Context-aware GPS Integrity Monitoring for Intelligent Transport Systems (ITS)

PhD Thesis

Tareq A. Binjammaz

This thesis is submitted in partial fulfilment of the requirements
For the degree of Doctor of Philosophy

Software Technology Research Laboratory
De Montfort University
United Kingdom

Spring 2015

Abstract

The integrity of positioning systems has become an increasingly important requirement for location-based Intelligent Transports Systems (ITS). The navigation systems, such as Global Positioning System (GPS), used in ITS cannot provide the high quality positioning information required by most services, due to the various type of errors from GPS sensor, such as signal outage, and atmospheric effects, all of which are difficult to measure, or from the map matching process. Consequently, an error in the positioning information or map matching process may lead to inaccurate determination of a vehicle's location. Thus, the integrity is require when measuring both vehicle's positioning and other related information such as speed, to locate the vehicle in the correct road segment, and avoid errors. The integrity algorithm for the navigation system should include a guarantee that the systems do not produce misleading or faulty information; as this may lead to a significant error arising in the ITS services. Hence, to achieve the integrity requirement a navigation system should have a robust mechanism, to notify the user of any potential errors in the navigation information.

The main aim of this research is to develop a robust and reliable mechanism to support the positioning requirement of ITS services. This can be achieved by developing a high integrity GPS monitoring algorithm with the consideration of speed, based on the concept of context-awareness which can be applied with real time ITS services to adapt changes in the integrity status of the navigation system. Context-aware architecture is designed to collect contextual information about the vehicle, including location, speed and heading, reasoning about its integrity and reactions based on the information acquired.

In this research, three phases of integrity checks are developed. These are, (i) positioning integrity, (ii) speed integrity, and (iii) map matching integrity. Each phase uses different techniques to examine the consistency of the GPS information. A receiver autonomous integrity monitoring (RAIM) algorithm is used to measure the quality of the GPS positioning data. GPS Doppler information is used to check the integrity of vehicle's speed, adding a new layer of integrity and improving the performance of the map matching process. The final phase in the integrity algorithm is intended to verify the integrity of the map matching process. In this phase, fuzzy logic is also used to measure the integrity level, which guarantees the validity and integrity of the map matching results.

This algorithm is implemented successfully, examined using real field data. In addition, a true reference vehicle is used to determine the reliability and validity of the output. The results show that the new integrity algorithm has the capability to support a various types of location-based ITS services.

Acknowledgement

First and foremost, I would like to express my deepest gratitude to God for giving me the strength to carry on. I would also like to thank my supervisor **Dr. Ali H. Al-Bayatti** for guiding and encouraging me throughout my PhD. He always provide me useful comments and suggestions regarding my research. I would also like to thank **De Montfort University** for providing a studious environment and good facilities with which to complete this research. Finally, I would like to thank **Ashwaq ALHARGAN** and **University of Nottingham** for helping me throughout the testing phase and providing me the testing vehicle.

Dedicated to my family

Declaration

I declare that the work described in this thesis is original work undertaken by me for the degree of Doctor of Philosophy, at the software Technology Research Laboratory (STRL), at De Montfort University, United Kingdom.

No part of the material described in this thesis has been submitted for any award of any other degree or qualification in this or any other university or college of advanced education.

Tareq A. Binjammaz

Publication

- T. Binjammaz, A. Al-Bayatti and A. Alhargan. GPS integrity monitoring for an intelligent transport system. 10th Workshop on Positioning Navigation and Communication (WPNC), 2013.
- T. Binjammaz, A., Integrity for Drons. *The UAE Drones for Good Award.* September 2014.
- T. Binjammaz, A. Al-Bayatti and A. Alhargan. Context-aware GPS Integrity Monitoring for Intelligent Transport Systems (ITS). *Journal of Traffic and Transportation Engineering*, 2015. (*Accepted)*

Table of Contents

Abstract ii
Acknowledgement....iv
Dedication v
Declaration vi
Publication vii
Table of Contents viii
List of Figures x
List of Tables xi
List of Abbreviation xii
Chapter 1: Introduction....1
1.1 Introduction 2
1.2 Motivation 6
1.3 Research questions 6
1.4 Criteria of success 7
1.5 Thesis contribution 8
1.6 Research methodology 8
1.7 Outline of the Thesis 11
Chapter 2: Background and literature review....14
2.1 Introduction 15
2.2 Overview of location-based ITS services....15
2.2.1 Route Guidance....17
2.2.2 Emergency Management....20
2.2.3 Electronic payment....21
2.2.4 Public Transport Operations 23
2.2.5 Fleet Management....25
2.3 Overview of Required Navigation Performance (RNP) characteristics....26
2.3.1 Accuracy 27
2.3.2 Integrity....28
2.3.3 Availability....29
2.3.4 Continuity....30
2.4 Overview of RNP parameters for ITS 31
2.5 Positioning augmentation systems 34
2.6 Context-aware system overview....36
2.6.1 Context....36
2.6.2 Context classification....37
2.6.3 Context Acquisition 38
2.6.4 Context Modelling 39
2.6.5 Context Reasoning 41
Context reasoning technique....44
2.6.6 Context-aware Systems....45
2.7 Review of Map Matching (MM) algorithms....48
2.7.1 Definition of Map Matching (MM)....48

2.7.1 Geometric Map Matching (MM) Algorithms 49
2.7.1 Topological Map Matching (MM) Algorithms 53
2.7.2 Probabilistic Map Matching (MM) Algorithms 55
2.7.3 Advanced Map Matching (MM) Algorithms 55
2.8 Existing integrity methods 57
2.8.1 Positioning integrity methods 57
2.8.2 Map matching integrity methods 58
2.8.3 Hybrid Integrity methods 59
2.9 Summary 60
Chapter 3: System Architecture 62
3.1 Introduction 63
3.2 GPS Integrity monitoring system mechanism 64
3.3 System architecture 66
3.3.1 Sensing subsystem 67
3.3.2 Reasoning subsystem 69
3.3.3 Application subsystem 72
3.4 Summary 72
Chapter 4: Development of a GPS Integrity Monitoring Algorithm 74
4.1 Introduction 75
4.2 Integrity monitoring algorithm 75
4.2.1 Positioning integrity phase 78
4.2.2 Speed integrity phase 81
4.2.3 The map matching integrity phase 85
4.3 Summary 90
Chapter 5: Analysis and Discussion 91
5.1 Introduction 92
5.2 System analysis 93
5.2.1 Pilot study 93
5.2.2 Main data collection 93
5.2.3 Integrity level inference 101
5.3 Evaluation 107
5.4 Results 109
5.4.1 Effects of operational environment on system performance 110
5.5 Discussion 112
5.5.1 Key features of the speed integrity method 112
5.5.2 Fuzzy parameters' weights 114
5.5.3 Integrity scale threshold 116
5.5.4 Reliability 117
5.5.5 Location-based ITS applications 118
5.6 Summary 118
Chapter 6: Conclusion and Future Work 120
6.1 Conclusion 121
6.2 Success criteria revisited 122
6.3 Research limitations 125
6.4 Future work 126
References 128

List of Figures

Figure 1-1: Research methodology flow diagram…………………………………….. 11
Figure 2-1: In-vehicle route guidance system..19
Figure 2-2: Location-based public transportation system [23] 24
Figure 2-3: The Required Navigation Performance (RNP) parameters and its components………………..……………………………………………………………...30
Figure 2-4: SBAS coverage [49] …...…………………………………………………35
Figure 2-5: Abstract architecture for context-aware systems [20]................................47
Figure 2-6: Example of point-to-point MM algorithm [14]..50
Figure 2-7: Example of point-to-curve MM algorithm...51
Figure 2-8: Example of curve-to-curve MM algorithm. 52
Figure 2-9: Example of topological MM algorithm..54
Figure 3-1: GPS integrity monitoring system mechanism...65
Figure 3-2: GPS integrity monitoring system architecture. 66
Figure 3-3: GPS integrity monitoring algorithm..71
Figure 4-1: A flowchart illustrating the integrity monitoring algorithm........................76
Figure 4-2: Positioning integrity phase.…………………………………………………79
Figure 4-3: Speed integrity phase.…………………………………. ……………….82
Figure 4-4: Map matching integrity phase.……………………………………………86
Figure 4-5: The accumulated distance. 87
Figure 5-1: NGI test vehicle..95
Figure 5-2: Equipment inside the test vehicle [14]. …………………………………....95
Figure 5-3: The full test route in and around Nottingham.…………………………..…96
Figure 5-4: An example of the complexity in Data set 1…………………………….....97
Figure 5-5: The test route for the first data set...98
Figure 5-6: The test route for the second data set. 98
Figure 5-7: An example of the environment in Data set 2……………………..…...99
Figure 5-8: The test route for the third data set...99
Figure 5-9: An example of the environment in Data set 3……………………………100
Figure 5-10: Satellite availability for data sets 1, 2 and 3, respectively. ……………..100
Figure 5-11: Fuzzy membership functions. 105
Figure 5-12: Integrity threshold. 108
Figure 5-13: Error in Data set 1, green arrow is vehicle direction and black circle is the system error………………………………………………………………………...111
Figure 5-14: Error in Data set 2, green arrow is the vehicle direction and black circle is the system error……………………………………………………………………111
Figure 5-15: Error in Data set 3, green arrow is the vehicle direction and black circle is the system error……………………………………………………………………112
Figure 5-16: Miss detection errors in the system. 117

List of Tables

Table 2-1: Classifying ITS services based on its target user group……………………17
Table 2-2: Required Navigation Performance (RNP) parameters for ITS [5, 18, 45, 48]……………………………………………………………………………………31
Table 2-3: Available SBAS system [50].………………………………………………35
Table 2-4: Comparison of Context reasoning technique. ……………………………...44
Table 2-5: Context-aware system category……………………………………….……46
Table 2-6: The basic phases of context-aware systems………………………………...47
Table 2-7: Summary of the existing integrity methods…………………………...…60
Table 4-1: The Fuzzy inference system (FIS). 89
Table 5-1: Three different data sets collected in Nottingham. 97
Table 5-2:Fuzzy parameters and its weight. 103
Table 5-3: Fuzzy system rules.. 104
Table 5-4: Number of MD points, and false alarms raised by the system. 109
Table 5-5: Overall Correct Detection Rate (OCDR) of each data set. 110
Table 5-6: Overall Correct Detection Rate (OCDR) of the data set without using the integrity algorithm. 110
Table 5-7:Performance of existing integrity methods. 114
Table 5-8: Comparison between the integrity threshold value in this research and other research. 116

List of Abbreviation

ITS	Intelligent Transport Systems
GPS	Global Positioning system
RAIM	Receiver Autonomous Integrity Monitoring
DR	Dead Reckoning
GIS	Geographic Information System
RNP	Required Navigation Performance
OCDR	Overall Correct Detection Rate
MM	Map Matching
EPS	Electronic Payment Services
ETC	Electronic Toll Collection
DSRC	Dedicated Short Range Communication
OBU	Board Unit
RSU	Road Side Unit
AVL	Automatic Vehicle Location
ICAO	International Civil Aviation Organization
DOT	Department of Transpiration
IR	Integrity Risk
HAL	Horizontal Alert Limit

TTA	Time to Alert
AR	Availability Risk
SA	Signal Availability
CR	Continuity Risk
GNSS	Global Navigation Satellite System
HPL	horizontal protection level
FIS	Fuzzy Inference System
HDOP	Horizontal Dilution of Precision
WSSE	Weighted Sum of the Square
SDOP	Speed Dilution of Precision
INS	Inertial Navigation System
FAR	False Alarm Rate
MDR	Missed Detection Rate
UAV	Unmanned Air Vehicles
DOP	Dilution of Precision

Chapter 1

Introduction

Objectives

- Provide an introduction along with the motivation for this research.
- Highlight the research questions and the success criteria.
- Outline the main contribution of this research.
- Present the research methodology and thesis structure.

1.1 Introduction

The growth of intelligent transport systems (ITS) in the last decade has resulted in a significant improvement in road safety and monitoring, as they plays a key role in solving many transport problems such as road accidents and traffic congestion [1]. ITS services include traffic management, electronic payment, route guidance, fleet management and emergency management vehicle service. These services are mainly supported by positioning and navigation capabilities. In addition, most of them require real time positioning data, which refers to as location-based ITS services.

There are two main components for any location-based ITS used for vehicle navigation systems and services, which are (i) a geometric position system, such as a global positioning system (GPS) or an integrated navigation system, such as GPS/dead reckoning (DR); (ii) a geographic information system (GIS) based on road digital maps [2]. In addition, to determine the correct road segment and road link on which a vehicle is travelling, a map matching (MM) algorithm, which is used to integrate the positioning information into the digital road map, is also an essential component for ITS services [3].

ITS services (e.g. route guidance) depend primarily on the positioning data received from a positioning system (e.g., GPS) [4]. However, stand-alone GPS cannot provide the high quality positioning data required by most ITS services [5, 6]. This is due to the various type of errors related to the received positioning data such as signal outage, and atmospheric effects [7]. The digital road maps are more reliable compared to a stand-alone GPS, thus map matching algorithms can contribute in improving the accuracy of positioning data [8-10]. This is due to the fact that map matching algorithms consider different type of information including position, speed and heading in the matching process, in order to identify the location of the vehicle on the road segment [3]. However, map matching algorithms may locate the vehicle on a wrong road segment due to the quality of input data [3, 11-13], which can lead to a significant error in ITS services [14]. Therefore, it is important to check and monitor the quality of the positioning information obtained from the GPS sensor and the other input data to the map matching algorithm; in order to detect any misleading or faulty information and notify the user, which refers to as integrity [15]. According to Yu et al. [10], integrity "is to detect blunders in input data and faults in the map matching process".

According to the literature, there are some existing researches that have been carried out in order to monitor and improve the integrity of in-vehicle navigation system. These researches focus either on improving the integrity of raw positioning data such as Andrés [16], the integrity of the map matching process such as (Jabbour et al. [17], and Yu et al. [10]), or combination of both such as (Quddus [18] and Velaga [14]). Moreover, Velaga [14] have considered the complexity of road network (urban, rural areas) during the integrity process in addition to the integrity of raw positioning data and map matching process. These researchers including [18] and [14] have used speed to calculate the distance in map matching process, but no efforts has been found towards checking the integrity of speed. However, [18] mentioned speed as an essential factor to enhance the map matching algorithm. Indeed, monitoring speed integrity is vital. Moreover, Li et al. [6] states that failure in any factor in the map matching process leads to defects throughout the whole process. Thus, checking the integrity of the speed has the greater potential to improve the overall integrity process and lead to more accurate outcomes.

Yet, to the best of our knowledge there is no existing method for monitoring the integrity of in-vehicle navigation systems has taken into account the integrity of vehicle's speed during the integrity process.

Accordingly, the aim of this research is to contribute to improve the integrity of in-vehicle navigation systems by developing a robust and reliable GPS integrity monitoring algorithm based on the concept of context-awareness, in order to support the performance requirement of location-based ITS services. The context-awareness can provide adequate information about the current status of things in the environment (e.g. integrity of the navigation system) [19]. To achieve this aim a set of research questions are formulated (see Section 1.3).

The proposed algorithm will be able to ensure the integrity of in-vehicle navigation systems accurately by taking into account three types of information: vehicle position, vehicle speed and the result of map matching process. As mentioned earlier, existing methods focus on monitoring the integrity of positioning information, map matching process or both. Whereas, this algorithm incorporate a new layer to monitor the integrity of speed, which can significantly enhanced the performance of the map matching algorithm and the overall integrity process.

The algorithm is divided into three integrity phases. These are, (i) positioning integrity, (ii) speed integrity, and (iii) map matching integrity phase. Each phase uses different techniques to examine the consistency of the GPS information. A receiver autonomous integrity monitoring (RAIM) algorithm is used to measure the quality of the GPS positioning data. GPS Doppler information is used to check the integrity of vehicle's speed. The final phase in the integrity algorithm is intended to verify the integrity of the map matching process. In this phase, fuzzy logic is also used to measure the integrity level, which guarantees the validity and integrity of the map matching results.

The system architecture of the proposed GPS integrity monitoring algorithm will be designed based on the five layered context-aware framework [20]. The architecture is composed of three main subsystems: sensing, reasoning, and the application subsystem. These subsystems correspond to the main phases of the context-aware system. The first subsystem is used to sense the current context of the vehicle, including position, speed, and heading. The second subsystem performs the integrity algorithm to reason out the integrity of the collected information about the vehicle. The final subsystem is used to warn the driver about the integrity status of the in-vehicle navigation system.

The algorithm examined using real field data collected in Nottingham. A special vehicle equipped with high accurate sensors was used as "true reference" to assess the reliability and the validity of the output of the algorithm. The results suggest that the algorithm can verify the integrity with an accuracy of 98.6%.

1.2 Motivation

The map matching algorithm performance depends mainly on the input data [3, 12, 13]. For example, speed is used as an input in the map matching algorithm to identify the vehicle location on the road segment [14]. Therefore, checking the integrity of speed data can enhance the performance of the integrity algorithm. In addition, it is important for many ITS application. For example, traffic law enforcement systems (e.g., speed fining) have legal or economic consequences which require high quality speed information, in order to avoid errors when charging drivers.

As a result, by developing a robust and reliable mechanism to monitor the integrity of a navigation system including speed data based on the concept of context-awareness; it will be possible to avoid any misleading or faulty information provided by the navigation system.

1.3 Research questions

The main research question discussed in this research can be provided as follow:

How can the integrity of navigation system be improved in ITS services using the concept of context-awareness?

To address the above research question effectively, it is better to divide it into three questions and address each one individually. These questions can be given as follows:

- What type of information must be monitored in order to improve the integrity of in-vehicle navigation systems?
- How can the integrity monitoring system for location-based ITS services be designed, using the concept of context-awareness?
- How can the integrity monitoring algorithm be designed efficiently to determine the integrity of certain and uncertain navigation data?

1.4 Criteria of success

The success of the work presented in this research will be examined against the following criteria:

- The research questions listed in Section Research question must be answered.
- An investigation illustrating how the proposed integrity monitoring algorithm is different from other existing integrity algorithm is required.
- An investigation showing how the proposed integrity monitoring algorithm can be applied in ITS services is needed.
- An investigation illustrating how using fuzzy logic can positively affect the actual implementation of the algorithm.

- An investigation illustrating whether the integrity algorithm can be implemented in the real world, and thus can be used commercially.

1.5 Thesis contribution

The major contributions that have been reported in this thesis are given as follows:

- A novel algorithm for monitoring the integrity of in-vehicle navigation system in ITS services is proposed. The algorithm has taken into account the integrity of vehicle's speed, which adds a new layer of integrity and enhances the result of the map matching process.

- An integrity monitoring system architecture for location-based ITS services is provided. The system has been designed using the concept of context-awareness and can capture different type of information about the vehicle, such as position, speed and heading in order to reason out their integrity.

1.6 Research methodology

In this thesis, a constructive scientific research method is followed, where the contribution is represented in term of a novel architecture, model or technique [21]. Consequently, we develop an algorithm for monitoring the integrity of in-vehicle navigation systems, in order to support ITS services. The monitoring process will be based on using the concept of context-awareness.

Throughout this research, a methodology consisting of four stages was followed. The first stage represents the literature review and the related work of the research. The second stage concerns about the proposed system architecture. The third stage focuses on implementing and testing the proposed integrity algorithm. Finally, the last stage of the research deals with evaluating the proposed integrity algorithm. Figure 1-1 shows a flow diagram of the research methodology.

- **Stage 1:** Research background

  In this stage, a wide range of resources are used in order to carry out the literature review and the related work, this includes books, digital libraries, published articles, journals, etc. The stage begins with investigating the ITS services along with the RNP parameters. Then, review the context-aware systems and the existing map matching algorithms. Finally, it studies the existing integrity monitoring algorithm for in-vehicle navigation systems along with their limitation, as the main aim of this research is to develop an integrity monitoring algorithm for ITS services based on the concept of contexts-awareness.

- **Stage 2:** Architecture

  In this stage, the proposed integrity monitoring system is designed to capture the contextual data about the vehicle, using different type of sensors (GPS and Wheel speed sensors), and reason about its integrity and react upon it. In addition, it discuss the components of the system in details and describe how these components interact with each other.

- **Stage 3:** Experiment

  In this stage, the integrity monitoring algorithm is implemented using MATLAB. The required data for testing the strength of the algorithm were collected in Nottingham using GT-31 GPS receiver [22]. In addition, a special vehicle equipped with advanced GPS equipment was hired and used as true reference to make sure the data is collected properly and validate the proposed algorithm. There were several factors considered during the collection of data to verify and test each part in the algorithm. These factors are: the length of track, environment, and speed. Moreover, the values of the fuzzy rules inputs and output were identified empirically in this stage.

- **Stage 4:** Evaluation

  In the final stage, overall correct detection rate (OCDR) is adopted as an evaluation criteria, which is widely used, to evaluate the performance and the efficiency of the proposed integrity monitoring algorithm. This criteria has the features of combining false alarms rate with miss detection rate to produce the accuracy of the integrity algorithm.

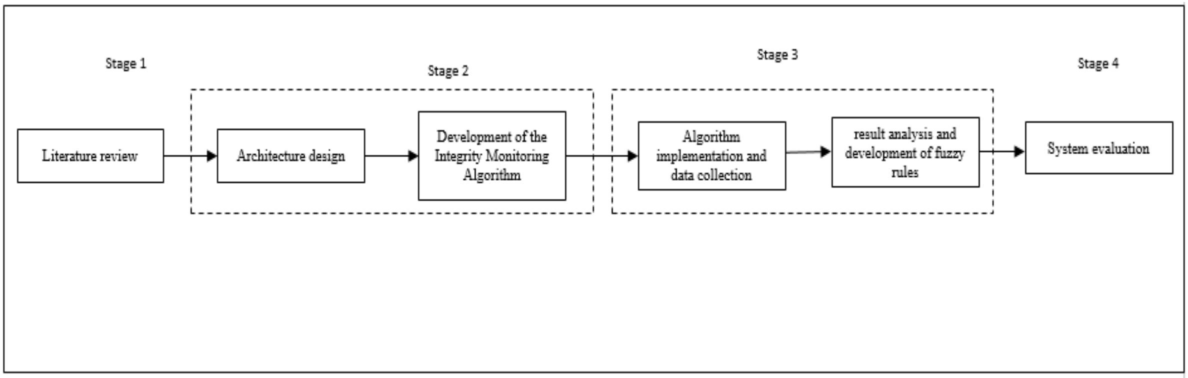

Figure 1-1: Research methodology flow diagram.

1.7 Outline of the Thesis

This chapter has provided an introduction to the research including research motivation, research questions, and the methodology. Moreover, the success criteria and the contribution of this research have been highlighted. The following illustrates the remaining chapters of this theses:

- **Chapter 2: Literature Review**

  This chapter is divided into three sections. The first section presents an overview of Intelligent Transport Systems (ITS); including a brief description of various types of ITS services that require the vehicle navigation information. This followed by defining the Required Navigation Performance (RNP) parameters. Then, reviewing the existing RNP parameters for ITS services. It also review some criteria's that have been used to derive the RNP for ITS services. The second section of this chapter reviews context-aware systems by providing definition of context, and explain how to capture, model and reason about given context, also it describe the context-aware system and its architecture. The third section gives an overview of map matching algorithms. This include, geometric, topological, probabilistic and advanced matching algorithms along with their limitations. Finally, exiting integrity algorithms available in literature are discussed.

- **Chapter 3: System Architecture**

This chapter presents the system architecture for monitoring the integrity of in-vehicle navigation systems. It begins with explaining the mechanism of the proposed system. In addition, the architecture of the integrity monitoring system that is designed based on the concept of context-awareness is presented. This includes, detailed explanation of the three main subsystems, which are sensing, reasoning, and application subsystem along with their components. Moreover, it shows how the system components interact with each other.

