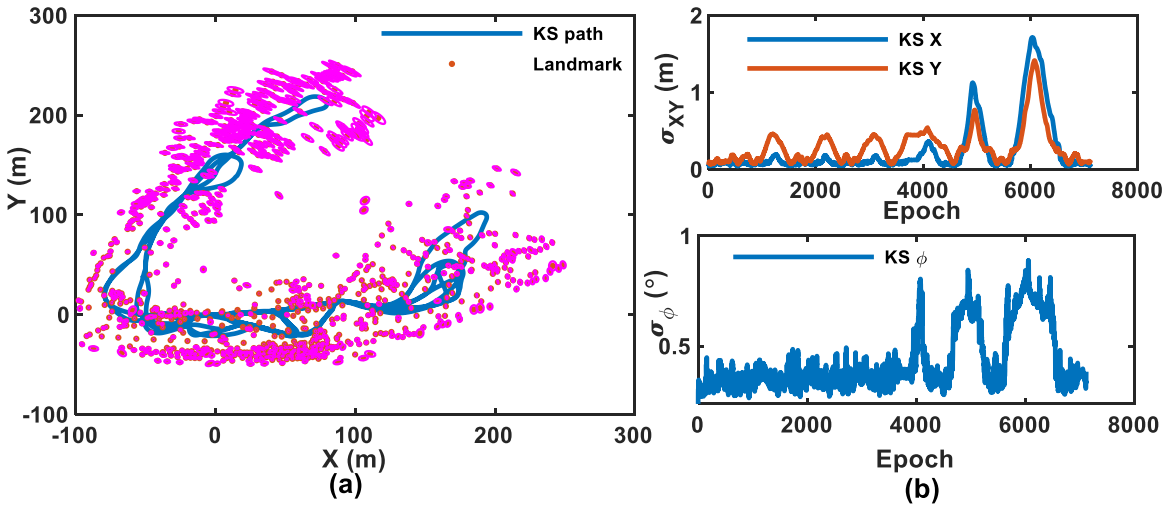


Figure 5.7 (a) Estimated trajectory and estimated landmark position by Kalman smoothing SLAM and (b) their uncertainty for the Victoria Park dataset

Kalman Smoothing based Quality Control (KS QC) test is undertaken to analyse the Victoria Park dataset. The quality control methods for the two SLAM estimation algorithms (EKF SLAM and KS SLAM) are different when testing the dynamic information. For the EKFQC, the predicted states are tested within the measurement update step, the processing of prediction is not tested. Therefore, if there is an outlier within the odometry model, the EKFQC method cannot directly detect this outlier, but will only detect the possible influence caused by this outlier upon the prediction state parameters. However, the KSQC will directly test the dynamic model with the odometry measurement and detect the possible outlier within this kind of model.

According to the outlier detection and identification results of the smoothing based SLAM estimation for the Victoria Park dataset (Figure 5.8), more outliers, both in observations and in odometry measurements, are detected by KSQC than by EKFQC, indicating the smoothing based method is more sensitive to the outliers and is more likely to detecting them. According to the filter results, the detected outliers are all within the bearing measurements, not within the ranging measurements or the predicted state models, whereas for the smoothed result, there are also possible outliers within the ranging measurement and the odometry measurements other than the bearing measurements that can be detected and identified.

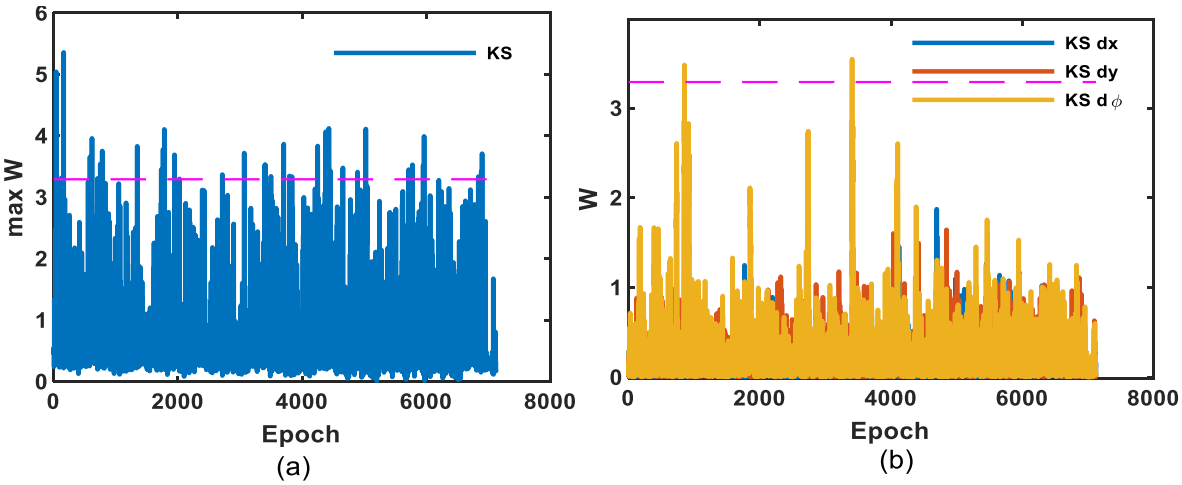


Figure 5.8 Outlier statistic test (W statistic test for outlier identification (a) within landmark detection model, (b) within vehicle odometry model) by KS for the Victoria Park dataset

Figure 5.9 shows the influence of the identified outliers within ranging and bearing measurements upon the final vehicle pose estimations. It is found these identified outliers will cause very low influence of the final estimation when compared to the KS SLAM estimation uncertainty. These influence values upon position estimation are around 0.01-45% of the position estimation uncertainty, while that upon heading are around 3.3-124% of the heading estimation uncertainty. These identified outliers will also influence the vehicle position parameters at neighbourhood epochs since the models are correlated. This influence will reduce when the time gap increases.

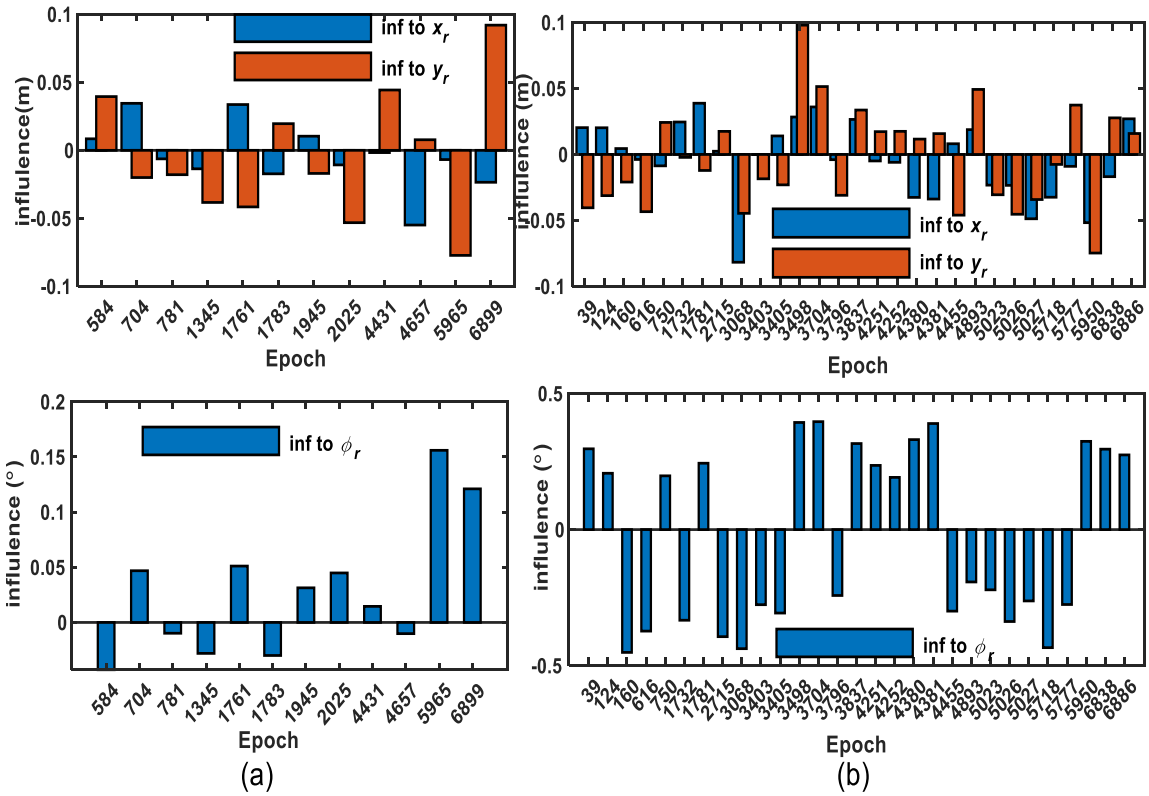


Figure 5.9 Influence of KS identified outliers within (a) ranging and (b) bearing measurement upon the final vehicle pose (position x_r and y_r and heading ϕ_r) estimation for the Victoria Park dataset

Figure 5.10 and 5.11 show the results after correcting the outliers identified by KSQC method. Thus, the influence of identified outliers can be successfully mitigated with the proposed KSQC method.

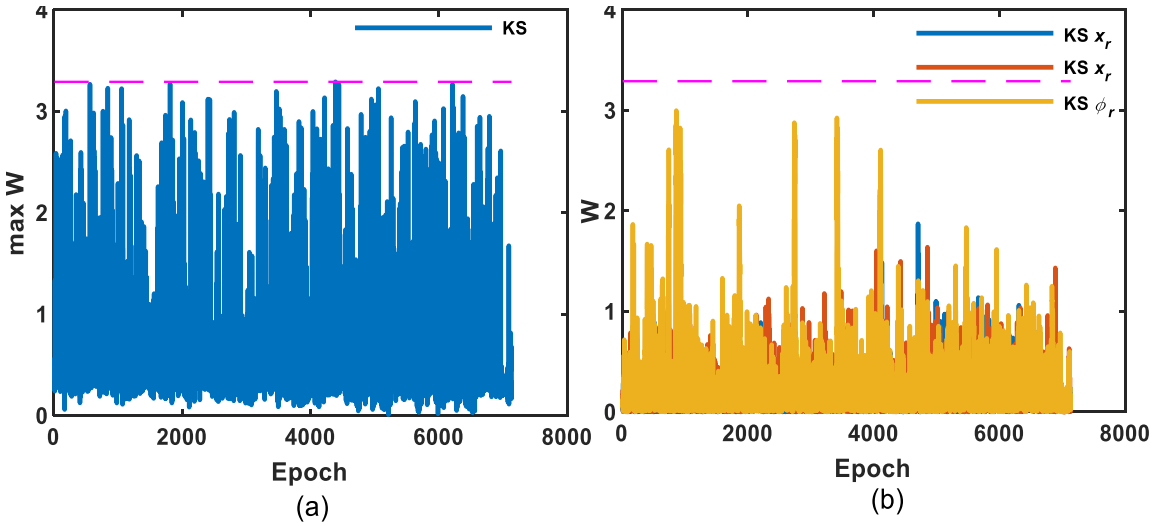


Figure 5.10 Outlier statistic test after correction of the influence of identified outliers for (a) landmark detection model, and for (b) odometry model by KS for the Victoria Park dataset

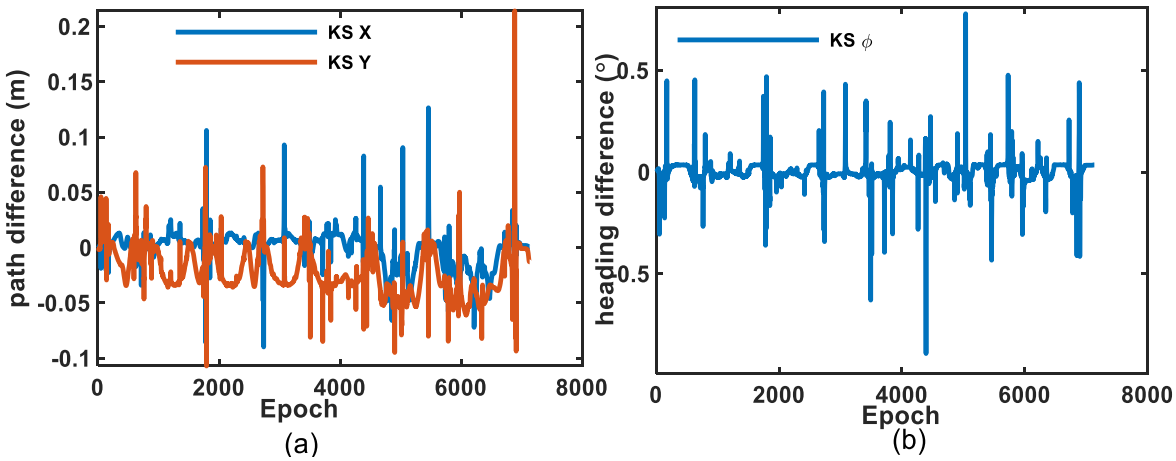


Figure 5.11 Differences in the estimated vehicle pose ((a) position x_r and y_r; (b) heading ϕ_r) with and without correcting the influence caused by KS identified outliers for the Victoria Park dataset

Figure 5.12 and Figure 5.13 show the Reliability of the Victoria Park dataset under the KS SLAM framework. The smoothing based method has slightly lower MDB values for the landmark detection measurements (ranging and bearing) than the EKF SLAM results, indicating the KS method has the capability to detect outlier with lower value within the detection measurement. This can also be proved by the outlier detection results that KS method has detected more outliers than the EKF method with higher W test values.

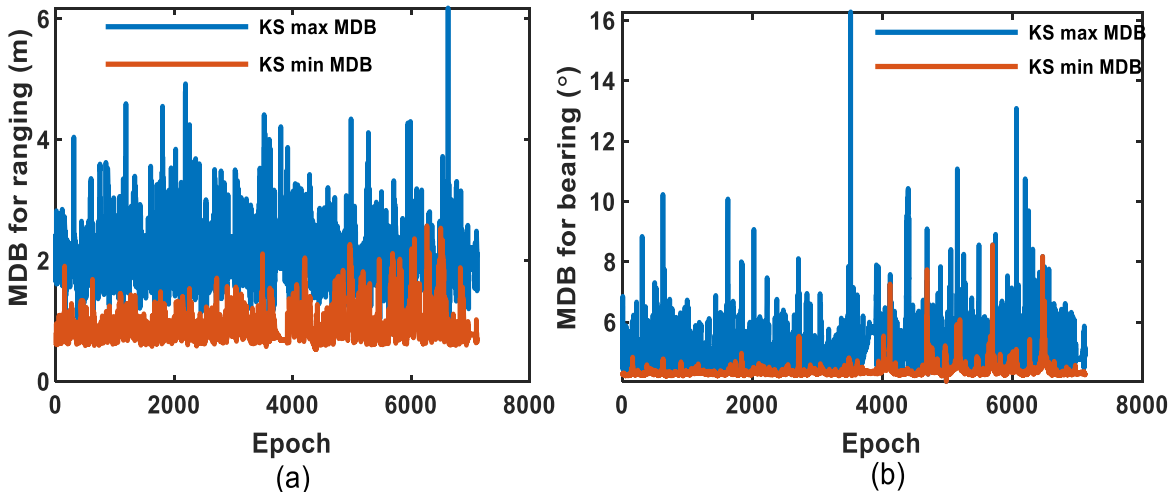


Figure 5.12 Minimum detectable bias (MDB) results for the landmark detection models of (a) ranging measurement and (b) bearing measurement by KS for the Victoria Park dataset

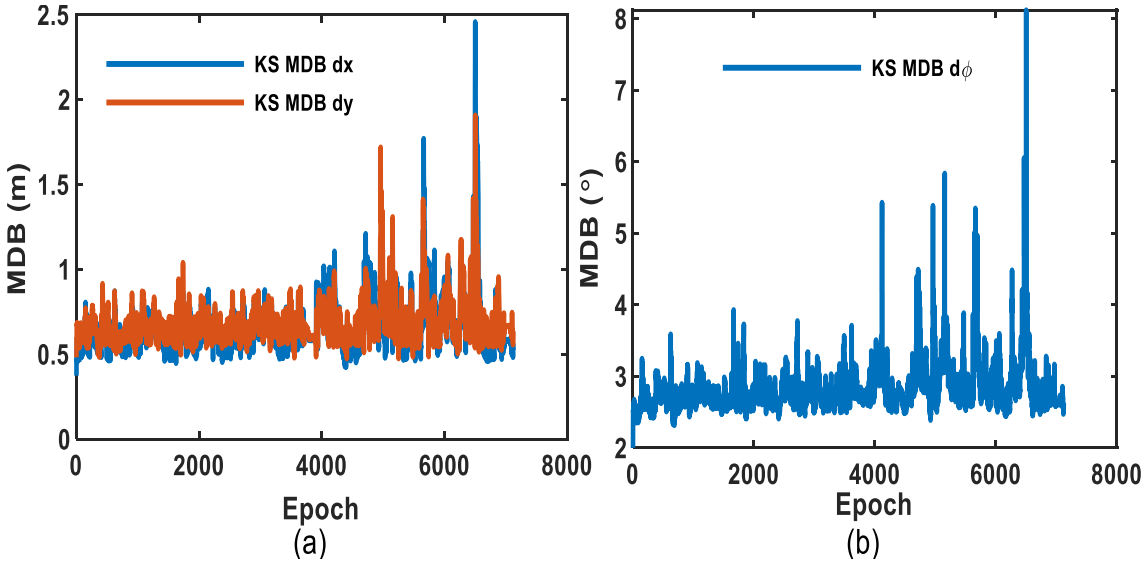


Figure 5.13 Minimum detectable bias (MDB) results for the vehicle dynamic model of (a) dx and dy and (b) orientation dϕ by KS for the Victoria Park dataset

The significant improvement by KS methods for the system reliability happens during the period with long loop closing, and also some other periods that have poor geometry strength (such as low feature number, revisiting of landmark after a long travel as mentioned in Section 4.3.3) (compare Figures 4.7 and 4.8 with Figures 5.12 and 5.13). For instance, according to Figure 5.13, the smoothing based SLAM system's reliability are less influenced by the large loop with long travel into unknown area (epoch 4500 to

6700), which is because it can utilize the revisiting measurements after the loop closed. Therefore, the KS SLAM is more reliable than EKF SLAM

The smoothing based method also has better External Reliability results (Figure 5.14) than the filter-based method. The influences of the undetected biases within the landmark detection and odometry measurements upon the vehicle states are smaller. However, for the landmark position estimation, the EFK results show no impact of measurement outlier when a landmark is not detected at a current period. The smoothing results are different, since all the observation and odometry information are used together for offline optimization, and since the landmarks are correlated between each other, the outlier within the observation and odometry measurement at one epoch will affect the final estimation of all the visited landmarks (Figure 5.14c and Figure 5.14f).

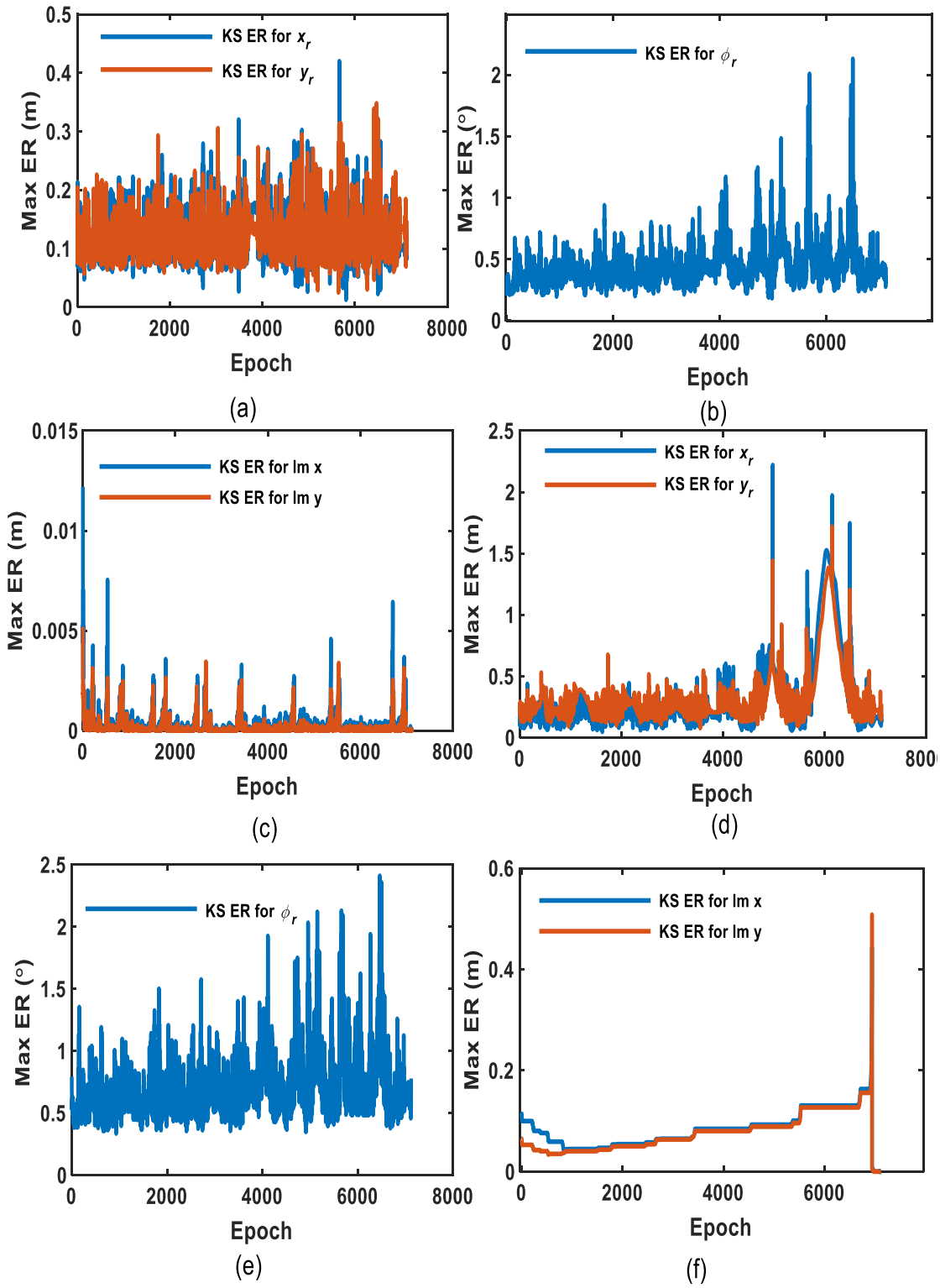


Figure 5.14 The External Reliability results: Influence of outlier with MDB value within landmark detection measurement upon the final vehicle (a) position, (b) heading and (c) landmark lm1 position; Influence of outlier with MDB value within vehicle odometry model upon the final vehicle (d) position, (e) heading and (f) landmark lm1 position estimation of the Victoria Park dataset by KS SLAM

Although smoothing based SLAM can enhance the system quality, the performance of the EKF based SLAM estimation results may be acceptable during most of the travelling epochs according to different application requirements. It is because during the EKF process for the Victoria Park dataset, there are many small loop closures that can frequently enhance the EKF estimation. The next section with testing the DLR dataset will show how smoothing based estimation and quality control method will improve the EKF based SLAM for case of large loop closure.

5.4.2 Kalman Smoothing Quality Control for DLR dataset

The DLR dataset is also analysed with the KS SLAM method. Figure 5.15 shows the estimated trajectory and its uncertainty by the smoothing method for the DLR dataset. The estimated trajectory with the DLR dataset by EKF method (Figure 4.2b) is significantly improved by the smoothing based SLAM, better accuracy (Figure 5.15b) can be acquired than that acquired by the EKF results. EKF results are inconsistency at some travelling periods, especially the period that the robot combing back to the building after a journey outside of the building with no overlapping landmarks, whereas for the smoothing based estimation, the estimated trajectory is smoother and thus no unnormal displacement is found.

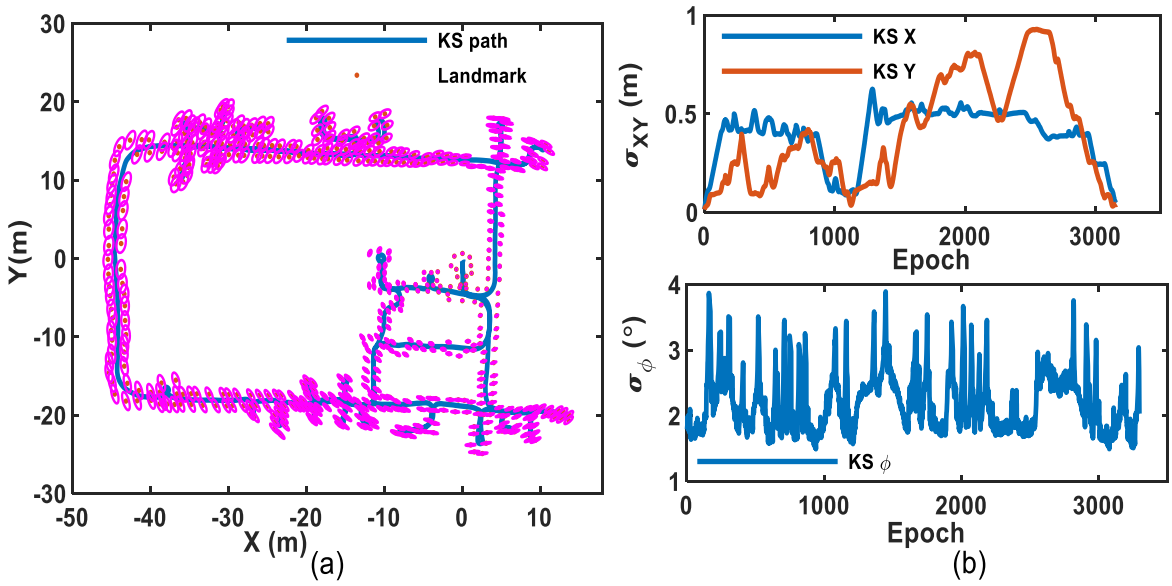


Figure 5.15 (a) Estimated trajectory and estimated landmark position by Kalman smoothing SLAM and (b) their uncertainty for the DLR dataset

For the Kalman Smoother (KS) based SLAM dealing with the DLR dataset, its quality control results also show no outlier is detected within the observation model (Figure 5.16a) which is similar to the online EKF test.

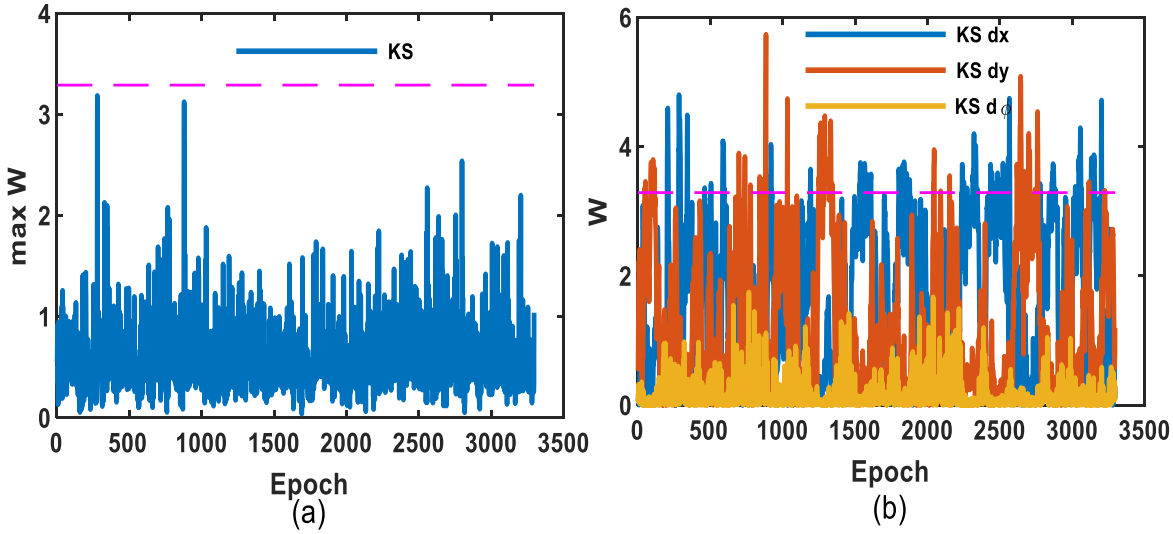


Figure 5.16 Outlier statistic test for outlier detection within (a) landmark detection model, and (b) within robot odometry model by KS for the DLR dataset

As aforementioned, the EKF method cannot test the odometry measurements, its quality control results for the predicted states are correlated to the landmark detection measurement. With EKF QC, only isolated outliers are detected within the predicted states. However, for the KS case, the W test results show that many outliers may exist within the odometry measurements along the whole trajectory (Figure 5.16b). These odometry outliers will cause the drift of the whole trajectory estimation but can be corrected.

