



**Politecnico
di Torino**

Politecnico di Torino

Degree in Automotive Engineering

A.Y. 2020/2021

Graduation Session October 2021

**Collaborative Positioning in low
latency vehicular networks:
Cooperative Integrity Monitoring**

Advisors:

Carla Fabiana Chiasserini

Fabio Dovis

Daniel Aloï

Candidate:

Sara Golisciani

© Copyright by Sara Golisciani,2021
All rights reserved

To my family and boyfriend

ACKNOWLEDGMENTS

My first thanks go to my family, who gave me the chance to improve myself and grow up, supporting me in the choice to change universities and cities. It is only thanks to them that I am here now.

I would also like to thank professors Giovanni Belingardi and Maria Pia Cavatorta, the Polytechnic of Turin and Stellantis, who allow students who deserve it, to repay the efforts made in previous years. This year, Covid pandemics has not allowed us to live this experience at 100%, to grow looking to a new culture and a new lifestyle. Nevertheless, Covid has done far more damage than that, and so I am still grateful for being able to participate to this double degree project. Even behind a screen, it has transmitted me so much. With my laptop, I was able to take lessons held in a university located on the other side of the world, very different from those I was used to. This also means growing and improving.

I would also thank all my advisors: Carla Fabiana Chiasserini, Fabio Dovis and Daniel Aloï. I want to thank you for believing in my abilities despite all the initial difficulties. But the sincerest “THANK YOU” goes to Alex Minetto, Marco Malinverno, Francesco Raviglione and Simone Zocca. Entering in a world totally unknown to me, such as Electronics and Telecommunications, was possible only thanks to you. Despite your many commitments, you have always managed to help me and make me learn so much, dedicating your time to my growth.

Finally, the biggest thank you goes to my half, Antonio.

Sara Golisciani

ABSTRACT

COLLABORATIVE POSITIONING IN LOW LATENCY VEHICULAR NETWORKS: COOPERATIVE INTEGRITY MONITORING

by

Sara Golisciani

Advisor: Daniel Aloï, Carla Chiasserini, Fabio Dovis

This research investigates the potentiality of Cooperative Vehicular Positioning in Intelligent Transportation Systems (ITSs) to mitigate the impacts of road transportation including road injuries, energy waste, and environmental pollution. At present, vehicular system is mainly based on the use of Global Navigation Satellite System (GNSS), that provides an absolute positioning with a good performance in open outdoor environments but, with a significant degradation or unavailability in urban canyons. However, thinking about a further improvement of absolute positioning capabilities is something unrealistic. Several works have pointed to Cooperative Positioning (CP) in vehicular networks, as the enabling technology to improve the quality of the estimated vehicles position in terms of space/time availability and reliability. Indeed, accurate and reliable positioning of vehicles is a decisive factor for many Intelligent Transportation System applications. To guarantee the reliability, GNSS integrity problem for vehicular positioning has a great value.

Regular exchange of information between road users keeps them informed about each other's position, driving kinematics, and other attributes. The technological revolution in the automotive industry will see every vehicle connected through Vehicle-To-

Everything (V2X) technologies in the coming years. These technologies will overturn the mobility experience in all its forms. Extremely dynamic topologies, variable densities, high bandwidth demand with ultra-low latency expected, and relatively high-power availability, make vehicular networks something complex. This is the reason why various automotive applications are usually tested by simulations. In this work, a framework for simulating the network, coupled with SUMO (Simulation of Urban Mobility), will be utilized. Nowadays, one of the greatest challenges is to integrate real navigation data in a network of connected vehicles. Indeed, as future work, an important feature that could be added to the framework is the possibility of injecting real GPS traces in alternative to SUMO mobility trace generator. In that case, each node simulated in the network will send messages filled with positioning information coming from real devices.

This thesis will investigate the performance of a GNSS solution, developing an application able to monitor integrity capability. Thus, it has been presented a new algorithm to evaluate the integrity of real position data available from GNSS receivers. Furthermore, to enable this Integrity Monitoring application, an analysis has been made on how to disseminate information coming from satellites in a network of connected vehicles.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	iv
ABSTRACT	v
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF ABBREVIATIONS	xiv
CHAPTER ONE	
INTRODUCTION AND LITERATURE SURVEY	1
1.1 Research Motivation	1
1.2 Cooperative Positioning	6
1.2.1 GNSS-only collaborative positioning	8
1.3 GNSS Integrity	13
1.3.1 GNSS Integrity Monitoring Methods	17
1.4 Thesis Outline and Objectives	21
CHAPTER TWO	
METHODOLOGY	24
2.1 Simulation Network for V2X	25
2.1.1 Vehicle to Network/Infrastructure: Scenario and Application	27
2.1.2 Vehicle to Vehicle: Scenario and Application	28
2.2 From Cooperative Awareness to Cooperative Enhancement	30

TABLE OF CONTENTS—Continued

2.3 Differential Unit: Inter-vehicle Distance Estimation	31
2.3.1 Single Differencing and Double Differencing Ranging techniques	33
2.4 Cooperative Integrity Monitoring Unit (CIMU)	35
2.4.1 Covariance Intersection	39
2.5 Cooperative Integrity Monitoring Application	40
CHAPTER THREE RESULTS	52
3.1 Testing Integrity Monitoring Application	52
3.1.1 Data Preparation	52
3.1.2 CIM Algorithm Results Analysis	55
3.2 Network Performance Analysis	59
CHAPTER FOUR CONCLUSIONS	70
4.1 Collaborative Agents Number Choice	70
4.2 Cooperative Integrity Monitoring: Possible Implementations	74
APPENDIX	
Copyright Permission Letter	75
REFERENCES	81

LIST OF TABLES

Table 3.1	Baseline Vectors errors generation along each Trajectory section	54
Table 3.2	Total traffic offered by CAM and CEM services Increasing the number of vehicles in the simulation	63
Table 4.1	Integrity Mean values over the entire trajectory and over each section of it, varying the total number of collaborative agents	72
Table 4.2	Integrity Standard Deviation (std) values over the entire trajectory and over each section of it, varying the total number of collaborative agents	73

LIST OF FIGURES

Figure 1.1	Intelligent Transportation System (ITS)	2
Figure 1.2	GNSS-Reflectometry	3
Figure 1.3	Navigation Performance Hierarchy	4
Figure 1.4	Vehicle to Everything concept scheme	7
Figure 1.5	High level block scheme of a networked positioning system including collaborative ranging module	10
Figure 1.6	Pseudo-range and Inter-vehicle range estimation	10
Figure 1.7	Absolute Horizontal Position Error	11
Figure 1.8	Target and neighbor nodes in opposite direction	13
Figure 1.9	Faults modes: a. Satellite fault, b. Terrestrial link fault	15
Figure 1.10	Illustration of GNSS receiver estimated position versus real vehicle position	17
Figure 1.11	Vehicular Cooperative positioning in urban scenario	18
Figure 1.12	Cooperative Positioning using GNSS receiver and UWB transceiver	19
Figure 1.13	Thesis flow chart	23
Figure 2.1	Navigation data processing scheme	26
Figure 2.2	Area Speed Advisory application scenario	27
Figure 2.3	Speed areas' subdivision for Area Speed Advisory Application	28
Figure 2.4	Emergency Vehicle Alert application scenario	29
Figure 2.5	Road segment and actions logic in Emergency vehicle Application	30

LIST OF FIGURES—Continued

Figure 2.6	Flow chart of double differencing (DD)-based IVD estimation method	32
Figure 2.7	Single differencing method	33
Figure 2.8	Double differencing method	35
Figure 2.9	Bell-shaped surface representing distribution density of bidimensional random variable and its contours	37
Figure 2.10	IVD and GNSS position vectors estimates' State and Covariance	38
Figure 2.11	Overall inter-agent distance error covariance intersected With target agent position vector covariance	40
Figure 2.12	No intersection points: a. one ellipse contained in the other, b. ellipses are separated	41
Figure 2.13	One intersection points: a. one ellipse contained in the other, b. ellipses are separated	41
Figure 2.14	Pseudo-code finding intersection points in case of 0/1 Intersection points	42
Figure 2.15	Two intersection points case	43
Figure 2.16	Pseudo-code performing segment area evaluation	44
Figure 2.17	Ellipse segment area evaluated subtracting triangle area to sector area	45
Figure 2.18	Ellipse segment area evaluated adding triangle area to sector area	46
Figure 2.19	Pseudo-code finding right ordering of points in case of two intersection points	48
Figure 2.20	a. Three intersection points, b. Four intersection points	48

LIST OF FIGURES—Continued

Figure 2.21	Pseudo-code finding right ordering of points in case of 3 or 4 intersection points	49
Figure 2.22	Integrity Monitoring algorithm full logic scheme	51
Figure 3.1	Total integrity parameter along different target agent trajectory sections	56
Figure 3.2	Target and collaborative agents' position and uncertainties in a random instant of time along second trajectory section [2501-5000] s	57
Figure 3.3	Target and collaborative agents' position and Uncertainties in a random instant of time along third trajectory section [5001-7500] s	58
Figure 3.4	Integrity monitoring along target agent route	59
Figure 3.5	Simulation map used to test CE protocol combined to CA protocol	60
Figure 3.6	V2V simulation result: total traffic offered by CAM service as function of total number of vehicles present in the scenario	61
Figure 3.7	V2V simulation result: total traffic offered by CEM service as function of total number of vehicles present in the scenario	62
Figure 3.8	V2V simulation result: mean traffic per vehicle offered by CAM and CEM service as function of the total number of vehicles present in the scenario	64
Figure 3.9	Zoom of mean traffic plot in regions [0-0.7] kB/s and [5-5.7] kB/s	65
Figure 3.10	One-way delay measured for CAMs messages when CEMs dissemination is not allowed	66
Figure 3.11	One-way delay measured for CAMs messages when CEMs dissemination is allowed	67

LIST OF FIGURES—Continued

Figure 3.12	One-way delay measured for CEMs messages	68
Figure 3.13	Mean Latency over total number of vehicles present in the scenario	69
Figure 4.1	Integrity mean trend compared to total number of aiding agents	71
Figure 4.2	Integrity Standard Deviation trend compared to total Number of aiding agents	71

LIST OF ABBREVIATIONS

ABAS	Airborne Based Augmentation System
AL	Alert Limit
ARAIM	Advanced Receiver Autonomous Integrity Monitoring
ASN.1	Abstract Syntax Notation Revision One
CA	Cooperative Awareness
CAM	Cooperative Awareness Messages
CE	Cooperative Enhancement
CEM	Cooperative Enhancement Messages
CERIM	Collaboration-Enhancement Receiver Integrity Monitoring
CI	Covariance Intersection
CIM	Cooperative Integrity Monitoring
CIN	Cooperative Inertial Navigation
CIMU	Cooperative Integrity Monitoring Unit
C-ITS	Cooperative Intelligent Transportation System
CP	Cooperative Positioning
C-V2X	Cellular Vehicle to Everything
DD	Double Differencing
DENM	Decentralized Environmental Notification Message
DSRC	Dedicated Short-Range Communication
DGPS	Differential Global Positioning System
DGNSS	Differential Global Navigation Satellite System

LIST OF ABBREVIATIONS—Continued

DU	Differential Unit
ENU	East-North-Up
EV	Emergency Vehicle
EVA	Emergency Vehicle Alert
FCC	Federal Communications Commission
FDE	Fault Detection and Exclusion
GBAS	Ground Based Augmentation System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GUI	Graphical User Interface
HMI	Hazardously Misleading Information
HPE	Horizontal Positioning Error
IEEE	Institute of Electrical and Electronics Engineers
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
ITS	Intelligent Transportation System
IVD	Inter-Vehicle Distance
LIDAR	Laser Detection and Ranging
LTE	Long Term Evolution
MAC	Media Access Control
MI	Misleading Information

LIST OF ABBREVIATIONS—Continued

MS-VAN3T	Multi Stack framework for VANET applications
NLOS	Non-Line of Sight
ns-3	network simulator 3
OBU	On-board Unit
PD	Probability Detection
PL	Protection Level
RAIM	Receiver Autonomous Integrity Monitoring
RRAIM	Relative Receiver Autonomous Integrity Monitoring
RSS	Received Signal Strength
RSU	Roadside Unit
RTK	Real Time Kinematics
SBAS	Satellite Based Augmentation System
SD	Single Differencing
SONAR	Sound Navigation and Ranging
STD	Standard Deviation
SUMO	Simulation of Urban Mobility
TDOA	Time Difference of Arrival
TOA	Time of Arrival
TraCI	Traffic Control Interface
TTA	Time To Alert

LIST OF ABBREVIATIONS—Continued

UWB	Ultra-Wide Band
V2I	Vehicle to Infrastructure
V2N	Vehicle to Network
V2P	Vehicle to Pedestrian
V2V	Vehicle to Vehicle
V2X	Vehicle to Everything
VANET	Vehicular ad-hoc Networks
WAVE	Wireless Access for Vehicular Environment
WLAN	Wireless Local Area Network

CHAPTER ONE

INTRODUCTION AND LITERATURE SURVEY

A novel over two centuries long, which tells the story of one of the most resounding inventions of all time, is the profile of the automobile, that sees in the year 1769 the first step towards four-wheel mobility, thanks to the invention of the first steam tank. This date marked the beginning of the motorization history: it was the first demonstration provided by a vehicle “auto-mobile” in the literal sense of the term, that it moves by itself through a non-animal-drawn. The first form of life in automotive industry is dated back in 1886, when the German engine designer and automotive engineer Karl Benz had the opportunity to indulge his old passion of designing a horseless carriage. Twenty years later, in 1913, Henry Ford was intended to contain the prices of goods produced through the reduction of working time. He introduced the first assembly line for the mass production of an entire automobile. His invention delivered into the hands of the people first affordable motorized vehicles. Compared to the vehicles of a century or more ago, current ones show not only an evident functional and aesthetic progress, but also a remarkable technological evolution. With the conception of transmuting cities into digital societies, Intelligent Transportation System (ITS) has become the indispensable component among all. It is the key component for city mobility.

1.1 Research Motivation

ITS has been established to make future traffic more safe, more efficient, and more eco-friendly. It enriches users with prior information about traffic, local convenience real-time running information, seat availability etc. which reduces travel time of commuters as

well as enhances their safety and comfort. An ideal representation of smart city concept, in which ITS plays a fundamental role, is depicted in Figure 1.1. Some of the involved technologies in ITS include vehicle navigation and fleet management. Recent navigation technologies for transportation systems require the precise positioning of the user vehicle and neighboring vehicles to improve performance and safety. At present, positioning, navigation, and timing capabilities in the vehicular field are supported by the Global Navigation Satellite System (GNSS).



Figure 1.1 Intelligent Transportation Systems (ITSs)

GNSS means:

- Absolute positioning: It is the ability to precisely determine localization with reference to a worldwide coordinate system.

- Navigation: It establishes present and desired location, and then keeps track of the course, velocity, and position to get the desired location.
- Timing: It is the ability of discerning and keeping time accurately, no matter the location.

Thus, a GNSS receiver can provide Positioning, Velocity and Time (PVT) information relying on a set of estimated quantities:

- Satellite to receiver distance (pseudo-range)
- Doppler shift or pseudo-range rate
- “Timestamps” of navigation signals

However, urban environment presents great challenges to common GNSS receivers. This is mainly because the GNSS positioning performance can be severely degraded by: limited satellite visibility, multipath effect, intentional or unintentional interferences, inaccurate ephemeris data and corrections and other undesired impairments and error sources. Thus, GNSS signals can be received in a direct way, or reflected by other surfaces, as in the example represented in Figure 1.2.