- **Chapter 4: Development of the GPS Integrity Monitoring Algorithm**

This chapter presents the development of the integrity monitoring algorithm for in-vehicle navigation systems. The chapter start with illustrating the mechanism of the proposed integrity algorithm. This is followed by a detailed explanation of the three integrity phases, which are positioning integrity, speed integrity and map matching integrity.

- **Chapter 5: Experiment and Evaluation**

This chapter presents detailed information about the experiment that was conducted. It shows how the testing data was collected, presented, and processed by the algorithm. Results and analysis which were carried out based on the evaluation criteria are presented in this chapter. The chapter also introduces fuzzy logic factors, fuzzy rules, and factors weights. Finally, the result of the integrity algorithms is discussed. This section starts by stating the key features of the speed integrity algorithm, comparing the accuracy and performance of the algorithm with others' integrity algorithms, and justifying each distinctive points achieve better outcome as a result of using the proposed integrity algorithm in this research. It also discusses the impact of fuzzy factors used on the system. Other points such as integrity scale threshold, reliability, and suitability of the algorithm for location-based ITS applications are also discussed in this section.

- **Chapter 6: Conclusion and Future Work**

This chapter summarise the work that has been achieved throughout this research and draw a conclusion about the final results. Then, revisit the success criteria and finally, it explain the limitation of the thesis and provides some suggestions for further improvements in future.

Chapter 2

Background and literature review

Objectives

- Provide an overview of ITS services in the context of vehicle positioning systems.
- Brief explanation of Required Navigation Performance (RNP) parameters for location-based ITS services.
- Reviews the literature concerning context-aware systems, modelling and reasoning.
- Provide an overview of existing map matching (MM) algorithms.
- Provide an overview of existing integrity methods.

2.1 Introduction

This chapter start by providing an overview of location-based intelligent transport system (ITS) and services. Describing the performance requirements that used to assess positioning systems, which are defined as Required Navigation Performance (RNP) parameters. These parameters are: accuracy, integrity, availability, and continuity. The RNP parameters for some ITS services are then discussed. Secondly, the chapter gives an overview of the concept of context and context-aware systems. Then, a review of existing map matching (MM) algorithms which categorised as: geometric, topological, probabilistic and advanced MM algorithms along with their limitation are presented. Finally, integrity algorithms currently available in literature are discussed.

2.2 Overview of location-based ITS services

Intelligent Transport Systems (ITS) is referred to a transportation system that composed of advanced technologies such as positioning system, data processing, communication means, sensing technologies to enhance its efficacy and safety [23]. The key goal of ITS is to support and enhance services in transportation systems such as emergency management, public transport management, commercial vehicle operations, traffic control and vehicle safety [23]. In general, it can be said that ITS offers a great opportunity to solve many complex transportation problems. It has played a key role in reducing risks of accidents, control congestion, improve road safety and reliability, and reduce adverse environmental impacts [1]. According to National Research Council [24], the term ITS defined as: “ITS applies computers, information management, advanced electronics, and communications technology to reduce traffic congestion, enhance safety, save energy, and in other ways generally improve the performance of the nation’s highways and transit”.

A location-based ITS services are services that continuously requires vehicle geographical positioning information in order to track the vehicle and perform its operations [4]. Basically, a navigation system of an ITS composed of positioning system such as stand-alone GPS or roadside beacons and digital road map to provide the necessary spatial information about vehicle, including longitude, latitude and height, and support different type services [14]. As an example, a positioning system that installed in vehicles can be used by an ITS service to determine the driver location on the road network and guide him to reach the target destination. A classification of ITS services and its target user groups discussed in [14, 15, 18, 23, 25] are illustrated in Table 2-1.

Most of the services listed in Table 2-1 use navigation and positioning technologies to support the service functionalities in all environments. Some of these ITS services are described in the following sections.

ITS services	Target user groups
Pre-trip travel information En-route driver information Route guidance Ride matching and reservation Traveller services information Traffic control Incident management Travel demand management Emissions testing and mitigation On-board emissions monitoring (OEM) Highway-rail intersection (HRI) Travel and Traffic Management Traffic data quality management	Traffic Management
En-Route Transit Information Operation automations Personalized public transit Public travel security	Public Transport Management
Electronic payment systems	Electronic Payment
Commercial vehicle electronic clearance Automated roadside safety inspection On-board safety and security monitoring Commercial vehicle administrative processes Fleet management In-situ vehicle condition monitoring	Commercial Vehicle Operations
Emergency notification/personal Security Emergency vehicle management Disaster response and evacuation (DRE)	Emergency Management
Longitudinal collision avoidance Lateral collision avoidance Intersection collision avoidance Vision enhancement for crash avoidance Safety readiness Automated highway systems Pre-crash restraint deployment Stolen vehicle recovery	Advanced Vehicle Safety

Table 2-1: Classifying ITS services based on its target user group.

2.2.1 Route Guidance

Several investigations have shown that the route guidance service play an important role in reducing traffic congestion, travel time, and minimizing air pollution by saving energy as the drivers have prior knowledge about their target route [26-28]. Fundamentally, the route guidance service is used to provide detailed turn-by-turn guidance directions to reach the desired destination. The provided instructions are either based on static travel information such as historical travel jamming on specific road or real-time travel information, including current travel speed and road conditions. There are two type of route guidance services: Pre-trip travel guidance and en-route driver guidance [23].

In pre-trip travel guidance service, information about the target route and transportation system such as travel time, current road status, scheduled road constructions and transit routes are provided to the service users before the trip. The main purpose of this service is to make the users aware of the various possible travel options and help them to make travel decisions by providing enough information based on the current route conditions [23].

On the other hand, en-route driver guidance service provides the driver with real-time turn-by-turn directions after the trip being started [23]. En-route guidance normally refers to in-vehicle guidance service which required to be equipped with cars in order to provide step-by-step driving instructions to the driver. Principally, it aims to improve driver behaviour especially in unfamiliar areas and the safety of travelling vehicle. The main component in-vehicle route guidance are: position system to determine the vehicle location on the road, road map database to provide information about the road on which the vehicle is travelling (e.g., road name, road speed limit, and direction to the next turn), map matching technique to locate the vehicle location on the road map, and user interface to present information to the driver [18]. Figure 2-1 shows an example of in-vehicle guidance system. Generally, the system starts by locating the vehicle on the road and then shows its location on the digital road map network. According to Sheridan [29], in-vehicle guidance system can be divided into two types: autonomous and dynamic. In autonomous route guidance, the travel route is created based on the driver preference such as the least overcrowding path. Whereas, in dynamic route guidance traffic conditions are integrated automatically with the system and the route may be changed along the journey in response to new traffic conditions.

Figure 2-1: In-vehicle route guidance system.

In general, the performance of route guidance service depends on the quality of positioning information that is used to provide guidance to the driver. As a result, this may possibly affect the efficiency of the route guidance service and confuse the driver. Therefore, it is vital to check the positing information and inform the user of any potential uncertainty in order to avoid misleading information.

2.2.2 Emergency Management

Emergency management systems are one of the most significant ITS services due to its key role in saving human lives and timely responding to emergency events such as traffic accidents. It defined as "a discipline that deals with risk and risk avoidance" [30]. Emergency management include two important services that used facilitate safety of life applications. These are emergency notification and personal security service and emergency vehicle management service [23].

In emergency notification and personal security service, drivers have the ability to manually notify the emergency service provider about urgent incidents or non-urgent such as vehicle breakdowns [23]. It also provides automated notification service where a notification of serious accident is sent automatically to the emergency service provider. Systems that use this service should be equipped with navigation system (e.g., GPS), digital road map, communication system, and in-vehicle sensors to sense for accidents automatically. The information about the vehicle location should be included automatically with the notification massages in order to help the service provider (e.g., emergency control centre) to precisely know the location of the vehicle and provide the require service.

Emergency vehicle management service, on the other hand, mainly intended to reduce the travel time from receiving the emergency notification to reach the accident location [23]. This service usually uses other services such as route guidance to determine the shortest route to reach the accident and emergency vehicle fleet management to determine the nearest emergency vehicle.

As has been described, emergency management services depend mainly on the vehicle location information obtained from GPS, especially when it used in safety critical applications. However, location information might have some potential inaccuracy especially in urban areas due to different GPS errors (e.g., signals delays, reflection), which may confuse the emergency control centre. As a result, high quality positioning information is important to help the service provider to precisely identify the location of an accident and deliver the required service.

2.2.3 Electronic payment

The most common transport problem in the late 20th century is traffic congestion. This is due to the limited capacity of road networks and the increasing number of motor vehicles especially in large urban areas [31]. Traditionally, the method that used to reduce traffic congestion is to expand the capacity of existing transport infrastructure. However, many studies has shown that the expansion of road networks possibly will lead to increase the number of road users. As a result, many cities have introduce road charging schemas (e.g., congestion tolls) as an alternative way to reduce traffic congestion such as London [31]. As an example, the drivers may charged based on driving within a particular zone or on travelling distance on specific road.

To facilitate the automatic enforcement of the charging regulations and make vehicles data available to infrastructure providers, Electronic Payment Services (EPS) have been used [32]. EPS generally provides the infrastructure providers with information about the vehicle location to determine whether the vehicle uses a toll road. In addition, it provides the drivers with electronic payment methods (e.g., smart cards) to facilitate the payment process.

The most common EPS is Electronic Toll Collection service (ETC). The main concept of ETC is to enhance tolls collection process and eliminate travel delays by allowing the travellers to pay road tolls electronically. ETC systems mainly implemented using Dedicated Short-Range Communication (DSRC) technology which consists of on board unit (OBU) installed in vehicle and road side unit (RSU) installed in the road [1]. The DSRC system allows the drivers to automatically communicate with the RSU and pay road tolls without stopping at the toll stations. In recent years, most of the ETC systems are implemented based on GPS and mobile communication technologies (e.g., Global Systems for Mobile (GSM)) [33]. Where the current vehicle positioning information is used to determine whether the vehicle is within a charging zone and apply the correspond charges. However, an error from the navigation system may possibly cause an error in chargers to road users. Consequently, a mechanism to check the positioning information provided by GPS should be applied in order to avoid charging the road users wrongly.

2.2.4 Public Transport Operations

Public transport operation services has a central role in the management of the public transport facilities. Where the main goal is to encourage the use of public transport and provide more reliable transit services [1, 34]. The main services are public transportation management, en-route transit information, and personalized public transit service [1, 23]. The following discuss each of these services:

- **Public transportation management:** This service uses advanced communication technologies and vehicle tracking systems to collected data that can be used to enhance the operational planning, vehicle scheduling and facilities of public transports efficiency and effectiveness [35]. The most common used system to collect real-time data is Automatic vehicle location system (AVL) [34]. The AVL system generally tracks the vehicles location in real-time and send the information to the transit agency centre via communication network [23]. This real-time information helps the dispatcher to determine any possible deviation in the schedule and perform corrective actions to the service. Furthermore, real-time location information can be used to facilitate the use of public buses by providing the travellers with accurate timetable at the bus stop on digital screens. A simple version of location-based public transportation management system is shown in Figure 2-2.

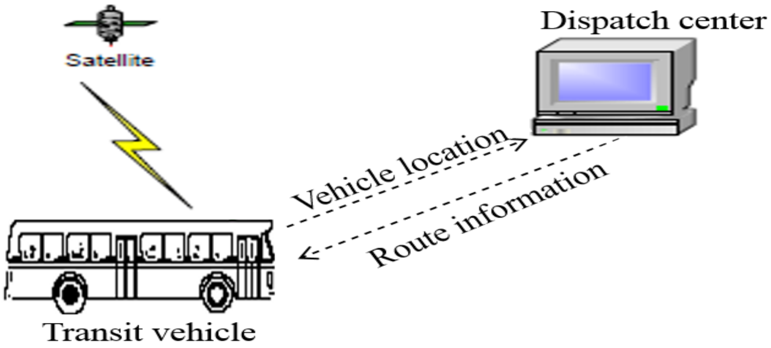

Figure 2-2: Location-based public transportation system [23].

- **En-route transit information:** This service is used to provide the travellers on board the transit vehicle with real-time information about the journey such as expected arrival time, current location, and next stop, in order to help them in making transfer decisions [23].
- **Personalized public transit:** This service provide passengers with on demand transit options. Which allow them to make a reservation for a trip in advance by sending to the dispatch centre information about the source and destination of the trip. The dispatcher then will assign the request to the nearest transit vehicle and inform the driver of the passenger’s location. In addition, the dispatcher would sent information about the expect arrival time to the passenger. The main goal of this service is to reduce waiting time and enhance the deployment of the schedule in real time [36].

Public transport operation services mainly rely on real time location information obtained from the navigation system in order to estimate the arrival/depart time of transit vehicles and stay to the planed schedule. However, as mentioned previously navigation systems may have some tracking problems that would probably make the public transport systems inefficient and unreliable. One solution to this reduce the problem is to check the positioning information and inform the dispatch centre of any significant inaccuracy.

2.2.5 Fleet Management

Fleet management is one of the most important services in the development of the business of the fleet vehicle operations. According to a recent market study [37], the fleet management market will continue to grow to reach $30.45 billion by 2018. In general, a fleet management system use to manage the vehicle fleet operations in an organised manner by using advanced technologies such as computer software, communication and navigation systems. It offers a wide range of services including vehicle tracking and security, driver monitoring and control, and dynamic routing that designed to help fleet operators to improve their operations in term of vehicle performance, reliability and safety of their operations [38].

A fleet management system typically consist of three subsystems: in-vehicle subsystem, communication subsystem, and dispatch subsystem [18]. The in-vehicle subsystem is used to track the vehicle and send real-time positioning information to the dispatch subsystem. The real-time position data including X, Y, and Z coordinates, heading, and speed are then used to determine the physical location on the road network. Furthermore, it helps the dispatch subsystem to fully control the fleet by automatically determinate the status of each vehicle such as whether the vehicle is on a highway or on a service road. The communication subsystem is used to handle the communication between feel vehicles and the dispatch centre subsystem.

The performance of the dispatch subsystem relies on the accuracy of the positioning information obtained from GPS. For example, locating the vehicle on the wrong road due to positioning error possibly will result in selecting the wrong vehicle to be send out. Therefore, a navigation system should include a robust mechanism to check the validity of the positioning information and deliver a warning massage of any fault to the dispatch centre subsystem.

ITS services are still under development as new services are emerging and new technologies are adapted. Future details about ITS services can be found in [1, 23, 30, 39-41]. As discussed previously, most ITS services are supported by positioning systems. Thus, they must satisfy the Required Navigation Performance (RNP) parameters of ITS services [15]. These requirements are discussed in the following section.

2.3 Overview of Required Navigation Performance (RNP) characteristics

The notion of Required Navigation Performance (RNP) parameters was firstly introduced in 1983, by International Civil Aviation Organization (ICAO), as part of the performance based navigation model [42]. Originally, the purpose is to provide navigation standards for aviation and defined the required level of performance measurements, in order to improve the aircraft navigation efficiency and safety [42]. The notion of RNP method has been extended over the last decade to be used in marina and land transportation [15].

The RNP parameters are divided into four measurements: Accuracy, Integrity, Availability, and Continuity, as shown in Figure 2-3. Generally, these four parameters are considered as a key measurement for evaluating the performance and improving the overall quality and efficiency of a positioning system. The performance measurements that have been addressed and defined in literature for land navigation systems are discussed below.Figure 2-3

2.3.1 Accuracy

According to Ochieng and Sauer [15], accuracy defined as the degree of conformance between the information provided from a navigation system at given time, such as position, speed and heading, and its real values. In simple terms, accuracy assessment can be achieved by comparing measured positions with a high accurate navigation source such as accurate GPS or high-resolution satellite imagery [14]. It has conclusively been shown that GPS accuracy has improved to reach 3-5 m [43]. In addition, the percentage of accuracy requirement for a GPS navigation system is specified at 95% [44]. This means that the degree of accuracy provided by the navigation system can be used as an estimated measurement of the system error. As an example, 95 percent accuracy means that the likelihood where the estimated error of a measured position at given location is within the accuracy requirement must be at least 95% [44].

According to Department of Transpiration, DOT [45], accuracy of navigation system can be divided into predictable, repeatable, and relative accuracy. Predictable accuracy refers to the accuracy of a positioning solution with reference to the charted solution (e.g. geodetic coordinates). Secondly, repeatable accuracy which refers to the ability in which the user can return to a previously defined position with the same navigation system. Finally, in relative accuracy a user can define positions that related to another user with the same navigation system simultaneously.

2.3.2 Integrity

Integrity of a navigation system refers to the ability of the system to identify any failure and notify the user when the system should not be used for navigation [15]. Integrity is also known as a measure of trust on the positioning information that are provided by a navigation system [44]. There are three components that can be used to measure the integrity of the system: integrity risk (IR), horizontal alert limit (HAL) and time to alert (TTA) [44]. Firstly, integrity risk (IR), which defined as the likelihood where the error exceeds the horizontal alert limit without raising any alarm to the user within the specified time to alert. Secondly, horizontal alert limit (HAL) is the maximum allowable error that can appear in a safe operation without warning the user. Finally, time to alert (TTA) which refers to the maximum allowed duration of time from a fault being detected to the time when the user being warned.

2.3.3 Availability

The availability of a navigation system refers to the percentage of time where the service is available and can be used for navigation at the beginning of the intended operation [46]. It has been identified as a critical parameter in navigation systems due to the high usage of navigation data in urban areas [15]. More precisely, availability measurements are used as indicator of services usability within a coverage area. In addition, the service is guaranteed to be available if the system accuracy, integrity and continuity requirements are satisfied [15].

According to [47], the author believed that availability is affected by both the physical features of the environment and transmitter capabilities. Hence, he defined two availability component given as follows [47]:

- **Availability Risk (AR):** is the probability that the necessary navigation information will not be available at the beginning of a specified task.

- **Signal Availability (SA):** is the percentage of time where navigational signals received from external source are available to be used.

2.3.4 Continuity

Continuity of a system is defined as the ability of the overall system to work continuously and provide the required information without disruption during the intended phase of operation [15]. The continuity measurement can be specified by the Continuity Risk (CR) which refers to the probability that the navigation system will be disrupted and will not provide necessary navigation information for the intended operation [47]. According to this definition of continuity risk provided by Hein [47], it is clear that the CR can be used as measurement of the navigation system reliability.

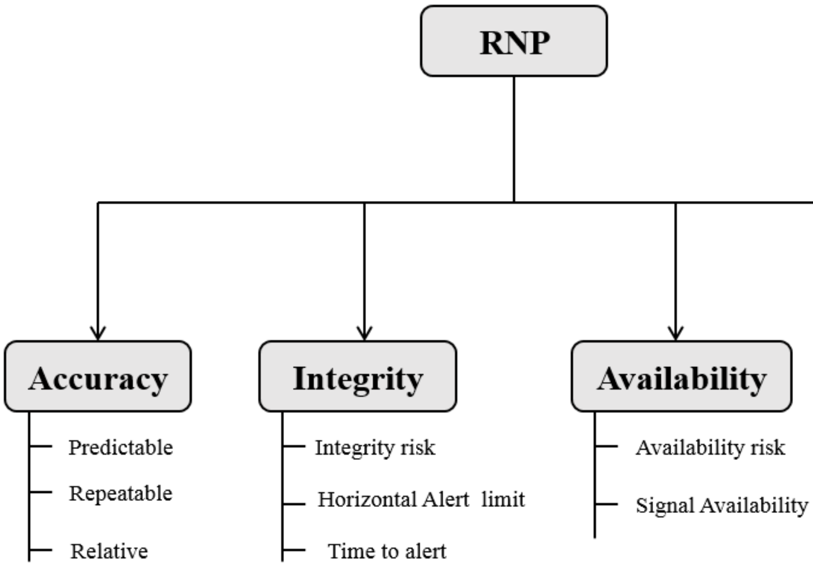

Figure 2-3: The Required Navigation Performance (RNP) parameters and its components.

2.4 Overview of RNP parameters for ITS

Positioning information is an essential requirement for supporting most ITS services. Consequently, it is important to define the RNP that should fulfil the needs of ITS services. As has been shown previously, the standard measurements of RNP parameters such as integrity risk, horizontal alert limit, continuity risk, and availability risk are well addressed in the literature for civil aviation [44, 45, 47]. However, the values of RNP parameters for ITS services are still under development as more services evolve and more applications continue to emerge. Some recommended values of RNP parameters available in literature for different ITS services are illustrated in Table 2-2.

ITS system	Accuracy (95%)	Integrity		Availability	Continuity (per 1 hour)
		HAL	TTA		
Navigation and route guidance	1 - 20 m	2 - 20 m	10 sec	99.7%	10^{-5}
Automated Vehicle Identification	1 m	0.2 -30 m	>= 5 sec	99.7%	Not applicable
Automated Vehicle Monitoring	5 m	3 m	10 sec	> 95%	10^{-5}
Collision Avoidance	1 m	2.5 m	1 sec	99.7%	10^{-5}
Intelligent Vehicle Initiative	0.1 m	0.2 m	5 sec	99.9%	Not applicable
Emergency Response	10 m	0.5 - 10 m	1 – 5 sec	99.7%	Not applicable
Accident Survey	1 – 4 m	0.2 – 4 m	30 sec	99.7%	Not applicable
Public Safety	10 m	.2 - 30 m	1 – 15 sec	95 - 99.7%	Not applicable
Vehicle Command and Control	30 – 50 m	Not available	Not available	99.7%	Not applicable

Table 2-2: Required Navigation Performance (RNP) parameters for ITS [5, 18, 45, 48].

It should be noted form Table 2-2 that the performance requirement for ITS services vary significantly based on the service functions. For instance, in safety of life applications, such as collision avoidance, positioning accuracy is considered as high requirement where the accuracy is required to be 1 m in 95% of the time [5, 18, 45].

Whereas in non-safety application, such as vehicle command and control system, accuracy is considered as low requirement where the value vary from 30 m to 50 m in 95% of the time [45]. This means that safety is an important factor to determine the values of RNP parameters that required by an ITS service, especially for those services that are safety critical. Some criteria that helps to derive the RNP for ITS services are summarised according to [14], as follows:

- **System performance requirements:** Operational requirements are considered as an influential factor in deriving the RNP parameters of an ITS service, in case the service aims to enhance the effectiveness of a transportation system. For example, the RNP parameters for public transport management systems that support bus priority at junction service should be high compared with bus arrival scheduling service as an error in the navigation system of the bus priority at junction service may cause an error in the bus schedule. As a result, this may lead to more delay and traffic congestion at junction.
- **Commercial issues:** ITS service that apply chargers to the road users based on their real-time location, such as electronic toll collection service, normally required to charge the users accurately in order to avoid wrong financial consequences. Thus, commercial issues are given the high priority to develop the RNP parameters, which needs to be relatively high [48].
- **Operational environments:** Operational environments in which ITS services are operating in also considered as a factor to derive RNP parameters. In case

the ITS service is operating in urban areas, and complex road networks that includes narrow streets, towers, and bridges are more likely to affect the performance of the navigation system. As an example, a small error in the navigation system that used for route guidance service in urban areas may locate vehicle on wrong road and misguides the driver. While in simple road networks (e.g., suburban and rural) this is improbable to occur. Therefore, the RNP parameters for ITS service that used in urban areas are relatively high compared with less complex environments.

- **Safety issues:** Generally, safety issues are given first priority in developing the RNP parameters for safety of life ITS services. For example, in collision avoidance high performance requirements are is essential, where the value of accuracy should be high and the values of integrity (HAL and TTA) should be low in order to avoid any possible collision.
- **Type of operation:** Type of operations which are fixed route and variable route operations can also be used to derive the RNP parameters. In fixed route operations, such as transit and rail, the RNP parameters are considered to be low as the driver use predefined routes. Therefore, a small error in the navigation system possibly will not affect the overall system performance. The RNP parameters for variable route operations, on the other hand, required to be high. The ITS services that which perform on variable routes includes fleet management services, electronic payment services, and route guidance.