With the KS outlier identification method, there are in total 440 odometry measurements with statistic test value higher than the threshold value (3.29) over the whole 3295 odometry models, these 440 measurements can be divided into several groups since some of them are within the continued time epochs, not isolated outliers. An outlier within one odometry measurement will influence the odometry model at the neighbourhood epochs due to the correlation between the models, especially for this DLR case that robot is travelling into new places with a small number of overlapping landmarks, and there is a very low number of loop closings to bound the accumulated trajectory error. The possible outlier within one odometry model will contaminate other models' statistic test results

and will cause misidentification of outliers at other epochs. Therefore, not all the 440 measurements with W value higher than 3.29 in this test are truly containing outliers. The measurements with the peak W test value within a group are the most possible places of outliers. Therefore, there are possibly 21 outliers within the robot odometry model with peak W test results. The values of these outliers can then be estimated together due to the high correlation between the outlier statistics for the odometry models.

Figure 5.17 shows the distribution of the identified outliers and their influence upon the final estimation of robot pose parameters at the epoch there are identified. These influences caused by outliers within odometry measurements are around 0.1-46% of the KS estimation uncertainty.

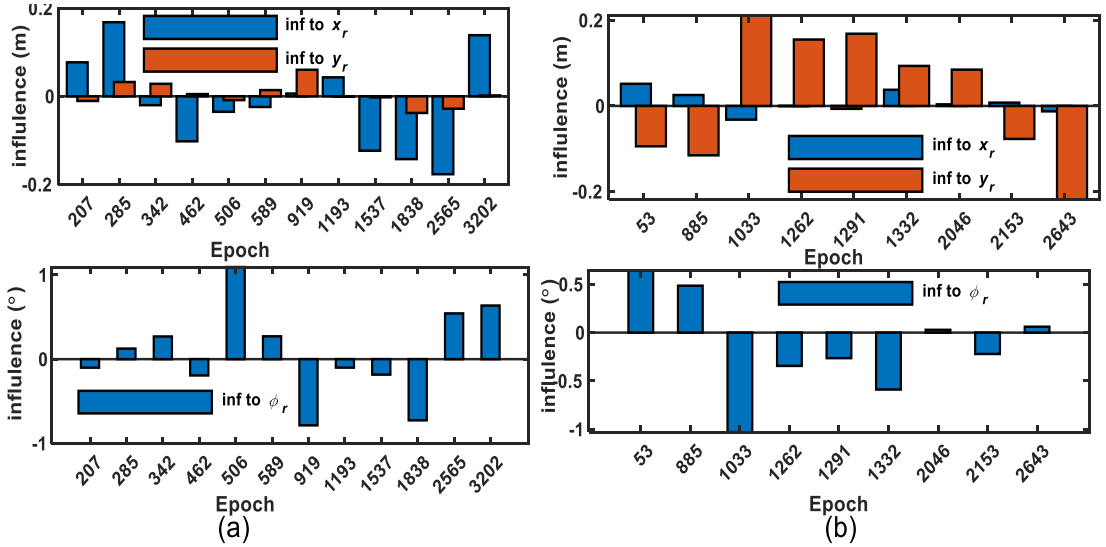


Figure 5.17 Influence of the KS identified outliers within (a) dx and (b) dy odometry measurement upon the final vehicle pose (position x_r and y_r and heading ϕ_r) estimation for the DLR dataset

Figure 5.18 shows the difference between trajectory estimations with and without outlier correction by the KSQC method, the differences are around $\pm$ 0.25 meters for position estimation and -0.5 to 2.5 degrees for heading estimation. After corrections for the identified outliers, no more outliers are detected (Figure 5.19).

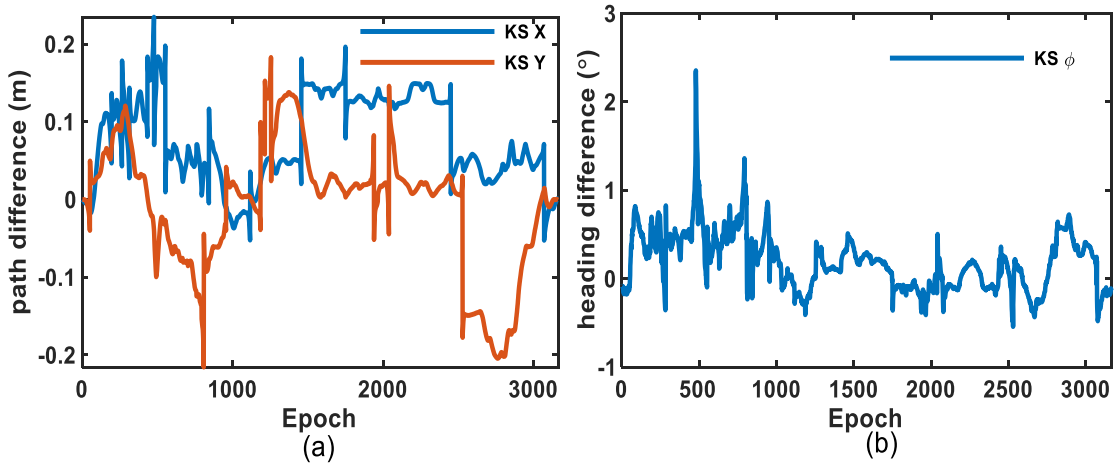


Figure 5.18 Differences in the estimated vehicle pose ((a) position x_r and y_r; (b) heading ϕ_r) with and without correcting the influences caused by the KS identified outliers for the DLR dataset

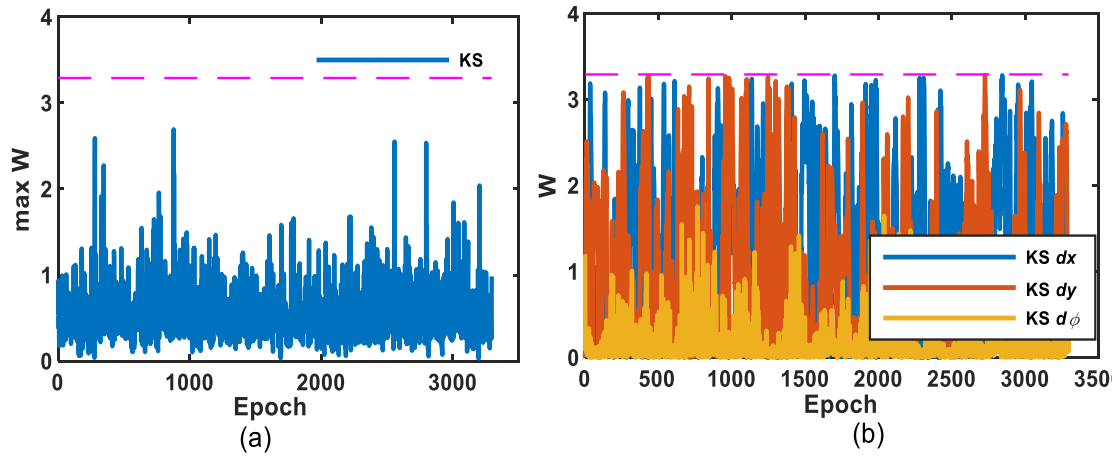


Figure 5.19 Outlier statistic test after correction of the influence of the identified outliers for (a) landmark detection model, and for (b) odometry model by KS for the DLR dataset

When compared to the reliability results of the EKF SLAM, lower MDB values (Figure 5.20, Figure 5.21) can be acquired with the Kalman smoother system, in average, the outlier value that can be detected from the landmark detection measurement is around 0.8 meters by EKF and 0.4-0.65 meters by KS. These outliers are around 4-5 times higher than the measurement noise which means this system can detect outliers within the measurement much lower than that with the EKF method, especially during the period when EKF faces poor geometry and has very high MDB values.

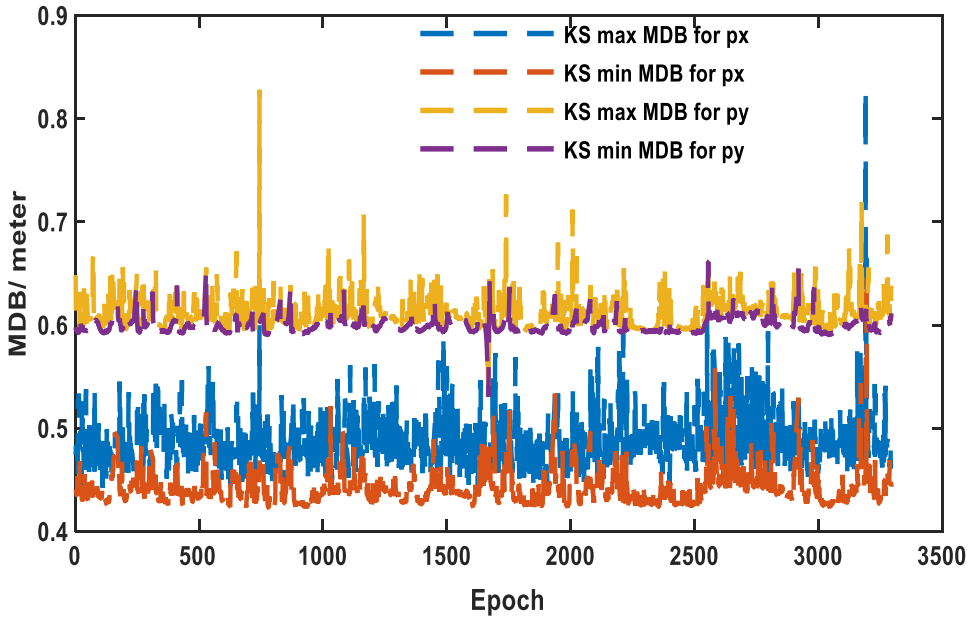


Figure 5.20 Minimum detectable bias (MDB) results for the landmark detection models px py measurements by KSfor the DLR dataset

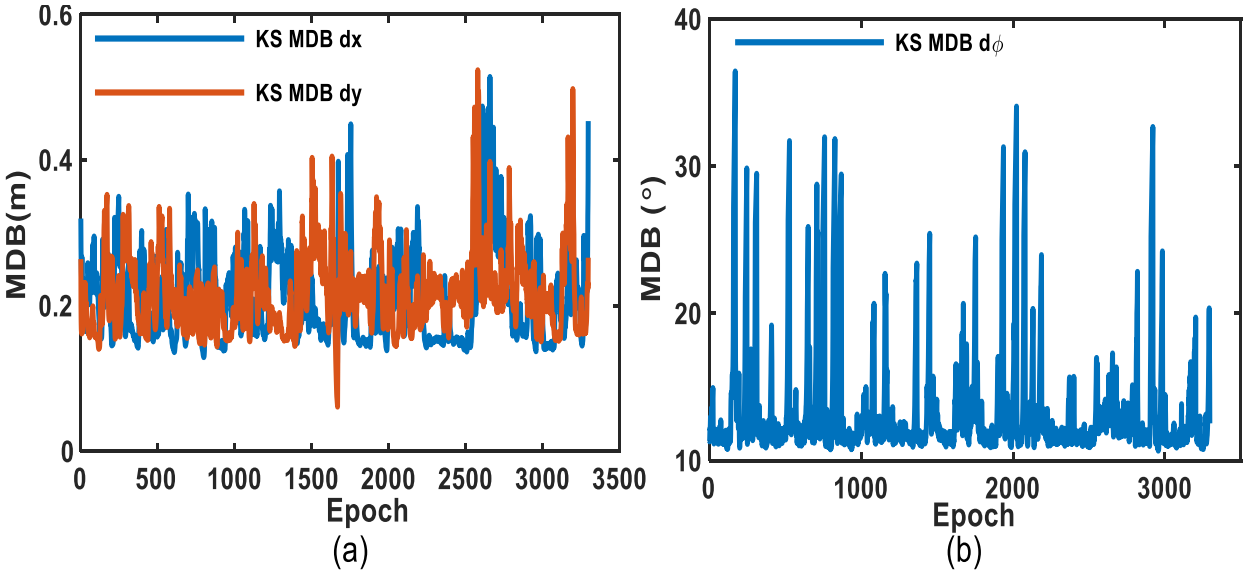


Figure 5.21 Minimum detectable bias (MDB) results for the robot odometry models of (a) distance dx and dy (a) and (b) turn angle dϕ measurement by KS for the DLR dataset

The External Reliability results of the smoothing based system are more stable and better with ER value (Figure 5.22) 2-5 times lower than that acquired by the EKF method. It is because the Full SLAM (least squares based or Kalman smoothing based) SLAM uses all

the measurement information and geometric constraints (loop closures). Therefore, it is less sensitive to the outlier and the poor geometry at a single epoch.

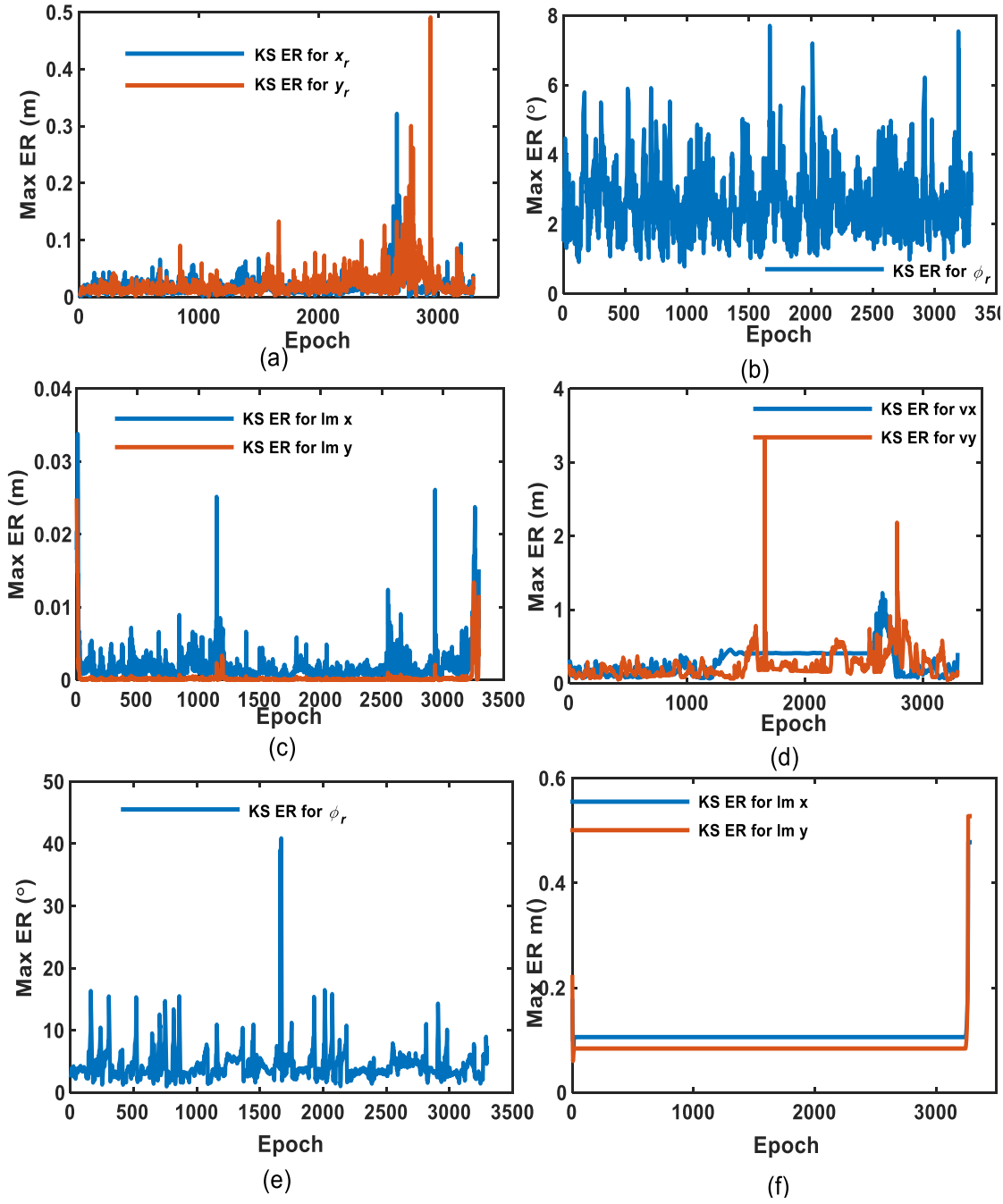


Figure 5.22. The External Reliability results: Influence of outlier with MDB value within landmark detection measurement upon the final vehicle (a) position, (b) heading and (c) landmark lm1 position estimation; Influence of outlier with MDB value within the robot odometry model upon the final vehicle (d) position, (e) heading and (f) landmark lm1 position estimation of the DLR dataset by KS SLAM

As aforementioned, when a landmark is not detected at one epoch for the EKF SLAM, the outliers within the measurements and predicted states at this epoch will not influence this landmark. However, for KS SLAM, this landmark will be influenced by other epochs' measurements. Here landmark lm1 is used to show this influence (Figure 5.22c and Figure 5.22f). Over the whole trajectory period, lm1 is only being detected around epoch 5-14, and epoch 2344-2395, therefore, it is found, during the period it is not viewed, its estimation is still influenced by the outliers. The influence of outlier within the odometry model will keep stable during the invisible period.

5.5 Summary

This chapter has proposed a new Kalman smoothing based SLAM algorithm which can conduct SLAM estimation and quality control recursively. A simulated numerical case study has shown the relationship between the EKF SLAM, KS SLAM and graph SLAM. The numerical results prove the equivalence between the KS SLAM and the ULS aided graph SLAM. The quality control results for this simulated case study also shows KS and graph SLAM can achieve same quality control results. These two methods can estimate the possible outlier value more accurate than EKF SLAM. By correcting the influence of the estimated outlier, KS/graph SLAM will achieve better estimation results. Therefore, the KS SLAM can replace the batch updated graph SLAM to evaluate the quality of the SLAM system and dataset, since it does quality control recursively and is much efficient.

According to the analysis with two real-word datasets, the smoothing based method has better Internal Reliability and External Reliability results than the filter-based method. The influence of undetected bias within the landmark detection and odometry measurement upon the vehicle/ robot related variables is lower and more stable. However, for the landmark position estimation, the EFK results show no impact of odometry error when a landmark is not detected at a current period. The smoothing results are different, since all the observation and odometry information are used together, and since the landmarks are correlated to each other, the outlier within the odometry measurement at one epoch will affect the final estimation of all the stored landmarks.

For the case of Victoria Park dataset test, although the smoothing based SLAM can enhance the quality of the extended Kalman filter based SLAM at poor geometry periods, the performance of the EKF based SLAM estimation results may be acceptable at most of the period according to the requirement of SLAM applications. It is because during the EKF process of Victoria Park dataset, there are many small loop closures that can frequently corrected the EKF estimation drift. The DLR dataset, however, contains a large loop. The Kalman smoothing method can significantly improve the estimation and quality of the full system with large loops.

The loop closure information applied in KS step is just same to the EKF at the same epoch, therefore, although the loop closure can correct the estimation of feature afterward during the EKF, the spatial relationship of feature and platform is not fed back to the previous epochs' measurement models by KS, but only the position information is fed back. The FLS SLAM in this thesis is also built up according to these spatial conditions at different time. In this research, the quality control on data association is not studied, which may need future study.

Chapter 6 Integration of High-definition Maps and Multi-Sensors for Localization- A Geometric Analysis

6.1 Introduction

Maps are often integrated with the sensors to provide road trend within a long distance. However the application of current commercial map is limited by its low accuracy and simple representation of road network. One of the emerging autonomous driving technologies is the High-definition map which is a reality-based map that provides high precision intelligence on the road features, and may be an optimal selection to fulfill the range and accuracy requirements of autonomous driving. Some mapping companies, such as HERE, TOMTOM, Google, Apple, have joined the study of HD map. HERE has already built up the mapping platform and provide HD mapping data to vehicle companies and research groups to test their highly automated vehicle (HERE, 2017). Unlike the coarse 2D maps currently used by vehicles, the HD map can provide detailed and precise 3D map of the road and surrounding environment and the map information is used by the navigation system, not by human driver. LiDAR, GPS/IMU, camera technologies are combined to generate the HD maps, therefore this kind of map can not only provide high density point cloud of the road environment with accurate position data, but also show the road network information, such as slope and curvature, lane marking types, roadside objects and intersection (HERE, 2017). The HD map can also be real-time or nearly real-time as it can be updated about the moving objects' (pedestrian or other car) information with the captured sensor data from the host autonomous vehicle. Hence, HD map shows high possibility of supporting very high accuracy vehicle localization and navigation ability for autonomous driving.

SLAM technique is currently one of the mostly studied strategies for autonomous driving with HD maps (Seif and Hu, 2016) as it can determine the location of the host vehicle and the relationship between the host vehicle and other entities around it at the same time. SLAM technique can also contribute to the update of the HD map in cloud with the real-

time road condition. However, SLAM requires high computing power and huge storage space. Real/near real-time map updating also depends on quick data transmission ability. Since the HD map is already containing very detailed road information, it shows potential of being utilized as a sensor to provide accurate position information of the static objects without very frequently updating. Therefore, a more reliable and lower computational cost vehicle localization strategy which using on-board HD map as a sensor is investigated in this study. Here, 'utilized as a sensor' means the map is only used to provide information, no map updating is conducted during this proposed HD map localization system. A perceptive sensor is used to detect landmarks and features around the vehicle, and the HD map will provide the position information of the detected features. As an initial study, the whole system will temporarily not use the GPS/IMU information. Least squares technique is used to perform vehicle localization.

Although HD map provides a vast amount of position information with high accuracy (10-20 cm) and can be used to estimate exact vehicle position, the study of vehicle localization with HD map is still in early stages. Mere localization computation without quality control is not enough to ensure acquiring expected lane level or even sub-lane level accuracy. The quality control technique is needed that contains outlier detection, reliability analysis and integrity monitoring which can be evaluated by reliability and separability (Wang and Knight 2012). The system reliability can be considered in two approaches, internal reliability and external reliability. The former one tests the ability of the system to detect a fault (MDB), while the latter one measures the influence of undetected fault upon the final solution. Wang and Knight (2012) have developed a new outlier separability test, which determines the minimum bias that can be separated for every pair of observations. In this study, the quality of the HD map based localization system will be evaluated by these quality control techniques.

The geometry of a positioning/localization system will significantly influence the final quality of the localization solutions. As the HD map is newly introduced and applied into the autonomous vehicle localization research field, the relationship between the geometry of the HD map based vehicle localization system and the quality of the results has not been well studied.

Therefore, this chapter aims to investigate a HD map-based vehicle localization system for highly automated driving. The influence of geometry to localization system is evaluated with different scenarios. The structure of this Chapter is as follows: Section 6.2 presents a brief introduction on mathematical models in vehicle localization with HD map and present the concept of Correlation Coefficient and MSB. Section 6.3 conducts a simulated analysis with different scenarios. Section 6.4 summarizes the geometry's influence on quality in HD map based localization and points out the problems that need to be explored further.

6.2 Mathematical Models for HD Map and Multi-Sensor Integration

Figure 6.1 shows the principle of the HD map based localization system. In the system, the vehicle is equipped with HD map to provide the position information of detected features, and also with an external perceptive sensor (laser) capable of measuring the relative position (for instance, d1- d6 in Figure 6.1) between the vehicle and its environmental features. The features could be trees, traffic light, traffic sign, buildings, etc. The mathematic relationship between the distance and 3D coordinates of vehicle and features can be used to estimate the vehicle position (x_{veh}, y_{veh}, z_{veh}) by least squares theory.

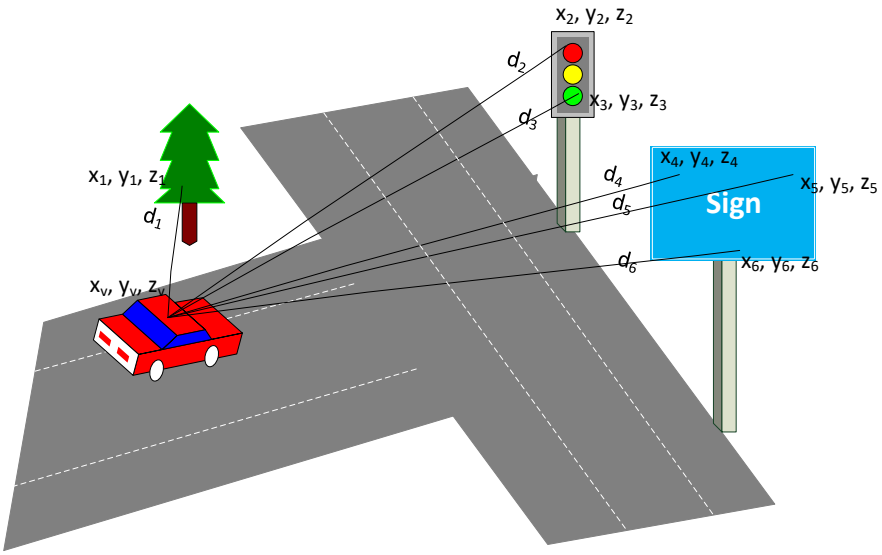


Figure 6.1The HD map based vehicle localization principle

The functional model for vehicle localization can be expressed as:

$$d_i = \sqrt{(x_i - \mathrm{x}_{veh})^2 + (y_i - \mathrm{y}_{veh})^2 + (z_i - \mathrm{z}_{veh})^2} \tag{6.1}$$

where x_i, y_i, z_i, i are the 3D coordinates and the identification of the features respectively. In order to estimate the position of the vehicle, more than 3 features are needed to be detected. x_{veh}, y_{veh}, z_{veh} are the 3D coordinates of the host vehicle. d_i is the distance between the vehicle and the feature (i) measured by the external sensor, such as laser. In this equation, the coordinates of the features are provided by the HD map and are treated as known values. The coordinates of vehicle are unknown.

Stochastic model represents the statistical properties of the errors in measurements and/or functional models by the variance-covariance matrix (VCV matrix). The stochastic model is:

$$D = {\sigma_0}^2 Q = {\sigma_0}^2 P^{-1} \tag{6.2}$$

where ${\sigma_0}^2$ is the priori variance factor which is set to one in this study. Q is the cofactor matrix, within which the diagonal elements have same values because only one type of measurement is tested, the off-diagonal elements is zeros. P is weight matrix and is the inverse of Q .

As using least squares to estimate the position information, the equation should be linearized and reasonable initial values (x_{veh0}, y_{veh0}, z_{veh0}) should be assigned to the vehicle coordinates. Therefore, for each captured feature, the model equation at approximate coordinates will be:

$$f_i = \sqrt{(x_i - \mathrm{x}_{veh0})^2 + (y_i - \mathrm{y}_{veh0})^2 + (z_i - \mathrm{z}_{veh0})^2} - d_i \tag{6.3}$$

Now, the design matrix A and l vector can be obtained by Equations 6.4 and 6.5:

$$A = \begin{pmatrix} \frac{\partial f_1}{\partial x_{veh0}} & \frac{\partial f_1}{\partial y_{veh0}} & \frac{\partial f_1}{\partial z_{veh0}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_{veh0}} & \frac{\partial f_n}{\partial y_{veh0}} & \frac{\partial f_n}{\partial z_{veh0}} \end{pmatrix} \tag{6.4}$$

$$l = (-f_1, \dots -f_n)^T \tag{6.5}$$

where *n* is the number of features.

Then, the correction *v* of the coordinates can be estimated by the following equation:

$$v = AX - l \tag{6.6}$$

here X is the estimated corrections of the initial vehicle coordinates.