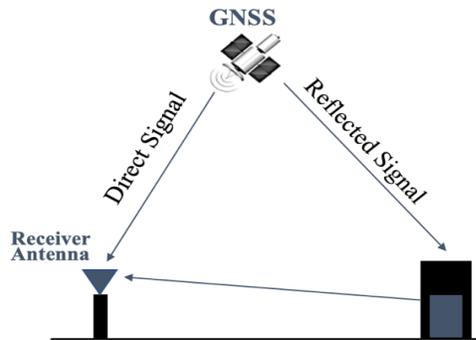


Figure 1.2 GNSS-Reflectometry

Generally, GNSS performance can be measured through four criteria [1] whose hierarchical positions have been represented in Figure 1.3:

- Accuracy: referred to a measured or estimated position and velocity of a vehicle, it is the “degree of conformance of these position and velocity with the true ones of the vehicle” [1].
- Integrity: conventionally defined as the “measure of trust that can be placed in the correctness of the information supplied by a navigation system” [1].
- Continuity: the “probability that the specified system performance (accuracy and integrity) will be maintained for the duration of a phase of operation, presuming that the system was available at the beginning of that phase of operation” [1].
- Availability: defined as the “percentage of time that the services of the system are usable by the navigator, which is an indication of the ability of the system to provide reliable information within the specified coverage area” [1].



Figure 1.3 Navigation Performance Hierarchy

Recent research works to go beyond the standalone GNSS weaknesses. The idea, to cope with GNSS shortcomings, has been to have networked Global Navigation Satellite System (GNSS) receivers supporting the sharing of raw measurements with other receivers within the same network. Such measurements can be processed with different techniques to retrieve inter-agent distances which can be in turn integrated to improve positioning performance. Differential GNSS (DGNSS) is one of these techniques helping in an accuracy improvement. It is based on relative positioning corrections from Geo-referenced stations. Indeed, the GNSS- standalone can reach 5-10 m of accuracy only in open sky, while, with a differential correction, it can be reached a meter-level accuracy. A further improvement can be obtained by the Real Time Kinematic (RTK) solution, based on corrections from a base station. This method is more precise and allows to reach a centimeter-level accuracy, but it could have unpredictable convergency time. However, also in this case, environmental features can highly limit navigation and positioning capabilities. Thus, “the accuracy and the availability of GNSS do not meet the requirements of the most demanding vehicular applications such as collision avoidance or lane-level positioning” [2]. In addition, the above methods may not be appropriate for low-cost commercial GNSS receivers in vehicular applications and, most of all, they ignore the potentiality of a collaborative localization in a network. In recent years, many cooperative approaches to improve the localization and navigation performance in harsh contexts has been presented. However, multiple vehicles dataset is rare and complex to obtain. Thus, until now, these solutions have been tested in a completely simulated environment in which data were unrealistic and at low rate.

In one hand, hoping for a further improvement of the current absolute positioning capabilities in terms of accuracy and availability (DGNSS, RTK) may be unrealistic for the more sophisticated navigation units. In the other hand, GNSS integrity is considered as one of the most essential performance parameters. Thus, extending the concept of integrity to a network of agents has recently attracted increasing interest and demand and it appears as a more feasible solution.

1.2 Cooperative Positioning

Regular exchange of information between road users keeps them informed about each other's position, driving kinematics, and other attributes. This is the basis of cooperative positioning: the cornerstone of road safety and traffic efficiency applications on the way towards autonomous driving.

Cooperative Intelligent Transportation Systems (C-ITSs) equip vehicles with the ability to communicate with each other and surrounding infrastructure wirelessly to enhance their positioning information determined by on-board sensors. The European Commission highlights that “depending on the nature of the applications (e.g. information supply, awareness, assistance, warning to avoid an accident, traffic management), C-ITS can contribute to improved road safety by avoiding accidents and reducing their severity, to decreased congestion, by optimizing performance and available capacity of existing road transport infrastructure, to enhanced vehicle fleet management, by increasing travel time reliability and to reduced energy use and negative environmental impact.” [3].

The sharing of information between vehicles can only be achieved through the availability of a communications infrastructure that supports information exchanges between vehicles and/or between vehicles and roadside infrastructure. Indeed, key parts of ITS framework

are concepts of ‘Vehicle to Everything’ (V2X) and ‘Connected Vehicles’. They refer to a communication system enabling the wireless data exchange between the vehicle and any other entity in its surroundings.

Vehicle to Everything communication system represented in Figure 1.4, integrates different types of communication including:

- Vehicle to Vehicle (V2V): to allow inter-vehicle communications
- Vehicle to Pedestrian (V2P): to facilitate interaction with pedestrian
- Vehicle to Infrastructure (V2I): to reach roadside facilities
- Vehicle to Network (V2N): to communicate with the network-based services and applications.

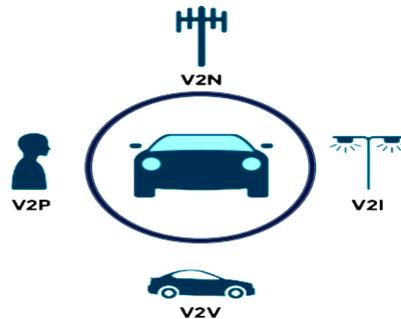


Figure 1.4 Vehicle to Everything concept scheme

Depending on the underlying technology used, there are two types of V2X communication technology: Wireless Local Area Network (WLAN)-based, and Cellular based. WLAN is particularly well suited for V2X communication due to its low latency, and it is based on

IEEE 802.11 family of standards. Based on IEEE 802.11, Dedicated Short-Range Communications (DSRC) standard aims to reduce as much as possible the size of the package to be sent via the radio link. In US, the Federal Communications Commission (FCC) has allocated for DSRC a dedicated bandwidth of 75 MHz for vehicular communications, from 5.850 to 5.925 GHz. In Europe, instead, the regulatory entity for vehicular networks is the European Commission, assigned the band from 5.855 to 5.925 GHz to vehicular applications. The Wireless Access in Vehicular Environments (WAVE) has been defined to enable mobile devices to operate in the DSRC band, ensuring the collection of traffic information, immediate and stable transmission, and secure information. WAVE contains a new amendment of the just mentioned IEEE 802.11 standard, IEEE802.11p. The main changes were made at MAC level as it was necessary to take action to make communication between vehicles faster and efficient at the same time. In Europe, 802.11p is used as the basis for the ITS-G5 standard. Instead, cellular-based V2X (C-V2X) is based on LTE as underlying technology.

1.2.1 GNSS-only collaborative positioning

Navigation and Positioning in urban traffic is mainly needed by vehicles and pedestrians. For this reason, “these systems have to be small, easy to use and requiring low power consumption” [4]. To tackle the navigation challenges, other methods to replace or to augment the use of GNSS have been studied by many researchers. Minetto et al. in [4] present an overview of GNSS-only collaborative localization in the context of cooperative connected cars. More in detail, they investigated the possible hybridization of GNSS with other sensors, with Inertial Navigation Systems (INS).

“The scope of such an integration is not only to improve the accuracy and precision of the solution, but also to provide a means to increase robustness and reliability of the positioning procedure to threats typical of urban environment such as signal obscuring, multipath etc.” [4]. In addition to INS, GNSS can also benefit from other complementary sensors such as visual navigation system, classified as proprioceptive sensors because they provide additional information about the state of the vehicle, and ranging sensors, such as LIDAR and SONAR, called exteroceptive sensors because they are able to sense the surrounding environment [4]. Unfortunately, ranging capabilities are highly limited by the Line of Sight (LOS) constraint. Furthermore, these sensors must be supported by high computational complexity algorithms. This high complexity can be overcome by exploiting connectivity solutions among different vehicles in the network. Thanks to cooperation among navigating units, it has been added the possibility to share additional ranging information. Thus, the relative measurements may be integrated in a hybrid absolute positioning to a cooperative solution as depicted in the block scheme in Figure 1.5. Thus, the hybrid measurement is composed by two estimations:

- Pseudo-range: distance between a satellite (s) and a user (i)
- Inter-Vehicle range: distance between a user (i) and an aiding agent (j)

According to what said so far, the computation of inter-vehicular ranges is of prominent importance for cooperative positioning of swarms of vehicles. As an example, the Double Differential (DD)-based method allows to extract a terrestrial collaborative range measurement whenever the agents share the LOS visibility of several common satellites as shown in Figure 1.6.

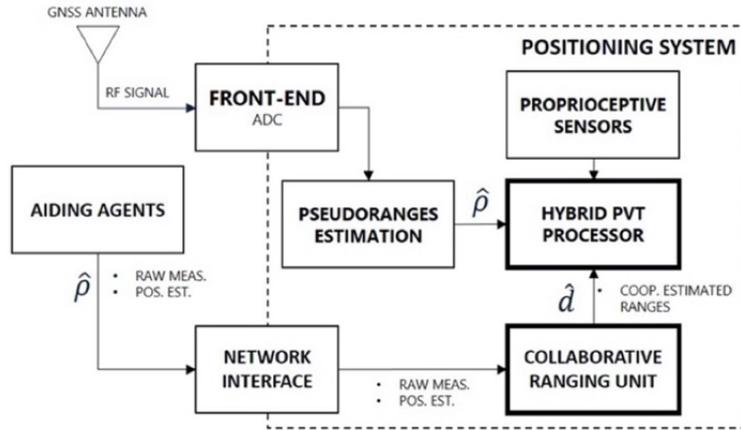


Figure 1.5 High-level block scheme of a networked positioning system including collaborative ranging module

Note: From “GNSS-only Collaborative Positioning Among Connected Vehicles” by M. A. et al., 2019, TOP-Cars’19, p. 38. Copyright 2000 by Minetto A. Reprinted by permission

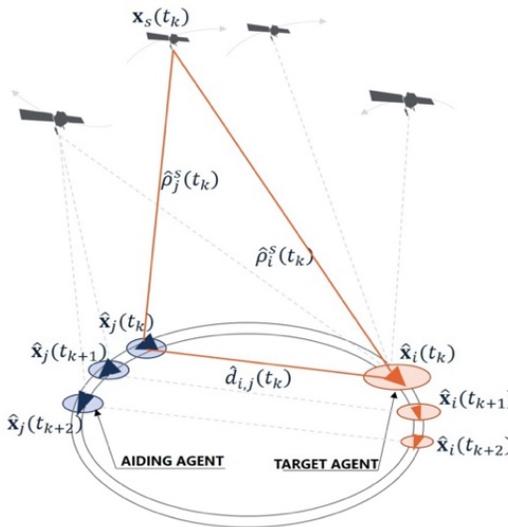


Figure 1.6 Pseudo-range and inter-vehicle range estimation

Note: From “GNSS-only Collaborative Positioning Among Connected Vehicles” by M. A. Author et al., 2019, TOP-Cars’19, p. 39. Copyright 2000 by Minetto A. Reprinted by permission

Effective implementation of GNSS-based cooperative navigation supported by DSRC shows a remarkable performance improvements w.r.t non-cooperative positioning solutions. The sample application proposed in [4] includes a set of aiding agents moving in two directions of a circular trajectory of 200 m of radius. In Figure 1.7 the behavior of the absolute Horizontal Position Error (HPE) over the whole timespan of the experiment is shown. It is possible to notice that the time-averaged accuracy of hybrid solution outperforms the standalone GNSS solution. Indeed, the mean errors are 8.25 m and 9.27 m respectively.

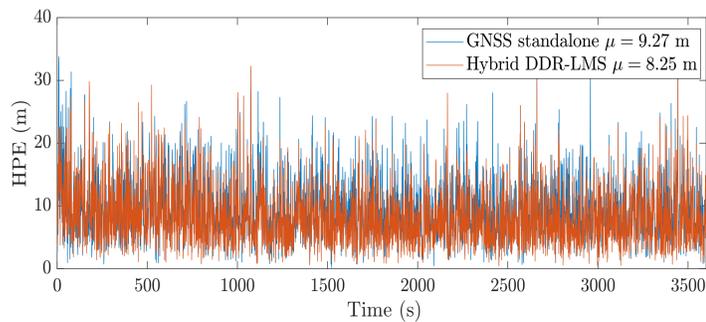


Figure 1.7 Absolute Horizontal Position Error

Note: From “GNSS-only Collaborative Positioning Among Connected Vehicles” by M. A. Author et al., 2019, TOP-Cars’19, p.41. Copyright 2000 by Minetto A. Reprinted by permission

A novel CP technique is presented by Alam et al. in [5]. Their cooperative inertial navigation (CIN) method can be used to enhance INS-based positioning in difficult GNSS environments, such as in very dense urban areas and tunnels. In the CIN method that is proposed, vehicles communicate their Inertial Measurement Unit (IMU) and INS based

position data with oncoming vehicles traveling in the opposite direction. For the CIN, it is assumed that vehicles broadcast their ID, position estimate provided by the INS, the Euler angles, and odometer-based speed estimates. Each vehicle receives data from all its neighbors. The CIN technique at a target vehicle should identify a passing neighbor vehicle and fuse the vehicle's own data with those received from the neighbor that passed by. In [5] two vehicles with and without CIN technique respectively have been tested. The corresponding INS- based positioning errors are the result of such experiment. The proposed CIN method allows to decrease the positioning error at the passage time of a vehicle in the neighborhoods of target agent, and in the subsequent epochs, as a result. The detection of another passing vehicle is mainly based on the usage of Doppler Shift effect.

Until now the most common techniques considered for radio ranging were Received Signal Strength (RSS), Time of Arrival (TOA), and Time Difference of Arrival (TDOA). However, the feasibility of these radio ranging methods in the harsh environment of vehicular networks is questionable. A new CP method is presented by Alam et al. in [6] for improving the GPS estimates using inter-node range-rates based on the Doppler shift of the carrier of Dedicated Short-Range Communication (DSRC) signals, the nominated medium for vehicular communication. The main reason behind this novelty is that the Doppler shift is considerably less distorted by channel fading and multipath which are dominant sources of errors in a vehicular environment. Basically, the Doppler shift requires relative mobility between users. Thus, after defining a target vehicle, neighbors are defined as those vehicles traveling in the opposite direction to the target vehicle, as depicted in Figure 1.8.

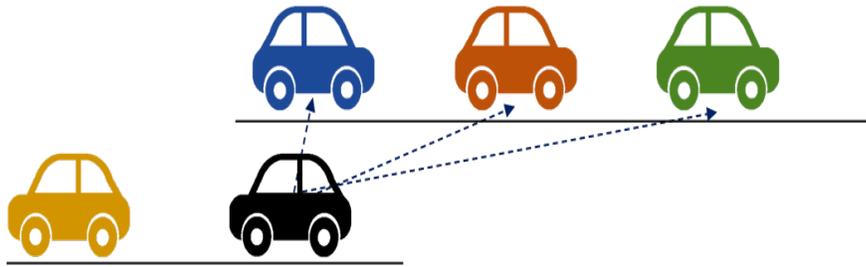


Figure 1.8 Target and neighbor nodes in opposite direction

In paper [6] it has been clearly showed the performance of proposed CP method at different conditions of speeds and traffic intensities. The traffic intensity is varied between 5 to 50 vehicles /km and the different average speeds of the traffic are 54, 72, and 90 km/h. The performance results almost constant from a traffic intensity of 15 vehicles /km on. Regarding the overall performance, enhancement of about 48% over the GPS-based position method is achieved for lower speeds. That result is due to a longer link lifetime between vehicles at lower speed. Although more intensive traffic leads to a higher number of neighbors, the uncertainty of the position and velocity of the neighbors limits the achievable performance. For this, the performance improvement is saturated for higher traffic densities. Thus, it is intuitable that this method is no suitable for dense urban areas.

1.3 GNSS Integrity

Up to now, focus of GNSS-based cooperative positioning applications has been to provide users with high positioning accuracy and availability. Compared with conventional GNSS applications, cooperative positioning applications are more vulnerable to the interferences or attacks from the ground. As matter of fact, in a cooperative positioning

system, the faults occurring at one device could also affect other devices because also errors related to ranging links between the devices are present. Thus, severe positioning errors of each node could result.