The RNP parameters accuracy, integrity, availability, and continuity for land transport systems for land transport systems are still under development as high performance requirements are important and differ from application to another. They should be set in such way to avoid any adverse effect if the system performance degrade. For this reason, the navigation system that support location-based ITS services should be improved to satisfy the performance requirement for the corresponding ITS service (as described in Section 2.4). As integrity parameters are most directly related to safety and confidence of the service [15], further enhancement in the positioning integrity is a possible approach to improve the performance of the navigation system and support the ITS services more effectively.

2.5 Positioning augmentation systems

The Satellite-based Augmentation System (SBAS) is a system used to aid existing navigation systems, such as GPS, to improve their accuracy, integrity, continuity and availability [49]. There are a range of SBAS currently available; these are: The European Geostationary Navigation Overlay Service (EGNOS), The USA Wide Area Augmentation System (WAAS), The Japan Multi-functional Satellite Augmentation System (MSAS) and The India GPS-Aided Geo Augmented Navigation (GAGAN). These systems are dedicated to regional operations and were designed principally to support navigation in civil aviation (see Figure 2-4). Table 2-3 identifies their key characteristics and limitation.

Figure 2-4: SBAS coverage [49].

System	Launch year	Characteristics	Limitations
WAAS	Fully launched in 2003.	- Provide positioning accuracy within 3 m. - Available in the US, Canada and Mexico.	- Used for civil aviation. - Requires ground navigation equipment.
EGNOS	Fully launched in 2009.	- Provide positioning accuracy within 3 m. - It can be used as a safety of life application. - Available in Europe. - Can support other types of transport.	- Currently used for civil aviation. - Requires ground navigation equipment. - Not suitable in urban areas.
MSAS	Launched in 2007.	- Provide positioning accuracy up to 2.2 m.	- Used for civil aviation. - Requires ground navigation equipment.
GAGAN	Launched in 2013.	- Provides horizontal positioning accuracy up to 1.5 m and vertical accuracy up to 2.5 m during a flight from take-off to landing. - Available in the Indian subcontinent.	- Used for civil aviation. - Requires ground navigation equipment.

Table 2-3: Available SBAS system [50].

As shown in Table 2-3, SBAS provides more accurate positioning data than standard navigation systems, such as GPS [49]. However, this accuracy is only guaranteed inside the coverage area, and its signal is subject to interruption due to tall buildings in urban areas, for instance.

EGNOS, which is operational in Europe, improved the accuracy of GPS to less than 3 m, and can be used for safety of life applications in civil aviation [49]. However, its utility for land navigation is still being developed and validated. In addition, its operation is currently regional and expansion is expensive. Thus, for the purposes of this research the EGNOS is not considered. GPS only is adapted here, as it has been widely used to acquire positioning information, and is currently the most available and widespread system in operation globally.

2.6 Context-aware system overview

2.6.1 Context

A number of definitions have been proposed to define 'what is context'. An early definition of context was offered by [51], who said that context is a location and includes the people and objects that are in that location. Similarly [52], say that context is the location, the time, the weather and the identity of users and an alternative view was offered by [53], who referred to context as the emotional state of users as well as the location, objects, date and time and other people in the context.

In [54], there is also reference to the user but with more of an emphasis on the computer itself, where context is the information that the computer acquires about its user environment and [55], says that context is about the aspects of a situation. Similarly [56] says that context is the information sensed by an application, this includes information about the environment or the user themselves. The free on-line dictionary

[57], defines context as something which surrounds, and provides meaning to other things. According to [58] context comprises of where someone is, who they are with and the available resources in a particular environment.

Perhaps the most clear and precise definition of context was proposed by [59], said:” *context is any information that can be used to characterize the situation of an entity. An entity is a person, or object that is considered relevant to the interaction between a user and application, including the user and applications themselves*”.

2.6.2 Context classification

It is important to categorise the different context types in order to assist application designers to ascertain the relevant aspects of context that they can consider in application development. A categorisation approach put forward by [59], considers the location, identity, activity and time of a context. Such an approach answers the questions of where, who, what and when and allows other sources of information from the within the context to be indicated. To provide more specific examples [59]; for an entity’s location we can derive numerous pieces of information about objects or people near the entity and processes that are taking place. For a person’s identity we can derive information including email addresses, phone numbers and friends, therefore, from primary entities in a context we can derive information about the secondary context.

2.6.3 Context Acquisition

The approach to acquiring context is an important part of the design and development of context-aware systems because it lays the basis of the style of architecture of the system and the method of gathering contextual information[60]. Usually, a context-aware system uses sensors to gather information about the context [61]. Sensors can be physical that can capture physical aspects of a context such as light and temperature, or software programs, sometimes referred to as virtual sensors which gather and collate contextual information using software applications. An example of the latter is a software program that can work out a user's location by checking their emails or by tracking a user's computer. Logical sensors are another type of sensor that work by combining information gained from the other aforementioned sensors, i.e. physical and virtual, to perform tasks. Which type of sensor to be used depends on the requirements of a system and [61] says that contextual data can be gained in the following ways:

- **Direct sensor access:** This approach, often found where devices have sensors built in, involves software directly accessing sensor information without the use of any middleware or processing of that information. This is achieved through having drivers that are directly hardwired to the application. It should be noted that these types of sensors are not suitable for use in distributed systems because of the fact that it uses direct access.

- **Middleware infrastructure:** Encapsulation is used in software design because there is a need to separate business logic and the user interface. In this approach, middleware is essentially a layer within the architecture of a context-aware system which hides low-level sensing details. It makes extensibility easier because the client code is not changed and it also makes reusing hardware much simpler, unlike direct sensor access [62].

- **Context server:** Middleware based architecture can be extended by using this distributed approach. This involves the utilisation of an access managing remote component. The function of gathering data from the sensors is placed in the context server to allow multiple access.

2.6.4 Context Modelling

It is necessary to model a context in order to establish and gather the context information that is gathered by sensors, according to [63] there are a number of different context modelling techniques available:

- **Key-Value models:** Key-value models show the minimum key data structure in context modelling. They are often employed in a wide range service frameworks, where simple values define system attributes and service values, these may include for example the name, time and location. Matching algorithms are then used for service discovery employing the aforementioned key-value pairs. Although this type of model is easy to use, it does not allow for advanced structuring characterised by a retrieval algorithm for an efficient context.

- **Mark-up schemes models:** Mark-up scheme models are essentially hierarchical data structures that include mark-up tags that show contents and attributes in a system.

- **Graphical models:** A powerful graphical modelling technique is Unified Modelling Language (UML) which is suitable for general modelling. Because it has a generic structure it can be used for modelling all types of context. Specifically, this approach to modelling approach is best suited for creating an Entity Relationship model ER-model, which clearly illustrates a relational database in a system that is structured around an architecture of context management.

- **Object oriented models:** This approach to modelling considers a context that utilises the full extent of object orientation which includes reusability, encapsulation and inheritance. Reusability and encapsulation are employed to resolve problems that are related to context dynamism. Various objects are used to represent the different types of context variables, these may include place and age, and the approach encapsulates information related to context presentation and processing. The information about the various aspects of the context can be accessed through a user interface.

- **Logic-based models:** Generally, this type of model defines context using rules, expressions and facts which can be added or updated or removed using a logic based system. The reasoning process involves deriving new facts based on existing system rules. Contextual information is represented formally as facts.

- **Ontology-based models:** An ontology-based approach provides a description of the ideas and relationships that are found within a context. Thus, this approach is a highly suitable instrument for modelling contextual information, especially due to the fact that expressiveness is highly formal. A classic example of an ontological approach is the Context Broker Architecture (CoBrA) system [61], this system uses a set of concepts to characterise entities which may include people and places.

2.6.5 Context Reasoning

The aforementioned context models have a limitation which is that they do not consider reasoning related to uncertain contextual information and they can simply capture and define contextual information which is certain such as place and temperature [64]. The main reason for this is that they lack the ability to reason. Unfortunately, it is an essential requirement of context-awareness that uncertain context can be derived from sensors, an example of such would be the location of a user within a context which can be uncertain [65]. There is available a wide range of reasoning mechanisms that have been mentioned in the literature for reasoning about uncertain context (see Table 2-4), below is an overview of these mechanisms provided by [65, 66]:

- **Fuzzy Logic:** Fuzzy Logic is a type of data processing used in advanced computer systems. For processors that process simple information, the likelihood that an event is a certainty, either true or false. In contrast, Fuzzy Logic systems aim to find a solution to complex problems, in other words information that is too complex or information that cannot be analysed using a conventional technique. Specifically, this approach considers the likelihood of something happening in degrees of truthfulness or falsehood. Therefore, this approach is applicable in systems that require multi-sensor fusion, need to consider a subjective context or where there is a conflict between different context variables. It is possible to combine two or more fuzzy sets to create a new fuzzy set.

- **Probabilistic Logic:** Probabilistic logic is about logical reasoning based on probability. Statements based on this approach may be, for example, 'the probability of X is more than 25 percent' and 'the probability of Y is half the probability of Z' when X, Y and Z are random occurrences. This approach allows for rules to be written in order that govern reasoning about the probability of events. Moreover, such rules can be used for better context information derived, multi-sensor fusion and for gaining higher-level probabilistic contexts. If there is conflicting information from different sources, these rules can provide a solution. According to [67] these rules can be reasoned about using rule engines such as prolog. Unfortunately, one of the disadvantages of this approach is that it does not give sufficient expressive power to capture variable dependencies and uncertainties [68]. In addition, does not consider the temporal aspect of the data [68].

- **Neural Networks:** Neural networks are comprised of interconnected constituents referred to as neurons. Such a network is actually based on the way that a human brain works when carrying out tasks. Through the use of these neurons they perform parallel and a non-linear computing and they are suitable for solving ill-defined that would usually need a large amount of computation. However, this approach has a number of disadvantages, including that not all architectures are suitable for its application and that its capability to predict is not as accurate as other reasoning techniques and the training of the network is slow [66].

- **Bayesian Networks (BNs):** This type of network is comprised of directed acyclic graphs, where nodes represent random variables which are the events and the causal relationships are represented by links between these nodes. A significant advantage of Bayesian networks is that a set of variables as a joint distribution can be shown as the result of the local distributions of the corresponding nodes. Such networks are effective for the representation and storing of probabilities. Furthermore, Bayesian networks have the ability to reason where contextual information is not certain and they can deduce results from causes and causes from results, moreover, they are particularly effective in deducing high level context from low level information. A disadvantage of Bayesian Networks is that they cannot cope with a continuous flow of data, therefore, they are not suited for representing contextual information that is continuously changing [64, 69].

Context reasoning technique	Advantages	Disadvantages
Fuzzy Logic	- Simple to use. - Suitable for modelling uncertainty. - Deal with ambiguities. - Suitable for real time systems.	- Cannot deal with inaccurate or incomplete data.
Probabilistic Logic	- The rules are suitable for context information derived. - Multi-sensor fusion. - Gaining higher-level probabilistic contexts.	- Cannot model dependencies and uncertainties between variables. - Does not consider the temporal aspect of the data.
Neural Networks	- Suitable for problems that need a large amount of computation.	- Less accurate. - The training of the network is slow.
Bayesian Networks (BNs)	- Suitable for modelling and reasoning about uncertainty in contextual information.	- Not suitable for system that continuously changing over time.

Table 2-4: Comparison of Context reasoning technique.

The integrity status of an in-vehicle navigation system is considered to be high-level context, this is because the map matching is not a certain process [70]. Therefore, for the algorithm of this study data from the navigation system should be combine so that its integrity can be reasoned. To put it simply, the algorithm should use logical rules to determine an output from multiple inputs. In summary, from the review of methods that can be used to make sense of uncertain data derived from context sensors, fuzzy logic is most appropriate because of the following:

- It can model uncertainty and express it in linguistic terms that have meaning [71].

- It can generate conclusions based on qualitative language and vague terms [72].
- It can be employed in decision making and planning for real world situations [73].
- It can deal with ambiguities in real world problems [74].

2.6.6 Context-aware Systems

Context-aware systems can sense and interpret data from within a context and then act according to changes in that environment [75]. Context-aware computing was first defined in 1994 by [51], who said that a context-aware system behaves dynamically in response to information coming from that environment. A more detailed definition is provided by [60]**,** who said that context-aware systems can adapt their behaviour or functions to their context without the user intervening and therefore, increases system effectiveness by considering environmental context. An alternative definition was provided by [59], who said that such systems use contextual information in order to provide users with information and services.

Generally, in the definitions presented above it is clear that context-aware systems are capable of offering automatic services to users, and that they react to their context based on information by sensors [76].

According to [77] there are two types of context-aware systems, as following:

- **Active:** A system that has this type of context-awareness changes its behaviour according to sensory input from the context and can automatically change its behaviour according to changes in that context.

- **Passive:** A system based on this type of context awareness can only sense a context and provide information to users, it cannot adapt itself according to changes in that context.

The use of active or passive context depends mainly on how the contexts is used in the application. Therefore, in this research the system can be either passive or active.

2.6.6.1 Context-aware System Categories

It has been suggested by [56], that how context-aware systems utilise and process data from the context can be divided into three category as shown in Table 2-5.

Category	Description
Presenting information and services to users	This approach is where context information is presented to the user, or context information is used to offer possible actions to the user so that services can be carried out.
Automatically executing the service	This involves systems that carry out commands by the user according to changes in the context. An example of this type system is the NeverLost application where a satellite navigation system automatically recalculates routes [78].
Attach context information for later retrieval	This approach involves tagging captured data that can be used later. An example of this type of system is Forget-Me-Not that tags context information in order to resolve issues, by for example, finding lost documents [79].

Table 2-5: Context-aware system category.

2.6.6.2 Context-aware System Abstract Architecture

Essentially, the main objective of context-aware systems is to gather contextual data and then act accordingly. During this operation there are three basic phases [57], described in Table 2-6.

Phase	Description
Sensing phase	This is where the sensors gather information about the context, this information can include temperature, light and people of a context. This information is used to decide the best course of action for that context.
Thinking phase	Once the contextual information has been gathered reasoning or thinking about that information has to take place, this will allow the system to reason more knowledge about the context.
Acting phase	Based on the reasoning, appropriate actions are carried out at this stage depending on the needs of the user.

Table 2-6: The basic phases of context-aware systems.

There are a number of different architectures of context-aware systems in the literature. Figure 2-5 shows a five layer architecture for a context-aware systems presented by [20].

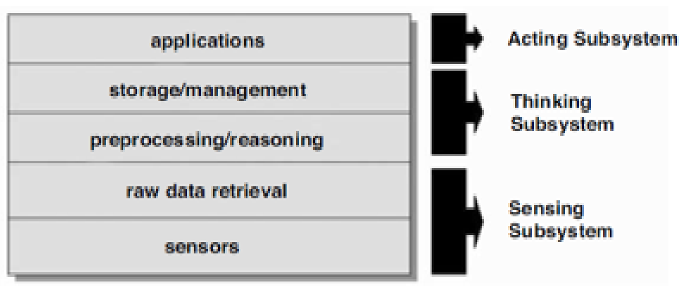

Figure 2-5: Abstract architecture for context-aware systems [20].

The first two layers constitute the sensing phase described above. The sensors layer includes different types of sensors, also discussed in the above. The raw data retrieval layer represents the gathering of the data by the sensors. Drivers are employed to retrieve the data from the sensors and Application Programming Interfaces (APIs) are used to retrieve data from virtual sensors [60]. The Pre-processing layer represents the processing of the raw data in order to derive a better understanding of the contextual information. Although this layer is not used by all context-aware systems, it is useful in providing information when collected raw data is not clear [60]. The storage and management represents where information is organised and then made available to users through an interface. Users can access the data in two different ways, firstly, synchronously whereby they request information from the server and wait for the information to be returned, and asynchronously, whereby, users state an event that they are interested in and when it occurs then they are informed [60]. The application layer represents the actual action that will take place in response to the contextual information.

2.7 Review of Map Matching (MM) algorithms

2.7.1 Definition of Map Matching (MM)

Map matching (MM) involves locating the position of a vehicle relative to a road digital map. This method involves the use of positioning data gathered from a navigation system and road networks data gathered from a digital map database [13, 80]. The initial part of the algorithm is to first find out the road segment where the vehicle is then finds the vehicle's exact position on that road segment [72] which is achieved through a comparison of the vehicle's trajectory with the potential pathways in the digital map database close to where the vehicle could be [81].

The literature [13] mentions numerous map matching algorithms including algorithms that use simple searches and those using more complex algorithms such as a Fuzzy logic. These algorithms are categorised as topological, geometric, probabilistic and advanced [13]. The following sections discuss each of these categories.

2.7.1 Geometric Map Matching (MM) Algorithms

A geometric map matching (MM) algorithm uses only the curves that are in the geometric information of the digital road network [82]. They were developed in 1990s and fall under three categories which are point-to-point, point-to-curve and curve-to-curve algorithms [83], each of which are discussed in more detail below.

2.7.1.1 Point-to-point MM algorithms

This type of MM algorithm, also referred to as the simple search method, the position of a moving vehicle gained from a navigation system is correlated with a node representing a section of a road [83]. These nodes is essentially comprised of longitude and latitude coordinates, and can be the starting or ending node of a link [84]. These links may be in a straight line or a curve, and they have middle points, referred to as shape points. Figure 2-6 provides an illustration of the point-to-point MM approach.

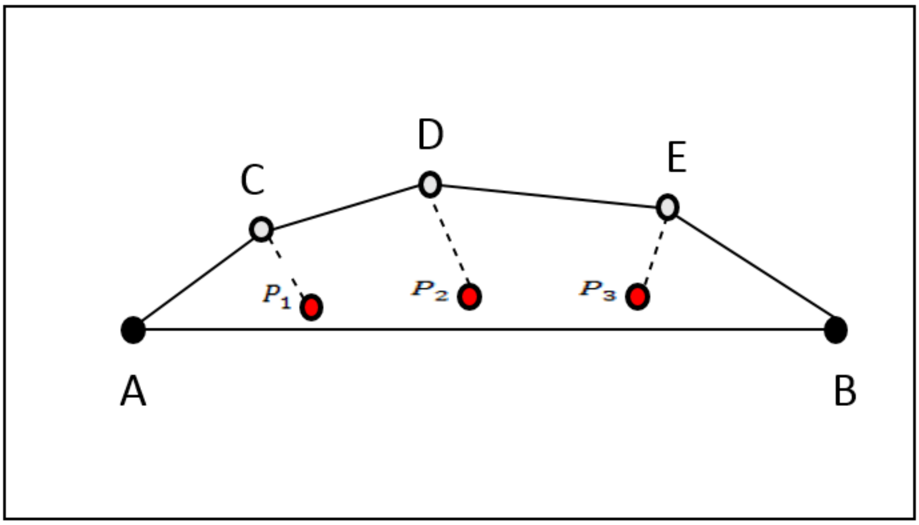

Figure 2-6: Example of point-to-point MM algorithm [14].

In the example provided here,P_1, P_2 and P_3 are the positioning data gathered from the navigation system A and B are the nodes and C, D and E are shape points. The estimated locationsP_1,P_2 and P_3 are matched by the algorithm to the nearest point in the road by calculating the distance between point P to all of the nodes in the road and then chooses the nearest one [85]. Here the link between A and B is where the vehicle is travelling, however, using the point-to point MM approach the route is actually shown by the links between C and D and D and E, because P_1 is near to C, P_2 is near to D and P_3 is near to E. Therefore, potentially the algorithms can identify the incorrect travelling route. Although, the point-to-point MM algorithm is widely adopted and easy and fast to use it is sensitive to the road network design and some problems may occurs during execution [13, 14].

2.7.1.1 Point- to- curve MM algorithm

In this type of algorithm, data gathered from the navigation system is matched against the nearest curve on the road network instead of a point as with the previous method [85]. The curves are comprised of line segments which are linear in nature [13]. The distance between a point P and a line segment is calculated by the algorithm. Following this, the algorithm chooses the line segment which is closest to where vehicle is travelling [83].

This approach has an advantage over the point-to-point approach because it provides the location of a vehicle on a link [14]. An example of this is illustrated Figure 2-7 where the closest link to the points P_1,P_2 and P_3 (which is A-B) is selected, which represents the route the vehicle is travelling. However, there are disadvantages to this algorithm which include that the positioning data is considered separately, if there is no available previous matching data and vehicle speed and heading are not employed during matching [11].

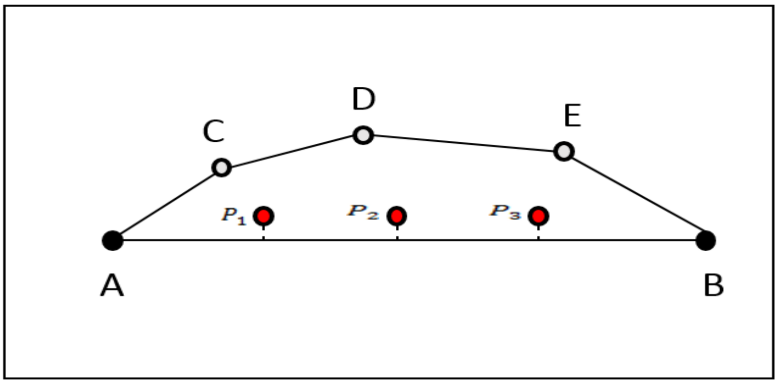

Figure 2-7: Example of point-to-curve MM algorithm.

2.7.1.1 Curve - to- curve MM algorithm

A curve-to-curve MM algorithm uses the position gathered from the navigation system as a curve [83]. As described by [13], the first part of this approach is to identify the candidate nodes by employing a point-to-point MM algorithm. Following this, piecewise linear curves are constructed for each candidate node by using the paths that start from that node. Thereafter, piecewise linear curves are constructed using the points, and the distance between them and the road segment curves are calculated. Finally, a road segment nearest to the curve is chosen, this curve is created from the positioning points from where the vehicle is travelling. Figure 2-8 illustrates this approach.

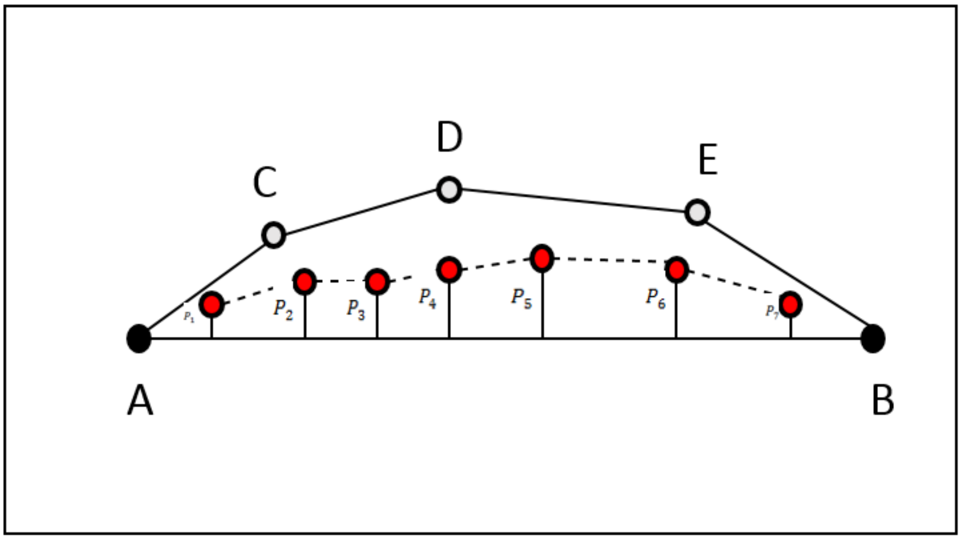

Figure 2-8: Example of curve-to-curve MM algorithm.

It is clearly illustrated in the figure above, that in the curve-to-curve approach the A-B link is selected because it is the link where the vehicle is actually travelling. The reason for this is that the first point of the vehicle trajectoryP_1 is both close to node A and the road segment that begins from node A.

When the curve-to-curve MM approach wants to identify candidate road segments, it is highly dependent on the point-to-point MM algorithm, and as a result the algorithm sometimes provides unexpected results. An additional problem is that this approach cannot give a real-time vehicle position, moreover, this approach does not utilise useful information including road connectivity, speed and direction [14].