Least squares quality control method presented in Chapter 2 is used to test this HD map based localization problem. Further-more, when two outlier detection statistics, w_k and w_i, are tightly correlated, outlier in one measurement may influence the analysis of other measurement. Therefore, the correlation coefficient of every two measurements' outlier detection statistics should also be taken into consideration when doing outlier detection. The correlation coefficient between the k^{th} and i^{th} outlier detection statistics can be obtained as (Förstner, 1983):

$$\rho_{ik} = \frac{{h_k}^T P Q_v P h_i}{\sqrt{{h_k}^T P Q_v P h_k}\sqrt{{h_i}^T P Q_v P h_i}} \tag{6.7}$$

As aforementioned, two outlier detection statistics may have high correlation coefficient value which will cause contamination of "good" measurement. This contamination may confuse the outlier detection statistics and manifest the "good" measurement as an outlier. (Hewitson and Wang,2006). Therefore, it is important to separate the contaminated and the "good" measurements so as to eliminate the influence of the outlier. Separability can be quantified by MSB, which stands for Minimal Separable Bias and is determined by:

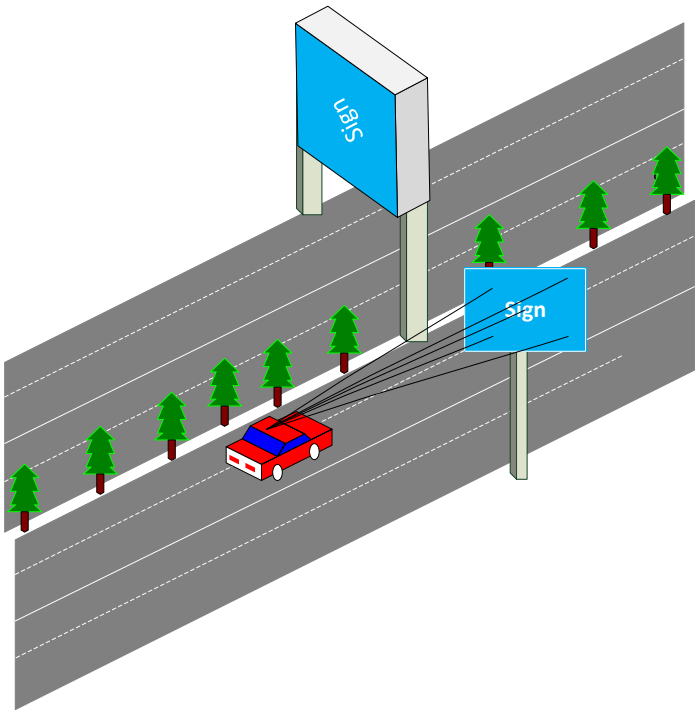
$$MSB_{ik} = \frac{\delta_s \sigma_0 \sqrt{2}}{\sqrt{{h_i}^T P Q_v P h_i (1 - |\rho_{ik}|)}} \tag{6.8}$$

where δ_s is the mean shift of the separability statistics determined by Type I and Type II errors (Wang and Knight, 2012). ρ_{ik} is obtained by Equation 6.7.

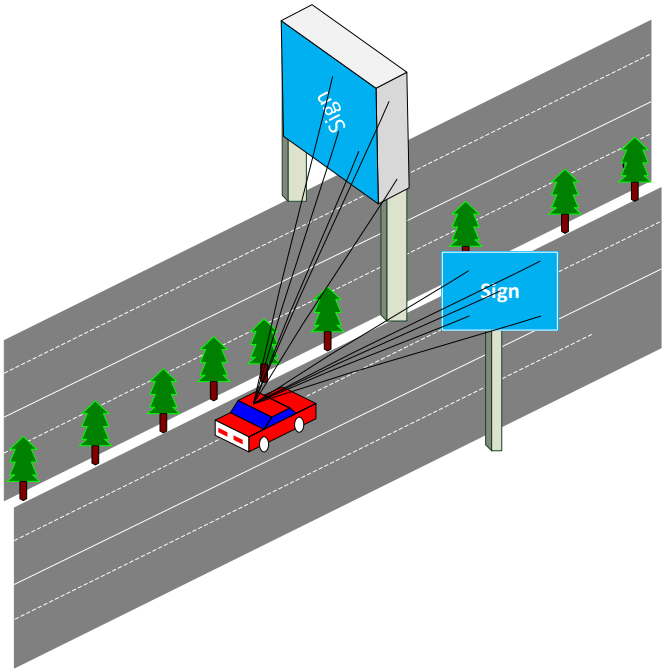
6.3 Experiments and Analysis

6.3.1 Different Vehicle Localization Scenarios

Two different scenarios were simulated and tested in order to analyse the influence of geometry on the estimation's quality. For both scenarios, the red vehicle is the host vehicle driving on the road. There are road signs, trees and buildings along the road and they may be on one side or both sides of the road. In Scenario 1, the detected features are on one side of the road (Figure 6.2a), and in Scenario 2, there are features from both road sides to be detected (Figure 6.2b). For each scenario, three factors (feature distribution type, feature number, and distance between the host vehicle and the feature) were taken into consideration that may affect the geometry of vehicle localization system. There are four types of feature distributions, including centrally distributed with features closed to each other (sub-meter level) (Figure 6.3a), distributed features with similar height level (Figure 6.3b), distributed features with similar horizontal level (Figure 6.3c), and randomly distributed (various horizontal and height level, meter level depart with each other) (Figure 6.2). In Scenario 2, the pair of features on one side of road can be treated as randomly distributed to the pair of features on another side, also, when analysing the influence of feature distribution type, this study considers that the features have the same type of distribution at each side. In each scenario, different number (5, 10, 15) of detected features will be tested. For scenario 2, the number of features on different side was set to be similar, for example, the left side has 7 features, right side has 8, therefore, the total number is 15. As assuming the external sensors on the vehicle can have a sensing range up to 150 m (such as Laser), we consider the situation that the vehicle detects some features with different distances from 150 m to 15 m.

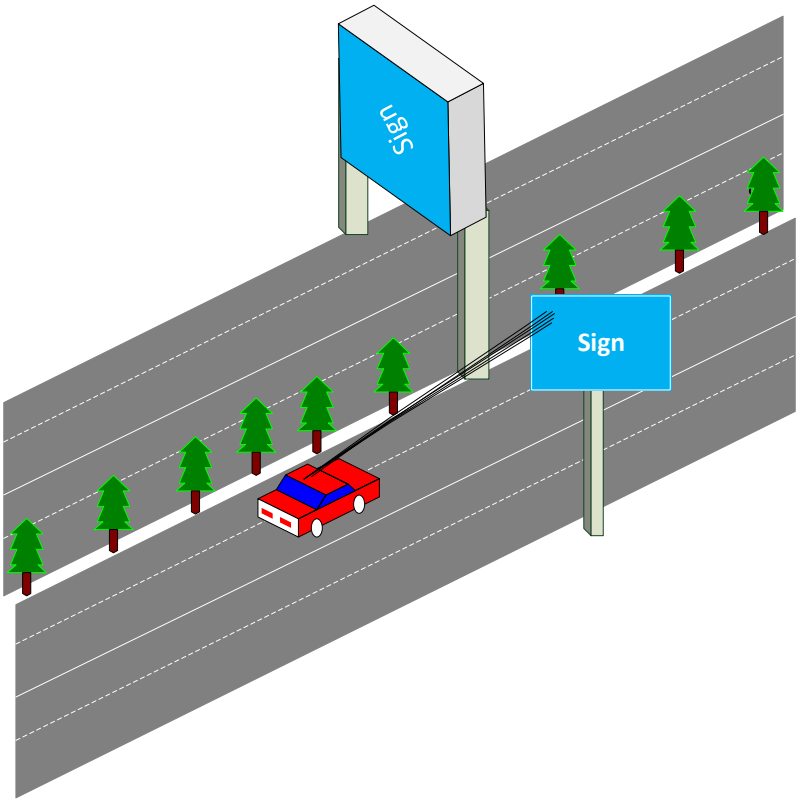


(a)

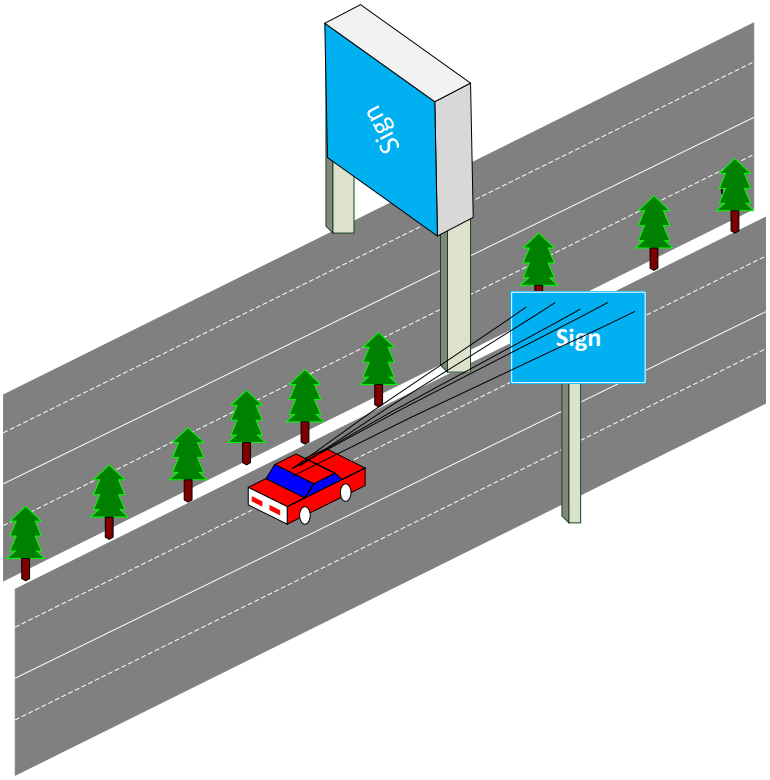


(b)

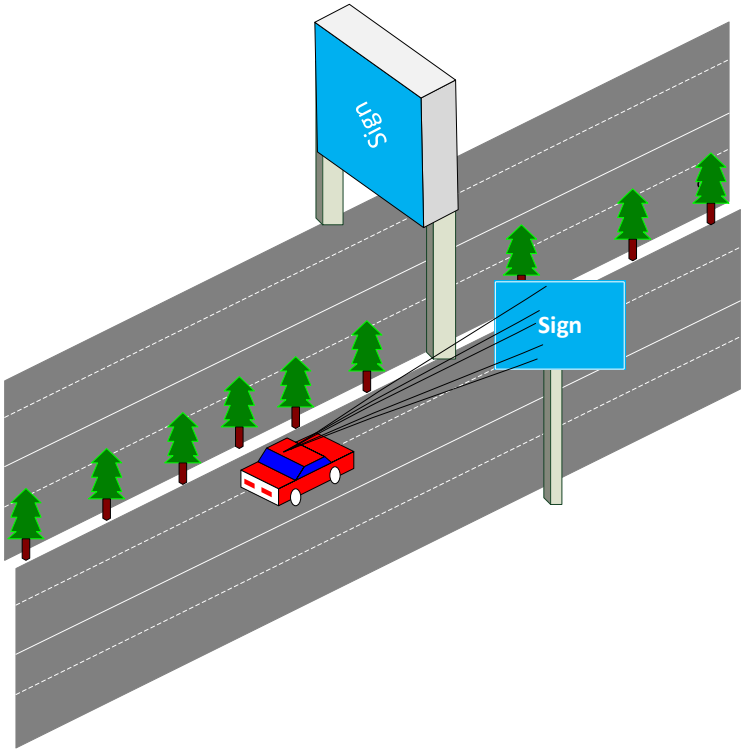
Figure 6.2 (a) Scenario 1: vehicle detects features from one side of the road; (b) Scenario 2: vehicle detects features from both sides of the road



(a)



(b)



(c)

Figure 6.3 (a) The detected features are clustered and centralized; (b) Features are distributed with similar height; (c) Features are distributed with similar horizontal level

Figure 6.4 shows the overall simulated trajectory of the vehicle and the distribution of feature for different types of distribution. For different feature number cases and different scenarios, the corresponding feature set can be selected from these features.

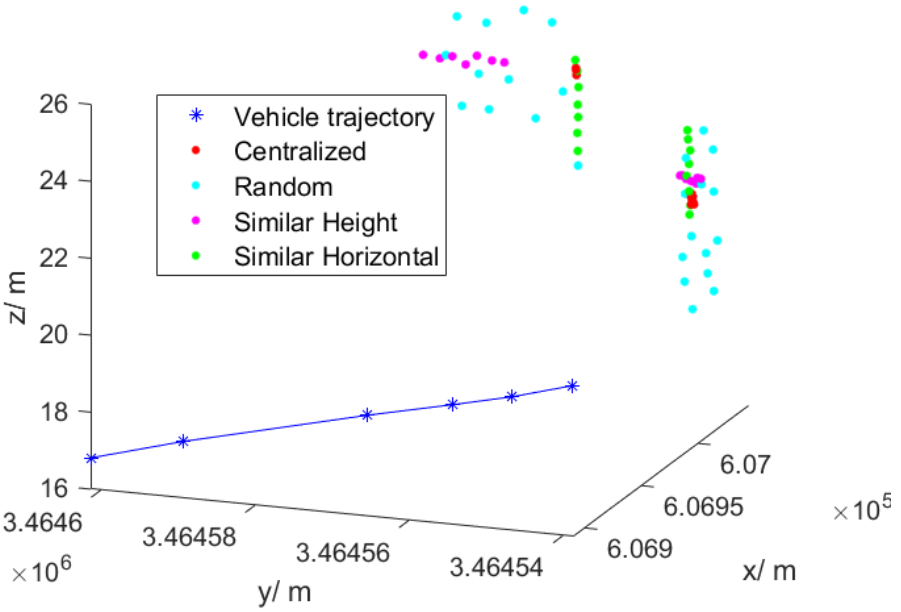


Figure 6.4 Vehicle trajectory and feature distribution with 4 different distribution types

6.3.2 Outlier Detection Statistics Analysis

For the majority of the laser scanners currently used in mobile device research, the distance accuracy is at centimeter level (Olsen et al., 2012), for instance, Velodyne LiDAR sensor has accuracy of less than 2 cm (Velodyne LiDAR, 2017). Therefore, in the simulated environment, the noise on the detection of feature was modeled as random error with a standard deviation of 2 cm. An example of Scenario 1 was introduced to illustrate the outlier detection test, Correlation Coefficient, MDB, and MSB. In this example, there are 5 features detected with a centralized distribution type, the distances between the features and vehicle are about 50 meters.

As for measurement 1, the minimal detected bias is 0.134m (Table 6.1), if an outlier of 0.05 m and 0.5 m is introduced into measurement 1, respectively, the outlier detection test results are shown in Table 6.2. It can be seen that the *w* value of measurements are all lower than 3.29 when the outlier of 0.05 m is lower than MDB value. Therefore, the system cannot identify the added outlier. For the outlier of 0.5 m, both measurement 1 and measurement 3 are possible of containing outliers with a high *w* value (16.991), and these two cannot be confidently separated, which may due to the high correlation between them (Table 6.3). According to Table 6.3, the pair of measurement 2, 4 and 5 is also not separable as they have correlation coefficient close to 1. Therefore, the Minimal Separable Bias (MSB) should be taken into consideration (Table 6.4). As the minimal separable bias to separate measurement 1 from measurement 3 is 2.823 m, an outlier higher than 2.823 m is introduced to measurement 1, and the outlier detection statistics test result is shown in Table 6.5, w_1 is the largest one, indicating that the outlier is in measurement 1. This identified outlier is corresponding with the true outlier.

Table 6.1 MDBs for all measurements in Example 1

Measurement ID	1	2	3	4	5
MDB (m)	0.134	0.143	0.105	0.106	0.310

Table 6.2 Outlier statistics for added outlier of 0.05 meters and 0.5 meters in measurement 1

Outlier (m)	**Measurement ID**				
	1	2	3	4	5
0.05	2.500	1.731	2.308	2.243	1.762
0.5	16.991	1.964	16.991	1.619	1.765

Table 6.3 Correlation Coefficients between outlier detection statistics

Correlation Coefficients ρ_{ik}		**Measurement ID**				
		1	2	3	4	5
Measurement ID	1	1.0000	0.0683	-0.9955	0.1422	0.0565
	2		1.0000	-0.1629	-0.9778	0.9999
	3			1.0000	-0.0475	-0.1512
	4				1.0000	-0.9802
	5					1.0000

Table 6.4 MSB (Minimal Separable Bias) between each pair of measurements

MSB_{ik} (m)		**Measurement ID**				
		1	2	3	4	5
Measurement ID	1	0	0.1968	2.8228	0.2052	0.1956
	2	0.2093	0	0.2208	1.3559	24.2005
	3	2.2154	0.163	0	0.1528	0.1619
	4	0.1624	1.0099	0.1541	0	1.0695
	5	0.4512	52.5127	0.4757	3.1158	0

Table 6.5 Outlier statistics for an outlier of 3 meters in measurement 1

Outlier (m)	**Measurement ID**				
	1	2	3	4	5
3	92.839	7.400	92.519	12.153	6.307

6.3.3 Geometry Analysis

6.3.3.1 Feature Distribution Type

When analysing the influence of feature distribution type on the quality of position estimation, 5 features were set to be detected with road sign about 40 meters ahead of the vehicle. For Scenario 1 and 2, when comparing the average value of MDBs of each measurement with different distribution types (Table 6.6), it can be found that each distribution type has a similar mean MDBs value, errors in the measurements that are larger than 13 to 15 cm can be detected by the outlier detection test. Some measurements in Scenario 2 have relatively low redundancy numbers in the Similar Height distribution case, which cause the average MDB values in this case slightly higher than that in Scenario 1. As MSB can be used to quantify the minimal bias that could be separated for each two measurements, it is observed that the distribution type does not have significant effect on MSB, similar to MDB. Decimeter to meter level bias in one measurement can be successfully separated from another measurement.

Table 6.6 Average values of MDBs contributed by features with different distribution types

Average MDB (m)		**Distribution Type**			
		Centralized	*Similar Height*	*Similar Horizontal*	*Random*
Scenario	1	0.137	0.140	0.151	0.142
	2	0.136	0.156	0.136	0.139

However, when comparing the external reliability (Table 6.7) for Scenario 1, the first three distribution types have very high mean external reliability values on all 3D coordinates, which means the minimal detected bias in the measurements can significantly influence the final position estimation, especially when the detected features are clustered together with very close distance (sub-meter level), the maximum external reliability values for 3D coordinates are 44.763 m, 32.094 m, and 89.880 m, respectively, much higher than the results from other distribution types. For the results in the second column (Similar height case), the external reliability value of z coordinate is far higher than that of x and y coordinates, while in the third column (Similar horizontal case) has opposite results, indicating that in order to get reliable position estimation for autonomous driving, the detected 5 features should not have very similar height level or horizontal level. This can also be demonstrated by the results of random feature distribution case, where the mean external reliability values are all less than 1 meter.

Table 6.7 External Reliability contributed by MDBs values with different feature distribution types

Distribution Type / **Coordinates**		**Scenario 1**				**Scenario 2**			
		Centralized	*Similar Height*	*Similar Horizontal*	*Random*	*Centralized*	*Similar Height*	*Similar Horizontal*	*Random*
X (m)	average	25.254	14.150	4.784	0.486	3.400	0.712	0.788	0.151
	max	44.763	23.371	10.356	1.170	4.534	2.235	1.369	0.351
	min	9.298	6.204	0.742	0.132	0.436	0.017	0.075	0.040
Y (m)	average	17.844	1.509	5.478	0.530	0.457	0.286	0.491	0.082
	max	32.094	3.892	10.165	0.948	0.712	0.654	0.975	0.121
	min	6.256	0.077	1.189	0.116	0.005	0.076	0.025	0.062
Z (m)	average	60.770	66.960	2.724	0.863	21.852	4.787	5.390	1.238
	max	89.890	106.930	6.193	1.420	29.265	14.486	9.521	2.655
	min	21.291	15.167	0.285	0.292	2.728	0.086	0.435	0.571

The external reliability results for Scenario 2 are similar to those of Scenario 1 that random feature distribution has best estimation quality. The influence of undetected bias on the horizontal coordinate estimation is much lower in Scenario 2 due to the large horizontal distance between the road signs. However, for the last two feature distribution types, the external reliability of vertical coordinate estimation is higher than that in Scenario 1, which indicates in this case that detected features from different road sides can improve the quality of horizontal coordinates estimation, but not vertical. The reason of this phenomenon is mainly due to the shape of the left sign. In our test, the left sign is much narrower than the right sign. Most of the randomly distributed or Similar Horizontal distributed features detected in left sign have height range around 22-24 meters, which is less diversity than that in right sign (19-24 meter). When combining the two signs as feature source, the variation on feature height is smaller than choosing the same number of feature only from the right sign.

6.3.3.2 Feature Number

The influence of feature number on the reliability and separability was analysed with various feature numbers from 5 to 15. For Scenario 1, it is observed that with more features being detected, the average MDB values and MSB values are getting smaller. More detected features can also reduce the influence the undetected bias onto the final position estimation (Table 6.8 and Table 6.9), which may be due to the higher redundancy of the system that makes the system more reliable. When comparing Table 6.8 and Table 6.9, it can be found that random distribution has surpassed centralized feature distribution type. If the features are centrally distributed, even when 15 features are detected by the sensor at this vehicle-feature distance (40 m), the undetected bias in the measurement may still cause meter level influence on position estimation, while for the random feature distribution, 10 features are enough to limit the influence of the undetected bias within the sub-lane level. Scenario 2 has similar results of Scenario 1 and has better estimation quality for horizontal coordinate estimation. When the features are randomly distributed, 5 features being detected are enough to acquire system external reliability at sub-lane level. Similar in Section 6.3.3.1, for vertical estimation, only detected feature from one side is better than from both sides.

Table 6.8 External Reliability contributed by MDBs values with different feature numbers when centrally distributed

Feature Number / Coordinates		*Scenario 1*			*Scenario 2*		
		5	10	15	5	10	15
X (m)	average	25.254	3.900	1.655	3.400	0.428	0.339
	max	44.763	13.263	5.703	4.534	1.038	1.091
	min	9.298	0.089	0.387	0.436	0.008	0.0001
Y (m)	average	17.844	3.202	1.555	0.457	0.073	0.063
	max	32.094	9.847	4.638	0.712	0.147	0.153
	min	6.256	2.106	0.206	0.005	0.008	0.005
Z (m)	average	60.770	7.060	2.409	21.852	2.801	2.236
	max	89.880	24.907	8.547	29.265	6.741	7.016
	min	21.291	0.815	0.077	2.728	0.205	0.024

Table 6.9 External Reliability contributed by MDBs values with different feature numbers when randomly distributed

Feature Number / Coordinates		*Scenario 1*			*Scenario 2*		
		5	10	15	5	10	15
X (m)	average	0.486	0.170	0.102	0.151	0.047	0.027
	max	1.170	0.293	0.169	0.351	0.095	0.073
	min	0.132	0.056	0.029	0.040	0.011	0.007
Y (m)	average	0.530	0.171	0.106	0.082	0.022	0.014
	max	0.948	0.350	0.182	0.121	0.051	0.029

	min	0.116	0.049	0.023	0.062	0.002	0.004
Z (m)	average	0.863	0.247	0.153	1.239	0.337	0.193
	max	1.420	0.515	0.275	2.656	0.719	0.522
	min	0.292	0.093	0.048	0.571	0.040	0.031

6.3.3.3 Detection Distance

MDB and MSB values change slightly when the vehicle has different distances to the features (Table 6.10). However, when the vehicle is far from the features, the external reliability values are large. It is interesting to find that when the vehicle is close (in this case, less than 20 meters) to the detected features, the External Reliability values are significantly larger than that at other distance (Table 6.11). This phenomenon can be found in all cases tested in Scenario 1. The reason may be that the distance would influence the redundancy number of each measurement. When the vehicle is close to the features, the redundancy number values of some measurements are less than 0.1 or very close to 0.1, which indicates very large MDB value for those measurements and will cause high external reliability. This influence of close distance on position estimation quality has been improved in Scenario 2 (Table 6.12), when the features are detected from both sides of the road. In that case, the vehicle has different distances to the two road signs at x and y directions, respectively, the redundancy number values for each measurement are all at an acceptable range.

Table 6.10 The average MDBs value contributed by features with different distance

Feature Distribution Type and Number			**Vehicle Distance (m)**					
			17	25	40	60	114	141
Scenario 1	Centralized feature	5	0.194	0.137	0.137	0.137	0.137	0.137
		10	0.111	0.102	0.102	0.101	0.101	0.101
		5	0.142	0.144	0.143	0.142	0.142	0.142

	Random distributed feature	10	0.010	0.099	0.099	0.099	0.099	0.099
Scenario 2	Centralized feature	5	0.143	0.136	0.136	0.136	0.137	0.137
		10	0.104	0.108	0.110	0.111	0.113	0.113
	Random distributed feature	5	0.144	0.138	0.139	0.139	0.139	0.139
		10	0.103	0.099	0.099	0.099	0.099	0.099

Table 6.11 External Reliability contributed by MDBs with different vehicle-feature distance in Scenario 1 (5 Features)

Coordinates		**Vehicle-Feature Distance (m)**											
		Centrally Distributed						*Randomly Distributed*					
		17	25	40	60	114	141	17	25	40	60	114	141
X (m)	ave	234.037	25.777	25.254	28.464	37.449	42.162	6.052	0.461	0.486	0.586	0.837	0.964
	max	482.362	46.012	44.763	50.294	66.064	74.360	14.182	1.178	1.170	1.342	1.811	2.055
	min	15.642	6.629	9.298	12.835	20.318	23.966	0.894	0.139	0.132	0.148	0.198	0.225
Y (m)	ave	74.312	8.927	17.844	30.694	60.431	76.088	1.735	0.307	0.530	0.866	1.655	2.073
	max	144.788	16.269	32.094	54.841	107.422	135.093	4.299	0.506	0.948	1.595	3.112	3.915
	min	4.907	1.570	6.256	12.755	27.684	35.537	0.116	0.070	0.116	0.186	0.352	0.440
Z (m)	ave	20.463	41.487	60.770	91.988	165.625	204.399	0.344	0.570	0.863	1.330	2.428	3.007
	max	60.754	63.302	89.880	135.843	244.103	301.070	0.738	0.955	1.421	2.164	3.927	4.858
	min	1.246	15.593	21.291	31.217	54.994	67.567	0.076	0.190	0.292	0.452	0.827	1.025

Table 6.12External Reliability contributed by MDBs with different vehicle-feature distance in Scenario 2 (5 Features)

Coordinates		Vehicle-Feature Distance (m)											
		Centrally Distributed						*Randomly Distributed*					
		17	25	40	60	114	141	17	25	40	60	114	141
X (m)	ave	2.994	2.909	3.400	4.076	5.422	6.079	0.027	0.144	0.151	0.167	0.203	0.223
	max	4.823	4.152	4.534	5.452	7.605	8.631	0.073	0.310	0.351	0.405	0.525	0.588
	min	0.689	0.496	0.436	0.396	0.375	0.380	0.007	0.050	0.040	0.033	0.021	0.015
Y (m)	ave	2.487	1.213	0.457	0.829	4.136	5.940	0.014	0.066	0.082	0.139	0.271	0.340
	max	3.956	1.809	0.712	1.251	5.998	8.653	0.029	0.106	0.121	0.234	0.616	0.815
	min	0.538	0.162	0.005	0.168	0.415	0.517	0.004	0.00722	0.062	0.061	0.052	0.048
Z (m)	ave	5.042	13.092	21.852	35.535	67.899	84.994	0.193	0.811	1.238	1.888	3.394	4.184
	max	7.936	18.722	29.265	46.473	93.046	117.848	0.522	1.693	2.655	4.023	7.121	8.735
	min	1.167	2.213	2.728	3.269	4.142	4.523	0.031	0.332	0.571	0.928	1.759	2.195

6.4 Summary

High-definition map can be used to act as an on-board sensor and assist autonomous vehicle localization with high accuracy. The geometry of HD based vehicle localization system can significantly affect the quality of position estimation of vehicle. Therefore, the analysis of geometry influence can provide reference for the further development of vehicle localization system to acquire appropriate geometry components. To meet the requirement of accuracy and reliability, factors like feature distribution type, number of detected feature and the distance between the vehicle and the features should be taken into consideration together. MDB and MSB are mostly influenced by the last two factors.

Generally, random distribution of features achieves better results. When the vehicle is close to the features, double-road-side feature detection is required to guarantee suitable

external reliability. It is obvious that more features being detected can enhance the quality of position estimation, however, it also leads to higher computational burden.

Chapter 7 LiDAR/GNSS/INS Integrated Mapping and Localization with Quality Control

7.1 Introduction

With the high demand of localization and navigation information such as high accuracy, high precision, long perception range, high data rate and within all kinds of road environment, standalone localization and navigation technique can hardly meet all these strict requirements for practical applications of autonomous driving. Light Detection and Ranging (LiDAR) system can obtain high density 3D point cloud with high sensing rate and high accuracy, which can provide high-quality localization and navigation information. The LiDAR based Simultaneous Localization and Mapping (SLAM) technology is widely studied and used for the robotics fields. Normally, the SLAM system contains estimation errors which may increase over time and distance travelled, thus it needs loop closure to correct the errors. Loop closure is much easier to form in indoor applications, or small-outdoor applications. However, it is hard to achieve in some large-scale outdoor applications, such as for driving on a highway, or long one-way trajectory in urban area, without a chance for loop closure. In addition, the LiDAR-only SLAM will only provide the relative localization information, and thus cannot provide the absolute localization information. Therefore, the combination of GNSS/INS to the LiDAR SLAM will effectively reduce the dependence of loop closure and provide the absolute position information. Furthermore, a LiDAR system can also support localization using existing HD maps when GNSS signal is not available.

Computation efficiency is one of the main factors that limit the practical application for autonomous driving. The high-quality map, such as the High Definition (HD) map can be generated offline by utilizing all the available sensor data and road information, which is less limited by the efficiency. However, for some online application, such as localization during driving, or HD map updating with real-time road detection results,

there may be not enough computation ability and storage size to support online SLAM for a long driving. As discussed in Chapter 6, a pre-generated HD map can be used as sensor source to support online Localization, a pre-generated map aided LiDAR/GNSS/INS localization system is proposed in this Chapter. An efficient dynamically loading method of pre-generated global map is utilized to reduce the estimation calculation burden.