Chen et al. in [10], underlined how it is important to consider all faults' modes to achieve integrity monitoring for GNSS receivers. In a GNSS-based cooperative positioning system, the pseudo-range measurements and terrestrial range measurements are utilized to calculate the locations of peer nodes. Both could be affected by interferences. Faults' modes are illustrated in Figure 1.9 and can be classified as follows:

- Satellite-related faults: satellite and/or constellation faults
- Node-related faults: terrestrial link and/or terrestrial node faults

Despite the existing difficulties, introducing the integrity concept to urban GNSS receivers is more and more attractive and could solve these issues. GNSS integrity concept firstly developed in aviation field for Safety of Life applications. It is defined as a “measure of trust which can be placed in the correctness of the information supplied by the total system” [7] and includes the “ability of the system to provide timely warnings to users when the system should not be used for navigation” [7].

Zhu et al. in [1] underline the importance to bound all possible errors related to a positioning solution or to ensure that the probability of errors not properly bounded is below a certain limit, to reduce the probability of the harmful effects and to guarantee the correctness and fairness of the decision.

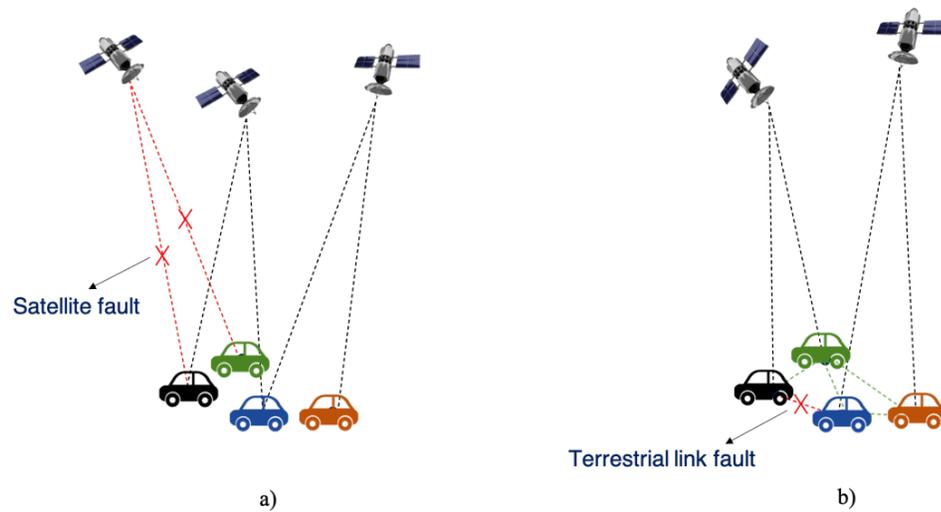


Figure 1.9 Faults' modes: a. Satellite fault, b. Terrestrial Link fault

Integrity concept can be further clarified thanks to four main parameters:

- Alert Limit (AL): the “largest position error allowable for safe operation. In urban context we are only interested in horizontal dimension. Thus, it is defined a Horizontal Alert Limit (HAL) is the radius of a circle in the horizontal plane, whose center is the real position, and which describe the region required to contain the indicated horizontal position with the required probability” [1].
- Time to Alert (TTA): the maximum allowable elapsed time from the onset of a positioning failure until the equipment announces the alert
- Integrity risk: the “probability of providing a signal that is out of tolerance without warning the user in a given period of time” [9]
- Protection Level (PL): a “statistical error bound computed to guarantee that

the probability of the absolute position error exceeding the said number is smaller than or equal to the target integrity risk” [9]

Generally, AL is defined by some applications, while PL is evaluated by users. Thanks to a comparison between these two parameters, the decision alert is done. Thus, if $PL > AL$ the alert triggers, otherwise not. In addition, a list of integrity events can be done:

- Integrity Failure: “integrity event that lasts for longer than the TTA and with no alarm raised within the TTA” [1]
- Misleading Information (MI): “integrity event occurring when, being the system declared available, the position error exceeds the protection level but not the alert limit” [1]
- Hazardously Misleading Information (HMI): integrity event occurring when, being the system declared available, the position error exceeds the alert limit.

Therefore, GNSS receiver can provide an estimate of position taken by a vehicle, which will be close or not to the real one depending on the integrity of the solution, as depicted in Figure 1.10. However, even if the integrity is a crucial measure of confidence of the information supplied by navigation system, “integrity monitoring algorithms developed in the aviation domain cannot be transported directly into the urban vehicle applications” [1]. This is because, on the one hand, the integrity monitoring algorithms developed in the aviation context are established on the fact that a high data redundancy exists, which is not the case in the urban context. On the other hand, the single-fault assumption made in the aerospace applications is not true for urban GNSS receivers due to the potentially large and frequent errors caused by multipath interference and Non-Line-Of-Sight (NLOS)” [1].

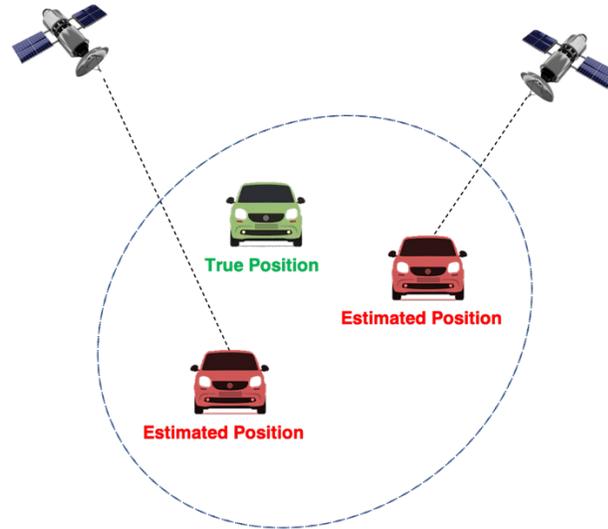


Figure 1.10 Illustration of GNSS receiver estimated position vs real vehicle position

1.3.1 GNSS Integrity Monitoring Methods

Many GNSS augmentation systems such as airborne based augmentation system (ABAS), ground-based augmentation system (GBAS) and satellite-based augmentation system (SBAS) were used to provide integrity information. GBAS and SBAS are Differential GPS systems (DGPS), developing corrections that improve the accuracy of the measurements and generate real-time error bounds. SBAS and GBAS are both very powerful means of guaranteeing integrity [13]. However, they are not suitable for low-cost commercial GNSS receivers in vehicular applications because they need complex and costly external aiding agents. At user level, the GNSS integrity can be monitored by exploiting the redundancy of the GNSS signals as collected at the receiver. Receiver autonomous integrity monitoring (RAIM) allows to monitor the residuals of the position solution in GNSS receivers. Relative to GBAS and SBAS, it is independent of external equipment allowing a rapid alert ability [8]. Many improved methods have been proposed

in recent years, such as advanced RAIM (ARAIM) [11] and relative RAIM (RRAIM) [12]. Xiong et al. in [8], since RAIM techniques only use information of a standalone receiver, proposed a new cooperative algorithm. As previously underlined, detection sensitivity is severely degraded by the limited number of available satellites, multipath and the non-line-of-sight (NLOS) as shown in Figure 1.11.

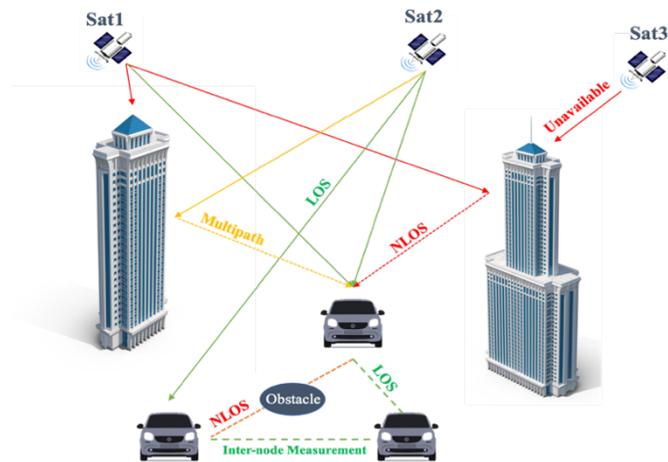


Figure 1.11 Vehicular Cooperative positioning in urban scenario

The detection sensitivity issue could be solved by augmenting the receiver with other sensors to increase the redundancy of the measurements. However, the sensors used are just belonging to a standalone vehicle, which does not take full advantage of the interaction with other vehicles in a Cooperative-ITS mechanism. A first idea to allow the full use of GNSS measurements from all collaborators is taking a collaboration-enhanced receiver integrity monitoring (CERIM). This method can improve the detection of faulty

GNSS measurements, but it cannot be used to detect faults of inter-vehicle measurements. Xiong et al. in [14] illustrated a cooperative integrity monitoring (CIM) method that cooperatively monitors the integrity for multiple sensor cooperative positioning (CP) systems.

The proposed method will not only improve the detection sensitivity of faulty GNSS measurements, but it is able to detect faults in sensors dedicated to inter-node measurements such as ultra-wideband (UWB) and dedicated short-range communications (DSRC). In paper [13] navigation measurements have been captured by GNSS receiver, adopted as basic positioning sensor, and UWB transceiver, used as inter-vehicle ranging sensor. The scenario is illustrated in Figure 1.12.

The fusion of information coming from both sensors is based on some innovations on the structure of a decentralized Kalman Filter and represents the first part of the integrity method. The second one is the fault detection and exclusion (FDE). FDE is an iterative procedure: if the test detecting the existence of faulty measurements satisfies specific conditions, navigation result is considered available.

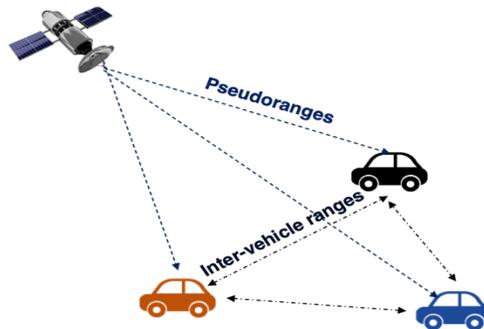


Figure 1.12 Cooperative positioning using GNSS receiver and UWB transceiver

Otherwise, the faulty hypothesis has been chosen and it must be conducted a fault exclusion. In each fault exclusion iteration, measurement that has the largest effect on test statistic is excluded, until the test statistic is lower than a detection threshold. There have been implemented two test statistics simultaneously:

- Common part: same measurement error observed for all collaborators due to a falling satellite
- Specific part: measurement error unique for the vehicle of interest

To evaluate performances of proposed CIM, Xiong et al. in [13] set a simulation of four vehicles in a heavy GNSS multipath scenario to compare the probabilities of detection (PD) for error coming from both common and specific part. Results are divided into two parts. In the first one only GNSS faults are present, while in the second both GNSS and UWB are simulated. During the simulation both GNSS signal and UWB ranging has been affected by existence of buildings and other obstacles. As expected, the proposed CIM method has a better fault detection ability in case of common GNSS errors. As an example, when GNSS bias is up to 10 m, PDs found in [13] are:

- 100% for CIM
- 98.4% for CIM without UWB
- 90.38% for CERIM
- 60.77% for RAIM

As in the common case, CIM is obviously the better method also in case of specific GNSS errors. When GNSS bias is larger than 8 m, PDs found in [13] are:

- 96.52% for CIM
- 84.44% for CIM without UWB

- 65.48% for CERIM
- 40.67% for RAIM

To further verify the performance of CIM in a realistic scenario with multipath and NLOS, in [10] it has been compared the positioning error of 1 vehicle, among the four simulated, in two cases: the proposed CIM with FDE and without FDE. Results found in [8] show the better accuracy of CIM relative to the non-FDE case for a simulation of 800 s. As an example, at 374 s, the positioning error (m) is:

- 17.13 m for CIM without FDE case
- 2.37 m for CIM with FDE case

Thus, without FDE, results suffer from GPS and UWB errors.

1.4 Thesis Outline and Objectives

This work concerns the feasibility of collaborative positioning applications in urban scenarios where multiple nearby vehicles are present. The thesis will deal with:

- Simulating the network in which connected vehicles can exchange data
- Taking and assessing raw GNSS traces, from which a set of real vehicles position estimates, together with their uncertainties, has been obtained
- Testing an application which evaluates real GNSS observations integrity for a target vehicle through aiding agent external estimates: Cooperative Integrity Monitoring Application
- Testing a cooperative enhancement (CE) protocol in the framework, performed for exchanging packets equal by type and size to real positioning data, which allows to enable the Cooperative Integrity Monitoring Application
- Analyzing network performance

- Analyzing integrity of a GNSS solution

This thesis starts, in Chapter 2, with an overview of adopted simulation framework. Two applications have been presented. Then, following the flow chart depicted in Figure 1.13, raw GNSS traces have been processed. It has been explained how GNSS positioning data have been injected in the framework as dummy traces to evaluate the possibility of exchanging this type of packets in the network. This analysis allows to understand if some application can be tested with this new protocol. Indeed, at the end of Chapter 2, an accurate delineation of the new tested application evaluating GNSS receiver solution integrity has been made. Chapter 3 shows both results of the integrity algorithm tested for real GNSS observations and of network performance parameters. Finally, Chapter 4 concludes the thesis.

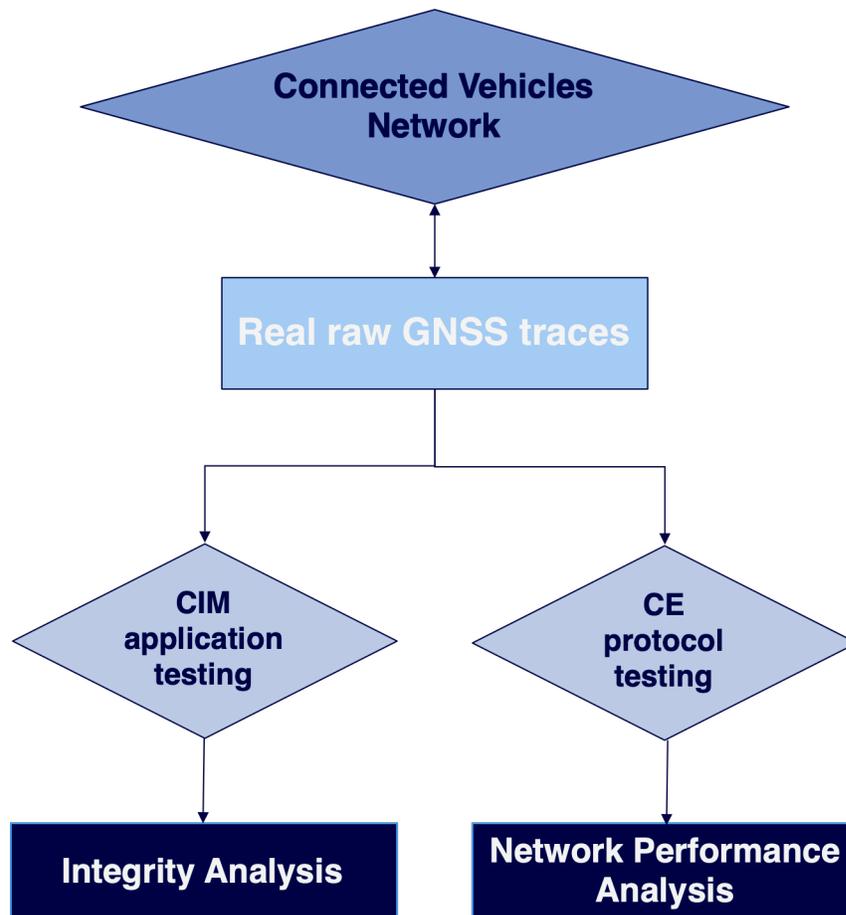


Figure 1.13 Thesis flow chart

CHAPTER TWO

METHODOLOGY

The applicability of cooperative integrity concepts underlined in Chapter 1 is subject to the availability of multiple observations of GNSS signals taken by different vehicles, which can be shared and combined to implement a collaborative spatial and temporal characterization. Indeed, the idea that stays behind the cooperative approach is to consider not only the single receiver, but also its nearby environment in its nominal conditions. Raw GNSS measurements coming from real trajectories covered by each vehicle i include:

- Pseudo-range estimate state $\hat{\mathbf{x}}_{i,k}$
- Pseudo-range estimate covariance $\mathbf{P}_{i,k}$

These data have been shared among the vehicles themselves, due to V2X system potentiality, and fused with other information sources to allow a cooperative localization. Processing these data, distances among vehicles themselves have been evaluated. In this work, this procedure has been made by a Differential Unit (DU), which adopts an inter-vehicle distance (IVD) estimation method: double differential (DD)-based method. Using this technique, available quantities will be:

- Inter-agent distance estimate state $\mathbf{d}_{ab,k}$
- Inter-vehicle distance estimate covariance $\mathbf{R}_{\mathbf{d},k}$

All mentioned estimated quantities, $\hat{\mathbf{x}}_{i,k}$, $\mathbf{P}_{i,k}$, $\mathbf{d}_{ab,k}$ and $\mathbf{R}_{\mathbf{d},k}$, represent the inputs of a collaborative integrity monitoring unit (CIMU). The output, rather, will be an accurate

integrity analysis. Furthermore, network performance analysis has been done. The procedure pattern followed is outlined in Figure 2.1.