Overall, an algorithm that only utilises geometric information is simple and fast, although geometric MM algorithms can often produce unexpected results [13]. The geometric MM algorithms can be improved by using additional information in road segment identification process [86].

2.7.1 Topological Map Matching (MM) Algorithms

A topological MM algorithm utilises the topological attributes of a road which include connectivity and orientation [87]. Furthermore, this approach utilises previous matching information, speed and direction, turn restriction as well as road geometry [82, 88]. Figure 2-9 is an illustration of the topological MM algorithm.

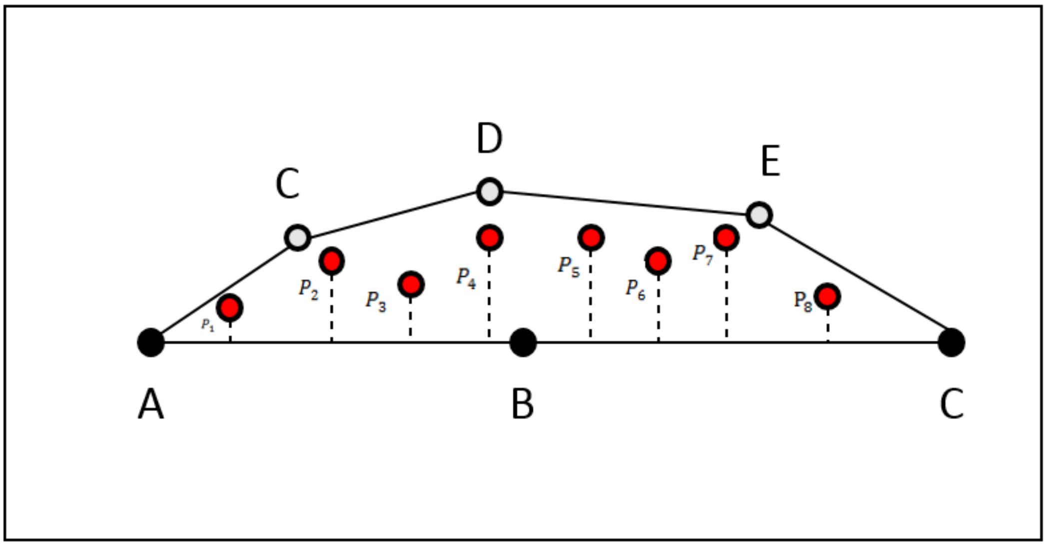

Figure 2-9: Example of topological MM algorithm.

As illustrated in Figure 2-9, the vehicle is on link A-B and B-C. If the topological MM approach is employed these links will be identified correctly because it considers both topological information and matching information derived from earlier analysis. To provide an example, if we consider pointP_5, although it is closer to link D-E, this link does not have a direct connection with link A-B on which the vehicle previously travelled on, therefore, the link B-C will be chosen as the travelling link for point $P_5$by the algorithm.

If the same scenario shown in the example above was applied to the geometric MM algorithm, specifically for point-to-curve for points P_5, P_6 and P_7, the corresponding link would be D-E, however this is not correct, and therefore the topological MM approach is better at finding the correct link than the geometric MM algorithm. The main reason for this is that the topological MM approach considers previous matching information, as well as vehicle speed and direction. Furthermore, it has a more logical way of identifying the correct links [86], although it does depend on identifying the correct link in the first place [82, 89].

2.7.2 Probabilistic Map Matching (MM) Algorithms

This algorithm uses probability theory to identify the correct road segment by choosing a set of segments for each point and establishing the area of the error around each positioning point, it may be an oval, circle, square or a rectangle shape [90]. Then, the road segment found within this area is selected as the candidate for that particular positioning point. Zhao [72], says that the error variances for a particular point are used to derive the error area. Subsequently, the error area is superimposed onto the road network in order to reveal the road segment where the vehicle is travelling [72]. If there is more than one road segment identified in the error area, then other information such as distance, direction and connectivity should be used to determine the correct one [13].

2.7.3 Advanced Map Matching (MM) Algorithms

This algorithm employs sophisticated techniques such as artificial neural networks, fuzzy logic and Bayesian interference in order to identify the correct road segment [72, 91-93]. This algorithm using better matching techniques and is therefore, better than the aforementioned algorithms [86]. However, these type of algorithm are difficult and slow to implement because they use much more data [86]. Thus, they may not be suitable for transport applications that need to work in real time.

It fact real time positioning is absolutely necessary for such type of ITS services which include emergency management and route guidance. Moreover, because the position of a travelling vehicle is continuously changing, it is necessary for any system to be able to identify the vehicle's position instantly. From the above review of different algorithms, it can be seen that there are different types of approaches to solve the problems related to map matching and that they all have their limitations. In the algorithm of the present research, a topological map matching algorithm is used to locate the vehicle on the correct segment of road for the following advantages:

- Using topological information to identify the correct road segment can significantly improve the map matching process [89].
- The integrity of the map matching process can be improved by using information such as speed and direction.
- This algorithm is easy to use and very quick at identifying the correct road segment where the vehicle is travailing, and is therefore, suitable for supporting real time ITS services [86].

However, topological map matching algorithms have the following issues:

- The performance of algorithm degrade in very complex environments (e.g. junctions and deans urban areas) [14].
- The identification of the correct link depends on the first matched positioning point.

2.8 Existing integrity methods

Integrity of in-vehicle navigation systems is considered as an essential requirement for supporting most ITS services. Therefore, many research have been carried out in order to monitor and improve the integrity of in-vehicle navigation system using different methods. Some research have tried to monitor the integrity by checking the validly of raw positioning data (Andrés [16]). Other research have tried to monitor the integrity of the map matching process (Jabbour et al. [17] and Yu et al. [10]) or combination of both (Velaga [14] and Quddus [18]). In the following sections each of these available integrity methods are discussed along with their limitations.

2.8.1 Positioning integrity methods

Andrés [16] proposes a geo-object recognition algorithm for detecting vehicles and charged them for the price of a geo-object region whenever they travelled inside it, geo-object represents the tolled regions in ETC system. The author utilises Geo-fencing method to segment the road network into geo-object regions. In addition, he uses the RAIM approach to check the integrity and the validity of the vehicle's position inside the geo-object region. After determining the integrity of the vehicle's position, his algorithm decide whether to charge the driver or not. If the vehicle's position is not valid in the geo-object region the algorithm wait for the next positioning point.

The author considered RAIM method in the integrity process, however RAIM alone is not enough for checking the integrity of vehicle's position especially for critical applications. In addition, no map matching algorithm is used in the integrity process, which can help to improve the accuracy of positioning data [8-10].

2.8.2 Map matching integrity methods

Jabbour et al. [17] develops a map matching integrity method for land vehicle navigation systems, based on multihypothesis technique. The author introduced two different criteria to check the integrity of map matching process which are: number of efficient hypotheses and normalised innovation square. In addition, two threshold values are derived empirically and used in the monitoring process. The integrity method was tested in France using real filed data. Their results illustrate 88.8% valid integrity warnings. However, they did not consider the errors related to the road map.

Yu et al. [10] provides a curve pattern matching algorithm to detect mismatches and improve the reliability of the map matching result. The algorithm start by matching a vehicle position on the road map then forms two curves from the positioning point and the matched points, respectively. The algorithm detects the mismatch by comparing the two curves together. If a mismatch is detected the algorithm restart aging and correct the mismatch. The algorithm was evaluated using a large amount of data (3,000 km) collected in Hong Kong. The performance was evaluated using missed detection rate (MDR) only and was 1.41%. Errors related to positioning data and the road map were not considered.

2.8.3 Hybrid Integrity methods

Quddus [18] proposes a method for checking the integrity of map matching process. The author considered two sources of information: (i) errors related to positioning data and (ii) errors related to map matching process. A fuzzy inference system is used to combine these information and infer the integrity threshold. The threshold is calculated empirically to be 70. However, this value may vary depending on the type of operational environment and the sensors that are used. The method was tested using three different map matching algorithms. The result were found to be 91.2% valid integrity warnings for the topological map matching algorithm.

Velaga [14] presents an integrity algorithm which takes into consideration errors related to both positioning data, digital road network and the map matching process. In the algorithm, RAIM method is applied to check the integrity of the GPS positioning data, and fuzzy logic is used to determine the validity of the map matching process. In addition to these two attributes the complexity of road network during the integrity process is also considered. The algorithm was tested using field data collected in Nottingham. The result showed that the algorithm can provide 98.2% valid integrity warnings.

As discussed above, existing methods for checking the integrity of in-vehicle navigation systems are centred on positioning data and map matching integrity or both. Andrés [16] checked the integrity of raw positioning data and ignored the integrity of other information. In addition, he does not take the advantages of map matching algorithm in the integrity process. While, Jabbour et al. [17] and Yu et al. [10] focused on checking the integrity of the map matching process without considering the errors related to the positioning data or errors related to other inputs to the map matching process. Other researchers (Quddus [18] and Velaga [14]) tried to combine both errors related to positioning data and map matching process in order to enhance the performance of the integrity monitoring process. However, the quality of in-vehicle navigation system output depends on different source of errors, which are: (i) errors related to the positioning data, (ii) errors related to the map matching process, and (iii) errors related to other input to the map matching algorithm (e.g. vehicle speed). None of the above methods have considered errors in other input to the map matching process such as speed, see Table 2-7.

Reference	**Integrity method**		
	Position	Speed	Map-Matching
Andrés [16]	Yes	-	-
Jabbour et al. [17]	-	-	Yes
Yu et al. [10]	-	-	Yes
Quddus [18]	Yes	-	Yes
Velaga [14]	Yes	-	Yes
The proposed algorithm	Yes	Yes	Yes

Table 2-7: Summary of the existing integrity methods.

2.9 Summary

Speed is an essential factor in the map matching algorithm to identify the vehicle position on the link. Therefore, checking the integrity of speed can future improve the integrity of map matching process. In this research we will consider errors related to the positioning data, speed of the vehicle, and result of map matching process, simultaneously, in the integrity monitoring process. Clearly, considering these source of errors simultaneously can lead to a better result. In addition, the proposed solution can provide information about the integrity of vehicle's speed which is significantly important; especially for critical ITS applications such as pay as you speed [94].

This chapter has presented an overview of location-based ITS service. Some services that use navigation systems as a main component of the service were identified and discussed. This chapter also defined the RNP parameters for land vehicle navigation systems. Then, the RNP parameters for different types of ITS services ware discussed. Moreover, the strategies that can be used to derive the RNP parameters for each group of ITS services were presented. The second part of this chapter, reviewed the concept of context and context-aware systems including methods that have been used in modelling and reasoning about a context. The third part of this chapter, provided an overview of the map matching algorithms along with their limitation. Finally, a review of existing integrity algorithms was provided. In the following chapter, a detailed description of the GPS integrity monitoring system architecture will be presented.

Chapter 3

System Architecture

Objectives

- Propose a GPS integrity monitoring system architecture for land vehicle navigation systems.
- Explain the mechanism of the integrity monitoring process.
- Describe the three subsystems of the architecture and its components based on the concept of context-awareness.

3.1 Introduction

This chapter presents the design of a GPS integrity monitoring system based on the concept of context-awareness that senses the in-vehicle navigation data, reasons about its integrity and reacts upon it. It is built on a new technique to reason about the integrity status of an in-vehicle navigation system from the captured information, and avoid any misleading or faulty information by warning the driver.

The system is composed of three main subsystems: sensing, reasoning, and the application subsystem. These subsystems correspond to the main phases of the context-aware system, which are: sensing, thinking, and acting phases. The abstract layered context-aware framework is utilised to construct the components of this system (see Chapter 2).

As discussed earlier in Chapter 2, the integrity status of the in-vehicle navigation system is considered to be uncertain context, due to the fact that the map matching is not a certain process [70]. Therefore, a reasoning technique is essential in order to combine different contextual information and deduce the current integrity status of the in-vehicle navigation system. In the proposed system, fuzzy logic is applied in the integrity monitoring algorithm to combine the contextual data and deduce the integrity status of the navigation system. Two types of sensors are integrated to the system in order to collect the required contextual information about the vehicle (position, speed, and direction).

The following section provides the explanation of the integrity monitoring mechanism. A detailed description of the system and its components is then given. This includes different types of sensors to collect real time contextual information about the vehicle, an integrity monitoring algorithm to carry out the reasoning process and alert unit to trigger an in-vehicle alarm.

3.2 GPS Integrity monitoring system mechanism

This section presents the mechanism for monitoring the integrity of the in-vehicle navigation system. The flowchart of the integrity monitoring process is presented in Figure 3-1. The process starts with the sensing of the vehicle current context, including vehicle's location, heading and speed. Physical sensors including location sensor (GPS), and in-vehicle wheel speed sensor are used to capture the required context information about the vehicle. After collecting the raw contextual data, a context interpreter transforms it into a form that can be understood by the machine. This step can be accomplished by using different context modelling techniques as discussed in Chapter 2 (this step falls outside the scope of this research).

After interpreting the data the system performs the integrity monitoring algorithm based on fuzzy logic, in order to reason about the integrity status of the in-vehicle navigation system. The algorithm composed of three phases of integrity checks, which are:

Positioning integrity to check the consistency of the positioning data, Speed integrity to calculate the speed of the vehicle from the sensed contextual data and ensure its integrity. Finally, the map matching integrity to check the integrity of the map matching results. Each phase involves the application of different techniques to examine the consistency of the contextual data (a detail description of each phase will be given in Chapter 4). If there is no integrity in the positioning data, speed data, or the result of the map matching process is rejected, then a suitable in-vehicle alarm is sent out to the driver. Otherwise, no action is performed and the system will continue to sense new information about the vehicle. This system is based on the concept of context-awareness as the process sense, reason and act in real time, based on the acquired contextual data.

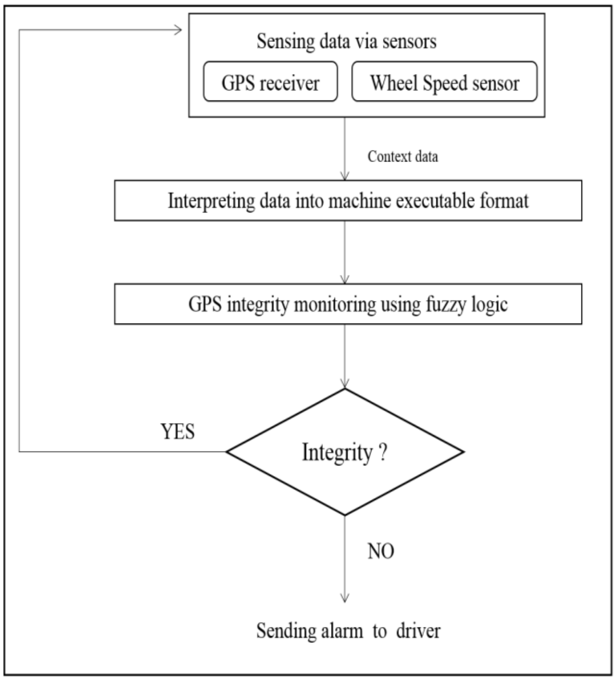

Figure 3-1: GPS integrity monitoring system mechanism.

3.3 System architecture

The components of the proposed GPS integrity monitoring system, as based on the concept on of context-awareness are describe in this section. Figure 3-2 illustrates the components of the system in detail, including the way that these components are connected to each other in order to monitor the integrity of the in-vehicle navigation system and warn the driver using in-vehicle alarms.

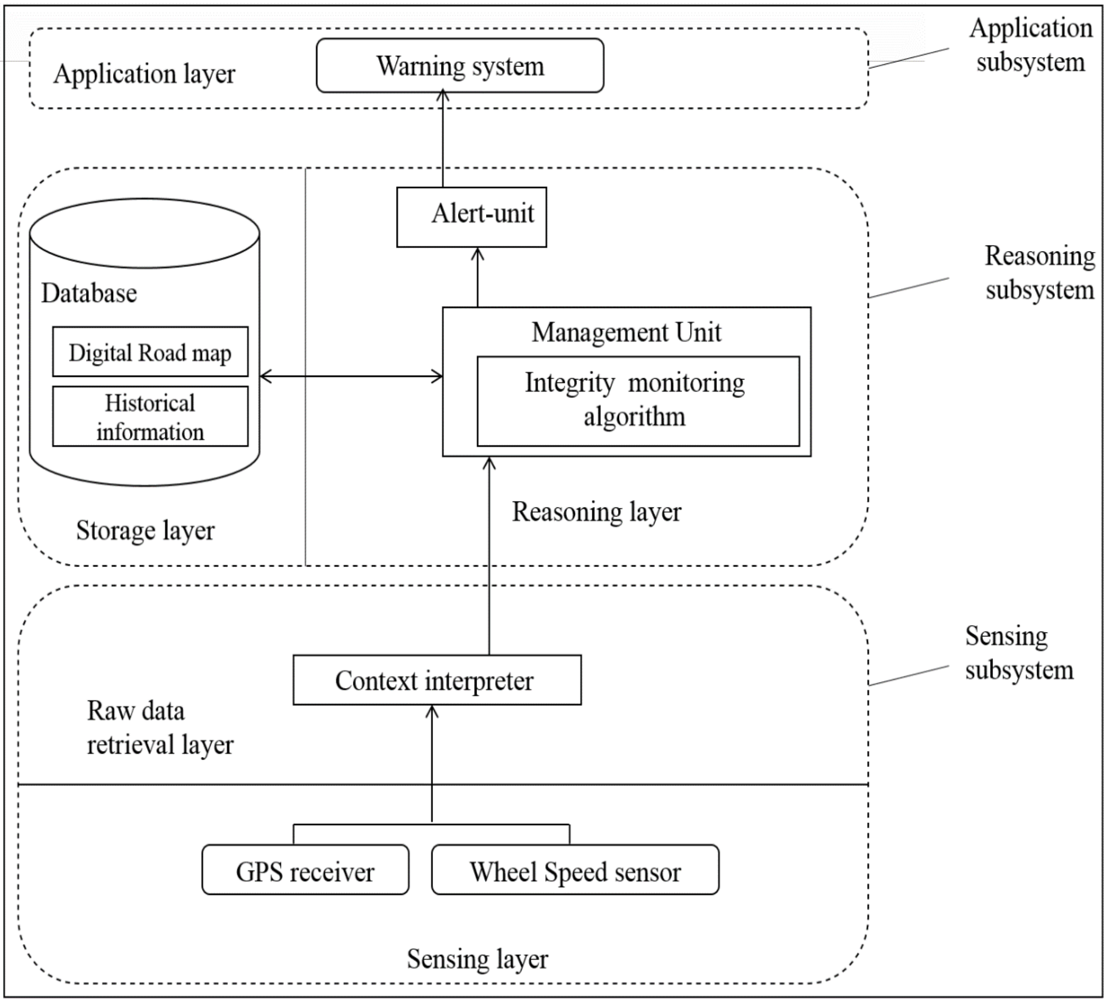

Figure 3-2: GPS integrity monitoring system architecture.

As shown in Figure 3-2 above, the proposed system is designed based on the five layer context-aware framework [20], and is composed of three main subsystems: (i) the sensing subsystem, (ii) the reasoning subsystem, and (iii) the application subsystem, which corresponds to the main phases of the context-aware system: the sensing, thinking, and acting phases, respectively as mentioned in Chapter 2.

Within the system the integrity monitoring process is carried out using six different components which are distributed over five layers according to the abstract layered context-aware framework. First one finds the sensors which represent the sensor layer. Next is the context interpreter component that represent the raw data retrieval layer, followed by the management unit, which is responsible for performing the integrity monitoring algorithm, and the alert unit that represent the reasoning layer. The digital road map database represents the storage layer. Finally, one finds the warning system component, which corresponds to the application layer. The following sections provide a detailed description of each subsystem and the related components.

3.3.1 Sensing subsystem

This section provides a description of the sensing subsystem, including how contextual data about the vehicle is sensed and the type of the sensors used to gather this information. The sensing subsystem, corresponds to the sensing phase in the context-aware system, responsible for sensing the current context of the vehicle, including position, speed, and heading, then interpreting this context into a machine executable format in order to be processed by the reasoning subsystem. This subsystem composed of:

- **Sensors:** Here, the contextual data about the vehicle is collected then transfer it to the next layer (raw data retrieval), as shown in Figure 3-2. It composed of set of sensors that is integrated to the vehicle and connected to the system. Generally, different type of sensors can be used (physical, virtual, or logical) in order to have access to different types of information according to the system needs (see Chapter 2). In the proposed system, two physical sensors are used to collect the required context information, these are:
    - GPS sensor: GPS provides information about the vehicle including current position, heading, and speed. In the proposed system, GT-31[1] (GPS receiver) is used to provide the required information for the integrity monitoring algorithm, including, Doppler speed data, which represented by the horizontal dilution of precision (HDOP)[2] value, speed dilution of precision (SDOP)[3] value, and other basic positioning information (such as X, Y, Z coordinates).
    - Wheel speed sensor: wheel speed sensor provides information about the current vehicle's speed from its wheels, which can be accessed from the

[1] See http://www.locosystech.com/

[2] Horizontal dilution of precision (HDOP) "allows to more precisely estimate the accuracy of GPS horizontal (latitude/longitude) position fixes" [95].

[3] Speed dilution of precision (SDOP) is a SiRF3 parameter that can be used to determine the accuracy of GPS Doppler speed [103].

in-vehicle sensor network. This sensor is used to check the status of the vehicle's speed before relying on GPS Doppler speed data; due to the fact that GPS Doppler measurements are not reliable at low speed (2 m/s [96]).

- **Raw data retrieval:** The purpose of this layer is to retrieve the raw contextual data from the sensing layer and abstract the low-level sensing details from the upper layers. It is comprised of one component as follows:
  - Context interpreter: This component is responsible for translating the acquired context data that has been received from the sensing layer into machine executable format. The data received from the GPS sensor should be transferred to a form that can be understood by the reasoning layer. This transformation process (modelling process) can be implemented using different modelling methods (e.g. object oriented), this is beyond the scope of this research.

3.3.2 Reasoning subsystem

This subsystem, corresponds to the thinking phase, responsible for checking the integrity of in-vehicle navigation system and warning the driver about its integrity status. As discussed in Chapter 2, the integrity status is uncertain information and consider to be high level context. Thus, an integrity monitoring algorithm based on fuzzy logic is performed in order to reason about the integrity status using the contextual data received from the sensing subsystem, in real time. This subsystem composed of:

- **Reasoning:** It reason about the integrity of the acquired context data about the vehicle. Then, warn the driver about the integrity status of the in-vehicle navigation system by triggering an in-vehicle alarm to avoid any misleading information. This layer is composed of:

    - Management unit: It manage and control all the activities of the reasoning layer. It performs the integrity monitoring algorithm in order to reason about the integrity status of the acquired contextual information. The algorithm is composed of three phases of integrity checks (see Figure 3-3). The positioning integrity phase is responsible for checking the consistency of the positioning data using RAIM method. The speed integrity phase is responsible for calculating the speed of the vehicle, form GPS Doppler data, and ensuring its integrity. Finally, the map matching integrity phase is responsible for checking the integrity of the map matching results using fuzzy logic. The fuzzy logic system is composed of eighteen rules based on three information: distance, speed, and heading of the vehicle, before and after the map matching process. If there is no integrity in the positioning data, speed data, or the result of the map matching process is rejected, then the management unit invoke the alert unit to send out a suitable in-vehicle alarm to the user.

Otherwise, no action is performed and the system will continue to sense new information about the vehicle. Detail description of each phase of the algorithm will be given in Chapter 4, which is the focus of this thesis. The algorithm has been implemented and tested using 80 km of real field data collected in Nottingham, further details are given in Chapter 5.