Another issue that associated to the application of localization and navigation for autonomous driving is the lack of quantitative evaluation of the estimation quality of the system. Some criteria results are hard to achieve for the real-word application. For instance, the ground truth of road map or an urban driving trajectory is mostly not available for the case of autonomous driving, especially the ground truth of a dense map for the complex road environment. Therefore, the accuracy of the localization and mapping system cannot be properly evaluated. There are some other qualities that may need to evaluate in order to ensure the safety of driving without knowing the accuracy, which can be monitored by the proposed quality control method.

In this Chapter, a modernized SLAM procedure combines Light Detection and Ranging (LiDAR)/ Global Navigation Satellite System (GNSS)/ Inertial Navigation System (INS)/High Definition (HD) map as a sensor is proposed and implemented. The whole procedure of offline mapping and online localization is represented. Quality control is undertaken within this integration system to monitor the quality of the mapping and localization process. The structure of the Chapter is as follows: Section 7.2 displays the experiment setup of a road driving test. In Section 7.3, the procedure of proposed offline LiDAR/GNSS/INS mapping system is explained, and the generated road map is shown. In Section 7.4, the pre-generated road map is utilized to demonstrate the proposed online LiDAR/GNSS/INS localization system with analysing the system quality.

7.2 Experiment setup

Land vehicle tests were conducted in Sydney to test the proposed LiDAR/GNSS/INS/HD map integration system. The vehicle was equipped with a VLP-16 LiDAR, a IMU sensor

and two GNSS antennas (Figure 7.1). The second antenna is also active as the function of dual antenna aided heading update is chosen by the online localization system. The sampling rate of LiDAR was 10 Hz, the sample rate of GNSS was 1 Hz, and for IMU it was 100 Hz.

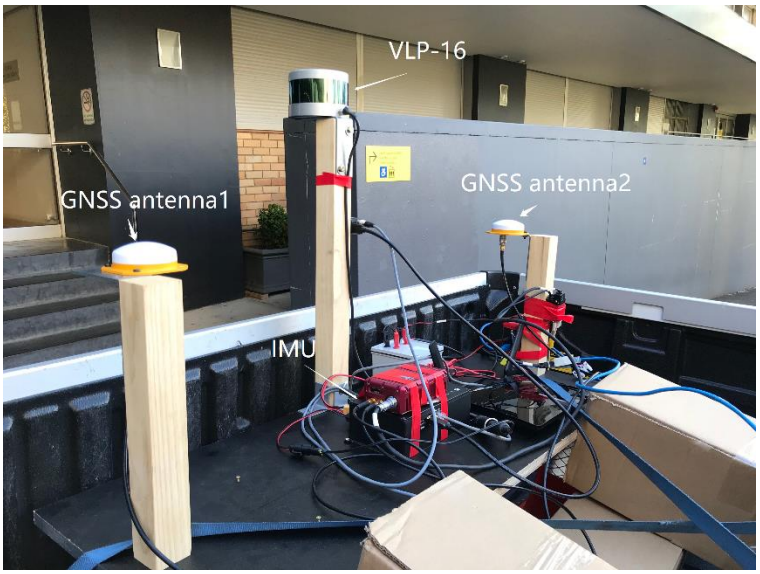


Figure 7.1 Experimental platform for road test

A road test was carried out in urban areas of Sydney, Australia. The trajectory is shown in Figure 7.2. The vehicle was driving from the University of New South Wales (UNSW) Kensington Campus to La Perouse (Section A), and then driving back to UNSW (Section B). In this study, the forward driving from UNSW to La Perouse was used to produce high precision 3D point cloud map of the road, and the backward driving from La Perouse to UNSW was used to test the performance for LiDAR/3D point cloud map-based localization method.

Figure 7.2 The road test trajectory (in blue) on Google map

In order to do quantitative analysis of the localization performance, 3 sections of the whole trajectory were selected (Figure 7.3), which is same to the selected sections used in Chapter 3. For each of the three selected sections on the driving trajectory, the GNSS RTK status for the forward driving and backward driving were both integer-ambiguity fixed, therefore the offline mapping results are expected to be accurate at about 5cm, and on the backward driving from La Perouse to UNSW, the selected sections will have highly

accurate GNSS/INS positioning results as a reference to evaluate the performance of the LiDAR/3D point cloud map-based localization method.

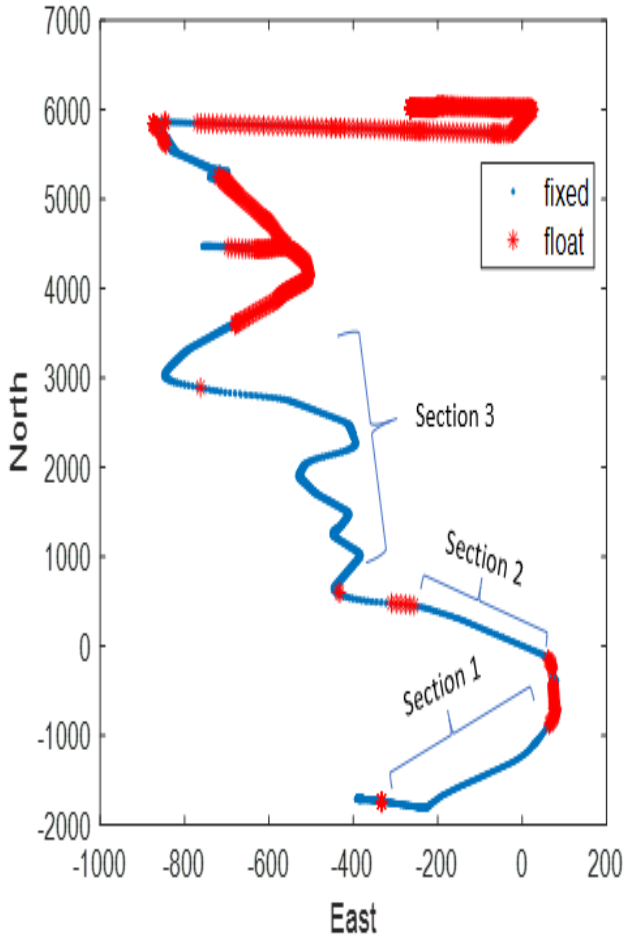


Figure 7.3 GNSS/INS localization of the whole trajectory with RTK positioning status

7.3 LiDAR/GNSS/INS Mapping with Road Test Dataset

7.3.1 Basic Technique used for Mapping

The acquired dataset of Section A was used to generate a georeferenced global point cloud map for the road environment. The georeferenced mapping was done based on LiDAR odometry frame-to-frame matching and GNSS/INS integration. Figure 7.4 shows the overview of the offline mapping system architecture.

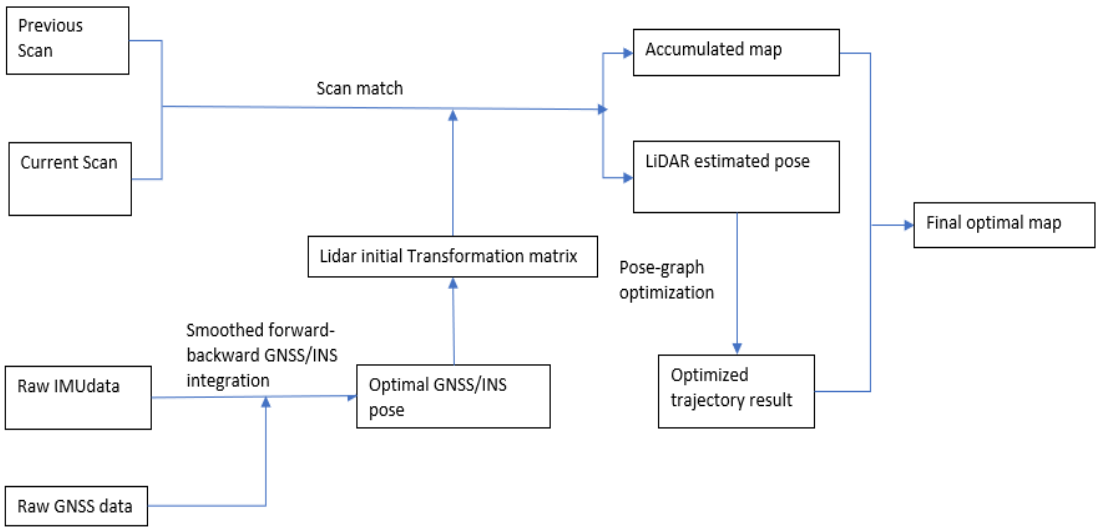


Figure 7.4 Overview of the proposed LiDAR/GNSS/INS mapping system architecture

7.3.1.1 GNSS/INS system and georeferencing

The GNSS/INS system can provide the geodetic positioning and attitude information. Since this map building was done offline, an optimal GNSS/INS trajectory can be obtained with a smoothed GNSS/ INS integration. The position and attitude results of GNSS/INS were used as the initial value of the frame-frame matching to transfer the newly merged point cloud to the referenced frame. Therefore, the point cloud can be georeferenced. Once the GNSS dataset is lost during driving, the Inertial navigation 6-DOF pose results can be used to generate the initial transformation until the GNSS signals come back.

There are several coordinate systems involved in this LiDAR/GNSS/INS system and need to be considered when doing georeferencing. The information from different sensor should be fused by different coordinate systems:

1) Vehicle body frame (b-frame): the testing vehicle's body frame, with X-forward, Y-right and Z-down pointing orientations.
2) Inertial Coordinate frame (i-frame): in this test the inertial frame is align to the body frame.
3) e-frame: ECEF(Earth-Center-Earth-Fix) frame is a Cartesian coordinate system with origin point as the center of mass of Earth.

4) n-frame: local navigation reference frame, with E-N-U (East-North-Up) pointing orientations.

5) LiDAR coordinate system (L-frame): the LiDAR coordinate system is moving with the vehicle. In this case study, y axis is toward the back of the vehicle, x axis for left and z axis for up direction of the vehicle.

The geo-referencing formula that describes the transformation of the LiDAR point cloud from the laser scanner frame to the ECEF frame will be:

$$p_{ecef} = R_n^{ecef} R_i^n(r,p,h)\left(R_L^i(\varphi,\omega,\kappa)\right)p_L + LA_l^l) + p_n^{ecef} \quad (7.1)$$

Here p_{ecef} is the positioning vector of the estimated current point in the ECEF frame; R_n^{ecef} is the rotation matrix from the navigation frame to the ECEF frame;

R_i^n is the rotation matrix from inertial frame to the local navigation frame with r, p, h the roll, pitch, heading angle providing by GNSS/INS system;

R_L^i is the boresight matrix that rotate the LiDAR frame to the navigation frame with φ, ω, κ the boresight angles between the laser scanner and the IMU sensor;

p_L is the position of points in LiDAR frame

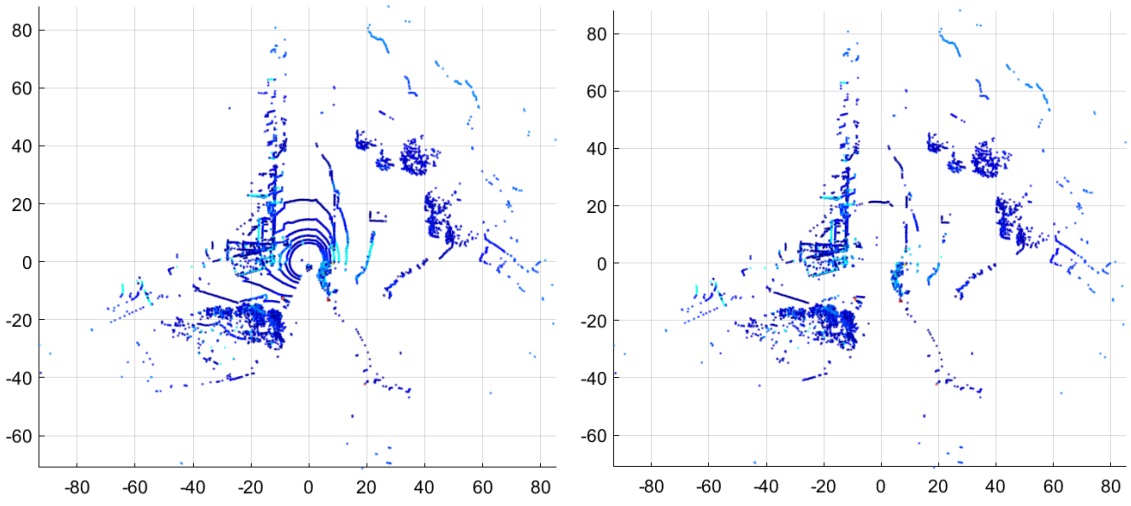
LA_l^l is the lever arm between the LiDAR scanner and the center of the inertial frame in the IMU frame.

p_n^{ecef} is the position of the vehicle in ECEF frame.

7.3.1.2 LiDAR Odometry

When doing LiDAR Odometry, each current frame was matched to the previous frame with Normal Distributions Transform (NDT) scan matching algorithm with the initial transformation information provided by GNSS/INS.

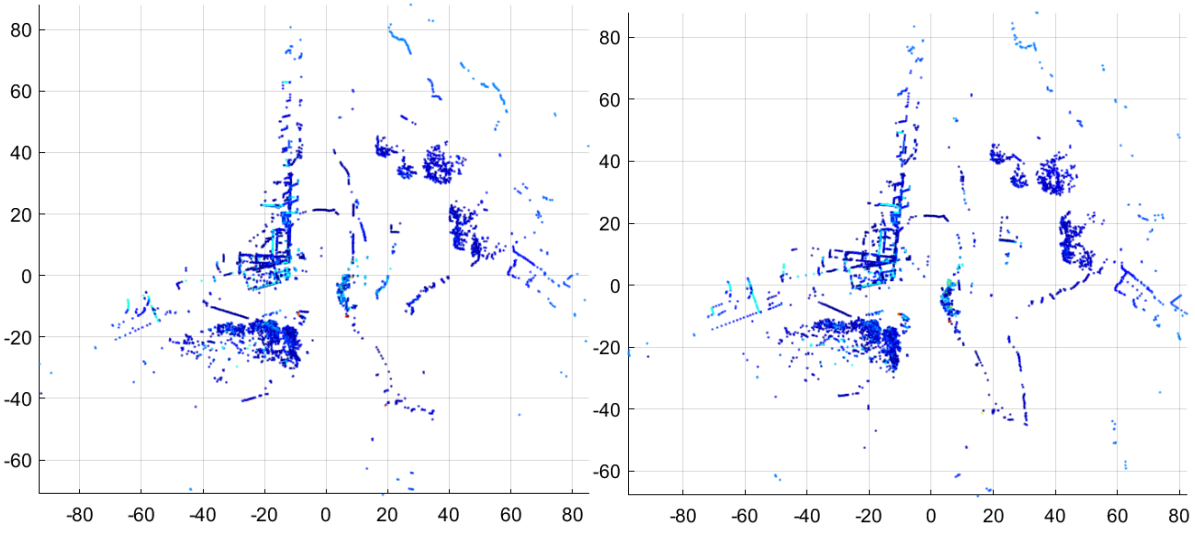
The point clouds were firstly pre-processed to remove the ground plane point (Figure 7.5) before matching by NDT to improve the accuracy of registration.



(a) (b)

Figure 7.5 Scan views of a LiDAR scan frame (a) the original scan view; (b) the view after pre-processing

Figure 7.6 show two scan views before being matched. It appears that these two scan views have slightly difference on the features (Figure 7.7).



(a) (b)

Figure 7.6 Scan views of two sequenced LiDAR scan frames ((a) previous scan frame; (b) current scan frame) for scan matching

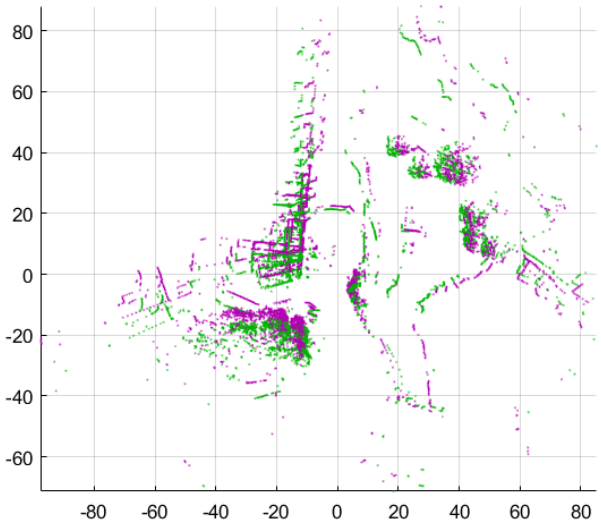


Figure 7.7 Comparison of the two sequenced LiDAR scan frames, green points: previous scan frame; red points: current scan frame

The matched point cloud from the two scan views can be generated (Figure 7.8).

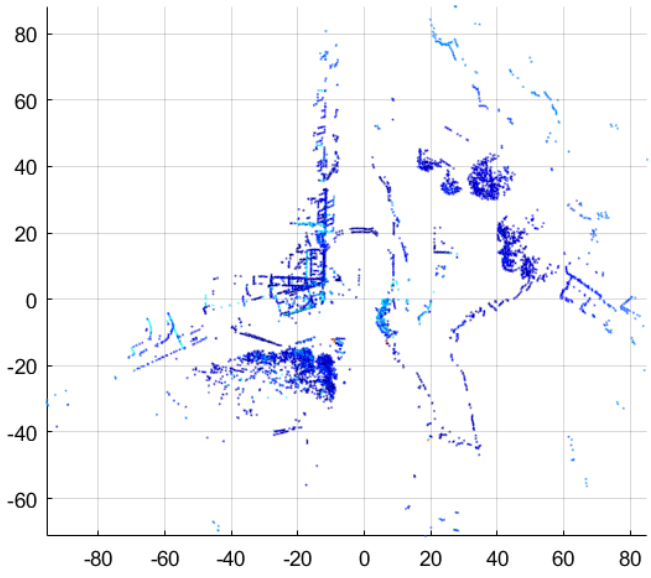


Figure 7.8 Generated map point cloud after matching two sequenced LiDAR scan frames (unit: meter)

By doing LiDAR odometry one by one with all the available LiDAR scans, the newly matched point cloud can be merged to the previously generated point cloud maps and

finally the accumulated map of the whole trajectory can be obtained and georeferenced (Figure 7.9).

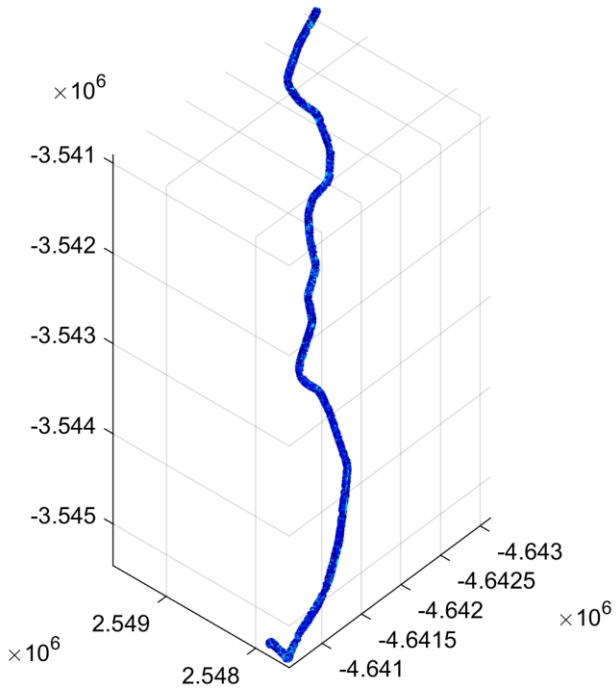
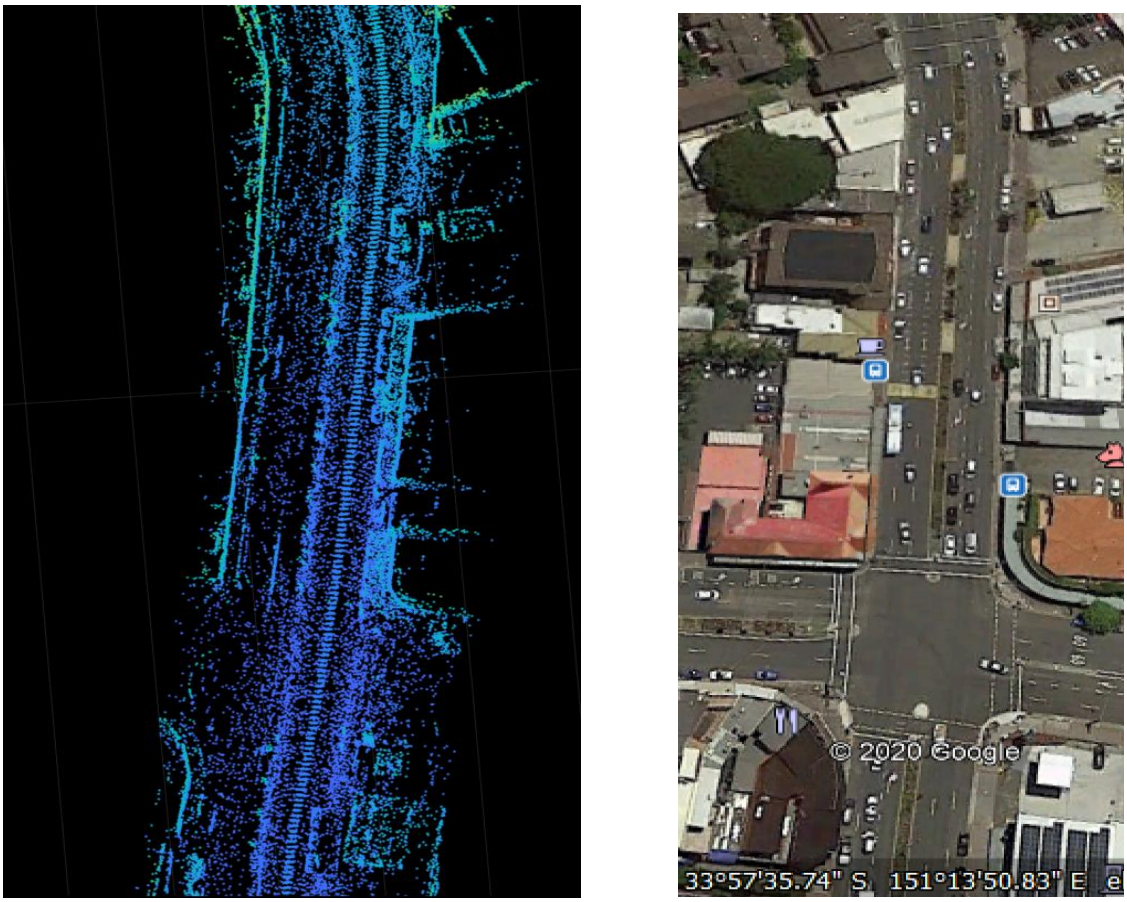


Figure 7.9 Global georeferenced road map from UNSW to La Perouse (frame: ECEF, unit: meter) the 3D point cloud based map for Section A dataset

By enlarging Figure 7.9, details of the road map can be seen, and its corresponding real-world road view can be found in Google Earth since this map is georeferenced. Figure 7.10 shows a comparison of one enlarged section of this generated map and its real view in google earth.



(a) (b)

Figure 7.10 (a) A section of the generated map and (b) the google earth view for the same location

Pose graph optimization can be run after trajectory estimation with sufficient number of loops to reduce the drift in trajectory estimation, and to generate more accurate final map. However, since our quality control method cannot monitor the dynamic model without any supported observation, the quality of the pose graph optimization cannot be tested.

7.4 Localization with LiDAR Scans and the geo-referenced 3D Point Cloud Map matching

The geo-referenced 3D Point cloud map produced from the data of Section A (the forward driving from UNSW to La Perouse) can then be used to support the LiDAR based localization for Section B (the backward driving from La Perouse to UNSW) by matching

the LiDAR scans to the map. The procedure of the online LiDAR/3D map matching based localization method is shown in Figure 7.11.

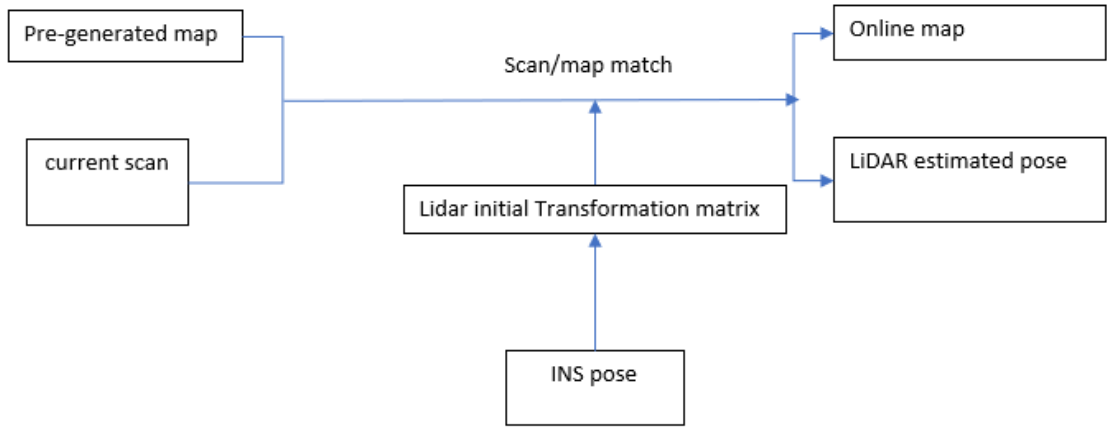
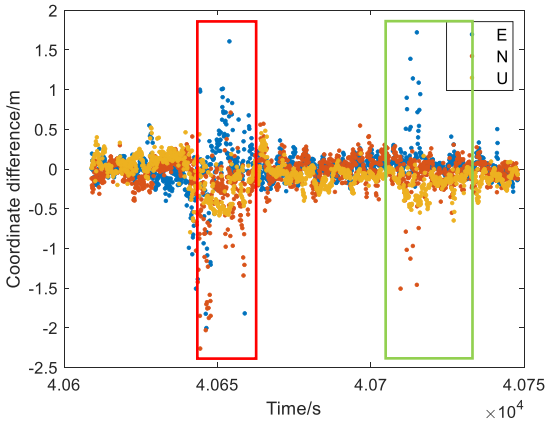


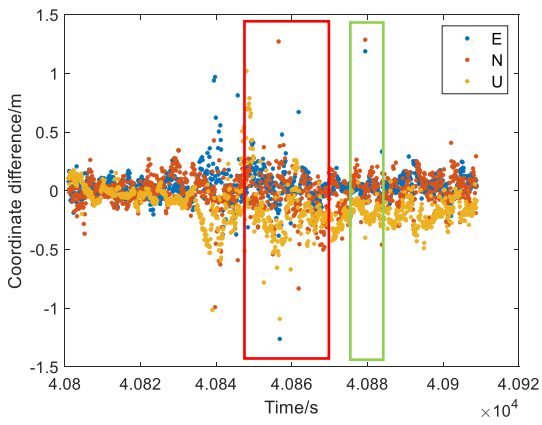
Figure 7.11 Overview of the proposed LiDAR/3D map matching based localization system architecture

7.4.1 Estimation Results of LiDAR/3D Map based Localization System

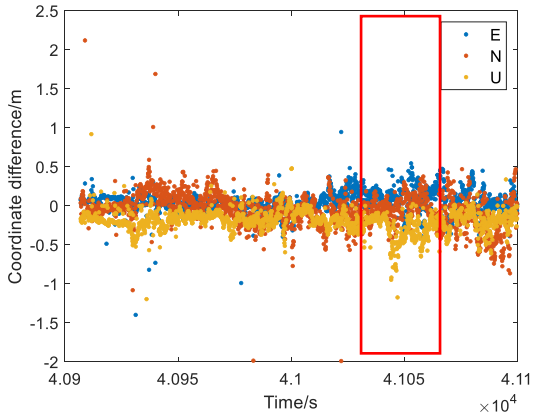
Since there is no ground truth information for this urban road test, the LiDAR/map matching based solutions are compared with the reference GNSS/INS solution within the three selected Trajectory Sections (Figure 7.12), during which the RTK status is fixed.



Trajectory Section 1



Trajectory Section 2



Trajectory Section 3

Figure 7.12 Coordinate difference between the proposed LiDAR/Map matching based localization method and the reference GNSS/INS localization solution

Table 7.1 Mean and standard deviation for the difference between LiDAR/Map matching based localization and the reference GNSS/INS localization results (remove periods with outlier)

		Trajectory Section 1	**Trajectory Section 2**	**Trajectory Section 3**
Mean (m)	East	0.020	-0.036	0.051
	North	-0.035	0.0031	-0.048
	Up	-0.084	0.140	-0.189
Stdev (m)	East	0.142	0.099	0.128
	North	0.162	0137	0.188
	Up	0.182	0.151	0.123

Table 7.1 shows the comparison between the LiDAR/Map matching based localization and the reference GNSS/INS localization results. The differences are fluctuated around zero, and their mean values are around centimeter to decimeter level. The standard deviation for all the three sections is around 0.1-0.2 meters, therefore we treat the coordinates difference larger than 0.6 meters as possible outliers. The epochs that have outliers is about 1.7% of the total testing period, which means the happen of outliers are rare. The reason of the outlier will be discussed in next Section.