2.1 Simulation Network for V2X

To develop and test different V2X applications, it is important to consider a simulator both for network and mobility aspects. Among all possible solutions allowing the communication between vehicles in a simulated environment, a multi-stack framework for VANET applications (MS-VAN3T) has been used in this work [14] [15]. This framework is composed by two pieces of software, whose bidirectional coupling is enabled by the so called TraCI (Traffic Control Interface) interface:

- Simulator of Urban Mobility (SUMO): It is a free, open, microscopic, and continuous road traffic simulation suite designed to handle large road networks. It allows modelling of intermodal traffic systems including road vehicles, public transport, and pedestrians. It also has a Graphical User Interface (GUI) available, to better visualize simulation results
- Network simulator (ns-3): It enables the creation of complex scenarios and allows to model all the aspects of communication among vehicles

With ns-3 it is possible to make use of different models to simulate vehicles' communication: a model for 802.11p, for LTE communications and for C-V2X. The key feature that distinguishes MS-VAN3T from other simulation framework is the possibility to not be limited to the usage of a single access technology but to be able to switch among different ones [15]. More in detail, for V2I/V2N communications it is possible to choose over LTE and 802.11p standards, instead, for V2V communications, over C-V2X and 802.11p standards.

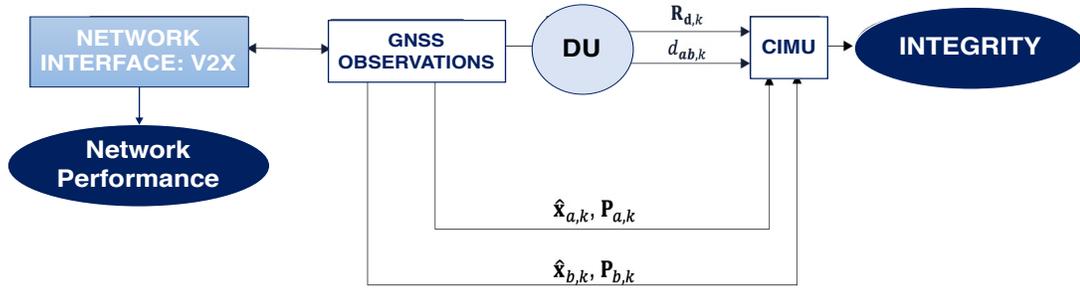


Figure 2.1 Navigation data processing scheme

In the adopted framework each vehicle can broadcast two messages' typologies:

- Cooperative Awareness Messages (CAMs): They are sent with a frequency going from 1 Hz to 10 Hz and allow to create or maintain awareness of each other in the network. Principal information included are about the status, as position, speed, time, and acceleration, and about vehicles attributes such as their dimension and type.
- Decentralized Environmental Notification Messages (DENMs): They are event-based messages, sent if a vehicle senses special conditions or incidents. Thus, they are used for emergency situations. DENMs are sent in addition to CAMs, and not instead of them.

Therefore, V2V and V2I/V2N communication systems can be employed to disseminate time-critical or awareness messages toward different nodes of the network. Vehicles can broadcast messages among themselves, through on-board units (OBUs), or to remote hosts, connected to a roadside unit (RSU). To show potentiality of adopted simulation framework, two sample applications, one for V2I/V2N and one for V2V communication models, have been illustrated.

2.1.1 Vehicle to Infrastructure/Network: Scenario and Application

For what concern V2I/V2N communication, vehicles periodically send CAMs messages, while an RSU broadcasts DENMs messages to all vehicles in a delimited zone [14]. For such a case, tested application is called “*Area Speed Advisory*”, whose scenario is depicted in Figure 2.2.



Figure 2.2 Area Speed Advisory application scenario

Note: From “Safety Applications and Measurement Tools for Connected Vehicles” by M. M. Author, 2021, Doctoral Thesis Polito, p. 56. Copyright 2000 by Malinverno M., Reprinted by permission

The map used in this context includes two road crossings connected through a central two-way street. The main logic adopted in this application is to divide the map into two different zones, depending on allowed maximum speed: in the central area limit is 25 km/h, while, in the outer area reachable speed is 75 km/h. The map representing this areas’ subdivision is depicted in Figure 2.3.

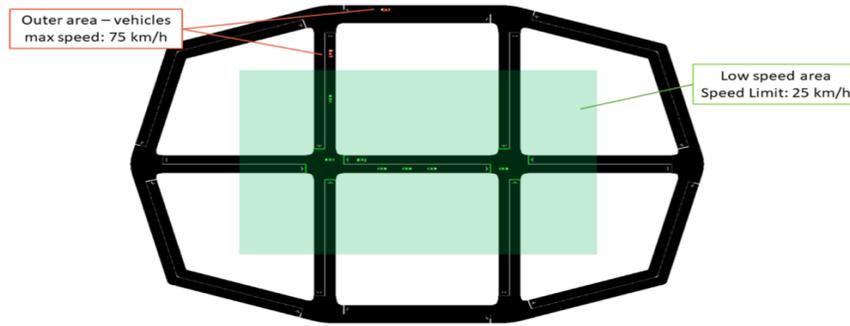


Figure 2.3 Speed areas' subdivision for Area Speed Advisory application

Note: From “A Multi-stack Simulation Framework for Vehicular Applications Testing” by M. M. et al. Author, 2020, Proceedings of the 10th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, Copyright 2000 by Malinverno M., Reprinted by permission

The logic of V2I and V2N is similar but it differs slightly depending on whether 802.11p (V2I) or LTE is used (V2N) [14]. The main difference among access technologies LTE and 802.11p is sending messages in unicast or in broadcast mode respectively [15]. In the first case, the server tracks vehicles through their CAMs sent and, in the case, advise to lower their speed due to restriction area entrance [14]. Vehicles send unicast messages to the server, which is the only one aware of two areas' boundaries. In the second case, broadcasting capabilities benefit has been exploited. Server sends a unique message imposing central bounded area as its destination. In this case, server cannot control each vehicle position, but deliver messages at a fixed frequency [15].

2.1.2 Vehicle to Vehicle: Scenario and Application

For what concern V2V communication, vehicles periodically send CAMs messages directly among themselves [15]. For such a case, tested application is called “*Emergency Vehicle Alert (EVA)*”, whose scenario is depicted in Figure 2.4:

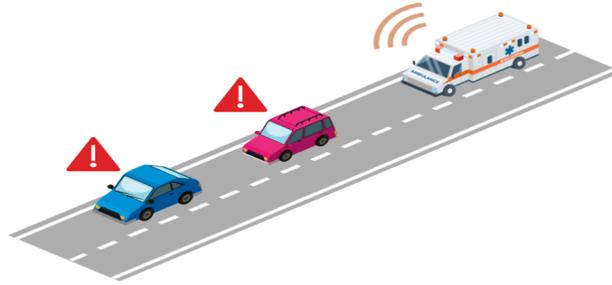


Figure 2.4 Emergency Vehicle Alert application scenario

Note: From “Safety Applications and Measurement Tools for Connected Vehicles” by *M. M. Author*, 2021, Doctoral Thesis Polito, p. 56. Copyright 2000 by Malinverno M., Reprinted by permission

Vehicles can be divided into two categories: passenger and emergency type. An example of emergency vehicle (EV) can be the ambulance. The scenario modeled is such that EVs periodically send DENM messages to all nearby vehicles to warn them [14]. Accordingly, alerted vehicles try to limit their hindrance to the EV. The map representing this scenario, depicted in Figure 2.5, consists of circular road segment, with two lanes for each direction of travel. Action taken by normal vehicles depends on the case [14]:

- Normal vehicle and Emergency vehicle on the same lane: normal vehicle will try to change lane. If it is no possible, it will speed up (green vehicle)
- Normal vehicle and Emergency vehicle on a different lane: normal vehicle will slow down (orange vehicle)
- Normal vehicle and Emergency vehicle are far away: no action taken

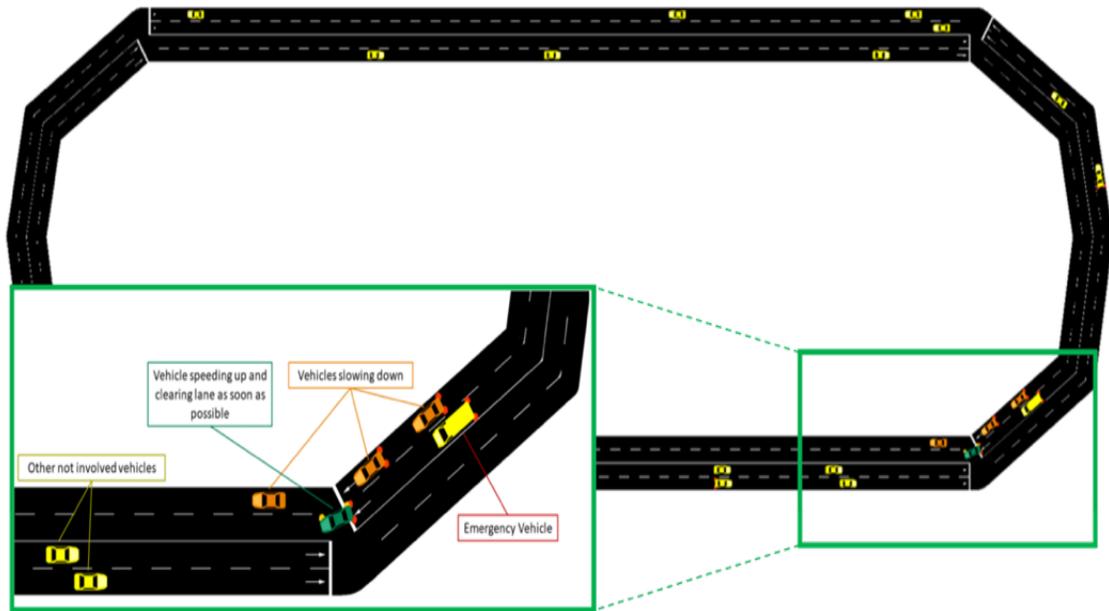


Figure 2.5 Road segment and actions logic in Emergency Vehicle Alert application

Note: From “A Multi-stack Simulation Framework for Vehicular Applications Testing” by M. M. et al. Author, 2020, Proceedings of the 10th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, Copyright 2000 by Malinverno M., Reprinted by permission

2.2 From Cooperative Awareness to Cooperative Enhancement

Applications described in 2.1 are enabled by the CA protocol for the information messages exchange. A protocol is used to define different rules about communication among entities in the network. For instance, CAMs generation interval is managed in between 100 ms and 1000 ms, while CAMs packet length is around 120 Bytes. In this work a new protocol, added to CA one, called Cooperative Enhancement (CE) protocol for messages exchange, will be addressed. In particular, the structure of CEM messages has been defined through ASN.1 specification. Through this protocol, it has been possible to define CEMs packets

length and to manage the way and the time in which they are generated and disseminated.

More in detail, CEMs packets can be of two types:

- Full precision intra-frames, called *I*-frames, whose dimension is almost 800 bytes each
- Differential micro-frames, called *D*-frames, whose dimension is almost 400 bytes each.

Their dissemination rule stipulates that nine *D*-frames are sent between two consecutive *I*-frames, because the formers are differential relative to the latter. Indeed, *D*-frames do not contain full data, but they only provide additional details with respect to *I*-frames information data.

CEMs generation interval is fixed to 100 ms. All vehicles, considered as nodes in the network, are in radio-range among themselves. It means that each node can communicate with all the other nodes, even if they are far away or if some obstacles are present. The innovation introduced, relative to CAMs, is the data type and size. Real time positioning data coming from GNSS observations have been processed and injected in the framework as dummy traces. It means that these real data have not been used to provide positioning information but, their dimension and type (e.g., latitude, longitude) have been retrieved to be reproduced in CEMs packets. Thanks to this procedure, it has been analyzed performance of the network, forced to take charge of both CAMs and CEM transmission protocol.

2.3 Differential Unit: Inter-Vehicles Distance Estimation

The availability of a vehicles' position estimates relative to successive instants of time taken by a GNSS receiver observations, allows to evaluate inter-vehicles distance

through some ranging method. In this work, Double Differencing method, whose procedure is illustrated in Figure 2.6, has been adopted.

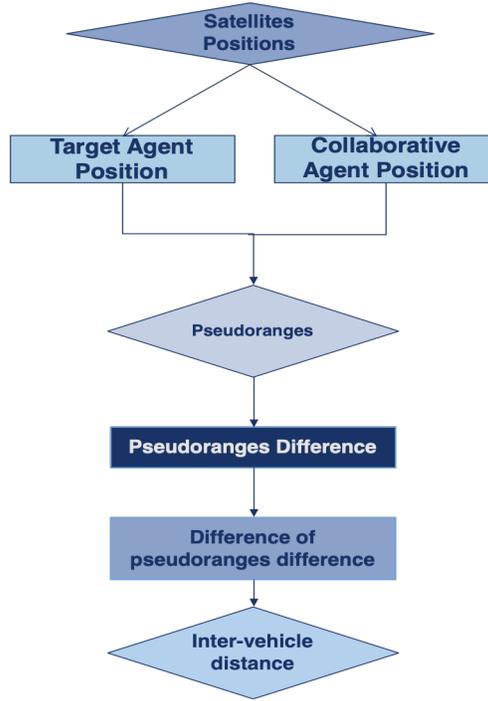


Figure 2.6 Flowchart of double difference (DD)-based IVD estimation method

In general, the aim of any ranging technique is to provide a distance estimation between two locations. Typically, at a given time k , one location is designated for the target agent, which will be called ‘ a ’, the other for the aiding agent, called ‘ b ’. Given two agents positions’ estimate, $\mathbf{x}_{a,k} = [x_{a,k} \ y_{a,k} \ z_{a,k}]$ and $\mathbf{x}_{b,k} = [x_{b,k} \ y_{b,k} \ z_{b,k}]$, distance has been evaluated in (1) as the norm of the displacement vector between these two positions:

$$d_{ab,k} = \|\mathbf{d}_{ab,k}\| = \|\mathbf{x}_{a,k} - \mathbf{x}_{b,k}\| \quad (1)$$

$$d_{ab,k} = \sqrt{(x_{a,k} - x_{b,k})^2 + (y_{a,k} - y_{b,k})^2 + (z_{a,k} - z_{b,k})^2} \quad (2)$$

The vector $d_{ab,k}$ is the so-called baseline vector. Assuming a as target agent, the receiver b must provide observable measurements synchronous to GNSS timescale.