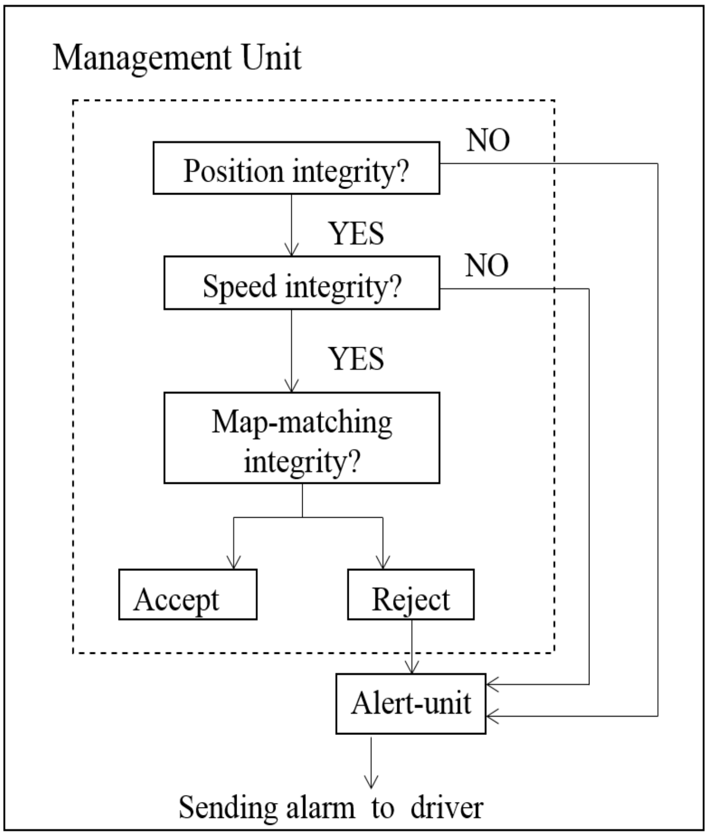

Figure 3-3: GPS integrity monitoring algorithm.

- Alert unit: The alert unit is responsible for sending a suitable in-vehicle alarm and warning the driver about the integrity status of the in-vehicle navigation system. It is invoked by the management unit in cases any misleading information is detected by the integrity monitoring algorithm.

- **Storage:** The storage layer is responsible for storing and retrieving the digital road map data and historical map matching information (such as road connectivity, and road direction).

3.3.3 Application subsystem

The application subsystem corresponds to the acting phase in the context-aware system. In the proposed system this subsystem is responsible for warning the driver about the integrity status of the navigation system; through sending in-vehicle alarms. In addition, any location based ITS application (such as electronic payment system) can be integrated with the proposed integrity monitoring system via this layer in order to utilise the integrity status of in-vehicle navigation system and avoid using any misleading or faulty information.

3.4 Summary

This chapter presented an integrity monitoring system architecture for land vehicle navigation systems. The system architecture is composed of three subsystems: sensing, reasoning and an application subsystem and was developed on the basis of the concept of context-awareness. In addition, the components of the integrity process were designed using the abstract layered context-aware framework. Here, an integrity monitoring algorithm was integrated within the reasoning layer in order to reason about the integrity status of the acquired context data. The main purpose of this system is to provide a robust and reliable mechanism for checking the integrity of in-vehicle navigation system and detect any misleading information based on the concept of context-awareness. In the following chapter a detailed description of the development of the integrity monitoring algorithm, including positioning integrity, speed integrity and map matching integrity phases will be presented.

Chapter 4

Development of a GPS Integrity Monitoring Algorithm

Objectives

- Propose a novel integrity monitoring algorithm for land vehicle navigation systems.
- Describe the mechanism of the integrity algorithm.
- Describe the steps for the development of the three phases of the proposed algorithm.

4.1 Introduction

As previously discussed in Chapter 2, the majority of existing integrity algorithms used for in-vehicle navigation systems focus on either the positioning integrity or the map matching integrity, or a combination of both. However, these algorithms are not adequate to support safety and critical ITS services, as they do not consider the integrity of other input to the map matching process such as vehicle's speed. Therefore, considering the errors related to vehicle's speed in addition to the errors in the positioning data and map matching process have the greater potential to lead to more accurate outcomes.

This chapter presents a novel algorithm for monitoring the integrity of the in-vehicle navigation system. The algorithm has the ability to detect any inconsistency related to the positioning data, speed data, and map matching process, using three phases of integrity checks. These consist of: (i) positioning integrity, (ii) speed integrity, and (iii) map matching integrity. The following section describes the proposed integrity algorithm. This is then followed by a detailed explanation of each phase.

4.2 Integrity monitoring algorithm

This section presents the steps for developing the GPS integrity monitoring algorithm. The algorithm will perform three phases of integrity checks in order to determine the integrity of the in-vehicle navigation data. These consist of: (i) the positioning integrity phase, (ii) the speed integrity phase, and (iii) the map matching integrity phase (as previously demonstrated in Chapter 3). A detailed flowchart for the designed integrity monitoring algorithm is illustrated in Figure 4-1, below.

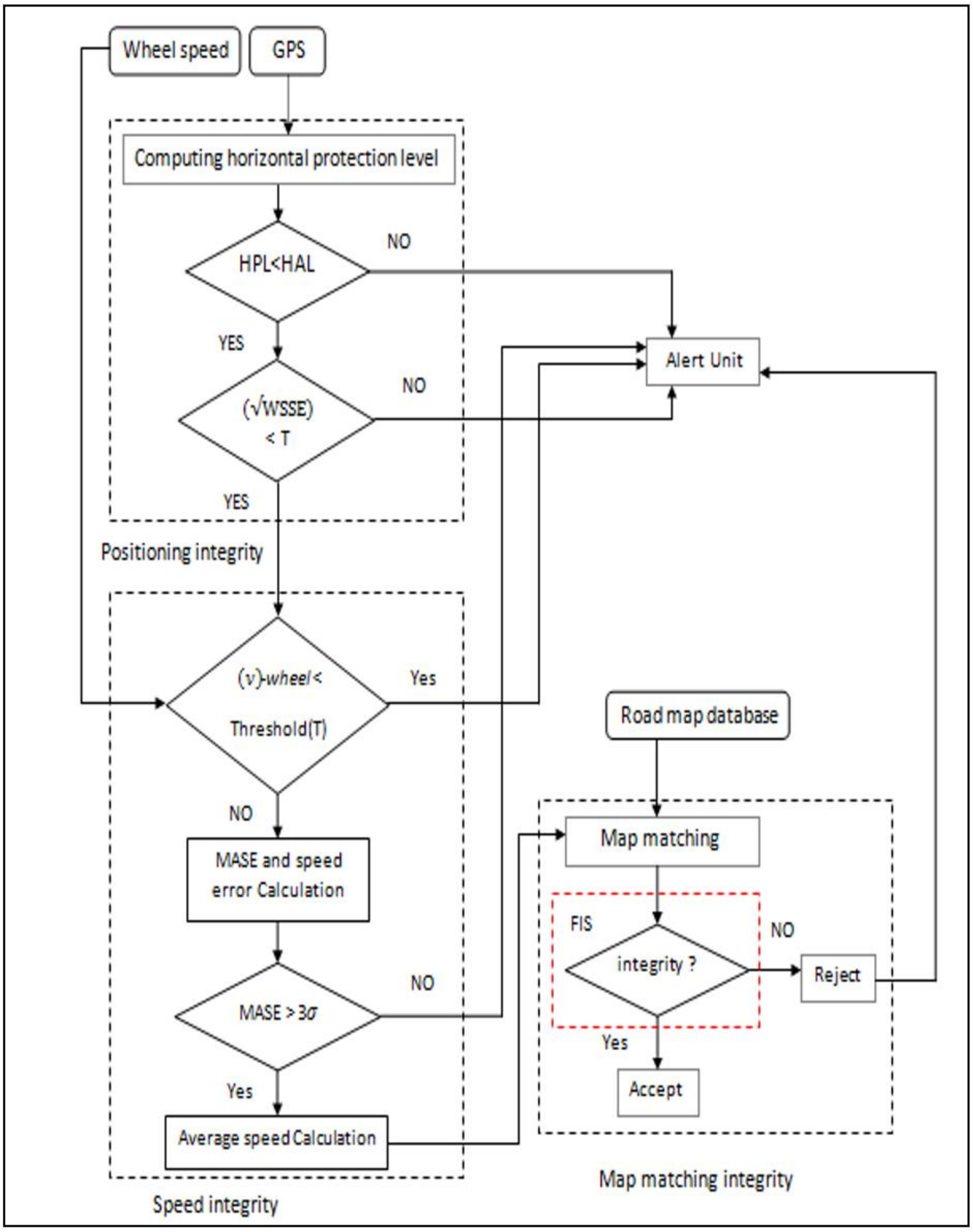

Figure 4-1: A flowchart illustrating the integrity monitoring algorithm.

As demonstrated in Figure 4-1, the algorithm starts by checking the integrity of the positioning data captured from the navigation sensor (GPS). Firstly, it calculates the horizontal protection level (HPL) based on the positioning information received from the GPS, which represents the upper limit for the GPS positioning error in the horizontal plane [14]. This is followed by checking the Receiver Autonomous Integrity Monitoring (RAIM) availability, in which the value of HPL is compared with the horizontal alert limit (HAL). The HAL represents the maximum allowable error along the horizontal plane and cannot be exceeded without alerting the user [14]. When the HPL exceeds the HAL, the RAIM is not available and a suitable alarm should be given to the user via the alert unit. If this is not performed, the RAIM is assumed to be available and thus the algorithm will continue to check the integrity of the positioning data. Any errors in the calculated position are detected by comparing a test statistic $\sqrt{WSSE}$ (see section 4.2.1) with a selected threshold (T). An alarm needs to be given to the user should the test statistic exceed the selected threshold. If it is not given, then the integrity of the positioning data is available, and thus, the speed integrity process is followed.

The integrity of the vehicle speed is also checked. Firstly, the wheel speed (v)-wheel is compared with a selected threshold (T). If the wheel speed is less than the threshold (T), then an alarm needs to be given to the user. Otherwise, the maximum allowable speed error (MASE), along with the estimated speed error, are calculated based on the GPS Doppler data. The MASE represents the upper limit for the GPS Doppler speed error that cannot be exceeded without alerting the user. The calculation of MASE, and the GPS Doppler speed error are explained in section 4.2.2. If MASE is three times greater than the standard error σ of the Doppler speed, then the integrity of the GPS Doppler speed is assumed to be available, and the average speed will be calculated. Otherwise, the speed integrity is unavailable and an alarm must be given.

When the speed integrity is available, a map matching algorithm is carried out to locate the travelling vehicle on the road map. A fuzzy inference system (FIS) is subsequently employed, in order to reason out the integrity of the map matching process and determine whether the result should be accepted or rejected. If the result is rejected, an alarm must be given. Otherwise, the map matching integrity is available.

The necessary input data for the integrity process is comprised of positioning data and satellite data from the GPS. This includes the following: the longitude and latitude coordinates of the vehicle position; the Doppler speed data; vehicle heading; the number of satellites; satellites' coordinates (X, Y and Z). Also inputted into the algorithm, is the wheel speed information from the in-vehicle sensor network used in the second phase. The HAL, probability of a false alarm rate (PFA), missed detection (PMD), and the minimum speed threshold are inputted into the algorithm as constant variables. The following sections outline each phase in detail.

4.2.1 Positioning integrity phase

The first phase in the proposed integrity algorithm is to examine the consistency of the positioning data. RAIM is one of the robust techniques that ensures the integrity of GPS data [97]. In this phase, a RAIM method provided by [97] is used to add a layer of integrity and verify the quality of the GPS positioning data.

The process of monitoring the integrity of the positioning data is divided into two steps (see Figure 4-2). The first is the computation of the HPL from the received positioning data in order to check RAIM availability. The second is the error detection process, based on using a weighted least squares approach [97]. The details of these two steps are set out below:

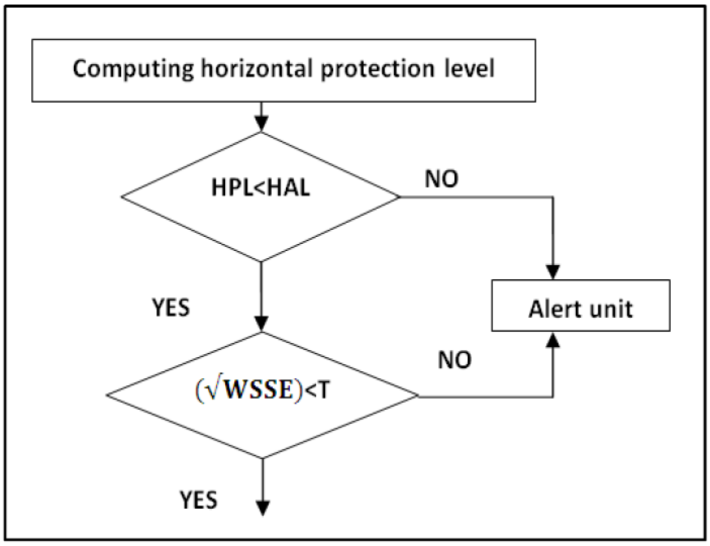

Figure 4-2: Positioning integrity phase.

Step 1: Checking RAIM availability

RAIM availability is checked using a horizontal protection level (HPL) parameter, the vertical protection level (VPL) is not required here as vehicles travel on the horizontal plane. The HLP parameter is calculated using the following formula, as shown in [97]:

$$\mathrm{HPL} = \max[\mathrm{Hslope}]\mathrm{T}(\mathrm{N}, P_{FA}) + \mathrm{k}(P_{MD})\mathrm{HDOP} \qquad (4.1)$$

Where HDOP is the horizontal dilution of precision and T is the threshold value[4] that is selected based on the number of satellites (N) and the probability of a false alarm rate (P_{FA}). k(P_{MD}) is the number of standard deviations for the missed detection probability (P_{MD}) and Hslope is the maximum allowable horizontal slope.

The values for (P_{FA}) and (P_{MD}) are chosen as 0.001 and 0.00001, as this is recommended by [5]. The maximum allowable horizontal slope is calculated, as shown in [97]:

$$\max[\mathrm{Hslope}] = maxumim \ \frac{\sqrt{[(GWG)^{-1}G^{T}\,W]_{1i}^{2}+[(GWG)^{-1}G^{T}\,W]_{2i}^{2}}}{\sqrt{I}-\,G(GWG)^{-1}G^{T}\,W_{ii}} \quad (4.2)$$

Where *G* is an observation matrix; *W* is the weight matrix; and *I* is the identity matrix.

After calculating the value of HPL, it is then compared to the maximum horizontal alert limit (HAL), in order to ensure whether or not the RAIM is available. The value for HAL is selected as 15 metres (as suggested by [4]), in order to support the majority of ITS services.

Step 2: Error detection

In this step, a weighted least squares approach [97] is applied to determine potential errors in the positioning data. A test statistic is compared with a selected threshold (T), in order to detect any errors. The test statistic is the square root of the weighted sum of the square errors (WSSE) [97]. The threshold (T) is selected based on the number of

[4] Further details concerning finding the values for T can be found in [97].

satellites and the probability of a false alarm rate. The calculation of the Weighted Sum of the Squared Errors (WSSE) is given as follows [97]:

$$WSSE = Y^T W[(I - P)Y] \quad (4.3)$$

$$P = G(G^T WG)^{-1} G^T W \quad (4.4)$$

Where G is the observation matrix; Y is the range of the residuals[5]; W is the weight matrix; and I is the identity matrix.

If the test statistic $(\sqrt{\mathrm{WSSE}})$ exceeds the threshold T, then the positioning solution is assumed to be unacceptable. Otherwise, the positioning solution is considered acceptable and the integrity process continues checking the speed integrity of the travelling vehicle. The following section provides a detailed explanation of the speed integrity phase.

4.2.2 Speed integrity phase

The second phase in the proposed integrity algorithm is to determine precisely the speed of the vehicle and ensure its integrity. One of the most accurate methods of estimating vehicle speed is using the Doppler Effect[6] [98]. In this phase, GPS Doppler speed measurements are used to provide an accurate estimation of the speed.

---

[5] Details of how the range of the residual is obtained can be found in [14].

[6] Doppler Effect refers to the difference between the calculated frequency at the GPS receiver and the satellite carrier frequency.

For integrity purpose, two steps are carried out before measuring the speed of the vehicle from the GPS Doppler data (see Figure 4-3). The first step is to check the vehicle speed status from the wheel speed sensor and the second is the speed error detection process based on the use of the SDOP parameter. The details of these two steps are outlined below:

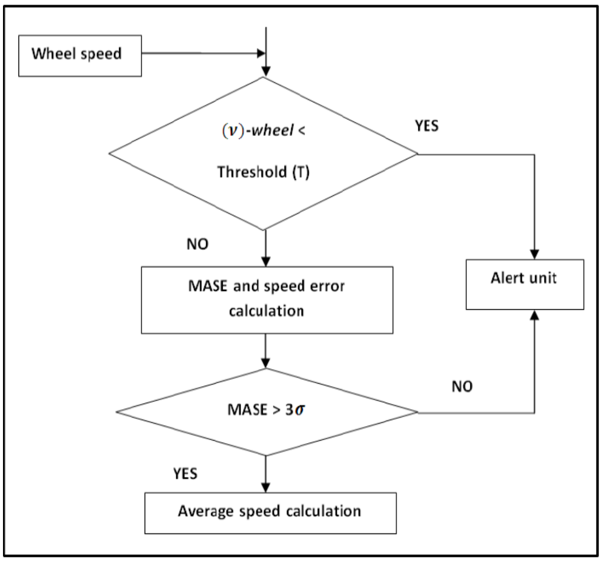

Figure 4-3: Speed integrity phase.

Step 1: Checking speed status

Despite the fact that the GPS can provide speed (based on the Doppler effect) with an accuracy of 2-5 cm/s and 4-10 cm/s on both the horizontal and the vertical axis, respectively [99], the magnitude of error in the estimated speed is inversely proportional to the actual speed of the vehicle [100]. As a result, GPS Doppler measurements are not reliable at low speeds, due to the significant increase in the error magnitude.

It is therefore vital to check the vehicle speed status before measuring the speed from the Doppler data. This step is carried out using wheel speed information, which can be accessed through the in-vehicle sensor network [101]. The wheel speed *(v)-wheel* is compared with a specified threshold (T). The threshold (T) is the minimum speed at which the Doppler data can be reliable. In this research, the value of T is chosen to be 2 m/s (as suggested by [96]).

If the (v)-*wheel* exceeds T, then the GPS Doppler data is assumed to be accurate, and can be used to estimate the real time vehicle speed. Otherwise, GPS Doppler data cannot be relied upon, and so is assumed to be untrustworthy. Additionally, GPS heading information is assumed to be inaccurate at low speeds, and so will not be used during the next phase [87].

Step 2: Speed error detection

Though GPS Doppler measurement can provide an accurate estimation of N second speed, each Doppler speed sample includes errors [98]. These include atmospheric and relativistic errors [102], their actual values cannot be established. In this case, the integrity of Doppler speed is difficult to check, as the actual value of the speed errors ((S_{nE}) of each speed sample) is not known. Therefore, the SDOP parameter is used in this step to estimate the maximum allowable speed error (MASE) for N Doppler speed measurements. The MASE can verify the integrity of Doppler speed, as is required to be three times larger than the standard error σ of the Doppler speed, determined using the central limit theorem [103]. This ensures a 99.9% confidence level when estimating the speed error [103]. The calculation of the average MASE for N second sample and the speed error as given in [103]:

$$MASE = \frac{\sum_{n=0}^{N} SDOP_n}{N} - \frac{SDOP_0 + SDOP_N}{2N} \quad (4.5)$$

$$speed\ error = \frac{MASE}{\sqrt{N}} \quad (4.6)$$

Where $SDOP_0$ and $SDOPN_N$ are the first and last SDOP values in the sampling interval, $SDOP_n$ is the SDOP value for each Doppler speed in the sampling interval, and N is the number of samples in the sampling interval.

After calculating the speed error, the integrity is verified by comparing the MASE value with the standard error σ for the N Doppler speed sample. If the MASE is less than three times the standard error σ, then the integrity of Doppler speed is assumed to be unenviable. Otherwise, it can be stated that the speed error for the N Doppler speed samples is calculated accurately with a 99.9% confidence level, and the true speed of the vehicle can be calculated precisely.

Here, vehicle speed is estimated using the average speed of N second GPS Doppler samples (rather than a single speed sample) in order to obtain higher accuracy [98]. The calculation of the actual average speed is given as follows [103]:

$$Average\ speed_{true} = Average\ speed_{Doppler} \pm speed\ error \quad (4.7)$$

$$Average\ speed_{Doppler} = \frac{\sum_{n=0}^{N} S_{nD}}{N} - \frac{S_{0D}+S_{ND}}{2N} \quad (4.8)$$

Where S_{0D} and S_{ND} are the first and last Doppler speed samples in the sampling interval, S_{nD} is the measurement of the Doppler speed in the sampling interval and N is the number of samples in the sampling interval.

4.2.3 The map matching integrity phase

In the final phase of the proposed integrity algorithm, the process continues with the map matching process only if the integrity of both the positioning and the speed data are available. This is due to the fact that the accuracy of the map matching process depends on the quality of input data [12]. Thus, checking the integrity of positioning and speed data will improve the performance and integrity of the map matching process and leads to more accurate outcomes. A flow chart of this phase is given in Figure 4-4.

In this phase, topological map matching algorithm provided in [85] is carried out in order to integrate the positioning data with the road map, and determine the road segment and the link where the vehicle is travelling. The fuzzy inference system (FIS) is then used to reason about the uncertainty related to the map matching results (see Chapter 2) and identify its integrity level, in order to decide whether to trigger an alert to the driver.

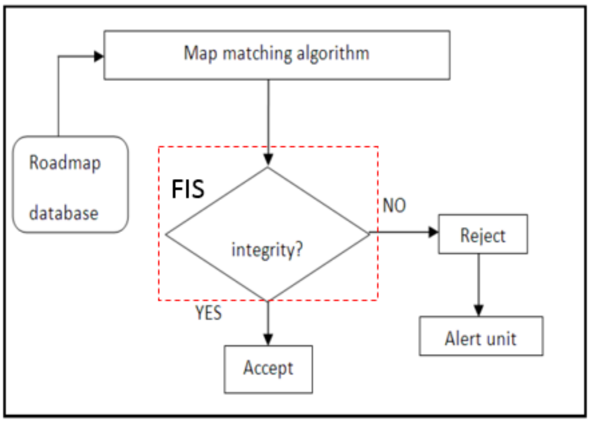

Figure 4-4: Map matching integrity phase.

The FIS used in this phase is based on three factors: distance, speed, and heading of the travelling vehicle, both before and after the map matching process. The calculation of these factors is given as follows:

Distance: The distance (computed using the accumulative distance between GPS points) always overestimates the real travelling distance, due to the fact that the line between all points is not smooth (zigzagging) [98]. To estimate the integrity of map matching process from the distance, the accumulative distance of the N positioning points should be compared with the accumulative distance of the corresponding N spatial positions on the road segment (after the map matching process). In this case, the errors related to the positioning data including errors from GPS receiver and digital road map should be considered. As illustrated in Figure 4-5, each positioning point has an error due to the error in the GPS receiver and the digital road map.

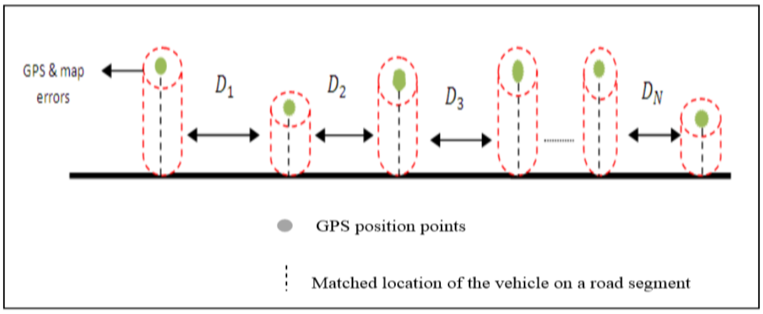

Figure 4-5: The accumulated distance.