7.4.2 Quality Analysis of the Numerical Results

The details of measurements during the epochs with big jumps (such as during red boxes Figure 7.12) are checked to find out the possible reasons for this kind of outliers detected. For Trajectory Section 1, it is found that when driving around a roundabout, there were some big outliers (red boxes Figure 7.12 Trajectory Section 1). The trajectory of the LiDAR/Map system and the GNSS/INS solution around this roundabout and their views in google map are shown in Figure 7.13. It can be seen the reference GNSS/INS solution are smoother at this area since GNSS integer-ambiguities are fixed, while the LiDAR/Map solution has some differences to this reference trajectory.

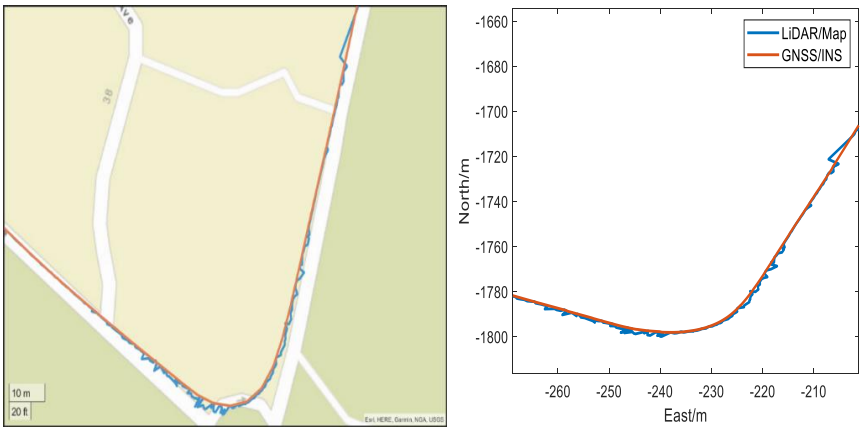
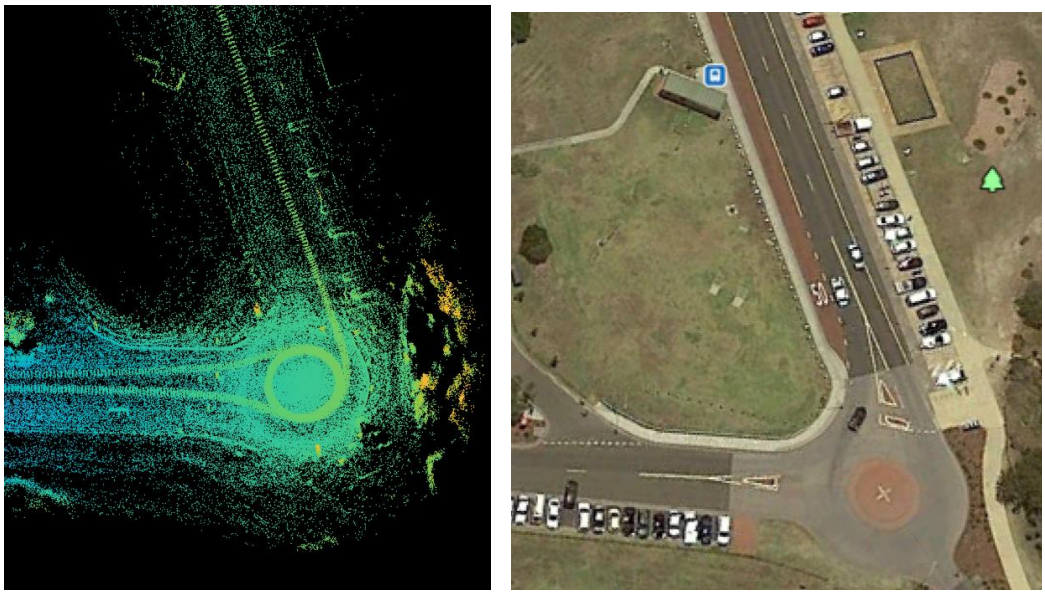


Figure 7.13 Trajectory of LiDAR/Map matching based localization (blue) and the reference GNSS/INS localization (red)

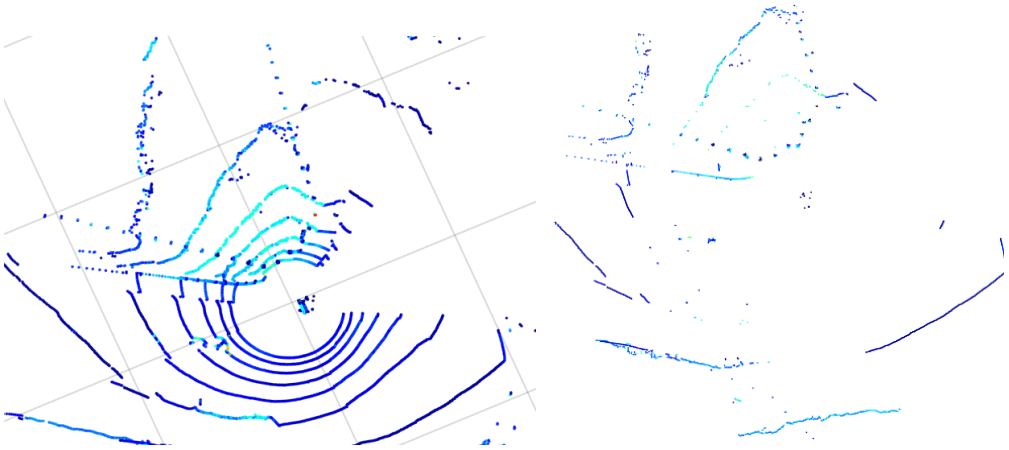
Figure 7.14 shows the LiDAR view at this roundabout. It is found from Figure 7.14 that the structure of the pre-generated map at the driving side of the road is not very clear since it is lack of features around the trajectory. The roundabout is located at a parking area of tourist attraction. There is no building or very few trees around this area. Since the testing was undertaken in the evening, there were even not many parking vehicles which could be used as features. Therefore, the quality of matching step may be poor, which results in poor localization accuracy.



(a) (b)

Figure 7.14 (a) A section of the generated map and (b) the google earth view for the same location at the La Pereous Roundabout

Figure 7.15 shows the LiDAR scan view at this point with a range threshold of 20 meters, it can be seen, this LiDAR scan has not much usable features, especially after pre-processing. Similar road environment with less feature can be found when another cluster of outliers happened in Trajectory Section 1 (green box) shown in Figure 7.12.



(a) (b)

Figure 7.15 Scan frame at epoch 40647 s with big outlier in Trajectory Section 1 ((a) the original scan view; (b) the view after pre-processing)

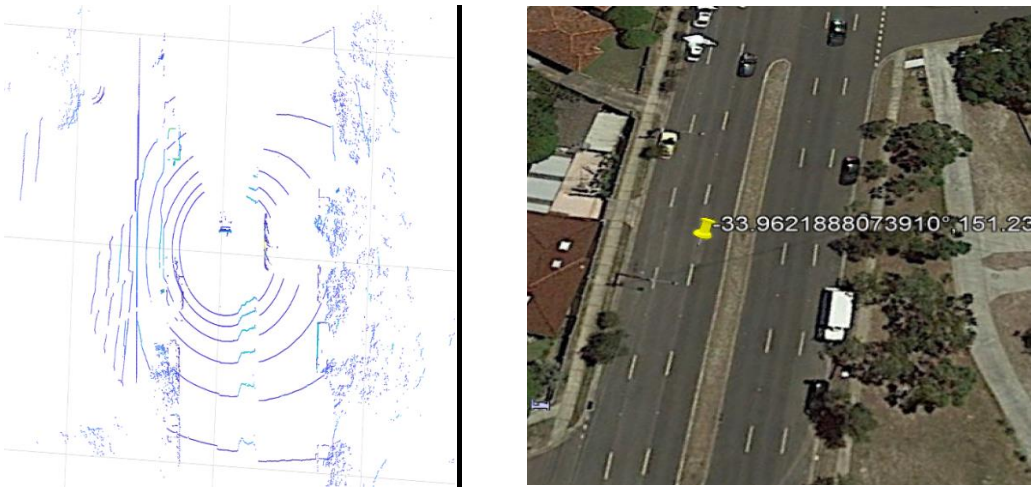
Another major source of outliers is the other moving entities around the host vehicle. Figures 7.16 to 7.18 shows the LiDAR views when there are big outliers in the localization stage.

No matter the other moving vehicle is at the same side of the road or at the opposite road side, such a moving vehicle will influence the LiDAR/Map matching-based localization quality. When the moving vehicle is at the same road side of the host vehicle, it will result in bad estimation when it is initially detected, has different speed of the host vehicle, or when it turns and drives to another road, which makes it no longer detectable anymore (red box in Figure 7.12 Trajectory Section 2 and Figure 7.16).

(a) (b)

Figure 7.16 Scan frame at epoch 40854s with big outlier in Trajectory Section 2: a following vehicle is driving to another road ((a) LiDAR scan; (b) google view)

Once the host vehicle detected an opposite driving vehicle, the localization estimation errors could reach to 1-1.5 meters (green box in Figure 7.12 Trajectory Section 2 and Figure 7.17).



(a) (b)

Figure 7.17 Scan frame at epoch 40888s with big outlier in Trajectory Section 2, an opposite driving vehicle is detected ((a) LiDAR scan; (b) google view)

The type of moving elements will also influence the happening of outliers. For most of time, vertical position estimation is less influenced by the moving elements. But when checking the red box in Trajectory Section 3 (Figure 7.12), it was found that the differences on vertical direction are much higher than other sections. By looking at the details of LiDAR view, it is found that, at that section, a tall bus was driving passed the host vehicle (Figure 7.18), which means these vertical differences of current LiDAR scan and the pre-generated map might have caused some systematic vertical biases.

(a) (b)

Figure 7.18 Scan frame at epoch 41103s with big outlier in Trajectory Section 3 with one tall bus driving passed ((a) LiDAR scan; (b) google view)

The moving objects within the road environment will be a major source of measurement outlier because the system treated the pre-generated map as a fixed reference map. Therefore, if there are any moving objects that cause the structures of the pre-generated map and the current scan frame different, it will cause outliers. The moving objects, such as other vehicles, may exist in both the previous road mapping stage and in the current road scans for use in localization. For the step of offline HD map generation, this kind of moving objects should be carefully identified and removed from the static 3D point cloud maps. For the online step, with the aiding of some quality control method, the influence of the possible detected moving objects could be eliminated, or the possible road environment change could be updated to the global map to enhance the accuracy of future driving around the same road path.

7.5 EKF based LiDAR/GNSS/INS/3D Map Localization within the pre-generated Global Map

Apart from directly using the INS pose information to support LiDAR/3D map localization, the LiDAR/map solution can also be used to correct the inertial system with under an Error-state EKF system. Figure 7.19 shows the architecture of the proposed EKF based LiDAR/GNSS/INS/Map localization system.

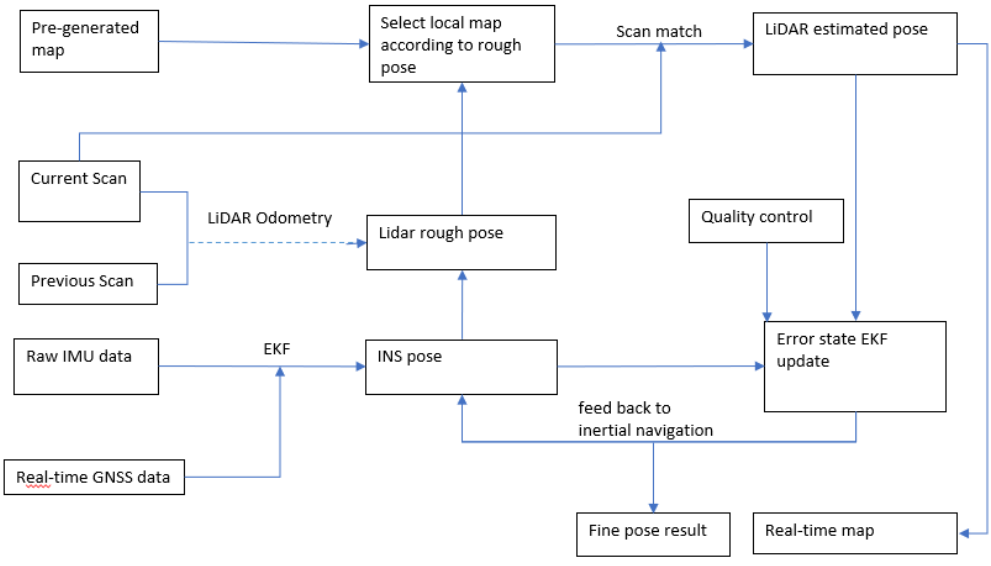


Figure 7.19 Overview of the proposed LiDAR/GNSS/INS localization system architecture within pre-generated map.

The localization system mainly contains two parts: scan matching and Error-state EKF updating. Here the Error-state EKF is an indirect Kalman filter that its measurement is not the directly IMU output, such as from gyro or accelerometer sensors, but is the difference between the IMU data and the external source data (GNSS data or LiDAR pose) (Roumeliotis et. al., 1999). This error-state EKF will estimate the errors in the inertial navigation results.

Firstly, if the inertial navigation information is not available, the frame-to-frame LiDAR odometry can be used to support localization. After initializing the Error-state EKF, the estimated pose of the inertial navigation will provide a rough pose for the current LiDAR scan frame, the LiDAR odometry can be shut down to save computation. With the rough position provided by the INS, a local map is searched and selected from the pre-generated global map in order to improve the matching efficiency. NDT based scan matching between the current LiDAR frame and then the local map is undertaken with the inertial based initial Transformation matrix. The LiDAR estimated vehicle posed can be achieved. A new real-time road map can also be generated during this real-time localization system with the SLAM algorithm.

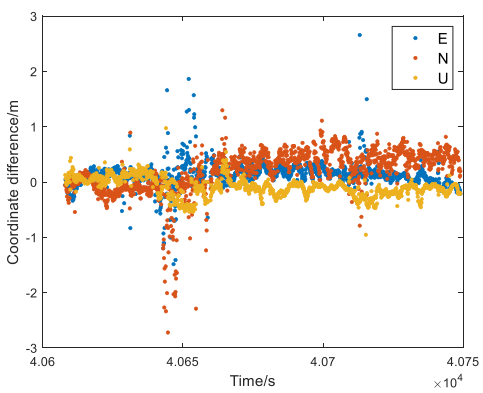
After achieving the LiDAR pose, the difference between the LiDAR pose and the inertial propagation pose can be obtained and the error within the inertial navigation information is estimated by the Error-state EKF and fed back to the inertial system to achieve fine pose results. When GNSS information is available, the positioning results of GNSS, such as RTK position results, can also be used to correct the inertial navigation information in order to improve the accuracy and reliability of the localization system. Quality control methods presented in Chapter 2 can be used to monitor this proposed localization system under the framework of the Error-state EKF.

7.5.1 Estimation Results of the EKF LiDAR/GNSS/INS/3D Map based Localization System

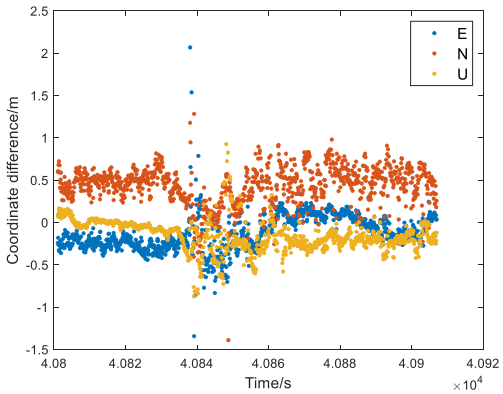
The Section B dataset of the road test can be used to test with the proposed EKF LiDAR/GNSS/INS/3D map-based localization system. In this testing, the LiDAR

frequency is 10 Hz, GNSS is 1 Hz, and IMU 100 Hz, therefore, it will output three sort of pose solutions: the LiDAR estimated pose (10 Hz), the LiDAR/INS (10 Hz), the GNSS/INS pose (1 Hz). Since there is no ground truth information for this urban road test, the estimated position solutions are compared to the reference GNSS/INS solutions used in Section 7.4 for analysis. The setting of measurement uncertainty is 20 cm for LiDAR based measurement and 0.02 cm for GNSS based measurement according to the results from Chapter 3 and Section 7.4.

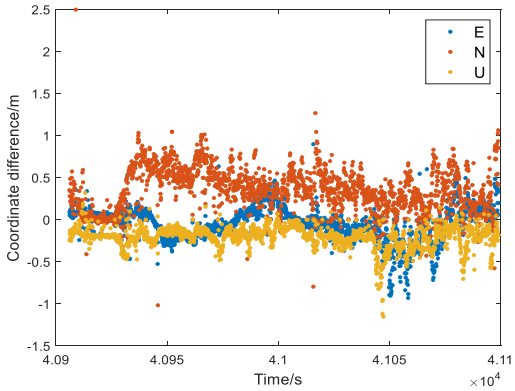
Figure 7.20 shows the LiDAR estimated position results that using the online INS pose for initial Transformation matrix and for local map matching. It is found in this EKF based localization system, the epochs in which outliers occur are at the similar times as shown in Section 7.4. This LiDAR estimated position is then be used as measurement for the inertial EKF system.



Trajectory Section 1



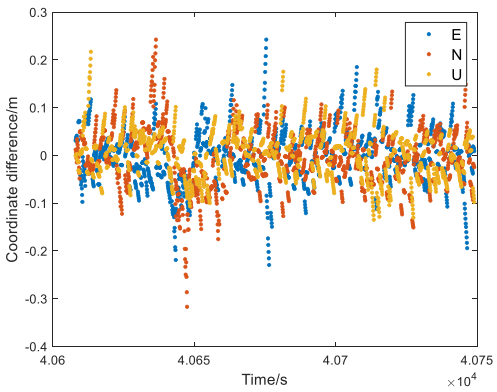
Trajectory Section 2



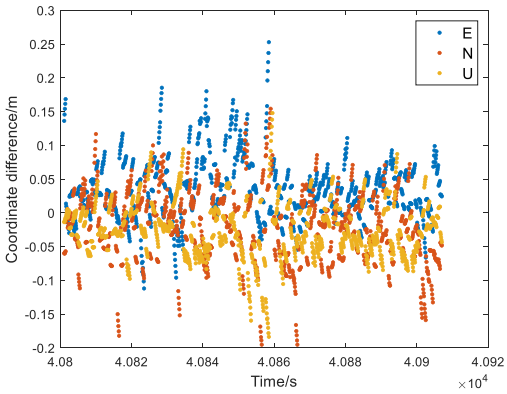
Trajectory Section 3

Figure 7.20 Coordinate differences between the LiDAR estimated position by the proposed LiDAR/GNSS/INS/Map method and the reference GNSS/INS localization solution

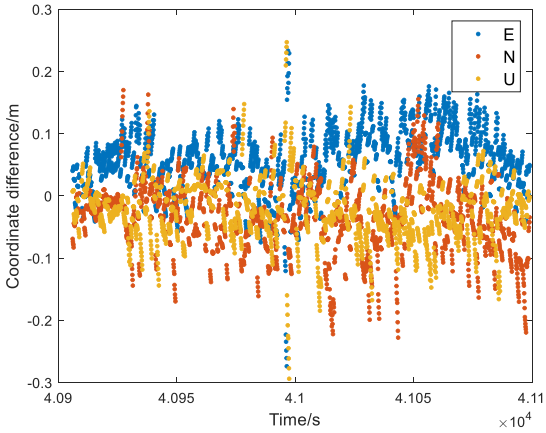
Figure 7.21 and Figure 7.22 compare the 10 Hz LiDAR based INS localization solution and 1 Hz GNSS based INS localization solution from this EKF LiDAR/GNSS/INS/Map system to the referenced GNSS/INS only system. By comparing Figures 7.21 and 7.22, it is found GNSS based solution has better accuracy (within 0.1 meters) than the LiDAR based accuracy (within 0.1-0.2 meters). Therefore, the final INS position is improved when GNSS measurement is coming in.



Trajectory Section 1

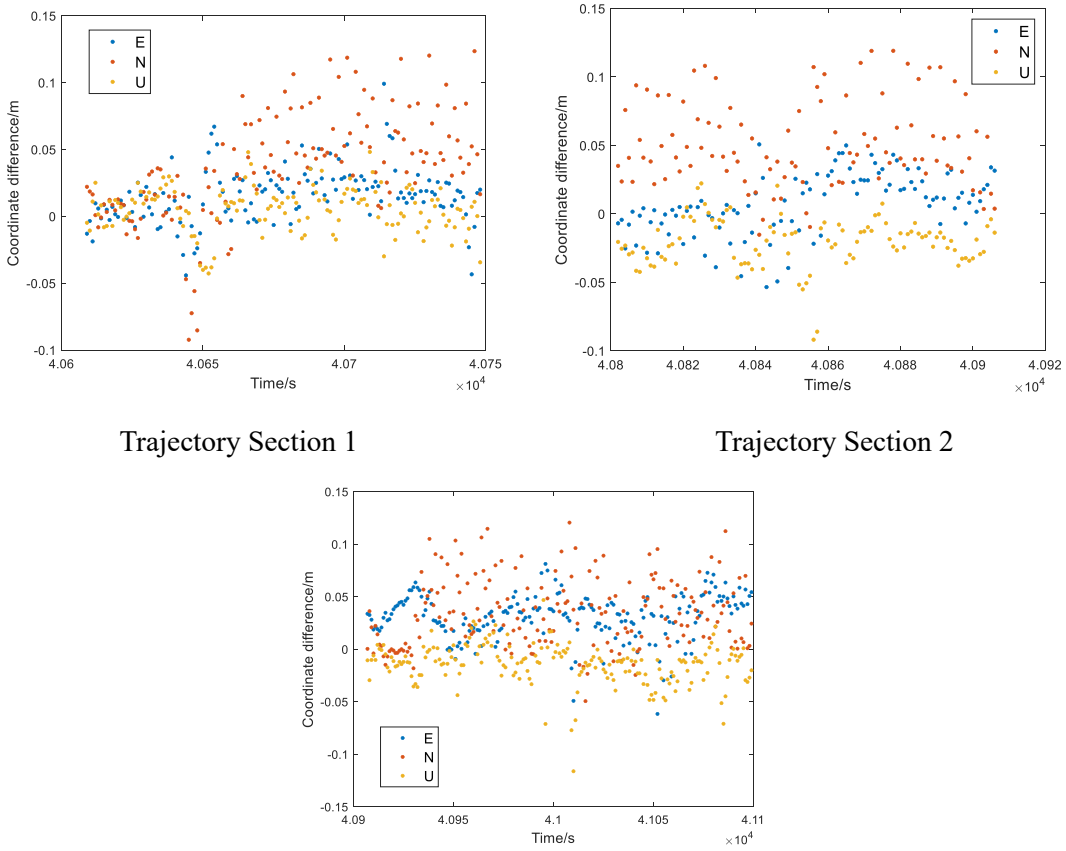


Trajectory Section 2



Trajectory Section 3

Figure 7.21 Coordinate differences between the final estimated INS position based on LiDAR measurement input by the proposed LiDAR/GNSS/INS/Map localization method and the reference GNSS/INS localization solutions



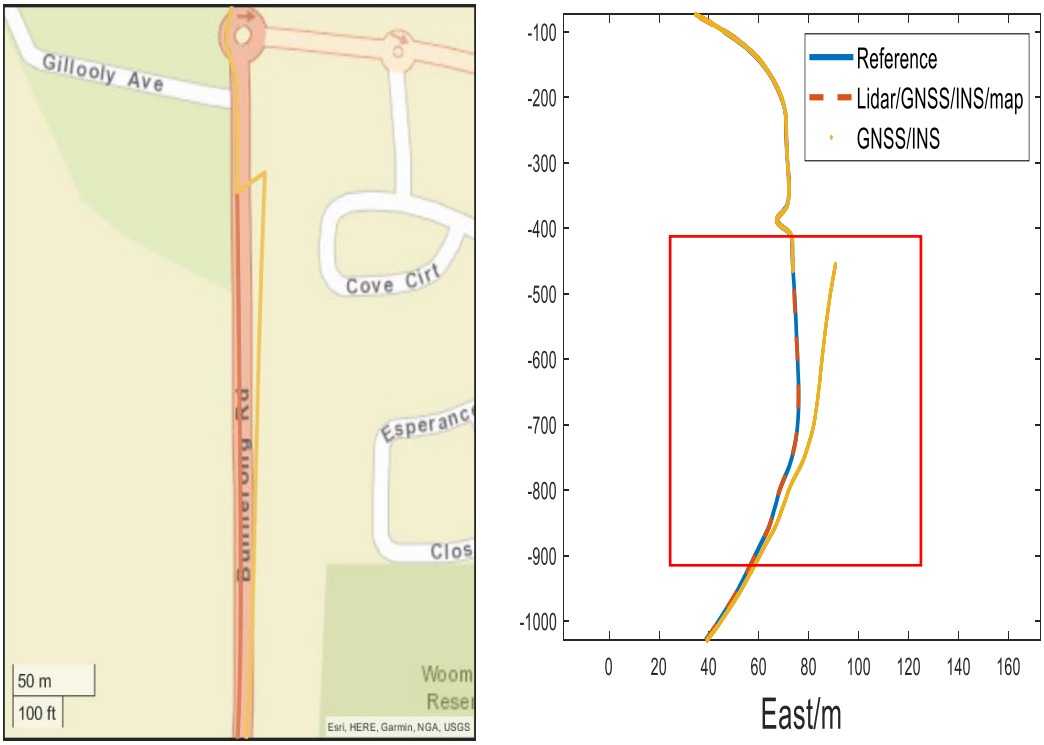
Trajectory Section 1

Trajectory Section 2

Trajectory Section 3

Figure 7.22 Coordinate differences between the final estimated INS position based on GNSS measurement input by the proposed LiDAR/GNSS/INS/Map localization method and the reference GNSS/INS localization solution

Since the GNSS signals may be frequently lost during driving within the urban area, a GNSS signal blockage period of 1 minutes was simulated between testing epoch of 40700s to 40760s. Therefore, the online GNSS/INS results during that simulated GNSS blockage period will have drifts to the referenced non-blockage GNSS/INS trajectory (Figure 7.23), these drifts, up to 20 meters, will be corrected once the GNSS signal comes back (see the red box in Figure 7.23). The LiDAR aided EKF solution has a much smoother trajectory during that period of simulated bad GNSS condition, indicating that the LiDAR based localization method can help to maintenance the localization accuracy when GNSS signal is lost (Figure 7.23).



(a) (b)

Figure 7.23 Estimated trajectory by the proposed localization method and the GNSS/INS localization method with simulated GNSS signal blockage period: trajectory viewed (a) on Google map; (b) on Local coordinates

The uncertainty results of the estimated states are shown in Figure 7.24-7.26. In this testing, since GNSS measurements have much higher precision than the LiDAR based measurements, much higher measurement covariance value is set for the LiDAR measurement. Therefore, when GNSS measurement available, it will make the EKF estimation's uncertainty significantly reduced, but during the 1s period, the LiDAR EKF results will have increasing uncertainty due to its higher uncertainty setting. The Standard Deviation trend of the EKF system will be like zigzag (Figure 2.24).

Frequently jumps can be seen when the host vehicle has a change on heading, such as turning. Heading has higher STD value than pitch and roll. According to Figure 7.26, Pitch and roll get convergent very quickly, while for heading, its STD has some increasing period, which may because the heading angle error is unobservable when the horizontal specific force components are zero (Farrell, and Wendel., 2017). The heading angle error

cannot be affected by the velocity errors in such condition. As a consequence, the uncertainty of the yaw error state will grow with time when the platform is static or driving in nonaccelerating motion (Farrell, and Wendel., 2017).