2.3.1 Single Differencing and Double Differencing ranging techniques

Single Differencing (SD) Distance. The SD-method estimates the inter-vehicle distance only using pseudo-ranges of two vehicles from the same satellite [16]. As shown in Figure 2.7, since the satellite is far away from the vehicles, pseudo-ranges are assumed to be in parallel.

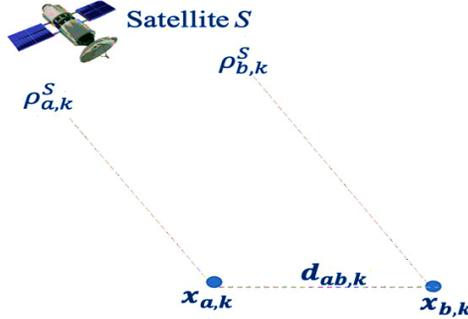


Figure 2.7 Single Differencing method

Pseudo-ranges relative to satellite S from vehicle V can be evaluated as in (3).

$$\rho_{V,k}^S = R_V^S + t_{V,k}^S + \varepsilon_{c,k} + \varepsilon_{uV,k} \quad (3)$$

Where:

- $R_V^S = \|\mathbf{x}_{s,k} - \mathbf{x}_{V,k}\|$ is the true distance between vehicle V and satellite S
- $t_{V,k}^S$ is the time delay error between the receiver and the satellite

- $\varepsilon_{c,k}$ is the correlated error such as ephemeris and atmospheric error
- $\varepsilon_{uv,k}$ is the uncorrelated error such as multipath error, difficult to model

Thus, the difference between pseudo-ranges can be computed as in (4), where common errors are eliminated [16], while uncorrelated ones are increasing.

$$\mathbf{S}_{ab,k}^S = \rho_{a,k}^S - \rho_{b,k}^S = \Delta R_{ab,k}^S + \Delta t_{ab,k} + \Delta \varepsilon_{ab,k}^S \quad (4)$$

where $\Delta R_{ab,k}^S$ is the difference among true distance of agent a and b and satellite S .

The relative distance between the receiver and the satellite is much higher than distance between the two agents [16], thus it is possible to estimate the pseudo-range distance through (5):

$$\Delta R_{ab,k}^S = [n^S]^T \cdot \mathbf{d}_{ab,k} \quad (5)$$

where n^S is the unit vector from the target agent a and satellite S . In a real scenario, many satellites are located in between two agents. In such a case, (4) becomes (6).

$$\begin{bmatrix} \mathbf{S}_{ab,k}^S \\ \mathbf{S}_{ab,k}^r \\ \vdots \end{bmatrix} \approx \begin{bmatrix} n^S & 1 \\ n^r & 1 \\ \vdots & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{d}_{ab,k} \\ \Delta t_{ab,k} \end{bmatrix} \quad (6)$$

From (6), it can be extrapolated the estimated value of $\mathbf{d}_{ab,k}$ [16].

Double Differencing (DD) Distance. In addition to SD, we can also take the advantage of multiple satellites measurements to achieve the IVD estimation. Indeed, DD-method combines pseudo-range information from two satellites S and R , as depicted in Figure 2.8. Both agents a and b can see all satellites simultaneously [16].

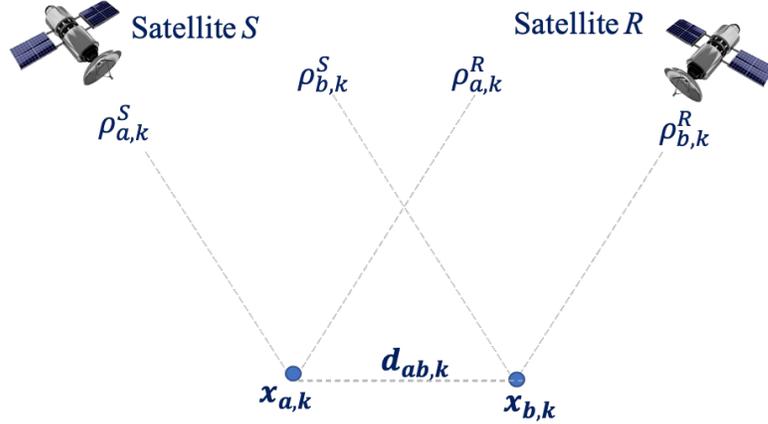


Figure 2.8 Double Differencing method

Difference between pseudo-ranges has been evaluated in (7):

$$\mathbf{D}_{ab,k}^{SR} = \mathbf{S}_{ab,k}^S - \mathbf{S}_{ab,k}^R = \Delta r_{ab,k}^S + \varepsilon_{ab,k}^{SR} \quad (7)$$

where:

$$\varepsilon_{ab,k}^{SR} = \Delta \varepsilon_{ab,k}^S - \Delta \varepsilon_{ab,k}^R \quad (8)$$

$$\Delta r_{ab,k}^S = \Delta R_{ab,k}^S - \Delta R_{ab,k}^R = [n^S - n^R]^T \cdot \mathbf{d}_{ab,k} \quad (9)$$

Taking one satellite O as reference, the double difference matrix is computed in (10):

$$\begin{bmatrix} \mathbf{D}_{ab,k}^{SR} \\ \mathbf{D}_{ab,k}^{SR} \\ \vdots \end{bmatrix} \approx \begin{bmatrix} n^S - n^O & 1 \\ n^R - n^O & 1 \\ \vdots & 1 \end{bmatrix} \cdot [\mathbf{d}_{ab,k}] \quad (10)$$

Solving (10), it is possible to evaluate inter-vehicle distance [16].

2.4 Cooperative Integrity Monitoring Unit (CIMU)

Both Single Differencing and Double Differencing give as result an estimate of the baseline vector, that is the inter-vehicle distance. Instead, GNSS receiver provides an estimate of vehicle position's vector. These two vectors can be expressed by three

coordinates with reference to the worldwide coordinate system [x y z]. Typically, the z-coordinate measured by a GNSS receiver is subject to major errors. In addition to that, to reduce the complexity of the problem, in this work measurements are considered as two-dimensions vectors. A general positioning vector can be considered as two-dimension variable whose characteristic parameters are:

- Mean along x and y : m_x and m_y respectively
- Variance along x and y : σ_x^2 and σ_y^2 respectively
- Covariance: $\sigma_{xy} = \sigma_{yx}$

Variance and covariance are also called second grade moments and they can be also considered as elements of a symmetric square matrix called variance-covariance matrix:

$$\mathbf{P} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{yx} & \sigma_y^2 \end{bmatrix} \quad (11)$$

Considering position vectors as two-dimensions variable of normal type, their distribution density $f(x, y)$ can be represented by a bell-shaped surface, centered at $x = m_x$ and $y = m_y$, [17]. In Figure 2.9, a multivariate distribution function has been represented, whose characteristics are the following:

- $m_x = 0$
- $m_y = 0$
- $\mathbf{P} = \begin{bmatrix} 0.5 & 0.3 \\ 0.3 & 1 \end{bmatrix}$

The intersections of a different planes parallel to the x-y plane with the surface of the frequency distribution of the two-dimensional variable are ellipses, as illustrated in Figure 2.9. The equation of these ellipses can be found by setting $f(x, y) = \text{constants}$.

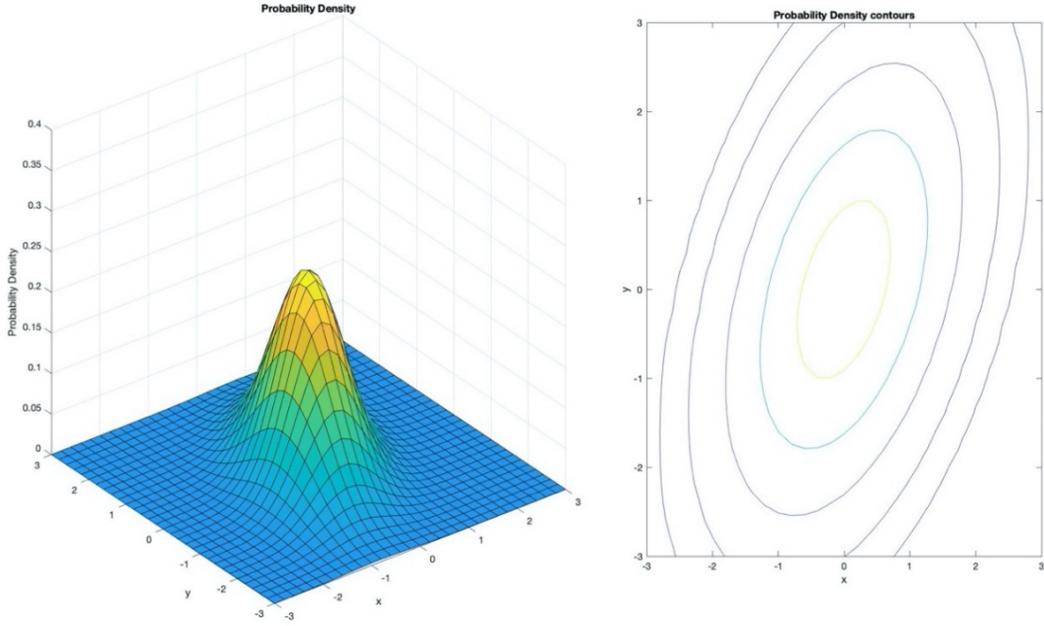


Figure 2.9 Bell-shaped surface representing distribution density and its contours

The inclination angle between ellipses principal axis and x-axis is found through (12), while semi-axes lengths of desired ellipse have been evaluated as eigenvalues of covariance matrix \mathbf{P} . Last important parameter to describe an ellipse is center location, which corresponds to position vectors' estimate state.

$$\tan 2\varphi = \frac{2\sigma_{xy}}{\sigma_x^2 - \sigma_y^2} \quad (12)$$

Applying the theory described above, it has been possible to associate an ellipse to each covariance estimation. As illustrated in the scheme in Figure 2.1, both inter-vehicle distance and GNSS receivers' position vectors' estimates' state and covariance represent the inputs of the Cooperative integrity Monitoring Unit. Such inputs have been represented in Figure 2.10.

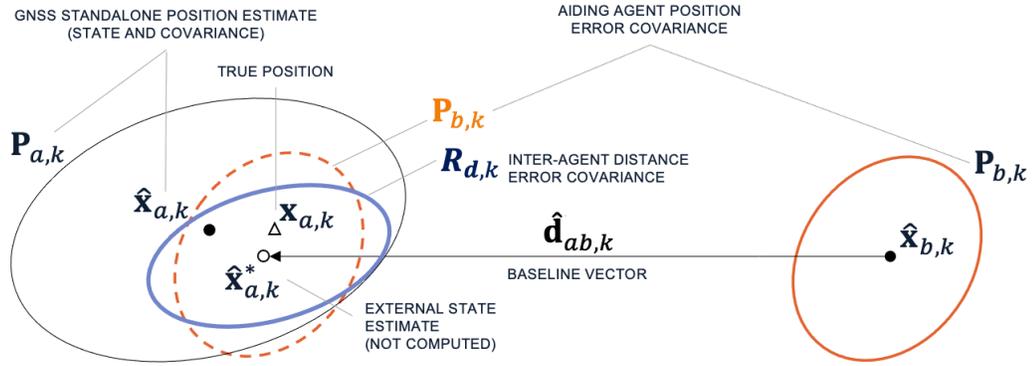


Figure 2.10 IVD and GNSS position vectors estimates' state and covariance

Instead, the exiting result of cooperative unit, which is one of the final objectives of this thesis, is the evaluation of integrity of target agent position solution. All position estimates have been exchanged among different receivers. Each collaborative agent can give an external estimate of target agent position to be integrated to original local measurement. Thus, an aiding agent b can collaborate to validate GNSS measurement provided for a target agent a . With such validation made, additional information has been added to local measurement and the original condition has been updated and improved. In one hand, it is intuitive that any bias affecting the state estimates of anchor agent b negatively affect state estimation of a . In the other hand, “when the aiding agent can rely on higher precision and accuracy of its estimates, a pairwise cooperation could considerably improve the performance of the aided/target agent” [18]. Assuming that an independent noise distribution is present on position and inter-vehicle distance measurements, it is possible to sum estimate covariances of the baseline vector $R_{d,k}$ and of aiding agent positioning vector $P_{b,k}$, to provide a more accurate error model. The result will lead to $R_{d,k}^*$, which is

the overall inter-agent distance covariance. It is equivalent to the external estimate of the target agent uncertainty. In addition, also the overall external estimate state has been found as difference among $\mathbf{x}_{b,k}$ and $\mathbf{d}_{ab,k}$, leading to $\mathbf{x}_{a,k}^*$. Both $\mathbf{R}_{d,k}^*$ and $\mathbf{x}_{a,k}^*$ says how anchor agent b “sees” the target agent a . Its representation is depicted in Figure 2.11.

2.4.1 Covariance Intersection

In the 1990’s, Jeffrey Uhlman, Simon Julier and their associates began promoting a data fusion technique termed "Covariance intersection (CI)" [19]. The idea is that, with V2V communication system, data is collected and processed locally on different receivers. Each node communicates its locally processed data to other nodes, allowing a fusion of local estimates. Nodes are generally intended to operate independently, but processed data are not independent of each other [20] [21]. CI technique has been applied to the local and external estimate covariances, $\mathbf{P}_{a,k}$ and $\mathbf{R}_{d,k}^*$ respectively, as illustrated in Figure 2.11. Therefore, the problem concerns the intersection between two ellipses.

Integrity of the position estimate state and covariance made by the GNSS receiver for target agent a can be verified by an aiding agent b with a certain confidence level expressed in percentage T (%). More in detail, verifying that the intersection ($\mathbf{R}_{d,k}^* \cap \mathbf{P}_{a,k}$) is larger than Confidence Level T (%), it means that integrity of local GNSS solution is actively verified by the external agent: $(\mathbf{R}_{d,k}^* \cap \mathbf{P}_{a,k}) > T(\%)$.

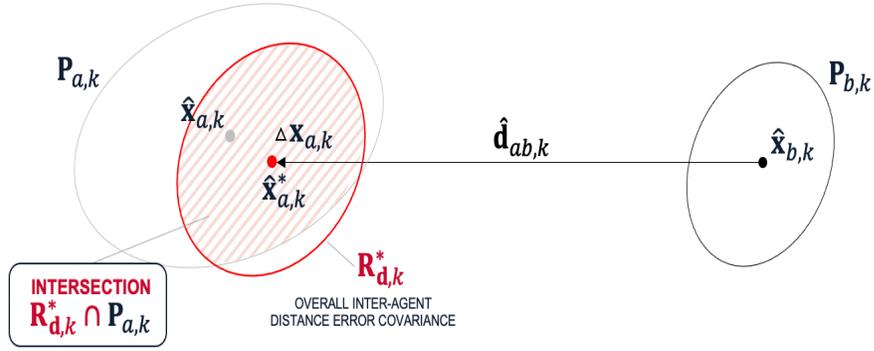


Figure 2.11 Overall inter-agent distance error covariance intersected with target agent position vector covariance

2.5 Cooperative Integrity Monitoring Application

As illustrated in section 2.4, given positions estimate state and covariance, error ellipses can be drawn. Their intersection state can provide useful information about integrity of GNSS solution. Essential parameters to define an ellipse are:

- Semi-axes lengths: a and b , respectively along x -axis and y -axis
- Inclination angle: φ
- Center location: (C_x, C_y)

In general, an ellipse is defined as the locus of points that satisfy parametrical equations (13) and (14).

$$x = a \cos \beta \cos \varphi - b \sin \beta \sin \varphi + C_x \quad (13)$$

$$y = a \cos \beta \sin \varphi + b \sin \beta \cos \varphi + C_y \quad (14)$$

where $0 \leq \beta \leq 2\pi$. Once ellipses are defined, it is possible to verify if there are any intersection points among them and, in positive case, to evaluate the overlapping area and the confidence level percentage. In this work, a robust algorithm that allows to retrieve

the mentioned quantities has been developed. In what follows, an easier explanation of its logic will be presented. In the discussion, the ellipses are named E_1 and E_2 . The first thing to verify is the presence of intersection points. Different cases can result:

1) *No intersection Points*

The case in which no intersection points can result either if one ellipse is strictly contained in the other, or if the ellipses are separated. These two subcases are depicted in Figure 2.12.