The accumulative distance is expressed as follows:

$$D_{GPS} = \sum_{n=1}^{N} d_{GPS} \quad (4.9)$$

$$D_{Map} = \sum_{n=1}^{N} d_{Map} - \sum_{n=1}^{N} \sigma_{GPSn+\sigma_{MAPn}} \quad (4.10)$$

$$\Delta D = \left| D_{Map} - D_{GPS} \right| \quad (4.11)$$

Where D_{GPS} and D_{Map} are the accumulative distance before, and after, the map matching process, respectively. The distance between two GPS points and the matched points on the road segment are d_{GPS} andd_{Map}, respectively. σ_{GPSn} is the error related to GPS for each point, and $\sigma_{MAPn}$is the error related to the digital road map for each point (considered to be 3% [104]), and N is the number of points in the sampling interval.

If the absolute value of the difference between (D_{Map} - D_{GPS}) is equal to, or approximately, zero, then the vehicle location on the map is considered to be identified correctly by the map matching process, and vice versa. The value of ΔD is needed in fuzzy phase to overcome the problem of zigzagging. Additionally, the value ofΔD, once obtained and checked, the distance after map matching can be used with confidence in speed calculation.

Speed: The speed calculated from the accumulated distance of N matched locations on the map should be equal to, or close to, the corresponding average Doppler speed, as computed in the previous phase. The computation of speed and distance are expressed as follows:

$$Speed_{Map} = Accumulated\ distance_{Map}/time \quad (4.12)$$

$$\Delta S = \left|Speed_{\ Map} - Average\ Speed_{Doppler}\right| \quad (4.13)$$

If the absolute value of the difference between $(Average\ speed_{\ Doppler} - speed_{\ Map})$ is equal to, or approximately, zero, then the result of the map matching process is assumed to be accurate, and vice versa. The value of ΔS is needed in fuzzy phase to verify the speed integrity.

Heading: The direction of the matched road segment and the travelling vehicle should be the same. Here, the travelling vehicle direction (azimuth angle) obtained from the GPS is compared to the direction of the matched road segment. If the absolute value of the difference between ($Heading_{Map}$ - $Heading_{GPS}$) is less than the threshold, then it can be said that the selection of the road segment by the map matching process is more consistent.

type fuzzy inference sys-demonstrates the Sugeno 1-Table 4tem (FIS) applied in this phase. The input factors are: (i) the difference in distance (ΔD), (ii) the difference in speed (ΔS), and (iii) the difference in heading (ΔA). The fuzzy subsets for ΔA are good and bad, and for ΔS and ΔD are low, average and high. The output of the fuzzy system is the integrity level, which refers to the confidence level of the map matching process. The fuzzy subsets for the integrity level are: very high, high, average, low, and very low. The formulations of the fuzzy rules are based on the state of each factor. Further explanations concerning the way that the values of each factor and the output values of the integrity level are identified, are provided in Chapter 5.

R1: If ΔA is good and ΔD is low and ΔS is low then integrity scale is very high
R2: If ΔA is good and ΔD is low and ΔS is average then integrity scale is very high
R3: If ΔA is good and ΔD is average and ΔS is low then integrity scale is very high
R4: If ΔA is good and ΔD is low and ΔS is high then integrity scale is high
R5: If ΔA is good and ΔD is average and ΔS is average then integrity scale is high
R6: If ΔA is good and ΔD is high and ΔS is low then integrity scale is high
R7: If ΔA is good and ΔD is average and ΔS is high then integrity scale is average
R8: If ΔA is good and ΔD is high and ΔS is average then integrity scale is average
R9: If ΔA is good and ΔD is high and ΔS is high then integrity scale is low
R10: If ΔA is bad and ΔD is low and ΔS is low then integrity scale is low
R11: If ΔA is bad and ΔD is low and ΔS is average then integrity scale is low
R12: If ΔA is bad and ΔD is average and ΔS is low then integrity scale is low
R13: If ΔA is bad and ΔD is low and ΔS is high then integrity scale is low
R14: If ΔA is bad and ΔD is average and ΔS is average then integrity scale is low
R15: If ΔA is bad and ΔD is high and ΔS is low then integrity scale is low
R16: If ΔA is bad and ΔD is average and ΔS is high then integrity scale is low
R17: If ΔA is bad and ΔD is high and ΔS is average then integrity scale is low
R18: If ΔA is bad and ΔD is high and ΔS is high then integrity scale is low

Table 4-1: The Fuzzy inference system (FIS).

4.3 Summary

In this chapter, an integrity algorithm for monitoring the land vehicle navigation systems was developed. The aim was to ensure that the navigation information is consistent and triggers alerts to the user when the information is not trustworthy. It described the ways in which the proposed algorithm is able to reason out the integrity of the in-vehicle navigation system by considering the errors related to positioning data, speed data and the map matching process.

Three phases of integrity checks were developed in the algorithm. A Receiver Autonomous Integrity Monitoring (RAIM) algorithm was used to measure the quality of the GPS positioning data. The GPS Doppler data was used to check the integrity of the vehicle speed, which added a new layer of integrity. In addition, it improved the performance of the map matching process, because the integrity of all inputs is checked. The final phase in the integrity algorithm was intended to verify the integrity of the map matching algorithm. Here, a fuzzy inference system was applied to ensure the validity and integrity of map matching results. In the following chapter, the experiments to validate the performance of the proposed integrity algorithm and the derivation of the integrity level will be examined. In addition, there will be an overall evaluation of the results.

Chapter 5

Analysis and Discussion

Objectives

- Show the collected data and the inference of the integrity level.
- Present the validity and the performance of the proposed integrity algorithm.
- Present and discuss the result.

5.1 Introduction

Integrity has been discussed in many studies. They cover three areas; integrity of raw positioning data [16], integrity of map matching [18], and both [14]. This research checks the integrity of raw positioning data and map matching with a new element added to improve the integrity of the positioning information by checking speed integrity. However, this factor has a huge impact on the correctness and integrity of raw positioning data, before and during integrity validation using the map matching process.

This chapter provides a pilot study, analysis of the collected data, algorithms, and system output. It also discusses the collected data tracks, operational environments, and additional data collection considerations. The system performance was examined in different environments and validated using a true reference GPS vehicle.

A well-recognised evaluation schema was used to evaluate system performance. Several factors in particular were evaluated, including speed integrity. The results obtained were good compared to related work [10, 14, 17, 18]. The results obtained from the developed integrity algorithms, and their performance are discussed. Moreover, the chapter discusses integrity threshold scale, the fuzzy inference systems and its parameters and their weight.

5.2 System analysis

5.2.1 Pilot study

Before performing the main data collocation procedure, a pilot study was carried out to test each component of the algorithm to ensure its functionalities work properly, and to establish the environmental requirements when collecting data, such as environment type (urban, rural), weather, time. Finally, the pilot results could be used to derive and examine the weights of fuzzy rules, delivering results that are more accurate.

Pilot data was collected in Leicester City, across 1146 positioning points and a distance of 4.2 kilometres. The weight of the fuzzy rules was derived based on this data, and then the performance of the system was tested.

The preliminary results indicated that validating the performance of the proposed algorithm required the collection of positioning data that is sufficiently precise to test each component of the algorithm thoroughly. The data collection environment should vary between urban, rural, and mix areas. Thus, it is important to thoroughly test GPS signal availability at the RAIM level. Vehicle speed is also a vital input for both the algorithm and the fuzzy system. The experiment also requires data to be collected using two different GPS receivers, as each receiver can then act as a true reference for the other.

5.2.2 Main data collection

- **Required data:** For the testing of the proposed integrity framework, the required data is:

1. **Satellite data:** This data includes satellite number or ID, satellite coordinates (X, Y, and Z in Earth-Centred Earth-Fixed (ECEF) coordinate system), receiver clock bias, satellite elevation and azimuth angles [4].
2. **User data:** This data relates to the user X, Y, and Z position in ECEF coordinate system [4].
3. **Vehicle speed:** obtained from the wheel speed sensor.
4. **Doppler spee**d: obtained from the GPS receiver.
5. **Map data:** This data represents the road network in terms of links. For each link, there is a link ID, a start node, and an end node.

- **Equipment**

A GPS receiver with certain capabilities is essential for collecting the data; a GT-31 GPS receiver from LOCOSYS[7] was used to accomplish this task. The GT-31 is equipped with highly advanced navigation technologies and has been adopted by major GPS manufacturers including TomTom, Garmin, and Magellan. GT-31 supports [22]:

1. Doppler speed (HDOP, SDOP, etc.).
2. SiRF Star III low power chip that decodes GPS signals at a very low signal level (-160dB).

The GT-31 alone is not sufficient to produce an accurate evaluation of the proposed integrity algorithm. Hence, there is a need for another source of data that is more accurate than the GT-31 to act as a true reference. The true reference GPS receiver was obtained from the Geospatial Institute at the University of Nottingham. It is a highly

[7] See http://www.locosystech.com/

advanced carrier-phase, single frequency high sensitivity GPS receiver, integrated with a high-grade Inertial Navigation System (INS). In addition, a gyroscope and integrated carrier-phase INS were used. Aponte et al. [105] found the accuracy of the integrated carrier-phase GPS and INS is more than 5 centimetres for 97.5% of the time, in X, Y and Z coordinates.

A special data collection vehicle, containing all the aforementioned true reference GPS devices, was hired from the University of Nottingham (see Figure 5-1 and Figure 5-2). Quddus [18] and Velaga [14] have also used true reference vehicle to validate their work.

Figure 5-1: NGI test vehicle.

Figure 5-2: Equipment inside the test vehicle [14].

- **Experiment Setup**

The GT-31 was mounted in a pre-determined location inside the true reference vehicle. Data collection took place during the morning over a period of approximately two hours, commencing at 10 am to avoid traffic congestion (rush hour). The weather was sunny and humidity low to ensure minimum atmospheric error. The experiment was carried out in the city of Nottingham and surrounding villages, in order to test the system in different operational environments. The total number of positioning points collected was 5,685, and the testing route covered was 80 km. Figure 5-3 shows the complete test route followed.

Figure 5-3: The full test route in and around Nottingham.

The journey was divided into three data sets, according to operational environment. Three types of environments were classified, as follows: (1) urban defined by high buildings, tunnels, narrow streets, and a maximum speed on average of 48km/h; (2) rural defined as less complex than urban areas, better satellite availability, and a higher maximum speed; and (3) mixed environment, combining both urban and rural features. Table 5-1 below details the tracks, environment, number of points, and length for each data set.

Data Set	Environment	Points	Length (km)
Data set 1	Urban	1,245	6.4
Data set 2	Rural	3,335	65
Data set 3	Mix	1,105	8.6

Table 5-1: Three different data sets collected in Nottingham.

The collection environment determined for the first data set was urban. This track was collected in an urban setting because its complexity was expected to obstruct GPS signals (e.g. buildings) and degrade the performance of the map matching process (e.g. intersections). Figure 5-4, provides an example of the complexity of the environment in Data set 1. The test route travelled is shown in Figure 5-5, and is 6.4 kilometres in length. The maximum number of available satellites on view was seven and the minimum was four (see Figure 5-10 (a)).

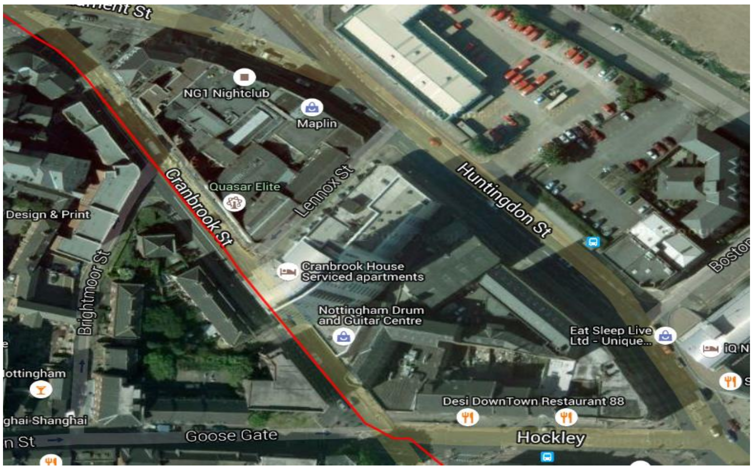

Figure 5-4: An example of the complexity in Data set 1.

Figure 5-5: The test route for the first data set.

The second set of data was collected in a rural environment, and consisted of 3,335 positioning points, covering 65 kilometres (see Figure 5-6). The maximum number of available satellites was 10, which is better than reported in data set 1 (see Figure 5-10 (b)). The track was selected to test the system's performance at high speeds and roundabouts. Figure 5-7 depicts the environment for Data set 2.

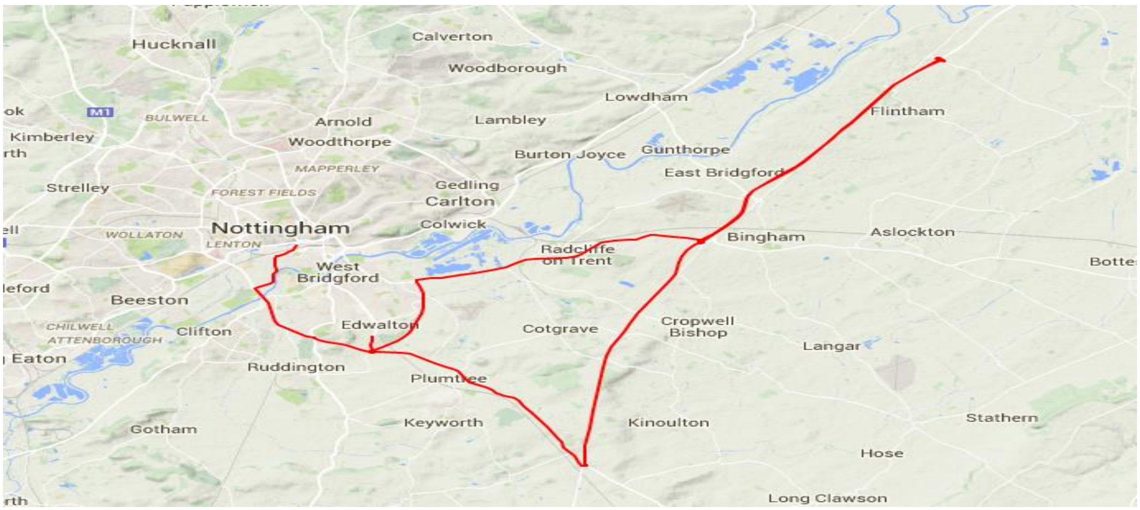

Figure 5-6: The test route for the second data set.

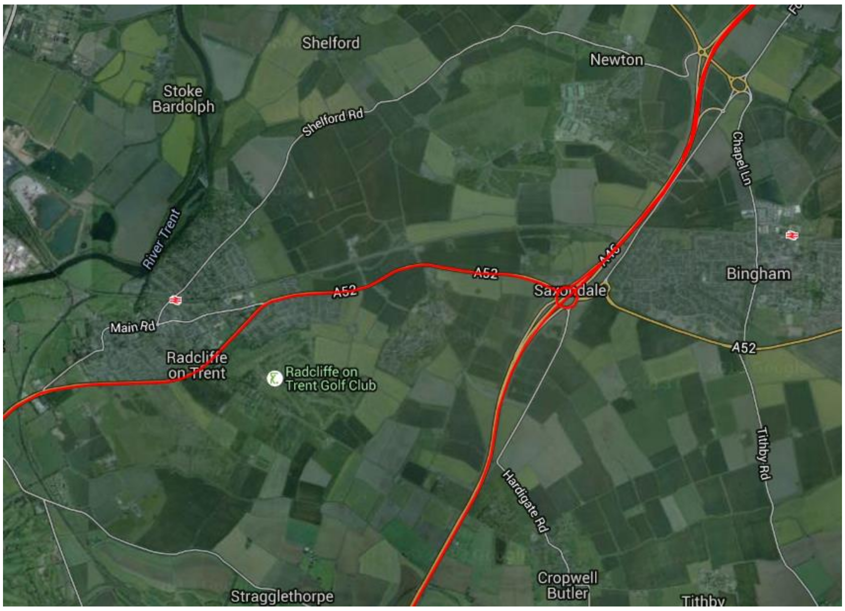

Figure 5-7: An example of the environment in Data set 2.

Figure 5-8 illustrates the test route for the third data set, which includes 1,105 positioning points, is 8.6 kilometres in length, and has a maximum number of nine available satellites (see Figure 5-10 (c)). The collection environment for this track combined urban and rural areas to cover the challenging features in these areas not considered in Data sets 1 and 2, such as roundabouts in urban areas (see Figure 5-9).

Figure 5-8: The test route for the third data set.

Figure 5-9: An example of the environment in Data set 3.

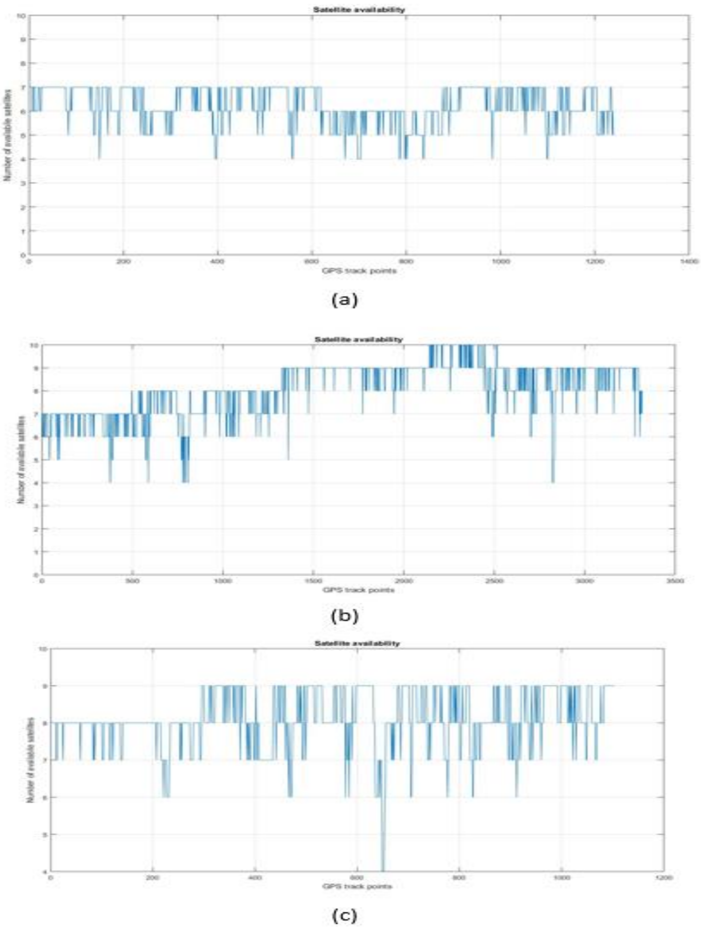

Figure 5-10: Satellite availability for data sets 1, 2 and 3, respectively.

5.2.3 Integrity level inference

There are some cases in which uncertainty remained after positioning data was validated against RAIM, Doppler and map matching process. These cases resulted from an error in the positioning data, the digital map, and the map matching process. Therefore, the best way to deal with such cases is to make use of fuzzy logic and develop rules to deal with all possible cases of uncertainties (see Chapter 4). Thus, an integrity scale is required here to reflect the confidence level related to each positioning data after being matched on the map. According to [18], the range of an integrity scale starts at “0” and ends at “100”. When the outcome of a fuzzy system equals “0”, this means that positioning data is un-trustable and has no integrity at any level in the system. The highest value the fuzzy system can indicate is “100”, and means the positioning data is wholly trustworthy. Based on this scale, the system makes it possible to alert the user when positioning data is lower than the integrity threshold

The fuzzy system developed follows the Sugeno-type fuzzy inference system. This has three input factors:

- **Δ Azimuth (Δ θ):** in terms of land navigation, an azimuth is the horizontal angle measured clockwise from a reference plan [85]. Based on observations in the collected data sets, a change in the azimuth angle can be used to indicate a vehicle’s heading.
- **Δ Distance (Δ *d*):** Refers to the variation in the difference in distance between two points before map matching and after map matching.

- **Δ Speed (Δ *S*):** Refers to the difference in vehicle speed from GPS Doppler speed and speed after the map matching process.

Considering these inputs and cases of uncertainties, the fuzzy system is comprised of 18 knowledge-based fuzzy rules, as listed in Table 5-3. The first factor in the fuzzy system is **Δ** θ. If the value for **Δ** θ is greater than 90 degrees, this indicates a change in the vehicle heading, and hence, not integrity. If it is less than 90 degrees, the system assumes the heading of the vehicle is unchanged. The second factor in the fuzzy system is **Δ** d. According to [98] the speed from the GPS receiver is exposed to a zigzag error. Therefore, speed validation requires distance validation. **Δ** d is the second factor to achieve validation, by checking the difference in distance before and after the map matching process. The closer the **Δ** d value to zero, the better the integrity. With this in mind, analysis of the positioning data sets shows that there are three clipping levels for **Δ** d; if **Δ** d is greater than 8, then the integrity based on **Δ** d is untreatable. If **Δ** d is less than or equal to 4, then the integrity based on **Δ** d is trustworthy, while if it is greater than 4 and less or equal to 8, then the integrity of **Δ** d is less trustworthy. Similarly, the integrity basis of **Δ** S was divided into three clipping levels, the first being the most trustworthy level for speed integrity, in which **Δ** S is less than or equal to 5. The second level is where **Δ** S is less trustworthy and arises when the **Δ** S is greater than 5 and less than or equal to 10. Finally, when **Δ** S is greater than 10, the integrity of a positioning data is considered untrustworthy.

Table 5-2 summarises the fuzzy factors and their weightings. Along with each factor there is a word to describe each value. For example, when the weight of Δd equals to 30, it is said to be low, which means trustworthy, while when it equals to 20 it is said to be average, which means less trustworthy. When the weight of Δd equals to 10, it said to be high, which means it is not trustworthy.

Factor	Total weight	Weight percentage distribution	
$\Delta \theta$	40	If $\Delta \theta < 90$	then w = 40 (good)
		If $\Delta \theta \geq 90$	then w = 0 (bad)
Δd	30	If $\Delta d \leq 4$	then w = 30 (low)
		If $\Delta d \leq 8$	then w = 20 (average)
		If $\Delta d > 8$	then w = 10 (high)
ΔS	30	If $\Delta S \leq 5$	then w = 30 (low)
		If $\Delta S \leq 10$	then w = 20 (average)
		If $\Delta S > 10$	then w = 10 (high)
Total	100		

Table 5-2: Fuzzy parameters and its weight.

Table 5-3 lists fuzzy conditions. The conditions cover all possible outcomes of the system. For each positioning point, delta heading, delta distance, and delta speed are calculated according to the given weight of each parameter. The total integrity weight for the positioning data is described by the summation of fuzzy parameters, and is used to raise an alarm if the value is under the integrity threshold. The rules were given in Chapter 4 and interpreted into mathematical conditions based on the weight of the fuzzy factor shown in Table 5-2. An example:

R1: If ΔA is good and ΔD is low and ΔS is low then integrity scale is very high

Becomes,

$$\text{If } \Delta\,\theta < 90 \text{ and If } \Delta\, d \leq 4 \text{ and If } \Delta\, S \leq 5$$

This rule works as follows: if the heading angle is less than 90°, and delta distance is less than or equal to 4, and the delta speed is less than or equal to 5, then the weights for heading, delta distance, and delta speed are 40, 30, and 30 respectively. Hence, the total weight becomes 100, denoting very high integrity. Similarly, the final rule would produce a total weight of 20 because the weight of delta angle, distance, and speed are 0, 10, and 10, respectively.