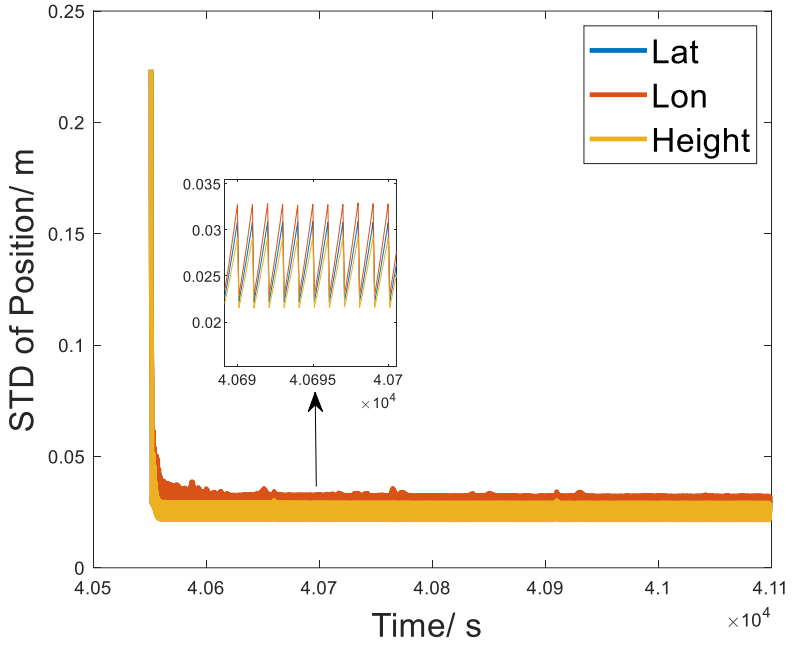


Figure 7.24 Standard deviations of estimated position error state by the proposed EKF LiDAR/GNSS/INS localization system within a pre-generated map

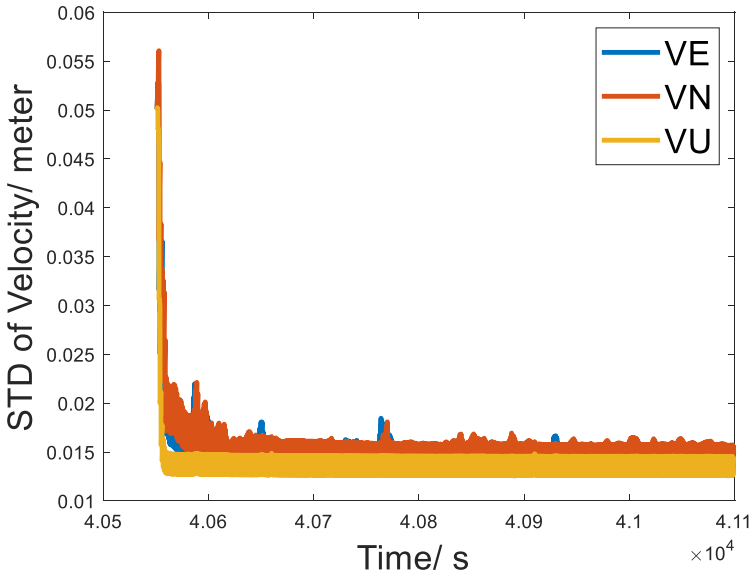


Figure 7.25 Standard deviations of estimated velocity error state by the proposed EKF LiDAR/GNSS/INS localization system within a pre-generated map

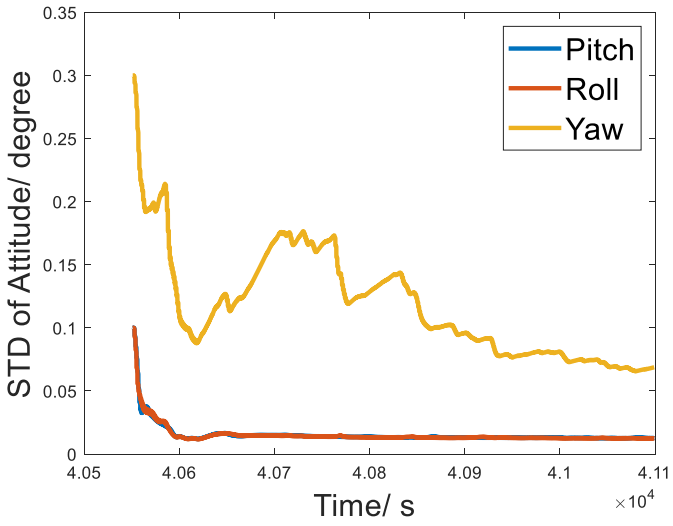
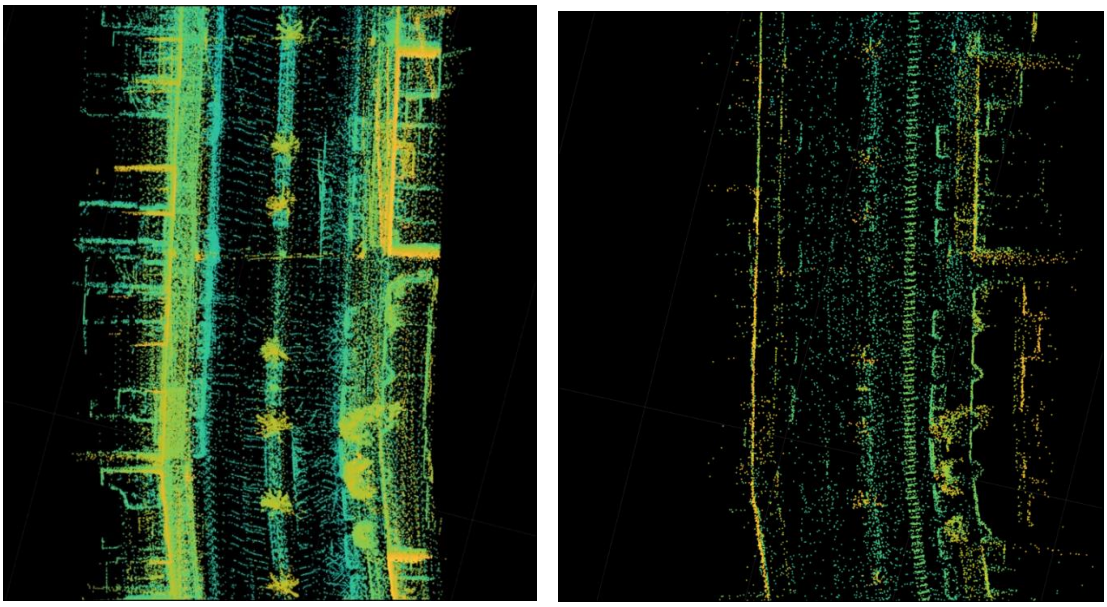
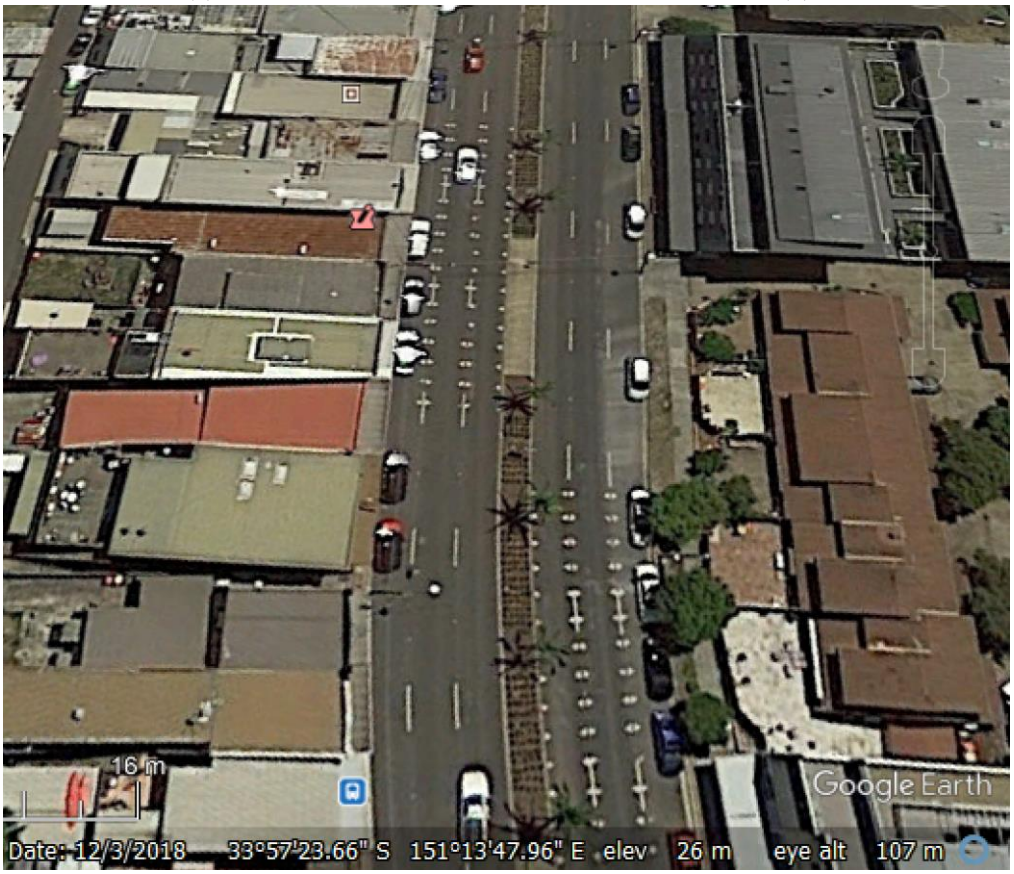


Figure 7.26 Standard deviations of estimated attitude error state by the proposed EKF LiDAR/GNSS/INS localization system within a pre-generated map

A real-time map can be generated simultaneously during the localization. Figure 7.27 is a comparison of the real-time generated map and the pre-generated map of around a position within the trajectory. As we can see from Figure 7.27, the pre-generated map has more features at the right lane, while the online map has more at the left lane. This is because in this test, for both of the mapping and localization steps, only one way driving trajectory is utilized, since there is no loop for this driving, the LiDAR view of the other side of road will be sparser than its driving side. Therefore, in the purpose of ensuring online localization accuracy and reliability, a forward-backward trajectory is needed for the offline mapping, or using the online real-time generate map to update the whole map, so that this update map can be serviced for future localization or for other road driving vehicles. Furthermore, since the road environment contains changing elements, such as temporary parked vehicle along road sight. This kind of real-time map from the real-time autonomous driving can be used to update the pre-generated map, or pre-stored HD map about this dynamic changing. However, for purpose of improving computation efficiency and reducing storage size, the real-time map can also be chosen to not be estimated and stored during driving.



(a) (b)

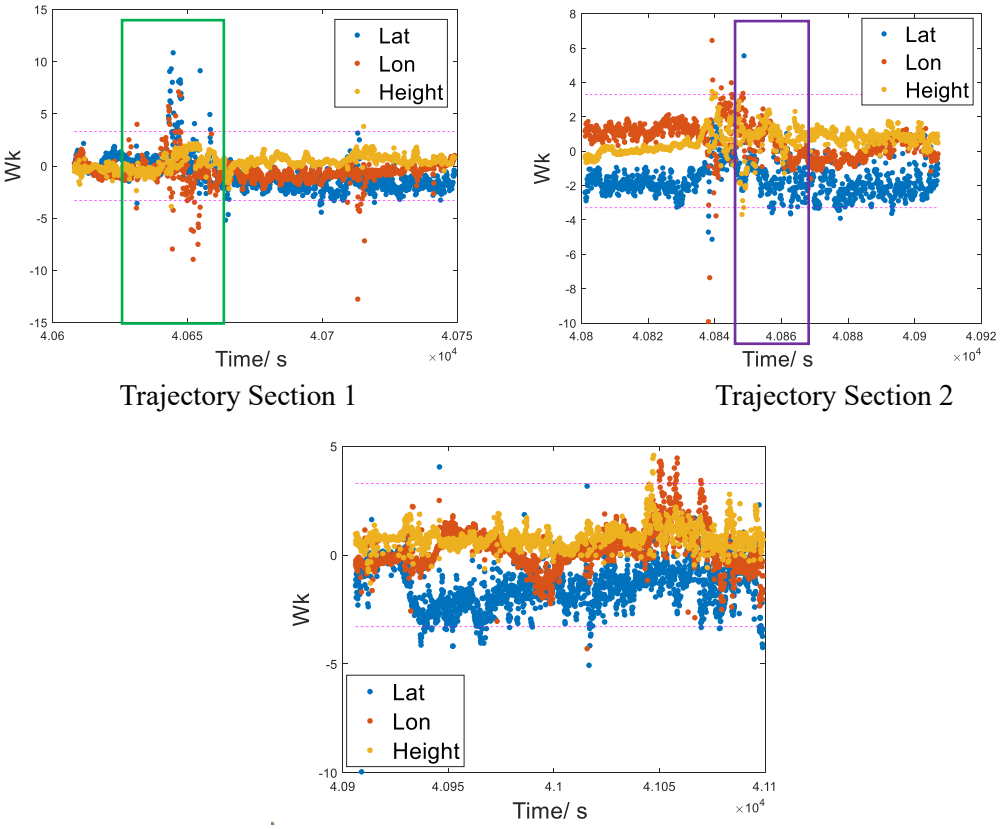


(c)

Figure 7.27 A section of real-time generated road map from La Perouse to UNSW when doing localization within a pre-generate map ((a) real time generated map; (b) pre-generated map; (c) real world road view in google earth)

7.5.2 Quality Control Results of the proposed Localization for Road Test dataset

Quality control can be conducted for this proposed EKF based localization method to monitor the quality of this urban road test. According to the outlier statistic test results, around 3-5% of the LiDAR based measurements have a W test value larger than 3.29. There are some jumps of W test results (Figure 7.28). Since the epoch with high Wk value is mostly corresponding to the epoch with outlier in Section 7.4, the possible reason of the outlier has already discussed in Section 7.4. Low number of features or the moving objects within the road environment will be a major source of measurement outlier because the system treated the pre-generated map as a fixed reference map. Therefore, if there are any moving objects that cause the structure of the pre-generated map and the current scan frame different, it will cause system outliers and will be detected by our Quality control system.



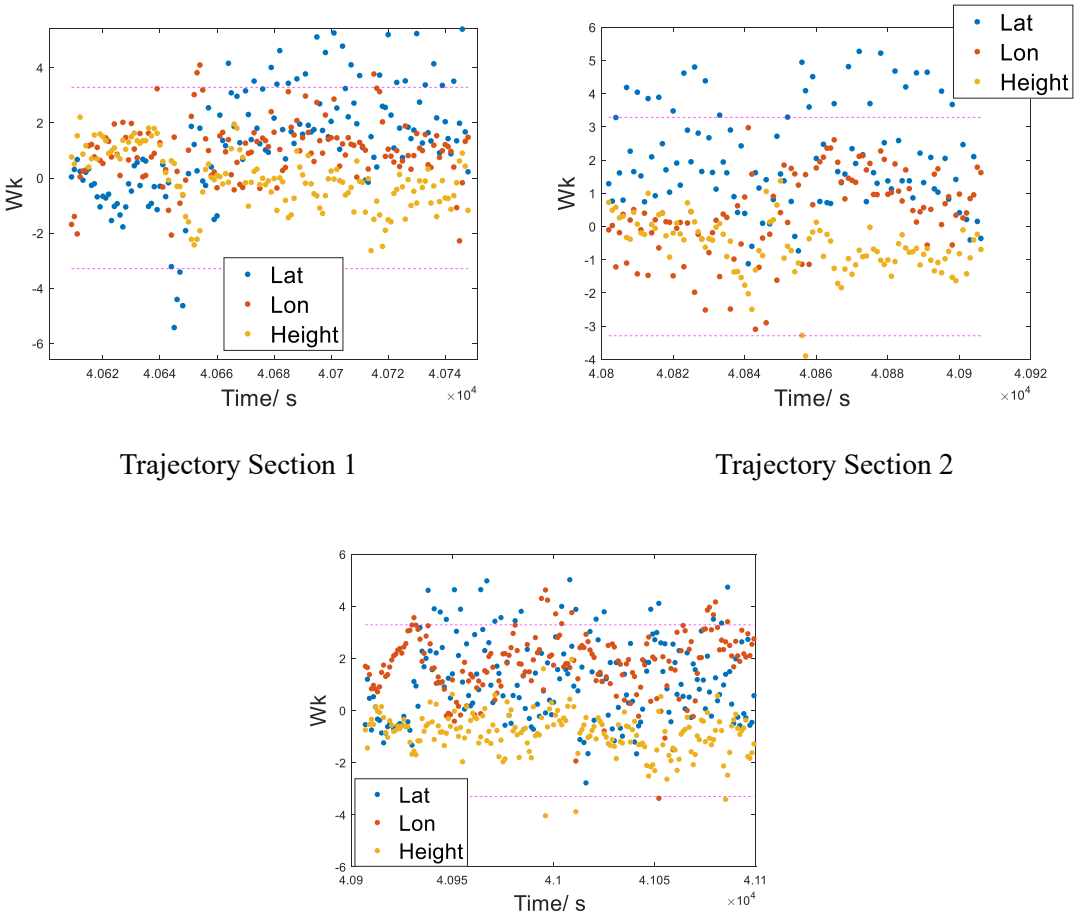
Trajectory Section 1

Trajectory Section 2

Trajectory Section 3

Figure 7.28 Outlier statistic test result of the proposed EKF localization method for the LiDAR/map matching-based position measurements

Figure 7.29 shows the outlier statistic test results for the GNSS based measurements during the EKF LiDAR/GNSS/INS/Map localization process. It is found many outliers are detected, which is more than the case of GNSS/INS only system (Chapter 3). The reason may be due to the higher LiDAR measurement uncertainty that will influence the whole integration system.



Trajectory Section 1

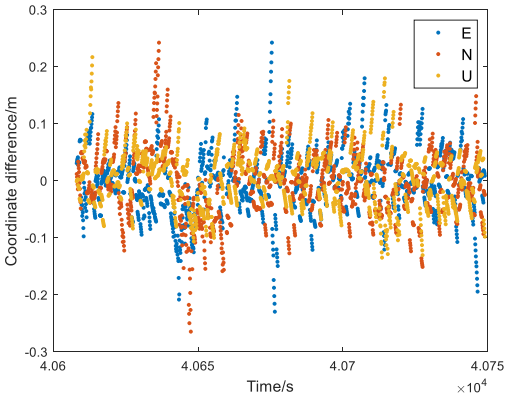
Trajectory Section 2

Trajectory Section 3

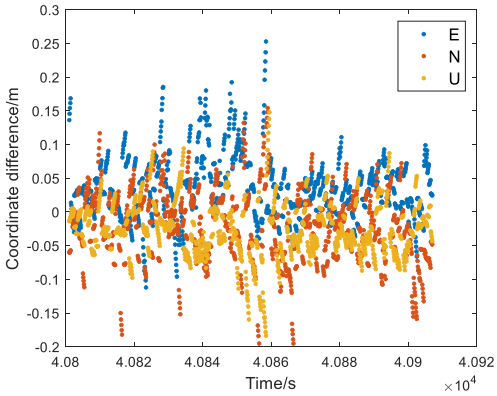
Figure 7.29 Outlier statistic test result of the proposed EKF localization method for the GNSS based position measurement

For the proposed online EKF based localization step, with the aiding of the quality control method, the influence of the detected moving objects can be eliminated. Figure 7.30 and Figure 7.31 shows the coordinate difference results after correcting any influence from the detected outliers during the proposed EKF LiDAR/GNSS/INS/Map localization system with the Adaption method. It is found LiDAR based solutions are

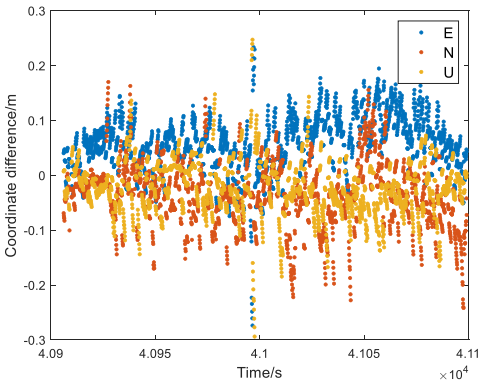
very slightly improved under this system after mitigating all the possible influence of the outliers. While for the GNSS based solution, it achieves much better improvements.



Trajectory Section 1

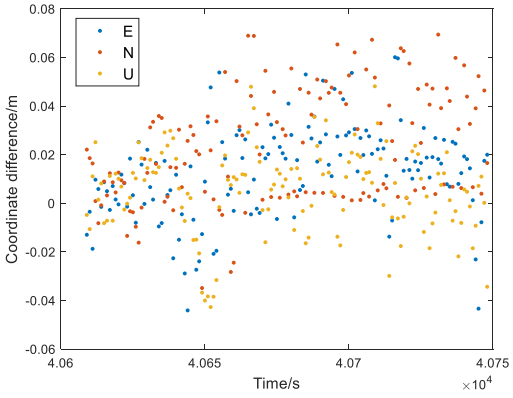


Trajectory Section 2

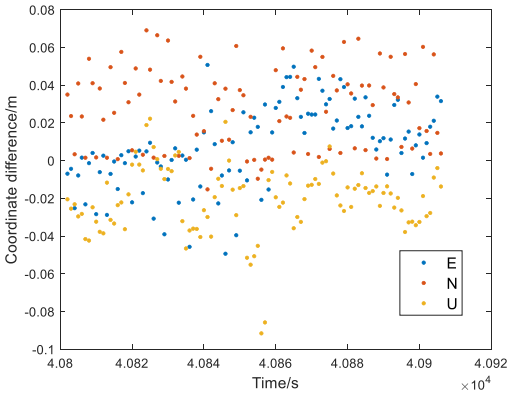


Trajectory Section 3

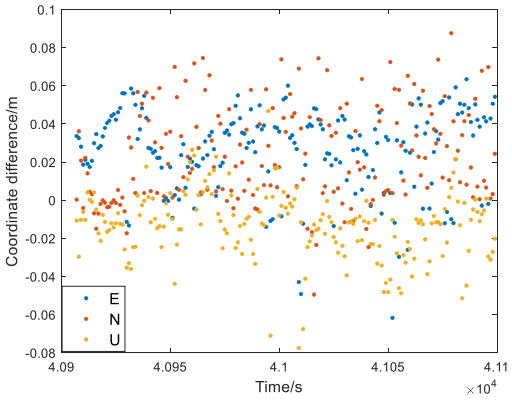
Figure 7.30 Coordinate differences between the final estimated INS position based on LiDAR measurement input by the proposed LiDAR/GNSS/INS/Map localization method and the reference GNSS/INS localization solution after correction with the Adaption method



Trajectory Section 1



Trajectory Section 2



Trajectory Section 3

Figure 7.31 Coordinate differences between the final estimated INS position based on GNSS measurement input by the proposed LiDAR/GNSS/INS/Map localization method and the reference GNSS/INS localization solution after correction with the Adaption method

Table 7.2 and Table 7.3 is the mean and Standard deviation values.

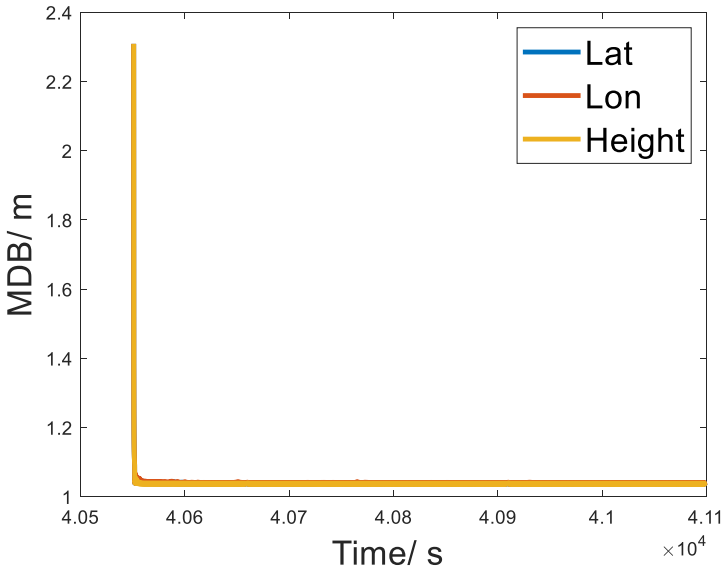
Table 7.2 Mean and standard deviation for the differences between LiDAR measurement-based LiDAR/GNSS/INS/Map localization results and GNSS/INS localization results before and after correcting the influence of detected outliers with the Adaption method

		Section 1		**Section 2**		**Section 3**	
		Before correction	After correction	Before correction	After correction	Before correction	After correction
Mean (m)	East	-0.0026	-0.0030	0.0358	0.0356	0.0571	0.0566
	North	-0.0052	-0.0043	-0.0221	-0.0212	-0.0371	-0.0359
	Up	0.0041	0.0041	-0.0250	-0.0248	-0.0228	-0.0225
Stdev (m)	East	0.0556	0.0551	0.0466	0.0462	0.0503	0.0498
	North	0.0605	0.0585	0.0530	0.0528	0.0574	0.0575
	Up	0.0481	0.0480	0.0410	0.0410	0.0486	0.0485

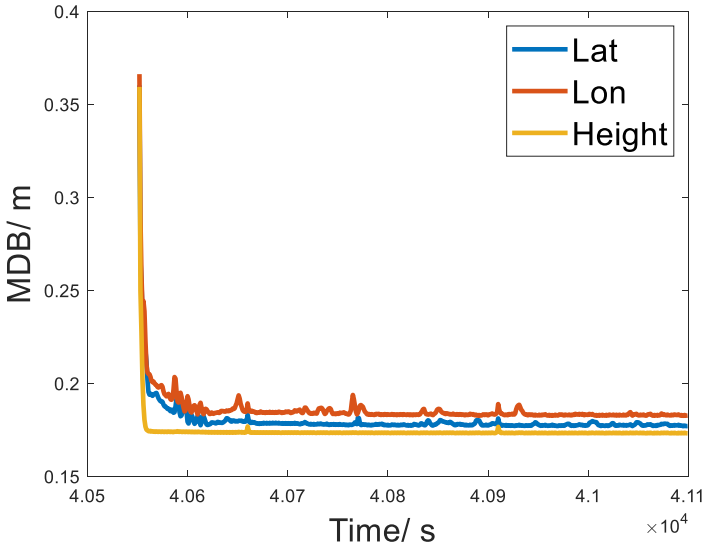
Table 7.3 Mean and standard deviation for the difference between GNSS measurement-based LiDAR/GNSS/INS/Map localization results and GNSS/INS localization results before and after correcting the influence of detected outliers with the Adaption method

		Section 1		**Section 2**		**Section 3**	
		Before correction	After correction	Before correction	After correction	Before correction	After correction
Mean (m)	East	0.0158	0.0128	0.0070	0.0070	0.0312	0.0260
	North	0.0371	0.0232	0.0511	0.0274	0.0343	0.0236
	Up	0.0051	0.0051	-0.0208	-0.0208	-0.0141	-0.0132
Stdev (m)	East	0.0210	0.0190	0.0225	0.0217	0.0216	0.0219
	North	0.0397	0.0228	0.0310	0.0216	0.0325	0.0250
	Up	0.0171	0.0171	0.0181	0.0180	0.0194	0.0175

Figure 7.32 shows the Minimum Detectable Bias (MDB) of the proposed EKF LiDAR/INS/GNSS localization method within the pre-generated map. It is found the MDB can converge very quickly from initial 2.3 meters to around 1 meter, indicating that around 1 meter outlier in the LiDAR based measurement can be detected by our quality control system. GNSS measurement has MDB value around 0.17 meters after convergence, which is slightly higher than the GNSS/INS only system in Chapter 3 (0.18-0.2 meters), indicating the aiding of LiDAR measurement may improve the GNSS/INS's reliability.



(a)



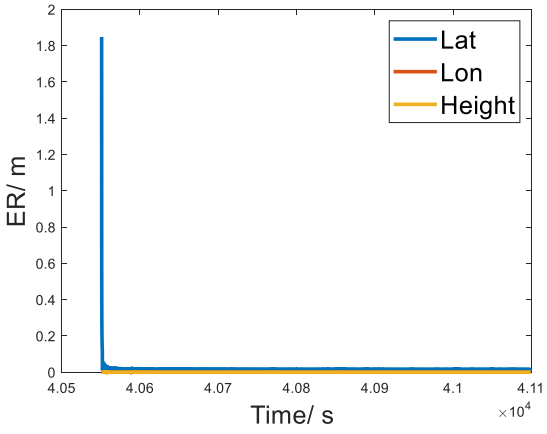
(b)

Figure 7.32 MDB values for the proposed LiDAR/GNSS/INS/Map localization method: (a) LiDAR based measurements, (b) GNSS based measurements

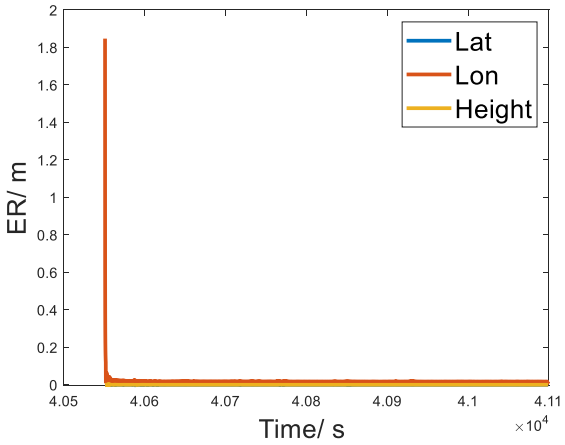
Any possible outliers within the EKF measurement model of the difference between inertial propagated position and LiDAR estimated position will cause bias on the final position estimations, no matter whether detected or not detected. The External Reliability (ER) results of the LiDAR based measurement models are shown in Figures 7.33-7.35,

indicating the influence of the undetectable outliers (with a value same to MDB value) upon the final error states' estimation. According to Figure 7.33, this influence upon latitude and longitude estimations will converge quickly to around 0.02m. The overall ER value is about 1-2 times of the STD value (sigma-range). The estimation of height states is less influenced by the outliers with ER value around 1 sigma-range. This means the estimation of vertical states is more reliable than that of the horizontal states.