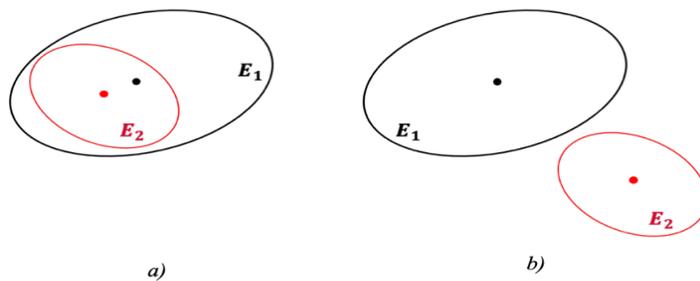


Figure 2.12 No intersection points: a) one ellipse is contained in the other b) two ellipses are separated

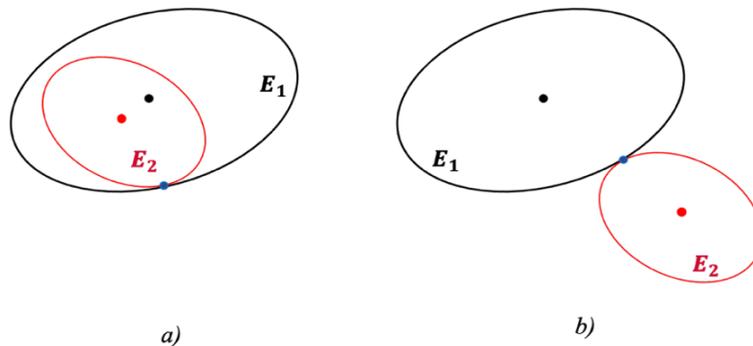


Figure 2.13 One intersection point: a) one ellipse is contained in the other b) two ellipses are separated

```

IF (Intersection points  $\leq$  1)

// Comparison between dimensions of the two ellipses allows to
understand which one can be contained in the other

    IF E1 is bigger than E2

        IF C2 is inside E1 //if center of E2 is inside E1

            THEN E2 is contained in E1 and Overlapping Area = Area of E2

        ELSE IF C2 is outside E1

            THEN two ellipses are separated and Overlapping Area = 0

    ELSE IF E1 is smaller than E2

        IF C1 is inside E2 //if center of E1 is inside E2

            THEN E1 is contained in E2 and Overlapping Area = Area of E1

        ELSE IF C1 is outside E2

            THEN two ellipses are separated and Overlapping Area = 0

```

Figure 2.14 Pseudo-code finding intersection points in case of 0/1 intersection point

2) *One Intersection Point*

Also, the case of one intersection point can result either if one ellipse is strictly contained in the other, or if the ellipses are separated, but in both cases, they are tangent [22] [23].

Such a situation is depicted in Figure 2.13. The logic of this case is the same as for case 1, and the logic implemented is shown in Figure 2.14.

3) *Two intersection points*

The situation for which the two ellipses intersect at two distinct points R_1 and R_2 is illustrated in Figure 2.15. For the overall goal of determining overlap areas between

ellipses, a useful measure is the area of chord region, also called ellipse segment area, that is the area between a secant line and the ellipse boundary.

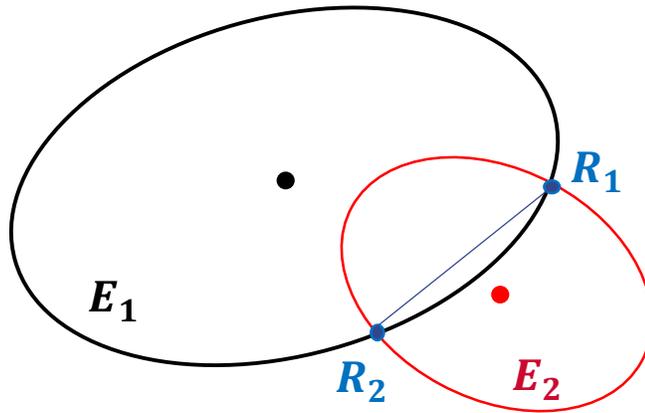


Figure 2.15 Two intersection points case

A secant line drawn between two points on an ellipse partitions the ellipse area into two fractions. The function “*Ellipse Segment Area*” proposed in Figure 2.16 for the evaluation of a segment area works with a standard ellipse, that is an ellipse centered in $(0,0)$ and not-rotated relative to x -axis. Thus, before performing calculation about ellipse segment areas, original intersection points R_1 and R_2 must be rotated and translated according to φ_1 and C_1 for the first ellipse, and to φ_2 and C_2 for the second ellipse, so that they lie on standard ellipses.

```

FUNCTION Ellipse_Segment_Area enters semi-axis lengths a and b of a
standard ellipse and the coordinates of points  $P_1 = (x_1, y_1)$  and  $P_2(x_2, y_2)$ .
It will display  $\theta_1, \theta_2$  and the Segment Area:
IF First point lies on the 1st or 2nd quadrant // ( $y_1 \geq 0$ )
     $\theta_1 = \cos^{-1} \left( \frac{x_1}{a} \right)$ 
ELSE //point 1 lies on the 3rd or in 4th quadrant
     $\theta_1 = 2 * \pi - \cos^{-1} \left( \frac{x_1}{a} \right)$ 
IF Second Point is in the 1st or 2nd quadrant // ( $y_2 \geq 0$ )
     $\theta_2 = \cos^{-1} \left( \frac{x_2}{a} \right)$ 
ELSE //point 1 is the 3rd or in 4th quadrant
     $\theta_2 = 2 * \pi - \cos^{-1} \left( \frac{x_2}{a} \right)$ 
END
Segment Area =  $\frac{1}{2} (\theta_1 - \theta_2) \cdot a \cdot b \pm$  Triangle Area

```

Figure 2.16 Pseudo-code performing ellipse segment area evaluation

New intersection points have been found in (15):

$$P_{1,i} = rot_i \cdot (R_1 - C_i) \text{ and } P_{2,i} = rot_i \cdot (R_2 - C_i) \quad (15)$$

where

$$rot_i = \begin{bmatrix} \cos(-\varphi_i) & -\sin(-\varphi_i) \\ \sin(-\varphi_i) & \cos(-\varphi_i) \end{bmatrix} \text{ for } i = 1, 2 \quad (16)$$

Therefore, new ellipse segment areas have been calculated from $P_1(x_1, y_1)$ to $P_2(x_2, y_2)$ always travelling along the ellipse in a counterclockwise direction from P_1 to P_2 . Chord region is drawn in blue in Figure 2.17, from which segment area can be evaluated as the difference of two areas: the ellipse sector area and the grey triangle area defined by the two points P_1 and P_2 and the ellipse center at the origin [24].

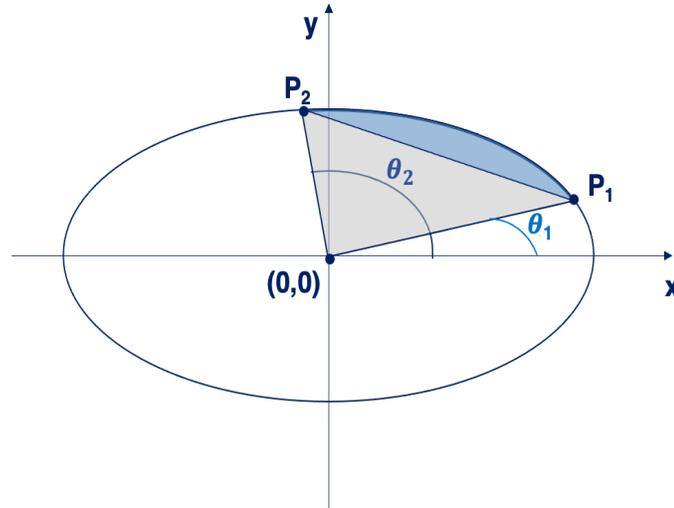


Figure 2.17 Ellipse Segment Area evaluated subtracting triangle area to sector area

Sector area is the area swept by a vector that begins at ellipse center and ends on the ellipse curve, starting the sweep at first point P_1 and travelling along the ellipse in a counterclockwise direction until second point P_2 . Sector area can be evaluated as in (17)

$$Sector\ Area = \frac{(\theta_2 - \theta_1) \cdot a \cdot b}{2} \quad (17)$$

However, in another case, as the one depicted in Figure 2.18, triangle area should be added instead of subtracted for ellipse segment area calculation. The key difference between the cases in Figure. 2.17 and 2.18 is the size of the integration angle. If the integration angle is less than π , then the triangle area must be subtracted from the sector area to give the segment area. If the integration angle is greater than π , the triangle area must be added to the sector area [24]. Thus, the function in Figure 2.16 enters rotated and translated points relative to standard ellipses and gives as output the segment areas always travelling in counterclockwise direction.

The region of intersection between two ellipses is bounded by two elliptical arcs, one from each ellipse. We must compute the areas bounded by the chord connecting the arc endpoints, one for ellipse E_1 and one for ellipse E_2 .

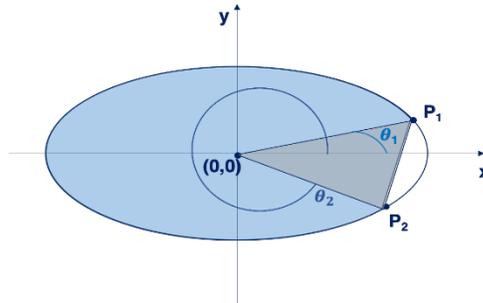


Figure 2.18 Ellipse Segment Area evaluated adding triangle area to sector area

The chord $\langle R_1 R_2 \rangle$ in Figure 2.15 partitions an ellipse into two subsets. New intersection points rotated and translated relative to the original ones, P_1 and P_2 must be passed to function “*Ellipse Segment Area*”, in the order that it will return the correct segment from each ellipse [24]. Only one area from each ellipse contributes to the overlap area. It is obvious that, for one ellipse taking P_1 and P_2 as first and second point respectively is a good choice, while for the other ellipse, this order must be reversed. The method proposed for the evaluation of the points ordering consists of finding an intermediate point $P_{mid} (x_{mid}, y_{mid})$ between $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ when travelling counterclockwise from P_1 to P_2 , which lies on the first ellipse E_1 [24]. Its coordinated are calculated in (18) and (19).

$$x_{mid} = a_1 \cos \frac{\theta_1 + \theta_2}{2} \quad (18)$$

$$y_{mid} = b_1 \cos \frac{\theta_1 + \theta_2}{2} \quad (19)$$

Once evaluated P_{mid} , the logic to choose the right ordering is well explained in Figure 2.19.

4) *Three and Four Intersection Points*

The situation for which the two ellipses intersect at three and four and distinct points is illustrated in Figure 2.20.a and 2.20.b respectively. In subcase a, the region of intersection is the union of a triangle, whose vertexes are R_1, R_2 and R_3 , and three chord regions. Instead, in subcase b, the region of intersection is the union of a convex quadrilateral, whose vertexes are R_1, R_2, R_3 and R_4 , and four chord regions. Each chord region is bounded by an elliptical arc and the line segment connecting the endpoints of the arc. In subcase a, the region of intersection is the union of a triangle, whose vertexes are R_1, R_2 and R_3 , and three chord regions.

As in previous case, function “*Ellipse Segment Area*” need intersection points to be rotated and translated according to both ellipses. Once new intersection points have been found, the key point is to find the order for each chord so that the right segment area is evaluated. More in detail, in the function proposed in Figure 2.16, a couple of points enters always in the same order, and the segment area has been evaluated travelling from the first point to the second point in a counterclockwise direction. Thus, for each chord endpoints, we must understand whenever to evaluate the segment area associated to the first or to the second ellipse [24]. The method proposed in case of 2 intersection points can be reported here, making some changes.

```

IF (Intersection points = 2)
    IF  $(x_{mid}, y_{mid})$  is inside E2
        P1 and P2 enters as 1st and 2nd point in Ellipse_Segment_Area for E1
    →Segment Area 1 evaluation
        P2 and P1 enters as 1st and 2nd point in Ellipse_Segment_Area for E2
    →Segment Area 2 evaluation
    ELSE
        P1 and P2 enters as 1st and 2nd point in Ellipse_Segment_Area for E2
    →Segment Area 1 evaluation
        P2 and P1 enters as 1st and 2nd point in Ellipse_Segment_Area for E1
    →Segment Area 2 evaluation
Overlapping Area = Segment Area 1 + Segment Area 2

```

Figure 2.19 Pseudo-code finding right ordering of points in case of 2 intersection points

Accordingly, we need to find an intermediate point $P_{mid,i} (x_{mid,i}, y_{mid,i})$ between each couple of intersection points $P_{i-1}(x_{i-1}, y_{i-1})$ and $P_i(x_i, y_i)$ when travelling in counterclockwise direction, which lies on the first ellipse E_1 .

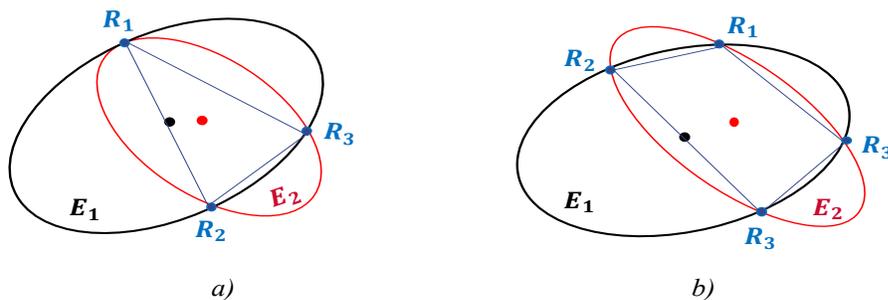


Figure 2.20 a) Three intersection points; b) Four intersection points

The parameter i can be from 1 to 3, or from 1 to 4, depending on number of intersection points. Once evaluated each $P_{mid,i}$, the logic to choose the right ellipse from which extrapolate each segment area and evaluate the total overlapping area is presented in Figure 2.21.

At the end, the confidence level T , given in percentage, has been evaluated in (20) as the ratio among the resulting Overlapping Area and the Area of a reference ellipse.

$$T(\%) = \frac{\text{Overlapping Area}}{\text{Reference Ellipse Area}} \cdot 100 \quad (20)$$

The reference ellipse between the two available is chosen, case by case, based on a dimension comparison

```

IF (Intersection points  $\geq 2$ )
  FOR each  $P_{mid,i} = (x_{mid,i}, y_{mid,i})$  //  $\underline{i} = [1,2,3]$  or  $\underline{i} = [1,2,3,4]$ 
    IF  $(x_{mid,i}, y_{mid,i})$  is inside E2
       $P_{i-1}$  and  $P_i$  enters as 1st and 2nd point in Ellipse_Segment_Area for E1
      →Segment Area ( $\underline{i}$ ) evaluation
    ELSE
       $P_{i-1}$  and  $P_i$  enters as 1st and 2nd point in Ellipse_Segment_Area for E2
      →Segment Area ( $\underline{i}$ ) evaluation
  Total Chord Area = Sum of all computed Segment Areas
  Overlapping Area= Total Chord Area + Area of polygon having intersection
  points as vertexes

```

Figure 2.21 Pseudo-code finding right ordering of points in case of 3 or 4 intersection points

Having uncertainty associated to smaller error ellipse means to retrieve a more accurate solution. Therefore, the chosen reference ellipse is always the smaller one.