	Condition
R1	*If Δ θ < 90 and If Δ d ≤ 4 and If Δ S ≤ 5*
R2	*If Δ θ < 90 and If Δ d ≤ 4 and If Δ S ≤ 10*
R3	*If Δ θ < 90 and If Δ d ≤ 8 and If Δ S ≤ 5*
R4	*If Δ θ < 90 and If Δ d ≤ 4 and If Δ S > 10*
R5	*If Δ θ < 90 and If Δ d ≤ 8 and If Δ S ≤ 10*
R6	*If Δ θ < 90 and If Δ d >8 and If Δ S ≤ 5*
R7	*If Δ θ < 90 and If Δ d ≤ 8 and If Δ S > 10*
R8	*If Δ θ < 90 and If Δ d >8 and If Δ S ≤ 10*
R9	*If Δ θ < 90 and If Δ d >8 and If Δ S > 10*
R10	*If Δ θ ≥ 90 and If Δ d ≤ 4 and If Δ S ≤ 5*
R11	*If Δ θ ≥ 90 and If Δ d ≤ 4 and If Δ S ≤ 10*
R12	*If Δ θ ≥ 90 and If Δ d ≤ 8 and If Δ S ≤ 5*
R13	*If Δ θ ≥ 90 and If Δ d ≤ 4 and If Δ S > 10*
R14	*If Δ θ ≥ 90 and If Δ d ≤ 8 and If Δ S ≤ 10*
R15	*If Δ θ ≥ 90 and If Δ d >8 and If Δ S ≤ 5*
R16	*If Δ θ ≥ 90 and If Δ d ≤ 8 and If Δ S > 10*
R17	*If Δ θ ≥ 90 and If Δ d >8 and If Δ S ≤ 10*
R18	*If Δ θ ≥ 90 and If Δ d >8 and If Δ S > 10*

Table 5-3: Fuzzy system rules.

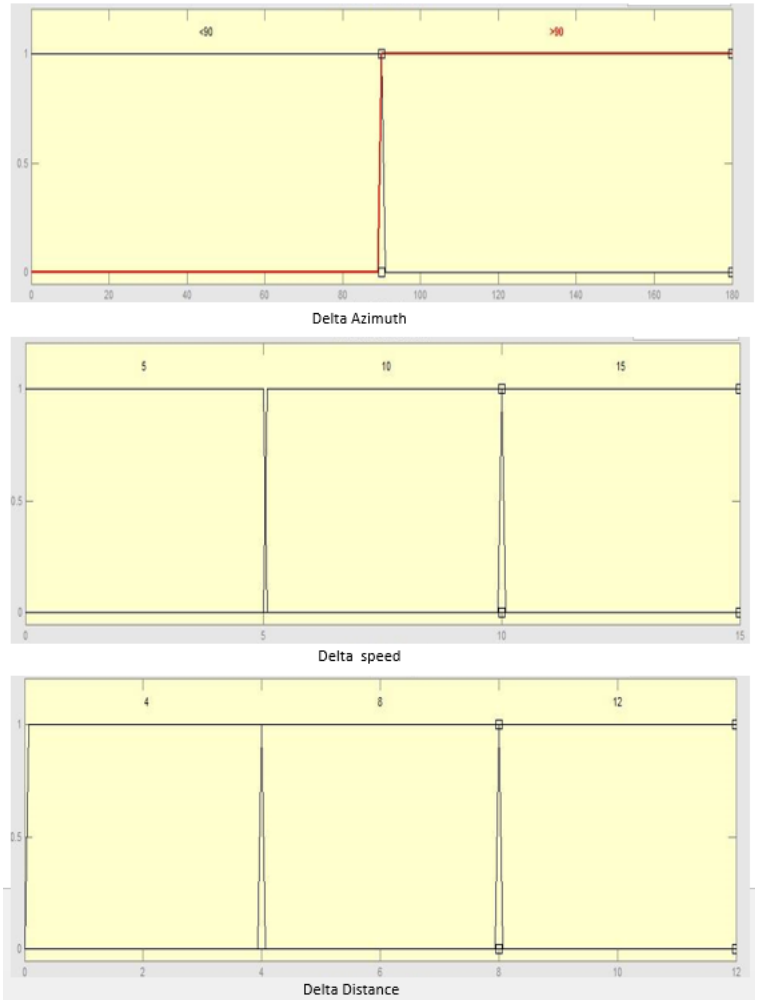

Figure 5-11: Fuzzy membership functions.

Figure 5-11 depicts the membership groups for fuzzy parameters[8]. There are three groups for integrity membership in the system: (1) high, (2) average, and (3) low. Each group maps the same integrity value to a truth value in the 0 to 1 range. The more closely the parameter value approximates 1, the higher the confidence that this value is trustworthy.

---

[8] The Matlab fuzzy toolbox was used in the fuzzy system. The scale of the digital map is 1:150,000, and was obtained from EdiNA.

For example, the delta azimuth belongs to two groups: high and low. The truth value upon which membership of the delta azimuth can be determined is 90º. If it is less than 90º, this means the delta azimuth is very close to 1; hence, it is a member of a high integrity group. If it is greater than or equal to 90º, then the value of the delta azimuth is very close to 0, which means that it is a member of a low integrity group.

Delta distance belongs to three membership groups: high, average, and low. If the value of the delta distance is less than or equal to 4, then delta distance is a member of a high integrity group and is as close as possible to 1 (i.e. integrity level). If the value of the delta distance is greater than 4, and less than or equal to 8, then delta distance is a member of the average integrity group. It is important to note here that the closer the value of the delta distance to 4, the closer it is to the high integrity group. The last case relating to delta distance is when delta is greater than 8. It is then said to be a member of the low integrity group.

Similarly, delta speed belongs to three membership groups: high, average, and low. If the value for delta speed is less than or equal to 5, then it is considered a member of a high integrity group and is associated with the integrity level, which is 1. If the value of the delta speed is greater than 5, and less than or equal to 10, then delta speed is a member of the average integrity group. It is important to note here that as the value of the delta speed moves closer to 8, it also moves towards low integrity. The last case affecting the delta speed is when it is greater than 10. It is then considered a member of the low integrity group.

5.3 Evaluation

The criteria that is widely used to evaluate the performance of the integrity algorithm is the Overall Correct Detection Rate (OCDR) [14, 17, 18, 106, 107]. The OCDR refers to the accuracy of an integrity system in providing correct alerts to users. Thus, there is a percentage for incorrect alerts. Incorrect alerts are of two types:

- **False Alarms (FA):** referring to the number of incorrect alerts that the system triggered, despite no error in the positioning points affecting their integrity.
- **Missed Detections (MD):** refers to number of the errors in the positioning output not identified by the integrity system.

Therefore, the performance of a proposed integrity algorithm can be measured with respect to False Alarm Rate (FAR), Missed Detection Rate (MDR), and Overall Correct Detection Rate (OCDR). As suggested by [14, 17, 18, 106, 107] the OCDR equation given as:

$$\mathrm{OCDR} = 1 - (\ \mathrm{FAR} + \mathrm{MAR}) \qquad (5.1)$$

$$\mathrm{FAR} = \frac{\mathrm{f}}{\mathrm{o}} \qquad (5.2)$$

$$MDR = \frac{m}{o} \qquad (5.3)$$

Where, o is the total number of observations, f is the total number of false alarms, and m is the total number of MDs.

The total number of false alarms (*f*) and the number of MDs (*m*), require positioning information from a highly accurate source, in the form of a true reference. The true reference is a highly accurate GPS carrier-phase observation integrated in high-grade INS devices to record positioning data.

When preparing an alert to indicate integrity, the threshold value must be identified. If a positioning point integrity value is less than this threshold, the point is considered to be not integrity, and an alert should be raised. In order to obtain that value, the positioning points in the collected data was used to observe changes in FA and MD. Figure 5-12 shows that while the MD changes slightly, the FA increases. Therefore, the best value to use as a threshold is the point of intersection between FA and MD. It can be seen that the intersection point is equal to 71. Therefore, in the case of any positioning point if the integrity scale was less than 71, an alert would be raised.

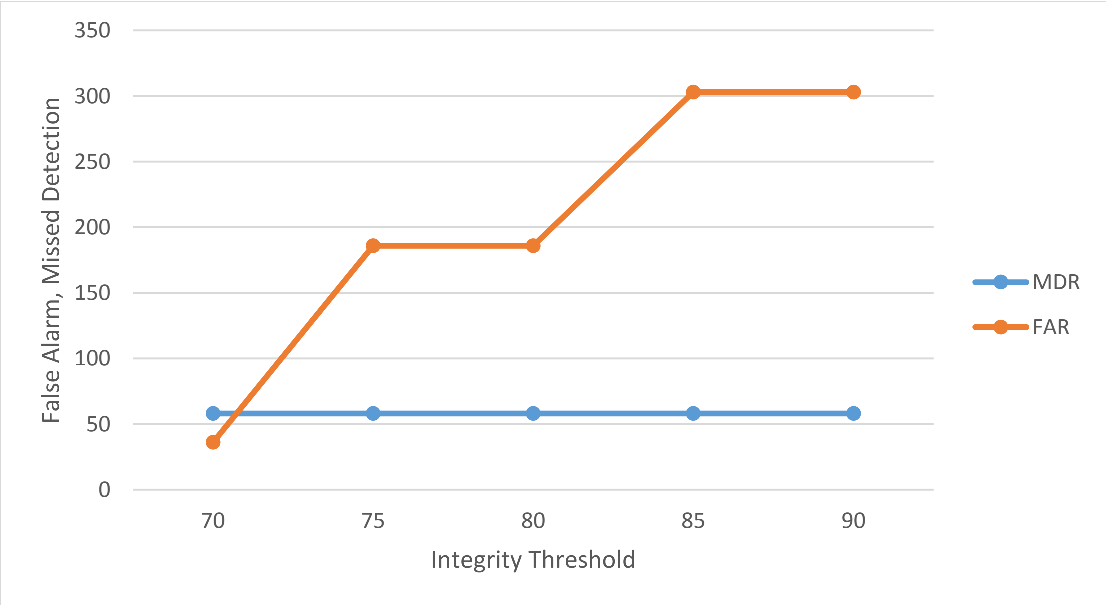

Figure 5-12:Integrity threshold.

5.4 Results

After application of the fuzzy rules, the data analysis showed the total number of false alarms and MDs varied from one track to another. Table 5-4 lists the MDs and false alarms reported in each data set. In the first track, there were 14 false alerts and 8 MDs. While in the second track, where the number of positioning points was much greater, the number of false alarms was 10, and the number of MDs was 16. In the final track, the number of false alarms was 11, and the number of MDs was 19.

Data Set	Misdetection	False alarm
Data set 1	8	14
Data set 2	16	10
Data set 3	19	11

Table 5-4: Number of MD points, and false alarms raised by the system.

The OCDR for each data set was 0.9824 for data set 1, 0.9923 for data set 2, and 0.9729 for data set 3. Table 5-5 shows the FAR, MDR, and OCDR for each data set. The OCDR indicates the accuracy of the developed integrity algorithm, and reflects the rate at which the system gave correct warnings.

Data Set	Number of Points	3-Levels Check (PASSED)	Environment	FAR	MDR	OCDR (%)
Data set 1	1,245	913 points	Urban	0.0112	0.0064	98.24%
Data set 2	3,335	2,994 points	Rural	0.0029	0.0048	99.23%
Data set 3	1,105	809 points	Mix	0.0099	0.0172	97.29%

Table 5-5: Overall Correct Detection Rate (OCDR) of each data set.

In Table 5-5, the third column shows the total number of points passing the three integrity checks. Without the integrity algorithm, it would be impossible to detect a fault in the data. Thus, the faulty data will be considered an MD. As shown in Table 5-6, 332, 341 and 296 points from data set 1, 2 and 3, respectively, are MDs, where no false alarm can be detect. The OCDR for each data set was found to be .7333 for data set 1, .9878 for data set 2 and .7321 for data set 3. This result is less accurate than the result obtained after applying the integrity algorithm. Hence, the proposed integrity algorithm can support location-based ITS service.

Data Set	MDs	MDR	OCDR without the integrity algorithm (%)
Data set 1	332	0.27	73.33%
Data set 2	341	0.10	89.78%
Data set 3	296	0.27	73.21%

Figure 5-6: Overall Correct Detection Rate (OCDR) of the data set without using the integrity algorithm.

5.4.1 Effects of operational environment on system performance

Junctions and tall buildings adversely affected the performance of the system in data set 1. In Figure 5-13 below, the green arrow indicates the direction of the vehicle and the black circle indicates the missed detected points. The system did not detect these errors, as the street is surrounded by tall buildings and is close to a junction.

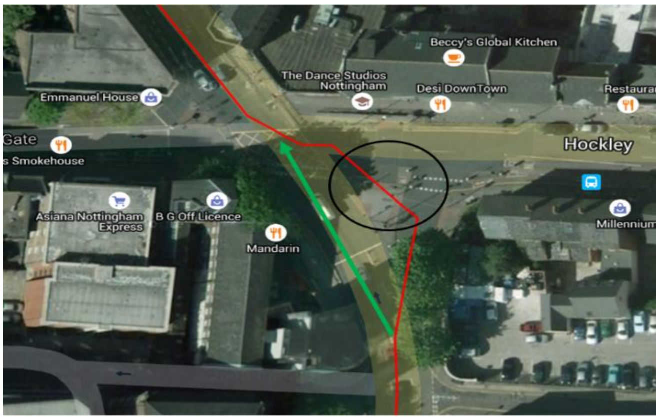

Figure 5-13: Error in Data set 1, green arrow is vehicle direction and black circle is the system error.

In rural environments (Data set 2) the system's performance was also degraded at roundabouts. As shown in Figure 5-14, the system identified the location of the vehicle incorrectly (black circle) without raising any alert; hence, this is considered an MD.

Figure 5-14: Error in Data set 2, green arrow is the vehicle direction and black circle is the system error.

In data set 3, the system's performance was less accurate at parallel roads. As shown in Figure 5-15, the system identified the location of the vehicle on the wrong road (black circle) and the vehicle was travelling on the opposite road; hence, this is considered an MD.

Figure 5-15: Error in Data set 3, green arrow is the vehicle direction and black circle is the system error.

5.5 Discussion

5.5.1 Key features of the speed integrity method

This research proposed, developed, implemented, and tested a new method to accurately check and validate the integrity of position points. This comprised three levels of integrity checks. Each level uses a different technique to check the consistency of the GPS information. The levels are:

1. A receiver autonomous integrity monitoring (RAIM) algorithm to measure the quality of the GPS positioning output.
2. GPS Doppler information to check the integrity of the vehicle speed, which contributes a new layer of integrity and could improve the performance of the map matching process.
3. The final level in the integrity check requires confirmation of the integrity of the

map matching algorithm.

In addition, a fuzzy inference system was developed to manage situations where the integrity of a given positioning point is uncertain. The system depends on three additional factors:

1. Delta azimuth.
2. Delta speed.
3. Delta distance.

As previously reported, the integrity method, when combined with the fuzzy inference system, can be used to check and validate the integrity of a given positioning point with an accuracy exceeding 98.6% a rate that is more accurate than many of the alternative newly proposed systems discussed in the literature review, as shown in Table 5-7 below. With this accuracy in mind, it can be stated with confidence that the integrity algorithm developed in this research is better than the majority of existing integrity algorithms, in particular that developed by [14, 18]. Using the same evaluation technique they applied the Overall Correct Detection Rate (OCDR), but the integrity method developed in this research offers better accuracy. Other integrity algorithms developed by other authors (e.g. [10]), is less accurate and uses different performance evaluation techniques.

Author of integrity method	Performance
Quddus (2006) [18]	91.1 % valid warnings[9]
Jabbour et al. (2008) [17]	88.8 % valid integrity warnings
Velaga (2010) [14]	98.2% valid integrity warnings[10]
Integrity method developed in this research	**98.6% valid integrity warnings**

Table 5-7:Performance of existing integrity algorithms.

Hence, the key features of the integrity method developed in this research are:

- Accurate identification of the final positioning point by minimising the effect of errors related to the positioning data, speed, and map matching process.
- Reliance on the Doppler speed to accurately validate the integrity of each positioning point before and after map matching.
- Development of a fuzzy inference system to deal with uncertainties related to the integrity of matched points and to measure integrity level.
- Use of new weight parameters, e.g. speed and distance in the integrity validation measurement.

5.5.2 Fuzzy parameters' weights

Three fuzzy factors were used in the fuzzy inference system:

- Delta azimuth indicates vehicle heading, and was given 40% of the total weight. This weighting is the largest attributed to any factor in the system. This value

[9] Using a weight based topological MM algorithm.
[10] Using a weight based topological MM algorithm.

was chosen empirically after analysis of the variation in the azimuth angle between the two points in each data set. Variation in the azimuth angle was found to be strong enough to be relied upon to identify the vehicle heading.

- Delta distance as an indicator of distance integrity. Chalko [98] observes that inaccuracies in track points lead to zig-zagging. The aim is that the distance after map matching would be based on a smooth and straight line, which is always less than a zig-zag line. The distance between matched points after map matching over a given link is compared to the distance between those points joined with a zig-zag, as an indication of the amount of distortion caused by the zig-zag and its effect on speed. Delta distance was awarded 30% of the system's total weight. Once the distance integrity was checked and validated, it can be relied on to validate the speed.
- Delta speed indicates the speed integrity. Previous researchers including [14, 18] have used speed to calculate the distance in map matching processing, but no one has used speed in integrity checking. However, [18] mentioned speed as an essential factor to enhance the map matching algorithm. Indeed, speed integrity monitoring is vital. Moreover, Li et al. [6] states that failure in any factor in the map matching process leads to defects throughout the whole process, thus the integrity of the speed was checked in Doppler level and using delta speed in the fuzzy system. Delta speed was awarded 30% of the system total weight. This value was obtained empirically based on a comparison of vehicle speed obtained from GPS Doppler and after map matching process. The absolute value of subtracting vehicle speed obtained from GPS Doppler and the one obtained based on

calculation from map matching process represents delta speed. The closer this value is to zero, the better the integrity. It is important to note that the speed obtained from map matching is calculated based on distance.

5.5.3 Integrity scale threshold

The integrity scale threshold was set at 71. Any point with an integrity weighting below this threshold will cause the system to raise an alert for the user. This value is obtained using a method similar to that used by [14, 18], which is the intersection point in the variation between false alarms and missed detection points.

Author of the integrity method	**Threshold value**	**Number of points used to obtain a threshold value**
Quddus (2006) [18]	70	2,040
Velaga (2010) [14]	82	10,347
Threshold obtained in this research	71	5,685

Table 5-8: Comparison between the integrity threshold value in this research and other research.

Table 5-8 shows that the difference between the threshold in [18] and the threshold obtained in this research is small despite the number of points used to empirically obtain the values. Therefore, the number of points used to obtain the threshold value is not that significant, and it cannot be used to indicate whether a threshold is more accurate than another, including the threshold used in this research. The calculation method of threshold was similar in all three studies (see Table 5-8) and the data used for the threshold calculation was collected in and around Nottingham's urban and suburban areas. The only justification for the differences in threshold values relates to the fuzzy factors and the weights that are used when validating positioning point's integrity.

5.5.4 Reliability

The author benefitted from the generosity of the University of Nottingham's loan of a dedicated vehicle equipped with highly accurate GPS carrier-phase devices integrated with a high-grade INS device. These highly accurate positioning data and collection devices could be used as a true reference. Aponte et al. [105] found the accuracy of the integrated carrier-phase GPS and the INS is more than 5 centimetres for 97.5% of the time in X, Y and Z coordinate. Without the existence of a true reference it would not be possible to confirm the accuracy of the system output. As shown earlier (Section 5.4.1), in critical cases, where GPS signal is low and the operational environment is complex (e.g. junctions), the system could have some mismatching errors that would be impossible to discover without a true reference (see Figure 5-16). In addition, the true reference could be used to confirm the speed of the vehicle, since it included INS that can accurately calculate the acceleration.

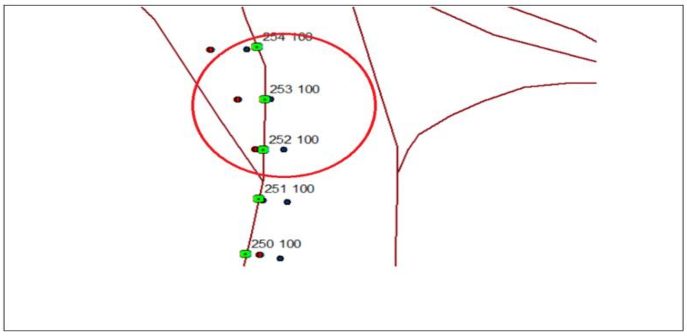

Figure 5-16: Miss detection errors in the system.

5.5.5 Location-based ITS applications

Intelligent transport systems (ITS) are widely used in transportation services such as those for emergency management, public transport management, commercial vehicle operations, and traffic control and vehicle safety systems. They can reduce the risk from accidents, control congestion, improve road safety and maintenance, and reduce adverse environmental impact.

Since location-based ITS services continuously require vehicle geographical positioning information in order to track a vehicle and perform their operations, the integrity algorithm developed in this research is of great value to such systems. It has the ability to check the integrity of positioning information and inform the user of any potential uncertainty in order to avoid misleading information (e.g. avoid confusing the driver in route guidance applications). In addition, it checks the integrity of vehicle's speed which is significantly important for critical applications such as pay as you speed.

5.6 Summary

This chapter provided detailed information about the experiment, data testing, and evaluation criteria. The data testing was conducted in Nottingham using special and advanced GPS equipment. 5,685 points were collected using the equipment and used as a true reference in evaluation. The overall correct detection rate (OCDR) was adopted as evaluation criteria. This criterion combines false alarms rate with a miss detection rate to produce accuracy rate in the integrity algorithm. Based on this, the OCDR of the first, second, and third data set was 98.24%, 99.23%, 97.29% respectively. A very high accuracy rate was obtained with an integrity threshold value equal to 71; a value obtained based on the analysis of false alarms and miss detection of all points.

A fuzzy logic system comprised of 18 conditions was used to contend with uncertainties in positioning points. The fuzzy logic system was constructed based on the azimuth angle, speed, and distance. A fuzzy system with three different factors was also implemented. The weight was divided over the fuzzy factors (i.e. heading 40%, speed 30%, and distance 30%). Given the integrity threshold (i.e. 71), any point with an integrity weight below this would raise an alert. The results suggest that the system can verify the integrity of positioning points before and after map matching with an accuracy of 98.6%.

Chapter 6

Conclusion and Future Work

Objectives

- Provide a summary of the work carried out in this research.
- Revisit the success criteria.
- Outline the limitations of this research.
- Provide direction for future work following this research.

6.1 Conclusion

Location-based ITS services require navigation system in order to perform its operations. The navigation system contains several errors either from GPS such as signal outages, receiver measurement errors, and multipath errors, or from map matching process; which can affect the quality of positioning data and may lead to inaccurate determination of a vehicle's location. It is therefore important to check and monitor the quality of the positioning information obtained from the GPS sensor and the other input data to the map matching process; in order to detect any misleading or faulty information and notify the user, which refers to as 'integrity'.

The main aim of this research was to improve the integrity of in-vehicle navigation systems by developing a robust and reliable GPS integrity monitoring algorithm based on the concept of context-awareness. Developing such a system that can be applied in real time ITS services could guarantee that navigation systems will not produce any misleading or incorrect information to the user.

To meet this aim, and after identifying limitations in the integrity algorithms outlined in existing literature, the system architecture for the proposed integrity monitoring algorithm was presented in Chapter 3. This architecture is able to monitor the integrity of in-vehicle navigation systems by capturing contextual information about the vehicle, including its position, speed and direction, and then determine their integrity. The architecture is composed of three subsystems: sensing, reasoning, and an applications subsystem, which is designed on the basis of context-awareness. The integrity monitoring algorithm was developed in Chapter 4. The algorithm takes into account errors related to positioning data, speed data, and the map matching process, using three phases of integrity checks, which consist of: (i) positioning integrity, (ii) speed integrity, and (iii) map matching integrity. In the first phase, a Receiver Autonomous Integrity Monitoring (RAIM) algorithm was used to measure the quality of the GPS positioning data. This was followed by a GPS Doppler data to check the integrity of vehicle speed, which added a further layer of integrity. In the final phase, a fuzzy inference system was employed, to ensure the validity and integrity of the map matching results.