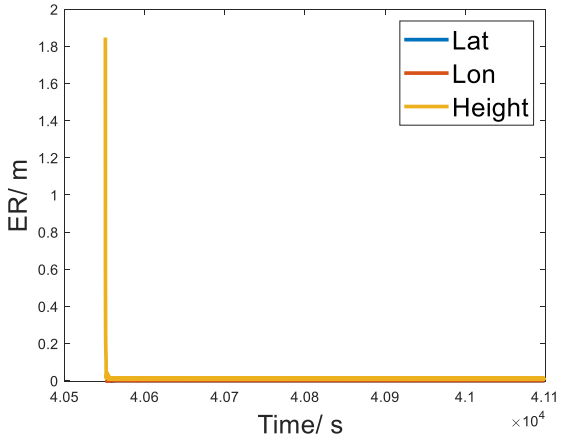
The outliers within the measurement model will also influence other error states that are not directly observable, such as the velocity and attitude. But these states are more robust to the outliers than the directly observed states (position). The ER values for velocity states are around 1 sigma-range, and for attitude states are 0-1 sigma-range. Heading is more likely to be influenced by the outliers than pitch and roll.



(a)

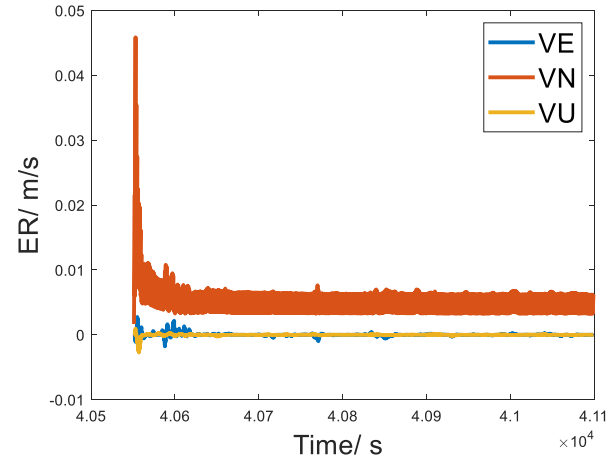


(b)

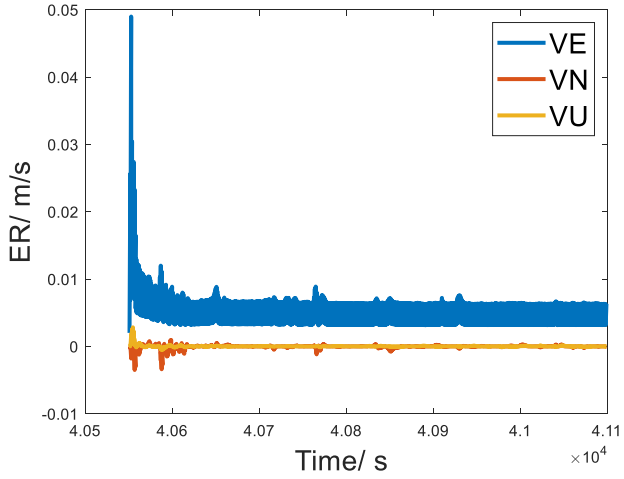


(c)

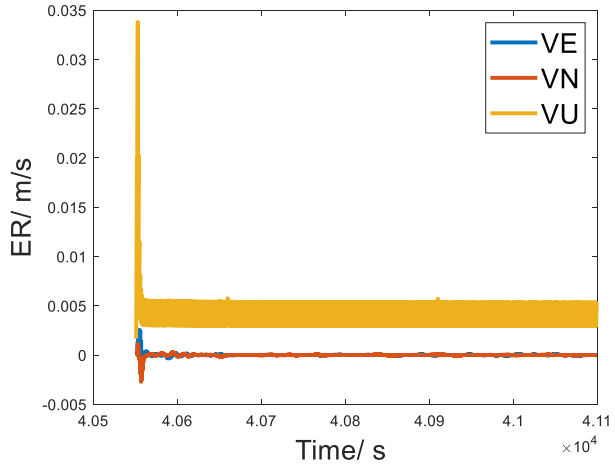
Figure 7.33 External Reliability values of the undetected outlier within EKF LiDAR based measurement models (a) Latitude difference, (b) longitude difference, (c) height difference toward position error states



(a)

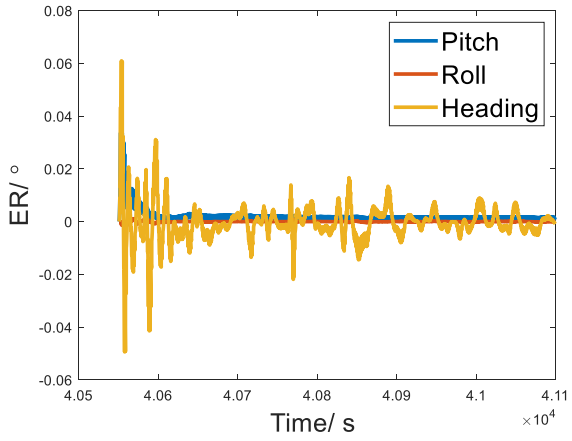


(b)

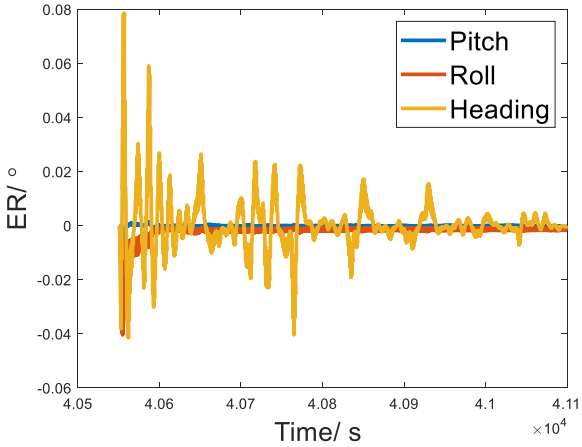


(c)

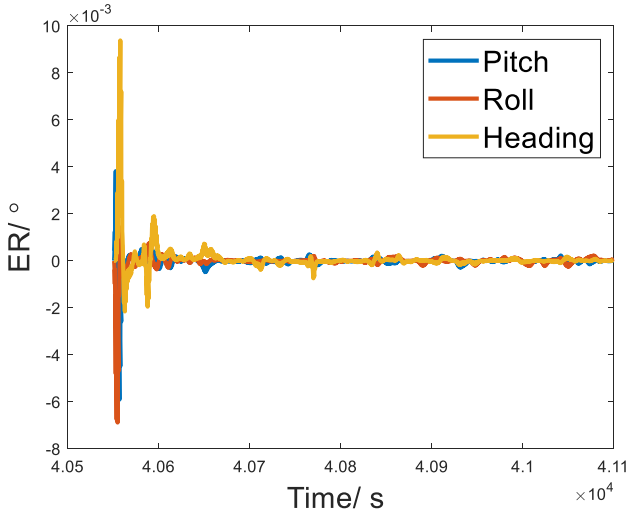
Figure 7.34 External Reliability values of the undetected outlier within EKF LiDAR based measurement models (a) Latitude difference, (b) longitude difference, (c) height difference toward velocity error states



(a)



(b)



(c)

Figure 7.35 External Reliability values of the undetected outlier within EKF LiDAR based measurement models (a) Latitude difference, (b) longitude difference, (c) height difference toward attitude error states

7.6 Summary

This Chapter presents a LiDAR/GNSS/INS/Map based Localization and mapping system for autonomous driving. LiDAR based SLAM technology is applied in the proposed system. The proposed LiDAR/GNSS/INS/Map system contains two steps. Firstly, LiDAR frame-to-frame Odometry mapping is aided with the optimal GNSS/INS trajectory to generate an offline map. Pose-graph optimization is used to optimize the final trajectory and point cloud map with enough number of loop closure. Secondly, the pre-generated offline map can be used as a 'sensor' to support the online localization at the same driving environment to provide high accuracy and robust position information, even during the period in which GNSS signal is lost.

An urban road testing is undertaken to test the proposed mapping and localization system. A point cloud map with clear road structure can be generated offline with this real-world dataset, which is then used to support the online localization for a car with driving in the same urban road.

There are two ways of doing LiDAR/map based localization, one is based on INS pose aided scan/map matching, another is under an error-state EKF system, thus LiDAR/map results will be used to feed back the inertial navigation system.

Due to the lack of ground truth information, the online localization LiDAR/map trajectory results are compared to the GNSS/INS results during the periods in which the GNSS RTK status is integer-ambiguity fixed. The results show that the first INS posed aided map-based LiDAR localization can achieve about 10-20cm accuracy, but with some big jumps which may be caused by outliers. For the EKF based system, centimeter lever accuracy can be achieved for the integration of the INS with both the high frequency LiDAR measurements and the lower frequency GNSS measurements.

A simulation of GNSS signal blockage has been tested, demonstrating that the LiDAR/map aided system are more stable and smoother during the period that GNSS signal is in bad conditions, indicating the LiDAR and map-matching based localization method is superior in the complex urban area.

Outliers within the LiDAR-Map-matching based position measurements have been detected for this road test. It is found that any dynamic objects, such as other moving vehicles on the road nearby, or sharp changing of road environments may cause outliers in the position measurements. The proposed EKF based localization system has the ability to detect outliers around value of 1 meter in the LiDAR-Map-matching based position measurements and 0.17 meters in the GNSS-RTK position measurements. Fortunately, the undetected outlier will only cause small position drifts. Any outliers above the MDB values can be detected with the defined success rate (power of the statistical testing) and their influence upon the final estimation can be eliminated with the correction by our quality control method.

Chapter 8 Conclusions and Recommendations

8.1 Conclusions

8.1.1 Quality Control Framework for optimal Filtering and Smoothing

Since smoothing estimation has very much potential for providing accurate and robust solution for various fields, such as navigation, a new Kalman smoothing based quality control procedure has been proposed to assist the smoothing method to deal with the faulty measurements and dynamic variations. The extended Kalman filter-smoother quality control method is developed according to the equivalence between the Kalman smoother and full least squares for a dynamical system. This quality control method is for the first time that applying the outlier detection, identification method and Reliability analysis for the smoothing process.

The numerical case studies in Chapter 2 have demonstrate the equivalence between the proposed smoothing method and the full least squares method, also the quality control results show these approaches are more robust to outliers, and have better reliability than the Kalman filter approaches. Other than the observation measurement model, the dynamic models can also be directly tested with the smoothing based quality control procedure, whereas the filtering-based procedure can only test the predicted states' quality.

A further variant of least squares quality control procedure (ULS QC) is developed for dealing with systems with ill conditioned matrix. For some practical applications, the system matrix may be singular, or constrained, such as the SLAM problem with constraint of loop closure. Since the condition number of the ill conditioned matrix are sometimes huge, it is sensitive to the inverse calculation, which may lead to wrong estimation solution and quality control results. With the proposed method, one new full rank covariance matrix is introduced to replace this ill conditioned matrix. By comparing the

three approaches, KS QC, ULS QC and FLS QC with SDV, with a simulated constrained system, the results demonstrate that the ULS method can successfully solve this singular system and is equivalent to the KS method under this special condition.

For the future test, the proposed quality control method can be used to analyse how geometry changes in the measurement setup could affect the Kalman filter-smoother estimation. Therefore, the analysis of geometry influence with quality control method can provide reference for the further development of dynamical system to acquire appropriate geometric strength components and estimation quality.

However, one drawback of the smoother method is that the KS method is generally applied as post-processing as it is a reanalysis process with measurement and dynamic information in future epochs. Therefore, for applications that need real time or near real time estimation and control, such as autonomous driving localization and navigation, the smoother method can only be used when a post data analysis is permitted, or a slightly delayed estimation is acceptable. For instance, this method can be applied for the offline map building to achieve better accuracy and quality, and then utilized this generated map to assist future real time positioning task. Therefore, more possible applications of the Kalman filter-smoother procedure and our extended quality control method was investigated in the following studies, such as achieving simultaneous localization and mapping (SLAM) with smoothing.

8.1.2 GNSS/INS Integration with Quality Control

Chapter 3 has discussed the application of the proposed EKF/KS system with quality control to some positioning strategies, such as RTK, PPK, and GNSS/INS with a real-world urban driving dataset. The smoothing based post processing GNSS double differencing method (PPK) can improve the estimation accuracy, precision, and reliability by utilizing a converged stable ambiguity resolution.

Different outlier influence mitigation strategies are also discussed in Chapter 3. The estimated influence of the identified outliers can be used to directly adapt to the final estimation results, or it can be used to correct the input measurement models. For most of

the cases, the mitigation method with outlier Adaption or outlier Correction will achieve similar final estimation results. However, since both the predicted state model and the measurement model may contain outliers, the predicted states may be illuminated by the outliers existing in the previous observations. If the system cannot successfully distinguish the outliers in the measurement model and the Kalman filter predicted state models, such as in some cases in which the outlier detection statistics for these models are highly correlated, just like the GNSS/INS loosely coupled integration system, misidentifications may happen. Therefore, the KF based quality control method may have mis-correction of these outliers which is not within the observation models, thus will influence the accuracy of the backward smoothing results.

8.1.3 Quality Control for Online and Offline SLAM

A quality control method is proposed in Chapter 4 and Chapter 5 for SLAM problems to deal with outliers within observation measurements and dynamic models and to evaluate the system reliability. This method can be applied for both online SLAM and offline post-processing. Kalman smoother SLAM is newly introduced in this study that can do estimation and quality control recursively and replace the full batch optimization based SLAM approach for large scale datasets.

Outliers within the measurements and the system model may seriously affect the performance of the SLAM approach and cause a failure of SLAM estimation. The quality analysis for two standard and widely used real-world SLAM datasets has demonstrated that outlier detection test statistic can successfully detect and identify the outlier within the measurements. The influence of the outlier on the final estimation can be mitigated by correcting the influence of the identified outlier with its estimated value. The results of the two tested datasets show that the proposed method is effective especially for the case in which it is hard to achieve drift correction by loop closure.

Reliability is an important system quality which is the capability of a system to detect outliers and evaluate the influence of the outliers towards final estimation. The reliability of the SLAM algorithm is analysed with Internal and External Reliability. The results show the reliability of a SLAM system depends on the geometry, redundancy, feature

revisiting and loop closing of the system. A detection of a long-term unobserved feature will have a high MDB value. Thus, an outlier within its observation will cause significant influence upon the SLAM estimation.

Three SLAM estimation methods are tested: 1) the Extended Kalman Filter SLAM, 2) Kalman Smoothing (KS) SLAM and 3) Full Least Square (FLS) SLAM. The performance of these three different SLAM frameworks can be compared with analysing the quality of the system. Therefore, the proposed quality control method can be applied as a benchmark method that gives quantified measure of the quality of different SLAM algorithms for evaluation and comparison. According to the results, Kalman smoothing (KS) SLAM and Full Least Squares (FLS) SLAM are equivalent and have the same results of both estimation and quality control if the FLS SLAM uses the EKF predicted state value as the initial approximate value. The offline systems are more reliable than filter-based system since they can detect outliers with lower value within both observation and odometry information, and achieve lower influence towards the final estimation results. The EKF SLAM has an advantage that it can detect the outlier and remove its influence online, the influence of the detected outlier to the following trajectory's prediction will be mitigated, while for smoothing or full least squares, the outlier detection can only be undertaken when all the data association has been done, this may contain some wrong statistic test results due to the contamination by the outlier, especially in the case that has high correlation between the outlier detection statistics for the odometry models. Thus, a multi-outlier detection and identification method should be utilized in this kind of cases after acquiring the possible number and location of the likely outliers.

This proposed quality control method can also be served to the SLAM designer at early stage of SLAM system application, as the reliability analysis can be used to evaluate the system based on the given system structure, such as the graph structure of full least squares SLAM, without having the real measurements available. For instance, the reliability results show that keeping frequently small loop closures can significantly increase the reliability of the EKF SLAM, which makes this online method suitable for large scale dataset with acceptable estimation and be less sensitive to the measurement outlier. Thus, more loop closures can be designed for a SLAM application when online

estimation is needed. The number and structure of the loop closure and the geometry of the measurement can also be chosen with analysing the possible outlier influence to match the requirement of the specific applications.

This study has demonstrated that the proposed quality control methods can be successfully utilized for both online and offline SLAM problem. It also has the potential to carrying out for delayed SLAM when best estimation is required, and additional computation is tolerable for every on-line step. Part of the variables can be marginalized out with the fixed-lag smoother approaches and quality control can be conducted within the smoothing lag.

8.1.4 HD Map based Localization with Quality Control and Geometry Analysis

High-definition (HD) map which is a reality-based high density map that provides high precision intelligence on the road features, may be an optimal selection to fulfill the range and accuracy requirements of autonomous driving. The main approach of generating and utilizing HD map is based on SLAM. This is suffered from the high cost of computation, storage space and transmission rate and limits the SLAM approach from online HD map application. However, HD map still has a potential to service as a sensor source to provide road feature location information and to support accurate and efficient online localization for autonomous driving.

Chapter 6 investigates a localization method that utilize the HD map as on board sensor to integrate with other perception sensors. MDB, MSB and correlation coefficients are utilized to support the explanation of the performance of outlier detection, identification, exclusion method in a case study.

The geometry will significantly affect accuracy of localization, mapping and navigation solution and performances of outlier statistics, including detectability of observations and reliability of outlier statistics. Therefore, the influence of the system geometry for this HD map-based localization system is essential to be comprehensively analysed. Geometry factors like feature distribution type, number of detected feature and the distance between the vehicle and the features were taken into consideration together in

this study. MDB and MSB are mostly influenced by the last two factors. Sparse random distribution of feature will contribute to decreasing MDB and ER value. More detected features and close distance between the host vehicle and the features may also contribute to good quality of vehicle position estimation.

According to the results, ER value seems more sensitive to the geometry change than the MDB and MSB. The results show with slight change of MDB and MSB value, the influence of the undetectable outliers upon the final position estimation are changing more dramatically. Therefore, the External Reliability should attract more attention when evaluating and judging the potential influence of the geometry when designing a HD map localization system.

The position estimated in this study is the relative position of the host vehicle within the HD map, not within the reality environment. As all the road network system is displayed within the HD map, the relative location between the host vehicle and the road network can also be estimated, such as the distance between the vehicle and the stop line. Therefore, this relative location information can be used to perform vehicle navigation. However, when there is a moving entity which is detected by the external sensor in real time but not displayed in the HD map, for example, another vehicle in front of the host vehicle, the relative location information should be estimated with considering the error of the HD map. In this situation, the influence of geometry on the relative location is also of importance to be studied and will be investigated in the future study.

When the vehicle is driving along the road, there may be situations where the external sensors cannot detect enough number of features with desired distribution type at an appropriate distance, therefore, the quality of vehicle localization would not be acceptable. As this study only consider laser and HD map as sensors for vehicle localization, other sensors such as camera, IMU, may be added into our HD map based vehicle localization system to overcome the problem of low geometric strength.

In this part of our study, the 3D features are treated as individual known points with the HD map, while such 3D features may not be enough, 3D point cloud-based features may be useful for localization as has been demonstrated in Chapter 7. In the future study, the

geometry influence of other sensors and the combination of different sensors will be further explored.

8.1.5 LiDAR/GNSS/INS integrated Mapping and Localization

Since each sensor and technique have their own advantages and drawbacks, the combination of multi sensors and different localization techniques are essential for fulfilling the high requirements of autonomous driving. A LiDAR/GNSS/INS integrated mapping and localization method is proposed in Chapter 7.

A LiDAR/GNSS/INS sensor system was set up on a vehicle and road testing was conducted for this study. SLAM algorithm can be used to generate an offline high density road map with the collected real-world dataset. The procedure of the map building is discussed in Chapter 7, including the LiDAR Odometry, GNSS/INS aided map matching, geo-referencing, pose-graph optimization. A road point cloud map with around 800 meters of driving trajectory was built up for further analysis of the localization and mapping performances.

Two localization systems are proposed by combining the LiDAR, GNSS, inertial navigation with the pre-generated HD map. One is directly using GNSS/INS pose solution to support LiDAR/Map matching based localization, the other one is an EFK based system that integrates Inertial navigation with both LiDAR and GNSS measurements. Estimation results show the proposed two online localization systems can both successfully and precisely locate the host vehicle with the aiding of LiDAR/map matching, especially during periods of GNSS single blockage. Therefore, this system is suitable for driving within a complex urban road environment. The EKF based system's solution has better accuracy, especially during period with outliers.

Quality of these two localization systems is analysed. It is found that other moving or changing entities along the road environment may be a source of outliers. Therefore, the dynamic entities should be detected and removed before conducting localization. The EKF based quality control method is used to quantitative test and monitor the performance of the proposed EKF based LiDAR/GNSS/INS/Map localization system.

Efficiency of the online localization can be improved by applying a proposed dynamic local map loading method. The local point cloud map is selected and extracted from the global map with the information of current GNSS/INS information and pre-set LiDAR distance threshold. Although the proposed online localization method can generate online point cloud map at the same time, the new map can be chosen not to be updated or stored for the purpose of efficiency. However, if for a more mature autonomous driving system, frequently updating of the global HD map with the dynamic road information from all of the available real-time map building candidates (automatic cars driving on the road) is necessary to support the safety and efficiency of the whole smart transportation system. Therefore, a cleverer online mapping and localization updating system should be considered for autonomous driving, such as updating and storing the real-time map on Cloud to avoid local platform calculation and storage burden.

8.2 Recommendations

Based on the research made in this thesis, the following recommendations can be made:

8.2.1 Near-real Time Optimization Estimation and Quality Control

According to the theorical analysis and numerical results, the smoothing based results are equivalent to that of the full batch update of least squares, and can achieve estimation with better robustness and reliability than the filtering method. However, this optimal solution can only be achieved offline since it uses the whole available measurements and models. For some applications that accept near real-time or small delayed optimal solution, the smoothing method can also be suitable to apply. For instance, a submap can be optimized with quality control firstly, and the optimal results can be used to enhance the estimation once loop closed for this optimized sub-map. In future, the application of smoothing based quality control for near real-time application should be studied.

8.2.2 Iterative Kalman Filtering and Smoothing

In this thesis, the equivalence between the Kalman smoothing and full least squares for a dynamic system has been proved by different localization or mapping systems. However,

it should be noticed that the least squares method can achieve optimal estimation results after a few iterations for the non-linear systems. If the initial approximate values for the variables are poor, it will take more iterations to achieve convergency and optimal solutions. For the proposed KS method, the EKF predicted state value is used as the initial approximate value. However, if the EKF stage does not converge, the initial KS results will not meet the final least squares results. Therefore, the iterative KF/KS system needs further investigations to achieve optimal solutions for various complex application scenarios. The influence of the iteration and the quality control results should also be evaluated and analysed for better understanding of the system.

8.2.3 Multiple Fault Detection for Smoothing based Localization and Mapping

The EKF based system can corrected the outlier immediately after being detected, and its influence upon the next epochs' estimation will be largely mitigated. However, for smoothing based system, since the estimation is undertaken offline with all the available models, the outliers within a model will contaminate the statistic testing of all other models. When the Correlation Coefficient between the outlier detection statistic for these models are low, the proposed single outlier detection, identification and correction or adaption method can work well to solve the outlier issue, such as outliers existing in the measurements of GNSS/INS system. However, if these models are highly correlated, such as the dynamic model in a SLAM system with no loop closure, the contamination of the outlier will mislead the quality control method, especially when there are multi-outliers occur. Therefore, a multi-outlier detection, identification and correction method is needed to be further investigated for this situation. Some earlier study on such topics may be found in, e.g., Wang and Knight (2012) and Knight et al. (2010).

8.2.4 Efficiency of HD Map Updating

Although the HD map can support accurate online localization, the dynamic of road environment will still influence the localization accuracy. An efficient HD map updating strategy is needed to upload timely information on some environmental changing, such as changing of parking entities, to ensure the autonomous driving. Since the proposed multi-sensor localization system has the ability to store the aligned point cloud with the

trajectory estimation, the accumulated point cloud map may be optimized and uploaded after a few driving periods and when parked. The procedure of this efficient HD map updating method will need investigation too.

References:

Asmar, D., 2006, *Vision-inertial SLAM using natural features in outdoor environments.* Dissertation, University of Waterloo, Canada, 2006.

Agarwal, P., 2015, *Robust Graph-based localization and mapping*. PhD thesis, 2015.

Agarwal, P., Burgard,W., and Stachniss, C., 2014, 'Geodetic approaches to mapping and relationship to graph-based SLAM'. in *IEEE Robotics & Automation Magazine*, vol. 21, no. 3, pp. 63-80, Sept. 2014.

Angelats, E., Colomina, I., 2014, One step mobile mapping laser and camera data orientation and calibration. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 40–3/W1, 15–20.

Alquarashi, M., Wang, J., 2015, 'Performance Analysis of Fault Detection and Identification for Multiple Faults in GNSS and GNSS/INS Integration'. *Journal of Applied Geodesy,* vol.9, pp. 35-48DOI: 10.1515/jag-2014-0019

Aravkin, A., Bell, B. B., Burke, J. V., & Pillonetto, G. 2013. 'Kalman smoothing and block tridiagonal systems: new connections and numerical stability results'. *arXiv preprint* arXiv:1303.5237.

Aravkin, A., Bell, B., Burke, J., Pillonetto, G., 2011 'An l1-Laplace robust Kalman smoother'. *IEEE Transaction on Automatic Control* vol.56, pp.2898-2911.

Auger, F., Hilairet, M., Guerrero, J., Monmasson, E., Orlowska-Kowalska, T., Katsura, S., 2013, ' Industrial Applications of the Kalman Filter: a review'. *IEEE Transactions on Industrial Electronics* vol.60, pp.5458-5471.

Baarda, W., 1968 'A testing procedure for use in geodetic networks'. *Netherlands Geodetic Commission*, New Series 2(4).

Bailey, T., 2002, *Mobile Robot Localisation and Mapping in Extensive Outdoor Environments*. PhD thesis, University of Sydney, Australian Centre for Field Robotics, 2002.

Bailey, T., Nieto, J. and Nebot, E., 2006, Consistency of the FastSLAM algorithm. Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006., Orlando, FL, USA, 2006, pp. 424-429, doi: 10.1109/ROBOT.2006.1641748.

Baritzhack, I.Y., Berman, N., 1988, 'Control theoretic approach to inertial navigation systems'. *J. Guid. Control Dyn.* 1988, 11, 237–245.

Bresson, G., Aufrere, R., Chapuis, R., 2015, 'A general consistent decentralized SLAM solution'. *Robotics and Autonomous Systems*, no.74, pp.128-147, 2015.

Bresson, G., Alsayed, Z., Glaser, L. Yu, S.,2017, 'Simultaneous localization and mapping: a survey of current trends in autonomous driving'. *IEEE Transactions on intelligent vehicles*, vol. 2, no. 3, pp. 194-220, Sept. 2017.

Bosse, M., Agamennoni, G., Gilitschenski, I., 2013 'Robust Estimation and Applications in Robotics'. *Foundations and Trends in Robotics*, vol. 4, pp. 225-269/

Cadena, C., and Neira, J., 2009, SLAM in O(log n) with the Combined Kalman - Information filter. 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, 2009, pp. 2069-2076.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., Leonard, J.J., 2016, 'Past, present, and future of Simultaneous Localization and Mapping: Toward the robust-perception age' . *IEEE Transactions on Robotics* vol.32, pp.1309-1332.

Carlone, L., and Calafiore, G. C., 2018, 'Convex Relaxations for Pose Graph Optimization With Outliers', in *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1160-1167, April 2018, doi: 10.1109/LRA.2018.2793352.

Carlone, L., Censi, A. and Dellaert, F., 2014, Selecting good measurements via ℓ1 relaxation: A convex approach for robust estimation over graphs. 2014

IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 2014, pp. 2667-2674, doi: 10.1109/IROS.2014.6942927.

Chen, C., Zhu, H., 2018, 'Visual-inertial SLAM method based on optical flow in a GPS-denied environment'. *Ind. Robot Int. J*. 2018, 45, 401–406.

Chen, C., Zhu, H., Li, M., You, S., 2018, 'A review of visual-inertial simultaneous localization and mapping from filtering-based and optimization-based perspectives'. Robotics, vol.7, 45, 2018.