As previously said, in this work, real position estimates state and covariances of a set of vehicles, provided by GNSS measurement, is available. Then, taking one vehicle in as target agent, and other vehicles as collaborative agents, a confidence level $T(i)$ associated to each aiding agent external estimate i has been evaluated. To retrieve a final integrity parameter, a weighted average of all confidence levels has been evaluated. The idea, to assign different weights, has been to relate each statistical distance between target and aiding agent i ellipses to the sum of all statistical distances. Statistical distance has been evaluated through Bhattacharyya formula [25], and it is intended as an approximate measure of the amount of overlap between two statistical samples. Therefore, the coefficient can be used to determine the relative proximity of the two samples considered [25]. Providing ellipses' center coordinates, μ_1 and μ_2 , and covariance matrices, Σ_1 and Σ_2 , adopted formula has been reported in (21):

$$d_s = \frac{1}{8}(\mu_1 - \mu_2)^T \Sigma^{-1}(\mu_1 - \mu_2) + \frac{1}{2} \ln \left(\frac{\det \Sigma}{\sqrt{\det \Sigma_1 \det \Sigma_2}} \right) \quad (21)$$

where:

$$\Sigma = \frac{\Sigma_1 + \Sigma_2}{2} \quad (22)$$

Thus, overall integrity parameter is evaluated as in (23):

$$\mathbf{Integrity}(\%) = \frac{\sum_i d_s(i) \cdot T(i)}{\sum_i d_s(i)} \quad (23)$$

The full logic of the entire proposed algorithm is well explained in block scheme in Figure 2.22

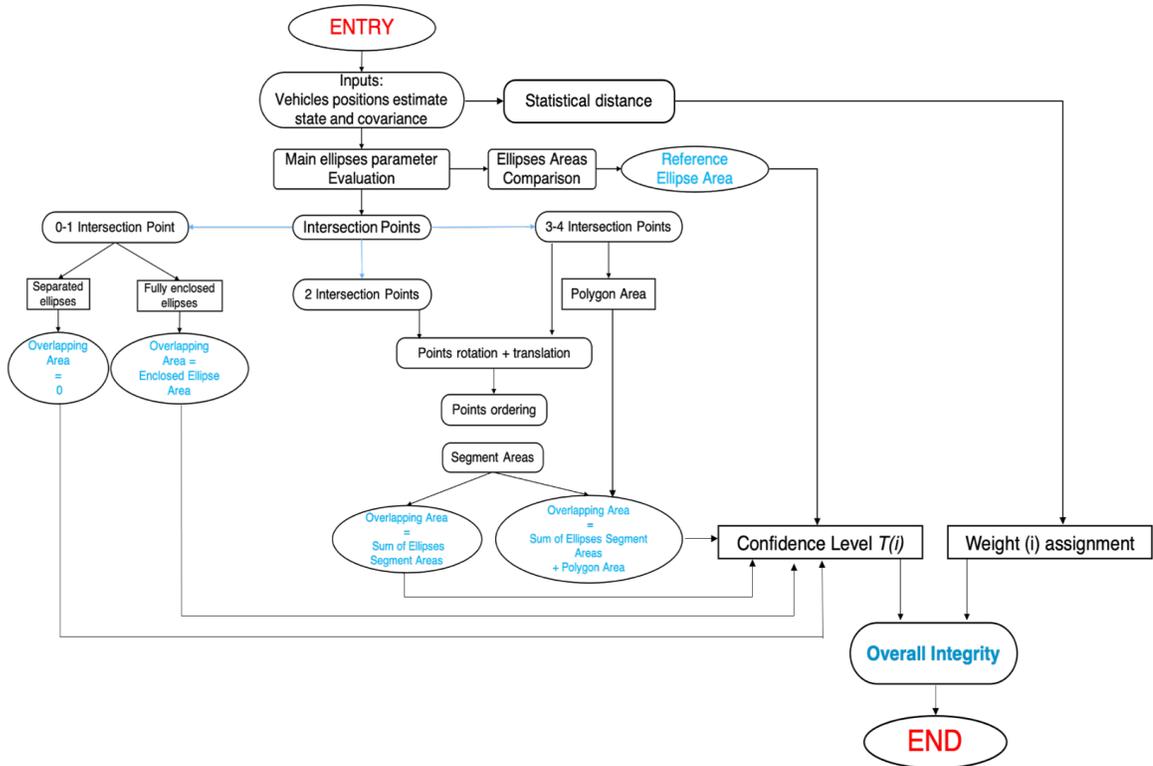


Figure 2.22 Integrity monitoring algorithm' full logic scheme

CHAPTER THREE

RESULTS

3.1 Testing Cooperative Integrity Monitoring Application

In this section, the proposed Cooperative Integrity Monitoring (CIM) algorithm is tested through a prepared dataset. The test involves 10 vehicles: 1 target agent and 9 aiding collaborative agents, which trajectories have been observed for a total time of 10000 s. The available dataset provides the following quantities for each epoch:

- Vehicles' position estimates state, relative to the ENU coordinate system. To work with 2-dimensional vectors, as explained in Chapter 2, Up (U) coordinate has been suppressed
- Covariance matrices of uncertainty about vehicles' position estimate
- Baseline vectors, all relative to target agent. Thus, first value assumed is zero, that is the baseline vector of target agent relative to itself
- Covariance matrices of uncertainty about baseline vectors

3.1.1 Data Preparation

Information about the way in which quantities are retrieved is given in what follows:

- Trajectories of involved vehicles are real and include dynamics occurred in the urban context of Pinerolo (TO). They have been estimated by a ruggedized multi-band, multi- constellation RTK GNSS receiver (Swift Piksi DURO) installed on each vehicle. They have been obtained with a centimeter accurate GNSS + INS Solution + RTK Corrections

- Baseline vectors are analytically simulated from vehicles positions retrieved by the GNSS receiver. Until now, it has been no possible to obtain real inter-vehicles distances based on available GNSS observations, but it can be assumed that they have been obtained through the double difference ranging technique, described in 2.3.1.
- Covariance matrices, both relative to uncertainty of GNSS position estimates and of baseline vectors are analytically simulated, because there was no way to retrieve them from Swift receiver. These covariance matrices have been created starting from eigenvectors whose magnitude is normally distributed with mean equal to 5, variance equal to 3 and a random direction (matrix orientation). Obviously, real covariance matrices are not always about this dimension, they could be even greater or smaller depending on GNSS receiver condition. However, they can be considered as realistic due to their dimension in terms of eigenvectors' magnitude, considering that a code based GNSS positioning has uncertainties above the meter.
- Even though baseline vectors' coordinates and covariance are no real, no loss of generality in application testing is achieved, knowing how errors are usually distributed on these quantities in the realty. Indeed, it has been possible to analytically generate errors so that they look like those normally observed. More details about that are summarized in Table 3.1.

As explained in Section 2.4, CIMU integrates collaborative agents position vectors with baseline vectors, finding in this way an overall inter-agent distance estimate.

Table 3.1 Baseline Vectors errors generation along each trajectory section

Trajectory sections (s)	0÷2500	2501÷5000	5001÷7500	7501÷10000
Error randomly added with non-zero mean		Mean: 5 m	Mean: 5m	Mean: 15 m
Error randomly taken from covariance matrices with varying variance dimension	Small Variance	Small Variance	High Variance	High Variance

Therefore, ellipse E_1 will always be associated to target agent's position vector and covariance matrix, while there will be one ellipse E_2 for each collaborative agent associated to following parameters:

- Center location: Difference between collaborative agent's position and baseline vectors
- Covariance Matrix: Sum of collaborative agent's position and baseline covariance matrices. This operation is allowed if there is no-correlation among two matrices. If baseline vectors were evaluated through different method, which no includes GNSS measurements, no-correlation would be sure. Even if a *DD* ranging technique has been used in this work, covariance matrices are assumed to be added together because errors generated during their simulation is independent from each other.

3.1.2 CIM Algorithm Results Analysis

The information retrieved by the available dataset are associated to a total time of 10000 s. In each time instant, CIM application evaluates total integrity parameter following the procedure depicted in Figure 2.22. The overall algorithm implementation has been made using MATLAB Software. To reduce computational time without losing accuracy, integrity of target agent estimate has been verified at positions of its trajectory occurred 25 seconds apart. Therefore, scatter plots with a total of 400 points have been obtained. The first goal is to verify the integrity of GNSS solution for the target agent along all its route. Thus, integrity parameter along each trajectory section has been plotted in Figure 3.1. It is clearly noticeable how this quantity varies along the time. More in detail, mean values for each section are the following:

- First segment [0-2500] s: 77.71 %
- Second segment [2501-5000] s: 43.16 %
- Third segment [5001-7500] s: 61.05 %
- Fourth segment [7501-10000] s: 7.91 %

This trend is due to the way in which errors have been generated, as explained in Table 3.1. Even though errors randomly taken by covariance matrices with small variance have been added in both first and second sections, integrity of GNSS solution is highly verified only in the first one. This result is due to the presence of another error randomly distributed in the second section. Same consideration can be done comparing third and fourth sections. Also in this case, difference stays in the mean of randomly distributed error, which is 5m and 15 m respectively.

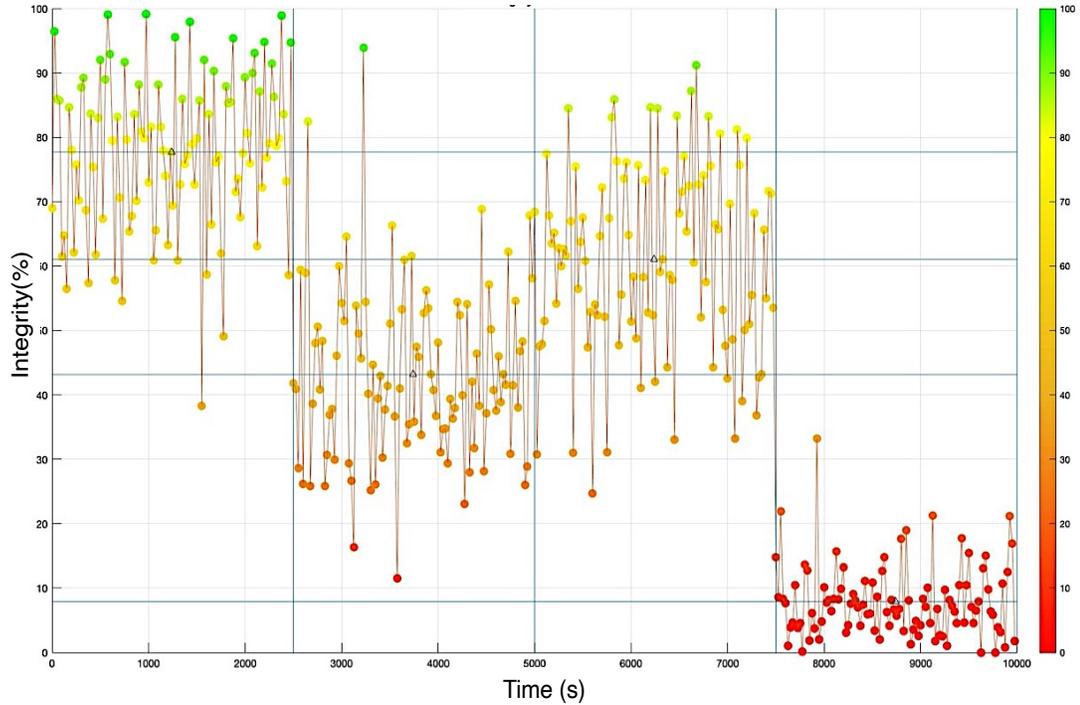


Figure 3.1 Cooperative Integrity Parameter in different target agent trajectory sections

Another observation can be done likening second and third sections. Indeed, mean of random bias is the same, while the variance of baseline vectors' covariance matrices has been increased in third section, resulting in higher uncertainties. To figure out the way in which this augmentation influences intersection states between ellipses, the proposed algorithm has been tested choosing a random instant time along the two inspected trajectory portions. The resulting situations are represented in Figure 3.2 and 3.3, respectively. By increasing the dimension of the baseline vector uncertainties, also the overall inter-agent estimates covariance matrices will be higher. This means to have a more favorable condition for ellipses intersection and a higher mean value of total integrity parameter, as consequence.

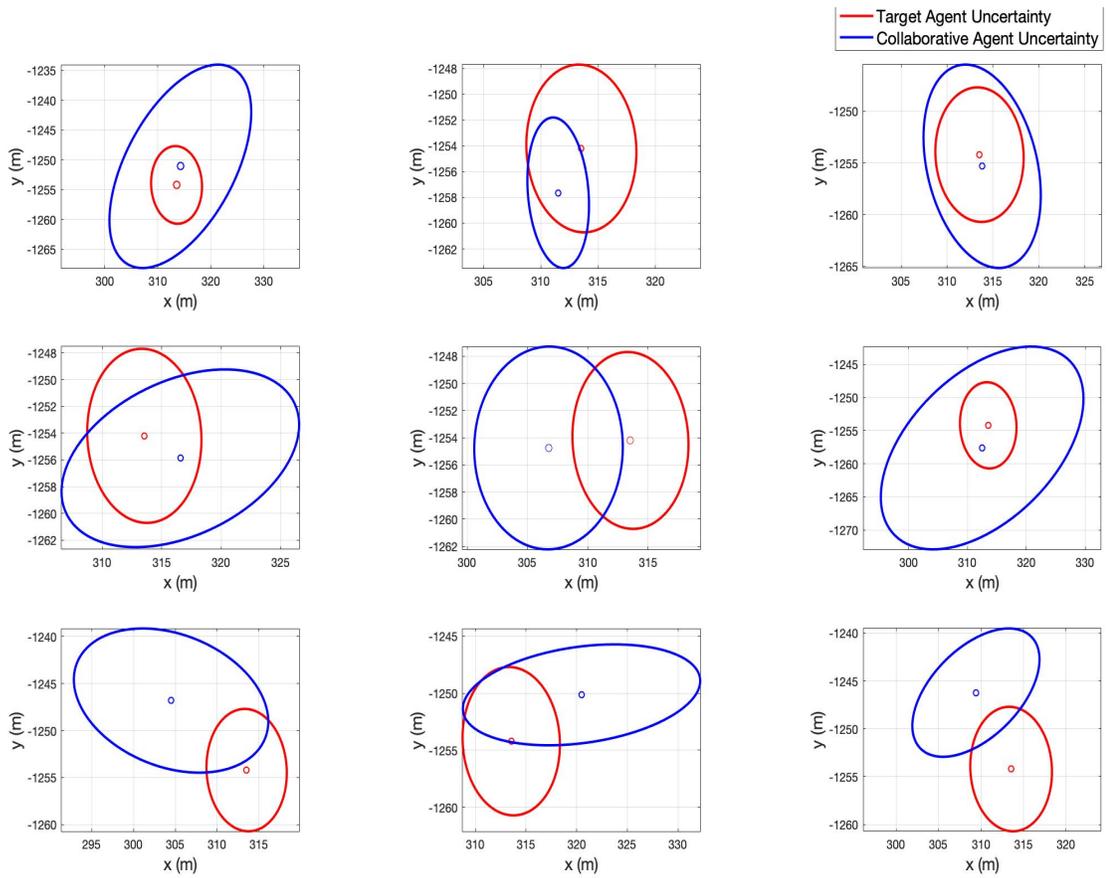


Figure 3.2 Target and Collaborative Agents' Position and Uncertainties in a random instant of time along trajectory section [2501-5000] s