The performance of the proposed integrity monitoring algorithm was tested and evaluated in Chapter 5, using real field data collected in Nottingham via a GT-31 GPS receiver. In the testing process, a single frequency high sensitivity GPS receiver integrates with a high-grade Inertial Navigation System (INS), and a gyroscope is used as a true reference, in order to validate the collected data. The evaluation result revealed that the proposed algorithm can accurately verify the integrity of positioning data both before and after the map matching process with 98.6% accuracy. The strengths of the algorithm arise from layered integrity checks that minimise the effect of errors related to positioning data, speed data, and the map matching process. The success criteria for the proposed integrity monitoring algorithm in ITS services will be revisited in the following section.

6.2 Success criteria revisited

As mentioned previously, the success of the work completed in this research was measured using criteria outlined in Chapter 1. The following section will discuss each of these criteria in turn, in order to judge the success of the research.

First, answers to the research questions outlined in Chapter 1 are presented below:

- **What type of information must be monitored in order to improve the integrity of in-vehicle navigation systems?**

  The main components of location-based ITS services, which are GPS, digital road map, and a map-matching algorithm, are liable to certain errors that might provide misleading or faulty information to the user. Therefore, information derived from these components is highlighted for monitoring, with the aim of improving the integrity of in-vehicle navigation systems. This information includes positioning data, and map matching result.

  In addition, speed data is considered as an essential factor for enhancing the map matching algorithm [18]. Thus, integrity of speed was monitored to improve the integrity process. A detailed explanation of this information is provided in Chapter 4.

- **How can the integrity monitoring system for location-based ITS services be designed, using the concept of context-awareness?**

  The design of a GPS integrity-monitoring system for ITS services is presented in Chapter 3. The architecture of the integrity monitoring system is designed based on the concept of context-awareness, in that it senses the in-vehicle navigation data, reasons about its integrity and reacts accordingly. The design is comprised of three main subsystems: sensing, reasoning, and an application subsystem. These subsystems correspond to the main phases of the context-aware system.

The abstract layered context-aware framework is used to construct the components of the proposed system.

- **How can the integrity monitoring algorithm be designed efficiently to determine the integrity of certain and uncertain navigation data?**

  A novel algorithm for monitoring the integrity of in-vehicle navigation systems is developed in Chapter 4. The proposed integrity algorithm has the ability to detect any inconsistency related to the positioning data, speed data, and map matching process, using three phases of integrity checks. These phases are: (i) positioning integrity, (ii) speed integrity, and (iii) map matching integrity. A fuzzy inference system is applied in the final phase to identify any uncertainty related to the map matching process; a detailed explanation of the proposed algorithm is provided in Chapter 4.

Second, an investigation illustrating how the proposed integrity monitoring algorithm is different from other existing integrity algorithm is required.

A review of the integrity algorithms exist in the literature, along with their limitations are provided in Chapter 2. In addition, the research contribution is given.

Third, an investigation showing how the proposed integrity monitoring algorithm can be applied in ITS services is needed.

Chapter 3 explains how an ITS service can be integrated with the proposed GPS integrity monitoring algorithm by utilising the concept of context-awareness.

The components of the proposed system are constructed using the layered context-aware framework, and the ITS services can be integrated via the application layer.

Fourth, an investigation illustrating how using fuzzy logic can positively affect the actual implementation of the algorithm.

Chapter 5 shows how the uncertainty related to the map matching process is resolved using a fuzzy inference system (FIS), which consists of three parameters: distance, speed and direction of the travelling vehicle, both before and after the map matching process. Each parameter is given a weighting based on an analysis of the collected data.

Finally, an investigation illustrating whether the integrity algorithm can be implemented in the real world, and thus be used commercially.

The proposed integrity algorithm was implemented and tested using data collected in a real environment. The results suggest that the integrity algorithm has the capability to support real world application, and thus can be used commercially.

6.3 Research limitations

As discussed in Chapter 5, the results of the proposed algorithm are more positive than the results of many existing integrity algorithms. Despite the fact that the proposed algorithm is able to provide valid integrity alerts to drivers 98.6 % of the time, the work achieved in this research does have some limitations, which are as follows:

- **Testing environment:** the performance evaluation of the proposed integrity algorithm is based on a small data set, collected in urban and rural areas in Nottingham. However, it is vital to validate the effectiveness of the algorithm in more complex environments, such as areas with high rise buildings, for example Hong Kong or New York, in order to increase the usability of the algorithm. Therefore, further investigations are required to test the performance of the system in such environments.

- **System continuity:** GPS signals can experience outage while driving on some roads, for example through tunnels. As a result, the system continuity will be affected. Many new navigation systems use INS to augment the GPS signal and predict vehicle positioning, in case of GPS outage. However, the proposed integrity algorithm uses only GPS sensor as the primary source of positioning data. Further research must be undertaken to address this issue.

6.4 Future work

The integrity of land vehicle navigation systems is an attractive research area in the field of intelligent transport systems (ITS). At present, the required integrity values are still under development, as more services evolve and new applications continue to emerge. However, high integrity navigation data is still yet to be achieved for most ITS services.

According to the work carried out in this research, further investigations are required in order to enhance the integrity monitoring algorithms used for in-vehicle navigation systems. These are as follows:

- To extend the integrity monitoring system by adding an inertial navigation system (INS) to the sensing layer, in order to provide required information, such as current position, speed, and direction of the vehicle, in case of GPS signal outage. This feature is anticipated to improve system continuity.

- To enhance the integrity algorithm by adding a new level to verify the integrity of the data provided by the INS sensor; further investigation is required to examine the validity of this layer.

- To enhance the transition process between GPS monitoring and INS monitoring algorithms to ensure that the transition takes place correctly, effectively, and smoothly.

References

1. Qureshi, K.N. and A.H. Abdullah, *A Survey on Intelligent Transportation Systems.* Middle-East Journal of Scientific Research, 2013. **15**(5): p. 629-642.

2. Taylor, G. and G. Blewitt, *Intelligent Positioning: GIS-GPS Unification*. 2006: Wiley Blackwell.

3. Quddus, M.A., W.Y. Ochieng, and R.B. Noland, *Intergrity of Map Matching Algorithms.* Transportation Research Part C, 2006. **14**(4): p. 283–302.

4. Velaga, N.R., et al., *Map-Aided Integrity Monitoring of a Land Vehicle Navigation System.* Intelligent Transportation Systems, 2012. **13**(2): p. 848 - 858.

5. Feng, S. and W.Y. Ochieng. *Integrity of navigation system for road transport*. in *In Proceedings of the 14th ITS World Congress*. 2007. Beijing.

6. Li, L., M. Quddus, and L. Zhao, *High accuracy tightly-coupled integrity monitoring algorithm for map-matching.* Transportation Research Part C: Emerging Technologies, 2013. **36**: p. 13–26.

7. *Understanding GPS: Principles and Applications*, ed. E. Kaplan and C. Hegarty. 2005: Artech House.

8. Bastiaansen, A. *The navigable digital street map is the critical success factor for vehicle navigation and transport information systems in Europe*. in *Proceedings of the 1996 IEEE Intelligent Vehicles Symposium*. 1996. Tokyo.

9. Bullock, J.B. and E.J. Krakiwsky. *Analysis of the use of digital road maps in vehicle navigation*. in *Position Location and Navigation Symposium, IEEE*. 1994. Las Vegas, NV

10. Yu, M., et al., *Improving Integrity and Reliability of Map Matching Techniques.* Journal of Global Positioning Systems, 2006. **5**: p. 40-46.

11. White, C.E., D. Bernstein, and A.L. Kornhauser, *Some map matching algorithms for personal navigation assistants.* Transportation Research Part C: Emerging Technologies, 2000. **8**: p. 91 - 108.

12. Chen, W., et al., *Effects of sensor errors on the performance of map matching.* Journal of Navigation,, 2005. **58**(2): p. 273 - 282.

13. Quddus, M., W. Ochieng, and R. Noland, *Current map-matching algorithms for transport applications: State-of-the art and future research directions.* Transportation Research Part C: Emerging Technologies, 2007. **15**(5): p. 312–328.

14. Velaga, N., *Development of a weight-based topological map-matching algorithm and an integrity method for location-based ITS services*. 2010: PhD thesis, Loughborough University.

15. Ochieng, W.Y. and K. Sauer, *Urban road transport navigation: performance of the global positioning system after selective availability.* Transportation Research Part C: Emerging Technologies, 2002. **10**(3): p. 171–187.

16. Andrés, C.D.S., *Integrity monitoring applied to the reception of GNSS signals in urban environments*. 2012: University De Toulouse, PhD Thesis.

17. Jabbour, M., P. Bonnifait, and V. Cherfaoui, *Map-Matching Integrity Using Multihypothesis Road-Tracking.* Intelligent Transportation Systems, 2008. **12**(4): p. 189-201.

18. Quddus, M.A., *High Integrity Map Matching Algorithms for Advanced Transport Telematics Applications*. 2006: PhD thesis, Imperial College London.

19. Hong, J.-y., E.-h. Suh, and S.-J. Kim, *Context-aware systems: A literature review and classification.* Expert Systems with Applications, 2009. **36**: p. 8509–8522.

20. Dey, A., G. Abowd, and D. Salber, *A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications.* Human-Computer Interaction, 2001. **16**(2): p. 97–166.

21. Iivari, J., *A paradigmatic analysis of contemporary schools of IS development.* European Journal of Information Systems 1991. **1**(4): p. 249–272.

22. Parra Alonso, I., et al., *Accurate Global Localization Using Visual Odometry and Digital Maps on Urban Environments.* Intelligent Transportation Systems, IEEE Transactions on, 2012. **13**(4): p. 1535-1545.

23. Mashrur, A.C. and A. Sadek, *Fundamentals of Intelligent Transportation System planning*. 2003, London: Artech House.

24. Council, N.R., *Development and deployment of standards for intelligent transportation systems: a review of the federal program*. 2004: Washington D.C.

25. Chadwick, D. *Projected navigation system requirements for intelligent vehicle highway systems (IVHS).* in *Proceedings of Institute of Navigation GPS-94*. 1994.

26. Arnott, R., A. De Palma, and R. Lindsey, *Does providing information to drivers reducetraffic congestion?* Transportation Research A, 1991. **25**(5): p. 309 - 318.

27. Liu, H., W. Recker, and A. Chen, *Uncovering the contribution of travel time reliability to dynamic route choice using real-time loop data.* Transportation Research A, 2004. **38**(6): p. 435-453.

28. Abdel-Aty, M., R. Kitamura, and P. Jovanis, *Investigating effect of travel time variabilityon route choice using repeated measurement stated preference data.* Transportation Research Record, 1995(1493): p. 39 - 45.

29. Sheridan, K., *Service requirements document (SRD) for vehicle performance and emissions monitoring system (VPEMS)*. 2001.

30. Haddow, G.D., J.A. Bullock, and D.P. Coppola, *Introduction to emergency management*. 2010, USA: Butterworth-Heinemann.

31. Walder, J. *Road User Charging*. in *Transportation Research Board 86th Annual Meeting*. 2007.

32. *Road user charging: Technical and Operational Development*. in *IEE Seminar* 2004.

33. Lu, S., T. He, and Z. Gao. *Electronic Toll Collection System Based on Global Positioning System Technology*. in *10th International Conference on Challenges in Environmental Science and Computer Engineering (CESCE)*. 2010. Wuhan, China.

34. Hammerle, M., M. Haynes, and S. McNeil, *Use of Automatic Vehicle Location and Passenger Count Data to Evaluate Bus Operations.* Journal of the Transportation Research Board, 2005. **1903**: p. 27-34.

35. Desaulniers, G. and M.D. Hickman, *Public Transit*, in *Handbook in OR & MS*, C. Barnhart and G. Laporte, Editors. 2007, Elsevier B.V. p. 69-127.

36. Yeh, J.H., et al., *Segment-based emotion recognition from continuous Mandarin Chinese speech.* Computers in Human Behavior, 2011. **27**(5): p. 1545-1552.

37. Markets, M.a., *Fleet Management Market (2013 – 2018)*. 2013.

38. Crainic, T.G., M. Gendreau, and J.Y. Potvin, *Intelligent freight-transportation systems: Assessment and the contribution of operations research.* Transportation Research Part C: Emerging Technologies, 2009. **17**(6): p. 541–557.

39. Ezell, S., *Explaining International IT Application Leadership: Intelligent Transportation Systems*. 2010.

40. An, S.-h., B.-H. Lee, and D.-R. Shin. *A Survey of Intelligent Transportation Systems*. in *3rd International Conference on Computational Intelligence, Communication Systems and Networks*. 2011.

41. Maccubbin, R., et al., *intelligent transportation systems Benefits, Costs, Deployment, and Lessons Learned: 2008 Update*. 2008: U.S.

42. Cassell, R. and A. Smith. *Development of Required Navigation Performance (RNP) for airport surface movement Guidance and Control*. in *Proceeding of Digital Avionics System Conference*. 1995. Cambridge, MA

43. Person, J. *Writing Your Own GPS Applications: Part 2*. 2008 [cited 2015 March]; Available from: http://www.developerfusion.com/.

44. Ochieng, W.Y., et al., *GPS integrity and potential impact on aviation safety*. The Journal of, 2003. **56**: p. 51-65.

45. DOT, *Radionavigation Systems Task Forc: A Report to the Secretary of Transportation*. 2004: USA.

46. Kibe, S.V., *Indian plan for Satellite-Based Navigation Systems for Civil Aviation* Current seience 2003. **84**(11): p. 1405-1411.

47. Hein, G.W., *From GPS and GLONASS via EGNOS to Galileo Positioning and navigation in the third millennium.* GPS Solutions, 2000. **3**(4): p. 39–47.

48. North, J.R., et al. *Technologies to measure indicators for variable road user charging*. in *11th World Conferece on Transport Research* 2007.

49. EGNOS. *What is ABAS?* ; Available from: http://egnos-portal.gsa.europa.eu/.

50. Džunda, M.J.M., *SATELLITE-BASED AUGMENTATION SYSTEMS.* 2013.

51. Schilit, B.N. and M.M. Theimer, *Disseminating active map information to mobile hosts.* IEEE Network, 1994. **8**(5): p. 22 - 32.

52. Brown, M.G. *Supporting User Mobility*. in *Proceedings of Technology, tools, applications, authentication and security IFIP World Conference on Mobile Communications*. 1996. Canberra.

53. Dey, A.K., G.D. Abowd, and A. Wood. *CyberDesk: a framework for providing self-integrating context-aware services*. in *Proceedings of the 3rd international conference on Intelligent user interfaces*. 1998. New York.

54. Ryan, N., J. Pascoe, and D. Morse, *Enhanced reality fieldwork: the context-aware archaeological assistant*, in *Computer Applications in Archaeology 1997.* 1998, Tempus Reparatum: Oxford. p. 182-196.

55. Hull, R., P. Neaves, and J. Bedford-Roberts. *Towards situated computing*. in *Proceedings of the First International Symposium on Wearable Computers*. 1997.

56. Salber, D. *Context-awareness and multimodality*. in *Proceedings of Colloque sur la multimodalité*. 2000. Grenoble.

57. Loke, S., *Context-Aware Pervasive Systems: Architectures for a New Breed of Applications*. 2006: Auerbach Publications.

58. Schilit, B., N. Adams, and R. Want. *Context-Aware Computing Applications*. in *Proceedings of the Workshop on Mobile Computing Systems and Applications*. 1994.

59. Dey, A.K. and G.D. Abowd, *Towards a Better Understanding of Context and Context-Awareness.* The 1st international symposium on Handheld and Ubiquitous Computing HUC, 1999: p. 304–307.

60. Baldauf, M., S. Dustdar, and F. Rosenberg, *A survey on context-aware systems.* International Journal of Ad Hoc and Ubiquitous Computing, 2007. **2**(4): p. 263-277.

61. Chen, H., *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*. 2004: PhD thesis, University of Maryland.

62. Hong, C.-S., et al., *Developing a Context-Aware System for Providing Intelligent Robot Services*, in *Smart Sensing and Context*, P. Havinga, et al., Editors. 2006, Springer Berlin Heidelberg. p. 174-189.

63. Strang, T. and C. LinnhoffPopien. *A Context Modeling Survey*. in *Proceedings of the First International Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp* 2004.

64. Truong, A., Y. Lee, and S. Lee. *Modeling and reasoning about uncertainty in context-aware systems*. in *Proceedings of the IEEE International Conference e-Business Engineering*. 2005. Beijing.

65. Bettini, C., et al., *A survey of context modelling and reasoning techniques.* Pervasive and Mobile Computing, 2010. **6**(2): p. 161–180.

66. Wu, H., *Sensor Data Fusion for Context-Aware Computing Using Dempster-Shafer Theory*. 2003: Phd thesis, Carnegie Mellon University.

67. Ranganathan, A., J. Al-Muhtadi, and R. Campbell, *Reasoning about uncertain contexts in pervasive computing environments.* IEEE Pervasive Computing, 2004. **3**(2): p. 62 –70.

68. Ji, Q., P. Lan, and C. Looney, *a probabilistic framework for modeling and real-time monitoring human fatigue.* IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans, 2006. **36**(5): p. 862 –875.

69. Kragt, M.E., *A beginners guide to Bayesian network modelling for integrated catchment management*. 2009.

70. Karimi, H.A., T. Conahan, and D. Roongpiboonsopit, *A Methodology for Predicting Performances of Map-Matching Algorithms.* Web and Wireless Geographical Information Systems 2006. **4295**: p. 202 – 213.

71. Konar, A., *Computational Intelligence: Principles, Techniques and Applications*. 2005, Berlin, heidleberg, New York: Springer.

72. Zhao, Y., *Vehicle Location and Navigation System*. 1997: Artech House.

73. Levinson, D., *Micro-foundations of congestion and pricing: A game theory perspective.* Transportation Research Part A: Policy and Practice, 2005. **39**(7-9): p. 691–704.

74. Syed, S. and M.E. Cannon. *Fuzzy Logic Based-Map Matching Algorithm for Vehicle Navigation System in Urban Canyons*. in *ION National Technical Meeting*. 2004.

75. Khedo, K. *Context-Aware Systems for Mobile and Ubiquitous Networks*. in *Proceedings of International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)*. 2006.

76. BROWN, P.J. *The stick-e document: a framework for creating context-aware applications*. in *Proceedings of EP''96*. 1995. Palo Alto.

77. Chen, G. and D. Kotz, *A Survey of Context-Aware Mobile Computing Research*. 2000: Hanover.

78. Hertz. *Hertz NeverLost*. Available from: http://www.neverlost.com/.

79. Lamming, M. and M. Flynn, *"Forget-me-not" Intimate Computing in Support of Human Memory*. 1994: CAMBRIDGE, ENGLAND.

80. Yang, D., B. Cai, and Y. Yuan. *An improved map-matching algorithm used in vehicle navigation system*. in *Proceedings of Intelligent Transportation Systems*. 2003

81. Joshi, R.R. *Novel Metrics for Map-Matching in In-Vehicle*. in *Intelligent Vehicle Symposium*. 2002.

82. Greenfeld, J. *Matching GPS observations to locations on a digital map*. in *proceedings of the 81st Annual Meeting of the Transportation Research*. 2002.

83. Bernstein, D. and A. Kornhauser. *An introduction to map-matching for personal navigation assistants*. 1996.

84. Wu, D., et al. *A Heuristic Map-Matching Algorithm by Using Vector-Based Recognition*. in *Proceeding of the International Multi-Conference on Computing in the Global Information Technology*. 2007.

85. Cossaboom, M., et al., *Augmented Kalman Filter and Map Matching for 3D RISS/GPS Integration for Land Vehicles.* International Journal of Navigation and Observation, 2012. **2012**(2012).

86. Velaga, N.R., M.A. Quddus, and A.L. Bristow, *Developing an Enhanced Weight Based Topological Map-Matching Algorithm for Intelligent Transport Systems.* Transportation Research Part C: Emerging Technologies, 2009. **17**(6): p. 672–683.

87. Ochieng, W.Y., M.A. Quddus, and R.B. Noland, *Map-matching in complex urban road networks.* Brazilian Journal of Cartography, 2004. **55**(2): p. 1-18.

88. Pink, O. and B. Hummel. *A statistical approach to map matching using road network geometry, topology and vehicular motion constraints*. in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems*. 2008. Beijing.

89. Quddus, M.A., et al., *A general map matching algorithm for transport telematics applications.* GPS Solutions 2003. **7**(3): p. 157-167

90. Honey, S.K., et al., *Vehicle navigational system and method,*. 1989: United States

91. Kim, S. and J. Kim, *Adaptive fuzzy-network based C-measure map matching algorithm for car navigation system.* IEEE Transactions on industrial electronics, 2001. **48**(2): p. 432-440.

92. Pyo, J., D. Shin, and T. Sung. *Development of a map matching method using the multiple hypothesis technique*. in *Proceedings on IEEE Intelligent Transportation Systems*. 2001.

93. Su, H., J. Chen, and J. Xu. *A Adaptive Map Matching Algorithm Based on Fuzzy- Neural-Network for Vehicle Navigation System*. in *proceedings of the 7th World Congress on Intelligent Control and Automation*. 2008.

94. Lahrmann, H., et al., *Pay as You Speed, ISA with incentives for not speeding: A case of test driver recruitment.* Accident Analysis & Prevention, 2012. **48**: p. 10–16.

95. Crow. *HDOP AND GPS HORIZONTAL POSITION ERRORS*. 2013 [cited 2015 May]; Available from: http://crowtracker.com/.

96. Bonnabel, S. and E. Salaun, *Design and prototyping of a low-cost vehicle localization system with guaranteed convergence properties.* Control Engineering Practice, 2011. **19**: p. 591–601.

97. Walter, T. and P. Enge. *Weighted RAIM for Precision Approach*. 1995.

98. Chalko, T.J., *High accuracy speed measurement using GPS (Global Positioning System).* NU Journal of Discovery, 2007.

99. Davidson, P., J. Hautamäki, and J. Collin. *USING LOW-COST MEMS 3D ACCELEROMETER AND ONE GYRO TO ASSIST GPS BASED CAR NAVIGATION SYSTEM*. in *15th Saint Petersburg International Conference on Integrated Navigation Systems*. 2008. Finland.

100. Jo, K., K. Chu, and M. Sunwoo, *Interacting Multiple Model Filter-Based Sensor Fusion of GPS With In-Vehicle Sensors for Real-Time Vehicle Positioning.* IEEE Transactions on Intelligent Transportation Systems, 2012. **13**(1): p. 329 - 343.

101. D'Orazio, L., F. Visintainer, and M. Darin. *Sensor networks on the car: State of the art and future challenges*. in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*. 2011. Grenoble.

102. Zhang, J., K. Zhang, and R. Grenfell. *On the relativistic Doppler Effects and high accuracy velocity determination using GPS*. in *Proceedings of The International Symposium on GNSS/GPS*. 2004.

103. Chalko, T.J., *Estimating Accuracy of GPS Doppler Speed Measurement using Speed Dilution of Precision (SDOP) parameter.* NU Journal of Discovery, 2009. **6**.

104. Lee, H., et al. *Ensemble of machine learning and acoustic segment model techniques for speech emotion and autism spectrum disorders recognition*. in *INTERSPEECH*. 2013.

105. Aponte, J., et al., *Assessing Network RTK Wireless Delivery.* The GPS world, 2009. **20**(2): p. 14-27.

106. Bonnifait, P., et al., *Multi-hypothesis Map-Matching using Particle Filtering*. 2009: University of Technology of Compiegne.

107. Toledo-Moreo, R., D. Betaille, and F. Peyret, *Lane-Level Integrity Provision for Navigation and Map Matching With GNSS, Dead Reckoning, and Enhanced Maps.* Intelligent Transportation Systems, 2010. **11**(1): p. 100 - 112.