Cheng, S., 2017, Quality analysis for satellite bias estimation and GNSS PPP ambiguity resolution. Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017) September 25 - 29, 2017 Oregon Convention Center Portland, Oregon, pp 2219-2234, 2017.

Chli, M. and Davison, A. J., 2009, 'Automatically and efficiently inferring the hierarchical structure of visual maps', in *IEEE International Conference on Robotics and Automation*, pp. 387–394.

Chiang, K., Duong, T., Liao, J., Lai, Y., Chang, C., Cai, J., Huang, S., 2012, 'On-line smoothing for an integrated navigation system with low-cost MEMS inertial sensors'. *Sensors* vol.12, pp.17372-17389.

Cipra, T., Romera, R., 1997, 'Kalman filter with outliers and missing observations'. *Test* vol.6 pp.379-395.

Colomina, I., Blazquez, M., 2005, On the stochastic modeling and solution of time dependent networks. 2005

Cosme E, Verron J, Brasseur P, Blum J, Auroux D, 2012, 'Smoothing problems in a Bayesian framework and their linear Gaussian solution'. *Mon Wea Rev* vol.140, pp. 683-695.

Cross, PA.,1994, *Advanced least squares applied to position fixing: working papers*. University of East London, School of Surveying.

Cucci, D.A., Rehak, M., Skaloud, J., 2017. 'Bundle adjustment with raw inertial observations in UAV applications'. *ISPRS J. Photogramm. Eng. Remote Sens*. 130, 1–12. https://doi.org/10.1016/j.isprsjprs.2017.05.008

Dellaert, F., Kaess, M., 2017, 'Factor graphs for robot perception'. *Foundations and Trends in Robotics*, vol. 6, no. 1-2, pp. 1-139.

Dedeoglu, P., Goel, G., Roumeliotis, S.I., and Sukhatme, G.S., 2000, Fault detection and identification in a mobile robot using multiple model estimation and neural network, in IEEE International Conference on Robotics and Automation, 2000, pp. 2302–2309.

Dissanayake, G., Huang, S., Wang, Z., and Ranasinghe, R., 2011, 'A review of recent developments in simultaneous localization and mapping', in Proc. Int. *Conf. Ind. Inform. Syst.*, pp. 477–482.

Durrant-Whyte, H., and Bailey, T., 2006, 'Simultaneous localization and mapping:part I'. *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99-110, June 2006.

Eade, E. and Drummond, T., 2006, 'Scalable monocular SLAM', in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 469–476.

Einicke, G., 2012, *Smoothing filtering and prediction: Estimation the past, present and future*. Rijeka, Croatia: InTech.

El-Mowafy, A., Imparato, D., Rizos, C., Wang, J, Wang, K., 2019,'On hypothesis testing in RAIM algorithms: Generalized likelihood ratio test, solution separation test and a possible alternative'. *Measurement Science and Technology*, vol. 30, 2019, http://dx.doi.org/10.1088/1361-6501/ab1836

Fahrmeir, L., Kaufmann, H., 1991 'On Kalman filtering, posterior mode estimation, and Fisher scoring in dynamic exponential family regression'. *Metrika*, pp 37–60.

Falquez, J.M., Kasper, M., Sibley, G., 2016, Inertial aided dense & semi-dense methods for robust direct visual odometry. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Daejeon, Korea, 9–14 October 2016; pp. 3601–3607.

Fang, W., Zheng, L., Deng, H., Zhang, H., 2017, 'Real-Time Motion Tracking for Mobile Augmented/Virtual Reality Using Adaptive Visual-Inertial Fusion'. *Sensors* 2017, 17, 1037.

Farrell, J.A., Givargis, T.D., Barth, M.J., 2000, 'Real-time differential carrier phase GPS-aided INS'. *IEEE Trans. Control Syst*. Technol. 2000, 8, 709–721.

Farrell, J. A. M. Todd, M. Barth, 2016, 'Best Practices for Surveying and Mapping Roadways and Intersections for Connected Vehicle Applications'. UC Riverside: Bourns College of Engineering, 2016. Retrieved from: http://escholarship.org/uc/item/4f88m75k

Farrell, J.A., Wendel, J., 2017, 'GNSS/INS Integration'. In: Teunissen PJG, Montenbruck O (eds) Chapter 28 in Handbook of Global Navigation Satellite systems, pp 834

Fauske, K. M., Hallingstad, O., Hegrenas, O., 2008, Toward a framework for high integrity navigation of autonomous underwater vehicles. IFAC Proceedings Volumes, 2nd IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles, vol. 41, pp. 184-189, 2008.

Fei, Y., 2012, *Real-Time Detecting and Tracking of Moving Objects using 3D LIDAR*, Zhejian University, Hangzhou, China, 2012.

Förstner, W., 1983, 'Reliability and Discernability of Extended Gauss-Marko Models', Deutsche Geodätische Kommission, Reihe A, No. 98, Muchen, 1983.

Fuentes-Pacheco, J., Ruiz-Ascencio, J., Rendon-Mancha, J. M., 2015, 'Visual simulataneous localization and mapping: a survey'. *Artifical Intelligence Review*, vol. 43, pp. 55-81.

Garcia, J., Mercadr, P., Muravchik C, 2005, 'Use of GPS carrier phase double differences'. *Lat.Am.appl.res*, vol. 35, pp.115-120, 2005.

Gargiulo. G., Leonardi. M., Zanzi. M., Varavalli. G., 2010, GNSS Integrity and Protection level computation for vehicular applications. In Proc. Of 16th Ka and Broadband Communications Navigation and Earth Observation Conference, 569-574, 2010.

Grisetti, G., Stachniss, C., and Burgard, W., 2007, 'Improved techniques for grid mapping with Rao-Blackwellized Particle Filters', in *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34-46, Feb. 2007.

Grisetti, G., Kümmerle, R., Stachniss, C., Burgard, W., 2010, 'A tutorial on graph-based SLAM', *In IEEE Intelligent Transportation Systems Magazine*, vol.2, no.4, pp. 31-43, 2010.

Groves, P., 2013, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition* , Artech, 2013

Guivant, J., and Nebot, E., 2001, 'Optimization of the Simultaneous Localization and Map Building Algorithm for Real Time Implementation'. *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, vol.17, pp.242-257;

Guivant, J.E, Masson, F.R., Nebot, E.M., 2001, 'Simultaneous localization and map building using natural features and absolute information'. *Robotics and Autonomous Systems*, vol.*40*, pp.79-90.

Han, H., Wang, J., Wang, J., Tan X, 2015, 'Performance Analysis on Carrier Phase-Based Tightly-Coupled GPS/BDS/INS Integration in GNSS Degraded and Denied Environment '. *Sensor* , vol.15, pp.8685-8711, 2015.

Hauschild,A., 2017, 'Basic Observation Equations'. In: Teunissen PJG, Montenbruck O (eds) Chapter 28 in Handbook of Global Navigation Satellite Systems, pp 561, 2017.

Hemann, G., Singh, S., and Kaess, M., 2016, Long-range gps-denied aerial inertial navigation with lidar localization, in Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016, pp. 1659–1666

HERE, 2017, 'HERE HD Live Map- the most intelligent vehicle sensor'. [Online] obtain from: https://here.com/en/products-services/products/here-hd-live-map at 19/03/2017

Hewitson, S., and Wang, J., 2010, 'Extended receiver autonomous integrity monitoring (eRAIM) for GNSS/INS integration'. *Journal of Surveying Engineering*, vol.136(1), pp. 13-22, DOI: 10.1061/(ASCE)0733-9453(2010)136:1(13)

Hewitson., S,, Wang. J., 2006 'GNSS receiver autonomous integrity monitoring (RAIM) performance analysis'. *GPS Solution* vol.10, pp.155-170.

Ho, B., Sodhi, P., Teixeira, P., Hsiao, M., T. Kusnur and M. Kaess, 2018, Virtual Occupancy Grid Map for Submap-based Pose Graph SLAM and Planning in 3D Environments, 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, 2018, pp. 2175-2182.

Hofmann-Wellenhof, B., Lichtenegger, H. and Collins, J., *Global positioning System: Theory and Practice*.

Hosseinyalamdary, S., Balazadegan, Y., Toth., C.K., 2015, 'Tracking 3D moving objects based on GPS/IMU navigation solution, laser scanner point cloud and GIS data'. *International Journal of Geo-information*, vol:4, pp:1301-1316. 2001, DOI: 10.3390/ijgi4031301

Huang, S., Lai, Y., Frese, U., Dissanayake, G., 2010, How far is SLAM from a linear least squares Problems? the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3011-3016, October 18-22, 2010, Taipei, Taiwan.

Huang, S., and Dissanayake, G., 2007, 'Convergence and consistency analysis for extended Kalman filter based SLAM'. *IEEE Trans Robot*, vol. 23, No.5, pp. 1036-1049.

Huang, S., Wang, Z., Dissanayake, G., & Frese, U., 2009, 'Iterated D-SLAM map joining: Evaluating its performance in terms of consistency, accuracy and efciency'. *Autonomous Robots*, vol 27 no.4, pp:409–429, 2009.

Huber, P., 1981, *Robust Statistics*, John Wiley and Sons, 1981

Ila, V., Polok, L., Šolony, M., Smrz. P., Zemcik, P., 2015, Fast covariance recovery in incremental nonlinear least square solvers. 2015 IEEE international conference on robotics and automation (ICRA), Seattle, WA, 2015, pp.4636–4643.

Ila, V., Polok, L., Solony, M., and Svoboda, P., 2017, 'SLAM++-A highly efficient and temporally scalable incremental SLAM framework'. *The International Journal of Robotics Research*, 36(2), 210–230, 2017.

Im, G., Kim, M., Park, J., 2019, 'Parking line based SLAM approach using AVM/LiDAR sensor fusion for rapid and accurate loop closing and parking space detection'. *Sensors*, *19*, 4811.

International Civil Aviation Organization, 2006, Aeronautical telecommunications: International standards and recommended practices: Annex 10 to the Convention on International Civil Aviation. Vol. I. Radio Navigation Aids, 6th ed. Montreal, Quebec, Canada: International Civil Aviation Organization, 2006.

James, M., 'The Generalised Inverse'. *The Mathematical Gazette*, vol.62, no.420, pp. 109-114, 1978.

Jo, K., Kim, C., Sunwoo, M., 2018, 'Simultaneous localization and map change update for the high definition map-based autonomous driving car'. *Sensor*, vol. 18(9), pp.3145.

Kaess, M., Ranganathan, A., and Dellaert, F., 2008, 'iSAM: Incremental Smoothing and Mapping'. *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1365-1378, Dec 2008.

Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J.U., Dellaert, F., 2012, 'iSAM2: Incremental smoothing and mapping using the Bayes tree'. *The International Journal of Robotics Research*, vol.31, no.2, pp.216–235.

Kaniewski, P., Gil, R., Konatowski, S., 2017, 'Estimation of UAV position with use of smoothing algorithms'. *Metrology and Measurement System* vol.24 pp.127-142.

Kaplan, E., and Hegarty, C., (Editors), 2006, *Understanding GPS: Principles and Applications*, 2nd ed. Norwood Massachusetts, USA: Artech House

Katrakazas, C., Quddus, M., Chen, W.H., Deka, L., 2015, 'Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions'. *Transportation Research* Part C, vol. 60, pp. 416-442, URL: https://doi.org/10.1016/j.trc.2015.09.011.

Kim, C., Cho, S., Sunwoo, M., Jo, K., 2018, 'Crowd-sourced mapping of new feature layer for high-definition map'. *Sensor*, vol, 18,1472, 2018.

Kitt, B., Geiger, A., and Lategahn, H., 2010, Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme. 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, 2010, pp. 486-492.

Klein, G., Murray, D., 2007, 'Parallel tracking and mapping for small AR workspaces'. Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on Nara, pp. 225-234.

Knight, N. L., Wang, J., and Rizos, C., 2010 'Generalised Measures of Reliability for Multiple Outliers'. *Journal of Geodesy*, vol.84(10), pp.625-635.

Kummerle, R., Hahnel, D., Dolgov ,D., Thrun, S. and Burgard, W., 2009a, Autonomous driving in a multi-level parking structure. 2009 IEEE International Conference on Robotics and Automation, Kobe, pp. 3395-3400, 2009a.

Kummerle, R., Steder, B., Dornhege, C., Ruhnke, M., Grisetti, G., Stachniss, C., Kleiner, A., 2009b, 'On measuring the accuracy of SLAM algorithms'. *Autonomous Robots*, vol. 27, pp. 387-407, 2009b.

Kurlbaum, J., & Frese, U., 2009, A benchmark data set for data association, Technical Report, University of Bremen, 2009 available online: http://www.sfbtr8.uni-bremen.de/reports.htm, Data available online http://radish.sourceforge.net/ .

Kuutti, S., Fallah, S., Katsaros, K., Dianati, M., Mccullough, F., Mouzakitis, A., 2018 'A survey of the state-of -the-art localization techniques and their potentials for autonomous vehicle applications'. *IEEE Internet of Things Journal*, vol.5, no.2, pp.829-846, April 2018.

Larson, C. D., 2010, *An Integrity Framework for Image-Based Navigation Systems*. PhD dissertation for Graduate School of Engineering and Management, Air Force Institute of Technology, Air University, Wright-Patterson Air Force Base, Ohio, 2010.

Latif, Y., Cadena, C., & Neira, J., 2013, 'Robust loop closing over time for pose graph SLAM'. *The International Journal of Robotics Research*, vol.32, no.14, pp. 1611–1626, 2013.

Lee, G. H., 2014, *Visual Mapping and Pose Estimation for Self-Driving Cars*. Thesis for Doctor of Science, ETH Zurich, 2014.

Lee, H., 2010, 'An integration of GPS with INS sensors for precise long-baseline kinematic positioning'. *Sensors* 2010, 10, 9424–9438.

Lee, H., Chun, J., and Jeon. K., 2018, Autonomous back-in parking based on occupancy grid map and EKF SLAM with W-band radar. 2018 International Conference on Radar (RADAR), Brisbane, QLD, 2018, pp. 1-4.

Levinson, L and Thrun, S., 2010, Robust Vehicle Localization in Urban Environments Using Probabilistic Maps. 2010 IEEE International Conference on Robotics and Automation (ICRA), pp. 4372-4378.

Li, H., and Zhijian, L., 2010, The study and implementation of mobile GPS navigation system based on Google Maps. 2010 International Conference on Computer and Information Application, Tianjin, 2010, pp. 87-90, doi: 10.1109/ICCIA.2010.6141544.

Li, H., Nashashibi, F., 2012, Multi-vehicle cooperative localization using indirect vehicle-tovehicle relative pose estimation, in IEEE International Conference on Vehicular Electronics and Safety, pp. 267–272, 2012.

Li, L., and Yang, M., 2016, Road dna based localization for autonomous vehicles, in 2016 IEEE Intelligent Vehicles Symposium(JV), pp. 883- 888,IEEE, 2016.

Li, Q., Song, Y, Hou., Z., 2015, 'Neural network based FastSLAM for autonomous robots in unknown environments'. *Neurocomputing*, vol. 165, pp. 99-110, 2015.

Li, Z., Wang, J., Alquarashi, M., Chen, K., Zheng, S., 2016 'Geometric analysis of reality-based indoor 3D mapping'. *The Journal of Global Positioning Systems* 14:1.

Li, Z., Wang, J., Chen, K., Sun, Y., 2017, 'New environmemtal line feature-based vision navigation: design and analysis'. *Journal of Navigation* vol.70, pp.1133-1152.

Litman, T., 2015, Autonomous Vehicle Implementation Predictions: Implications for Transport Planning. Transportation Research Board 94th Annual Meeting. Washington DC, United States.

Liu, R., Wang, J., and Zhang, B., 2020, ‘High definition map for automated driving, overview and analysis’. *Journal of navigation*, 2020, DOI: https://doi.org/10.1017/S0373463319000638

Lynen, S., Sattler, T., Bosse, M., Hesch, J., Pollefeys, M., Siegwart, R., 2015, Get Out of My Lab: Large-scale, Real-Time Visual-Inertial Localization. In Proceedings of the Robotics: Science and Systems, Rome, Italy, 13–17 July 2015.

Malis, E., Marchand, E, 2006, Experiments with robust estimation techniques in real-time robot vision. IEEE/RSJ Int.Conf. on Intelligent Robot and Systems, IROS’06, Beijing, China, pp223-228.

Martinez-Cantin, R., and Castellanos, J.A., 2005, Unscented SLAM for large-scale outdoor environments. 2005 IEEE/RSJ International Conference on. *International Conference on Intelligent Robots and Systems*, Edmonton, Alta., 2005, pp. 3427-3432.

Miller, W., ‘RoboSense develops $200 LiDAR system for autonomous vehicles’. Electronic Products, [available online]: https://www.electronicproducts.com/Automotive/RoboSense_develops_200_LiDAR_system_for_autonomous_vehicles.aspxMorales .

Molina, P., Silva, P.F., Bandeiras, J., Engenharia, D., Lisbon, P., Stebler, J., Skaloud, J., 2011, Integrity Aspects of Hybrid EGNOS-based Navigation on Support of Search-and-Rescue Missions With UAVs, Jan. 2011.

Morales, Y., Takeuchi, E., and Tsubouchi, T., 2008, Vehicle localization in outdoor woodland environments with sensor fault detection, in IEEE International Conference on Robotics and Automation, pp. 449–454, 2008.

Nebot, E. M., Durrant-Whyte, H., 1999, ‘A high integrity navigation architecture for outdoor autonomous vehicles’. *Robotics and Autonomous System*, vol. 26, pp. 81-97.

Ni, K., Steedly, D. and Dellaert, F., 2007, Tectonic SAM: exact, out-of-core, submap-based SLAM. Proceedings 2007 IEEE International Conference on Robotics and Automation, Roma, 2007, pp. 1678-1685.

Olsen, M. J., Roe, G. V., Glennie, C., et al., 2013, Guidelines for the use of mobile LiDAR in transportation applications. National Cooperative Highway Research Program (NCHRP) Report 748, Corvallis, OR, February, 2013.

Olson, E., and Agarwal, P., 2012, 'Inference on networks of mixtures for robust robot mapping', in : *Robotics: Science and Systems (RSS)*.

Paz, L., Piniés, P., Tardós, J.D., Neira, J., 2008, ' Large-scale 6DOF SLAM with stereo-in-hand'. *IEEE Trans Robot*, vol. 24, pp.946–957.

Pillonetto, G., Aravkin, A., Carpin, S.,2010, The unconstrained and inequality constrained moving horizon approach to robot localization. In: 2010 IEEE/RSJ international conference on intelligent robots and systems, Taipei, pp. 3830-3835.

Pinies, P., Paz, L. M. and Tardos, J. D., 2009, CI-Graph: An efficient approach for large scale SLAM. 2009 IEEE International Conference on Robotics and Automation, Kobe, 2009, pp. 3913-3920.

Press, W., Teukolsky, S., Vetterling, W., and Flannery, B., 1992, *Numerical Recipes*, 2nd Edition. Cambridge Univ. Press, 1992.

Qin, T., Li, P., Shen, S., 2017, 'VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator'. arXiv arXiv:1708.03852.

Rao, C. R., and Mitra 1971, 'Further Contributions to theory of Generalized Inverse of Matrices and its Applications'. *Sankhyā, A*, vol.33, pp. 289-300.

Realini, E., Reguzzoni, M., 2013, 'goGPS: open source software for enhancing the accuracy of low-cost receivers by single-frequency relative kinematic positioning'. *Measurement Science and Technology* 24(11):115,010.

Roumeliotis, S. I., Sukhatme, G. S. and Bekey, G. A., 1999, Circumventing dynamic modeling: evaluation of the error-state Kalman filter applied to mobile robot localization, Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C), Detroit, MI, USA, 1999, pp. 1656-1663 vol.2, doi: 10.1109/ROBOT.1999.772597.

Rousseeuw, P., and Leroy, A., 1987, *Robust regression and outlier detection*. John Wiley and Sons, 1987.

Rouzaud, D., Skaloud, J., 2011 'Rigorous Integration of Inertial Navigation with Optical Sensors by Dynamic Networks'. *Navig. J. Inst. Navig*. 58, 141–152, 2011.

Salzmanm, M., 1993 Least Squares filtering and testing for geodetic navigation applications. Netherlands Geodetic Commission, Publications on Geodesy, New Series, Vol. 37.

Sarkka S., 2013, *Bayesian Filtering and Smoothing*, Cambridge University Press, 2013

Seif, H. G., Hu, X., 2016, 'Autonomous driving in the iCity- HD maps as a key challenge of the automotive industry'. *Engineering,* vol. 2, pp. 159-162.

Seo, J., Walter, T., Enge, P., 2011, 'Availability impact on GPS aviation due to strong Ionospheric Scintillation'. In *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, pp. 1963-1973, July 2011, doi: 10.1109/TAES.2011.5937276.

Skog, I., and Händel, P., 2005, A low-cost GPS aided inertial navigation system for vehicle applications, *2005 13th European Signal Processing Conference*, 2005, pp. 1-4.

Smith, R., and Cheeseman, P., 1986, 'On the representation and estimation of spatial uncertainty'. *The International Journal of Robotics Research*, vol. 5, no. 4, pp. 56–68.

Speidel, J., Tossaint, M., Wallner, S., and Angel, Angelavila-Rodriguez J., 2013, 'Integrity for Aviation: Comparing Future Concepts,' *InsideGNSS*, vol. 2013, pp. 54–64, 01.07.2013.

Strasdat, H., Montiel, J., Davison, A.J., 2010. Real-time monocular SLAM: why filter? In: 2010 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 2657–2664.

Suganuma, N., Yamamto, D., Yoneda, K., 2015, 'Localization for autonomous vehicle on urban roads'. *Journal of Advanced Control, Automation and Robotics*, vol. 1, pp. 47-53.

Sünderhauf, N., and Protzel, P., 2012, Switchable constraints for robust pose graph SLAM, *2012 IEEE/RSJ* International Conference on Intelligent Robots and Systems, Vilamoura, 2012, pp. 1879-1884.

Sundvall, P., and Jensfelt, P., 2006 Fault detection for mobile robots using redundant positioning systems, in IEEE International Conference on Robotics and Automation, pp. 3781–3786, 2006.

Thrun, S, Montemerlo.M., Koller,D., Wegbreit.B., Nieto. J., Nebot, E.M., 2004, 'Fastslam: An efficient solution to the simultaneous localization andmapping problem with unknown data association'. *Journal of Machine Learning Research*, vol.4(3), pp.380-407.

Thrun, S., and Liu, Y., 2005, 'Multi-robot SLAM with Sparse Extended Information Filers'. *Springer Tracts in Advanced Robotics Robotics Research. The Eleventh International Symposium*. Vol 15, pp:254–266.

TomTom, 2017, 'RoadDNA, Robust and scalable localization technology', available online: obtained at 25/05/2019.

Tossaint, M., Samson, J., Toran, F., Ventura-Traveset, J., HernandezPajares, M., Juan, J. M., Sanz, J., and Ramos-Bosch, P., 2007, 'The Stanford - ESA Integrity Diagram: A New Tool for The User Domain SBAS Integrity Assessment'. *Navigation*, vol. 54, no. 2, pp. 153–162, 2007.

Teunissen, P., 2017, 'Batch and Recursive Model Validation'. In: Teunissen PJG, Montenbruck O (eds) Chapter 24 in *Handbook of Global Navigation Satellite Systems*, pp 187–229.

Teunissen, P., 1990, Quality control in integrated navigation systems. Proceedings of IEEE PLANS, Las Vegas, IEEE Press, New York, pp. 158-165.

Velodyne LiDAR, 'HDL-64E', [available online], http://velodynelidar.com/hdl-64e.html accessed by: 29/05/2017.

Vu, T.-D., 2009, *Vehicle perception: Localization, mapping with detection, classification and tracking of moving objects*, Ph.D. dissertation, Institut National Polytechnique de Grenoble-INPG, 2009.

Wagner, R., Frese, U., and Bauml, B., 2014, 'Graph SLAM with signed distance function maps on a humanoid robot,' in Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on *Intelligent Robots and Systems*, Chicago, IL,2014, pp. 2691–2698.

Wang, C.C., Thorpe, C., Thrun, S., Hebert, M., Durrant-Whyte, H., 2007, 'Simultaneous Localization, Mapping and Moving Object Tracking'. *The International Journal of Robotics Research*. 2007;26(9):889-916. doi:10.1177/0278364907081229

Wang, C.C., Thorpe, C.,and Thrun. S., 2003, Online simultaneous localization and mapping with detection and tracking of moving objects: Theory and results from a ground vehicle in crowded urban areas, in IEEE International Conference on Robotics and Automation, pp. 842–849, 2003.

Wang, J., Knight, N.L., 2012, 'New outlier separability test and its application in GNSS positioning'. *Journal of global positioning systems*, vol. 11, pp. 46 - 57, 2012, http://dx.doi.org/10.5081/jgps.11.1.46

Wang, J., and Ober, P.B., 2009, 'On the availability of Fault Detection and Exclusion in GNSS receiver autonomous integrity monitoring'. *Journal of Navigation*, vol. 62, no. 2, pp. 251-261, 2009.

Wang, J., 2008, 'Test statistic in Kalman filtering'. *Journal of Global Positioning Systems* vol.7, pp.81-90.

Wang, L., Zhang, Y, Wang, J., 2017, 'Map-Based Localization Method for Autonomous Vehicles Using 3D-LIDAR'. IFAC PapersOnLine, vol 50, pp:276-281, URL: https://doi.org/10.1016/j.ifacol.2017.08.046

Wang Y,2012, Position estimation using extended kalman filter and RTS-smoother in a GPS receiver. 2012 5th International Congress on Image and Signal Processing (CISP 2012), pp. 1718-1721.

Wangsiripitak, S., and Murray, D,W.,2009, Avoiding moving outliers in visual SLAM by tracking moving objects. 2009, IEEE International Conference on Robotics and Automation, Kobe, Japan, 2009.

Wei, L., Cappelle, C., and Ruichek, Y., 2013, 'Camera/laser/GPS fusion method for vehicle positioning under extended NIS-based sensor validation'. *IEEE Transactions on Instructiontian and Measurement*, vol.62, no.11, pp. 3110-322, 2013.

Williams, S.B., 2001, *Efficient Solutions to Autonomous Mapping and Navigation Problems*. PhD thesis, University of Sydney, Australian Centre for Field Robotics, 2001.

Wolcott, R.W., Eustice, R.M., 2014, Visual Localization within LiDAR maps for automated urban driving. In Intelligent Robots and Systems (IROS2014), 2014 IEEE/RSJ International Coference on *Intelligent Robots and Systems*, Chicago, IL,2014, pp.176-183.

Wörner, M., Schuster, F., Dölitzscher, F., Keller, C. G., Haueis, M. and Dietmayer, K., 2016, 'Integrity for autonomous driving: A survey'. *2016 IEEE/ION Position, Location and Navigation* Symposium *(PLANS)*, 2016, pp. 666-671, doi: 10.1109/PLANS.2016.7479759.

Xiao, W., Vallet, B., Schindler, K., Paparoditis, N., 2016, 'Street-side vehicle detection, classification and change detection using mobile laser scanning data', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, pp. 166-178.

Xie, L., Wang, S., Markham, A., Trigoni, N., 2017, GraphTinker: outlier rejection and inlier injection for pose graph SLAM. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, 2017, PP. 6777-6784.

Yan, G., Wang, J., Zhou, X., 2015, High-Precision Simulator for Strapdown Inertial Navigation Systems Based on Real Dynamics from GNSS and IMU Integration. In: Sun J., Liu J., Fan S., Lu X. (eds) China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III. Lecture Notes in Electrical Engineering, vol 342. Springer, Berlin, Heidelberg

Yang, L., Wang, J., Knight, N.L., Shen, Y., 2013, 'Outlier separability analysis with a multiple alternative hypotheses test'. *Journal of Geodesy*, vol.87, pp.591–604, 2013.

Ye, H., Chen, Y., Liu, M., 2019, 'Tightly coupled 3D lidar inertial odometry and mapping'. arXiv:1904.06993

Zhang, Z., Liu, S., Tsai, G., Hu, H., Chu, C.C., Zheng, F., 2017, 'PIRVS: An Advanced Visual-Inertial SLAM System with Flexible Sensor Fusion and Hardware Co-Design'. arXiv 2017, arXiv:1710.00893.

Zhang, J., and Singh, S., 2015, Visual-LiDAR odometry and mapping, lowdrift, robust and fast, in IEEE International Conference on Robotics and Automation, 2015.

Zhao, Y., 2013, *Key Technologies in Low-cost Integrated Vehicle Navigation Systems*. Thesis for Doctor of Philosophy, Royal Institute of Technology (KTH), Sweden, 2013.

Zheng, S., Wang,, J., 2017, High definition map-based vehicle localization for highly automated driving: Geometric analysis. 2017 International Conference on Localization and GNSS (ICL-GNSS), Nottingham, United Kingdom, 2017, pp. 1-8. doi: 10.1109/ICL-GNSS.2017.8376252