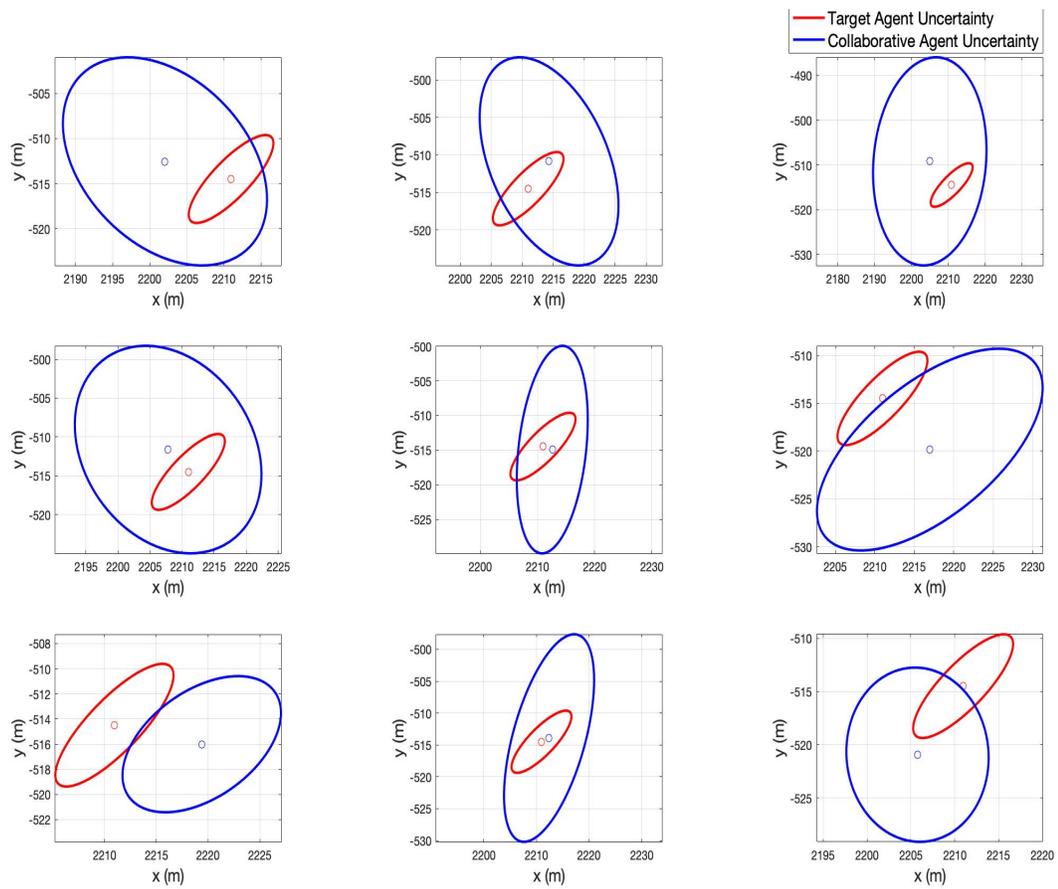


Figure 3.3 Target and Collaborative Agents' Position and Uncertainties in a random instant of time along trajectory section [5001-7500] s

The overall target vehicle route has been represented in Figure 3.4, in which the x and y coordinates of following positions taken by the agent are reported on x axis and y axis, respectively. Every position taken by the agent has been colored on basis of its monitored integrity. Same considerations made above about different path sections can be done. Indeed, GNSS solution integrity is poor in the last segment, where multiple errors typologies have been generated.

3.2 Network Performance Analysis

Adding the new CE protocol in the framework, it is important to evaluate network performance to understand if it could be implemented in some applications. Indeed, the main objective of vehicular framework is to provide safety to users.

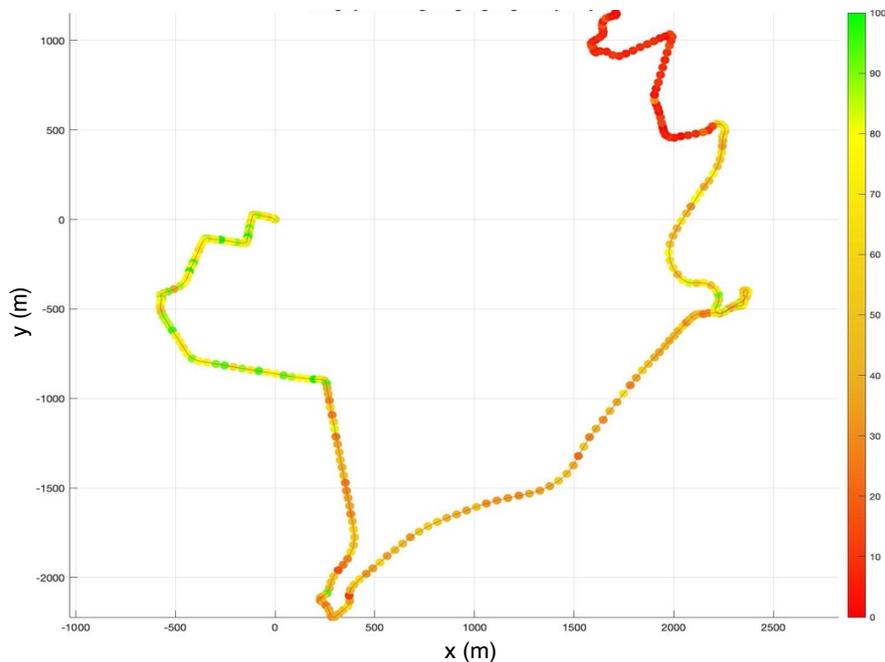


Figure 3.4 Integrity Monitoring along target agent route

Until now, MS-VAN3T framework achieves effective V2V communication by only using CA protocol. The integration of CE protocol stressed the network to take charge of new type of data, which can cause high delays in delivering packets or low utilization of available bandwidth. In this work, among all the measures of performance, results focus on network latency and on traffic offered by CAM and CEM service.

The traffic offered has been obtained by running 19 simulations, each one with a fixed duration of 100 s. Each simulation involves an increasing number of vehicles, starting from 10 to 100, with a step of 5 among them. The map, depicted in Figure 3.5, is the same used for V2V application. The access technology adopted is 802.11p, which allows vehicles OBUs to send both CAMs and CEMs while RSUs to send DENMs. Therefore, all delivered packets must share the channel. The simulations have been carried on by configuring the OBUs to transmit using one of the possible data rates available for 802.11p, which is 12 Mb/s.



Figure 3.5 Simulation map used to test CE protocol combined to CA protocol

Total traffic offered by CAM service is represented in Figure 3.6, in which the increasing number of vehicles involved in the simulation is reported on x axis, while the total offered traffic on y axis. Each displayed number represents a mean value over the elapsed time between the first and the last delivered packet. Thus, for each case, it has been retrieved as in (24).

$$\text{Total traffic offered by CAM service} = \sum \frac{\text{Transmitted CAMs Packets Length}}{t_{lastCAM} - t_{firstCAM}} \quad (24)$$

The same plot can be obtained for CEM service, considering that each value of total traffic has been evaluated as in (25), in a similar way to previous case.

$$\text{Total traffic offered by CEM service} = \sum \frac{\text{Transmitted CEMs Packets Length}}{t_{lastCEM} - t_{firstCEM}} \quad (25)$$

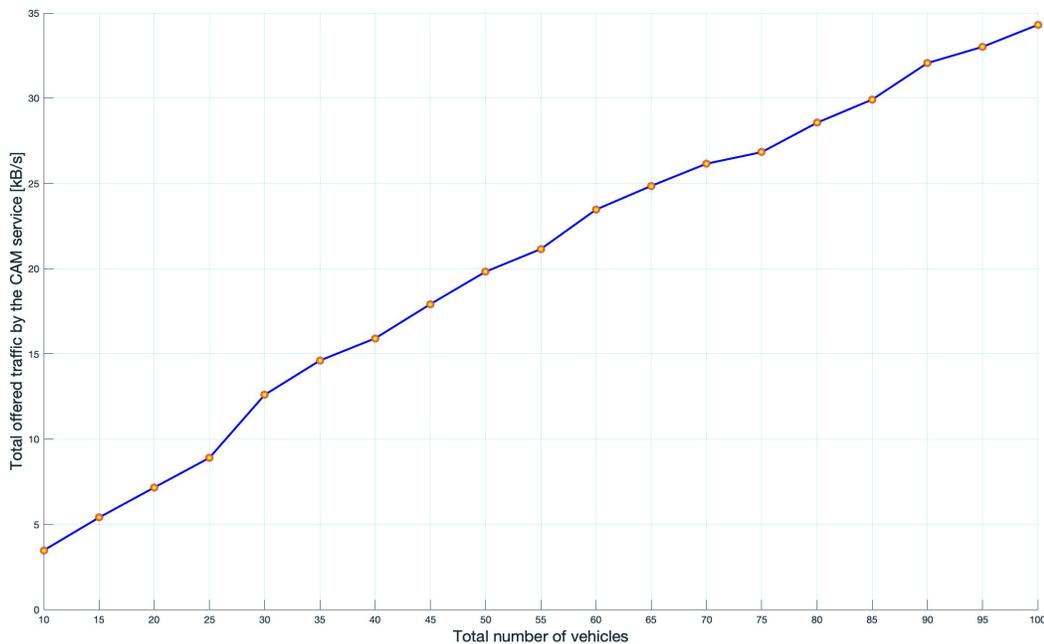


Figure 3.6 V2V simulation result: Total traffic offered by CAM service as function of total number of vehicles present in the scenario

Results of (25) are depicted in Figure 3.7, while both values retrieved by (24) and (25) are collected in Table 3.2. It can be observed that the overall packets traffic, both offered by CAM and CEM service, has an increasing trend. This result is due to the augmentation of transmitted packets over the network when a larger number of vehicles is involved. Thus, network must take charge of more and more data to transmit. Despite trend similarities, traffic offered by CEM service is one order of magnitude higher than one offered by CAM service. The reason is due to difference in packets' length and generation frequency. CEM packet length ranges from almost 400 to 900 bytes, while CAM packet length is about 121 bytes. Furthermore, CAMs packets are generated with an average frequency lower than CEM packets.

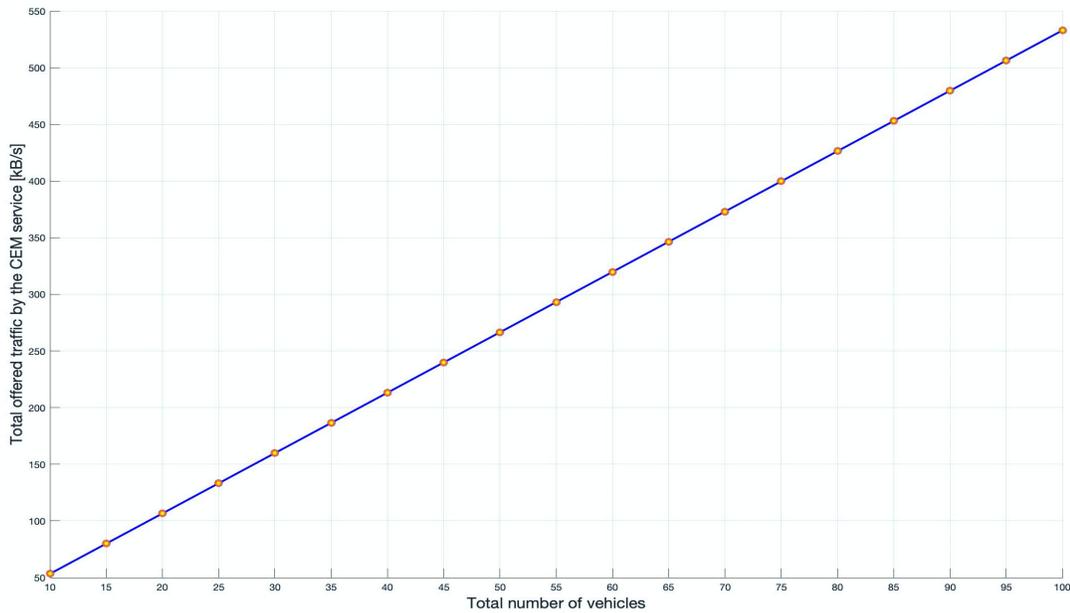


Figure 3.7 V2V simulation result: Total traffic offered by CEM service as function of total number of vehicles present in the scenario

Table 3.2 Total traffic offered by CAM and CEM services increasing the total number of vehicles in the simulation

Total number of vehicles	Total traffic offered by CAM service [kB/s]	Total traffic offered by CEM service [kB/s]
10	3.464	53.297
15	5.391	79.946
20	7.154	106.594
25	8.896	133.243
30	12.592	159.891
35	14.605	186.539
40	15.903	213.188
45	17.911	239.837
50	19.816	266.486
55	21.144	293.134
60	23.461	319.783
65	24.849	346.431
70	26.153	373.079
75	26.833	399.728
80	28.556	426.377
85	29.917	453.0262
90	32.0573	479.675
95	33.012	506.324
100	34.298	532.971

Indeed, CEMs generation frequency is fixed to 10 Hz, while CAMs generation frequency ranges from 1 Hz to 10 Hz, depending on the vehicle dynamics: higher the speed of vehicles, higher the messages generation frequency. Knowing both the total offered traffic in the network and the quantity of vehicles involved in each simulation, the mean offered traffic of each vehicle has been evaluated as their ratio. Mean traffic per vehicle offered by CAM and CEM service at different number of agents in the simulation is depicted in Figure 3.8. On average, CAM service generates 0.373 kB/s. Instead, CEM service produces 5.329 kB/s, requiring a higher bandwidth than CAM one. This result is due to differences concerning packets' lengths and generation frequencies, as explained above. Anyway, bandwidth usage of these services is lower than maximum value of throughput available with the 802.11p access technology.

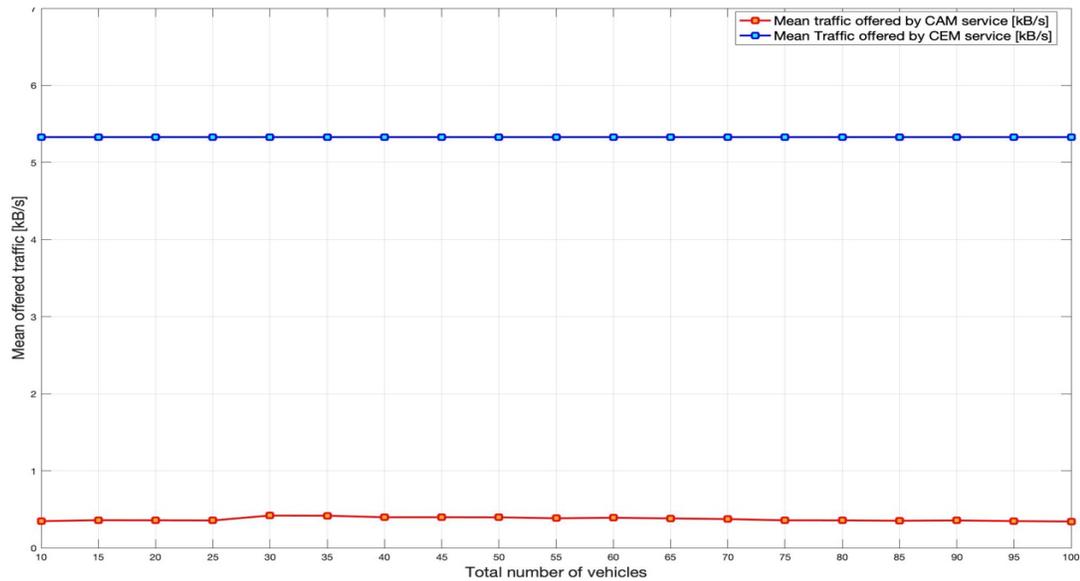
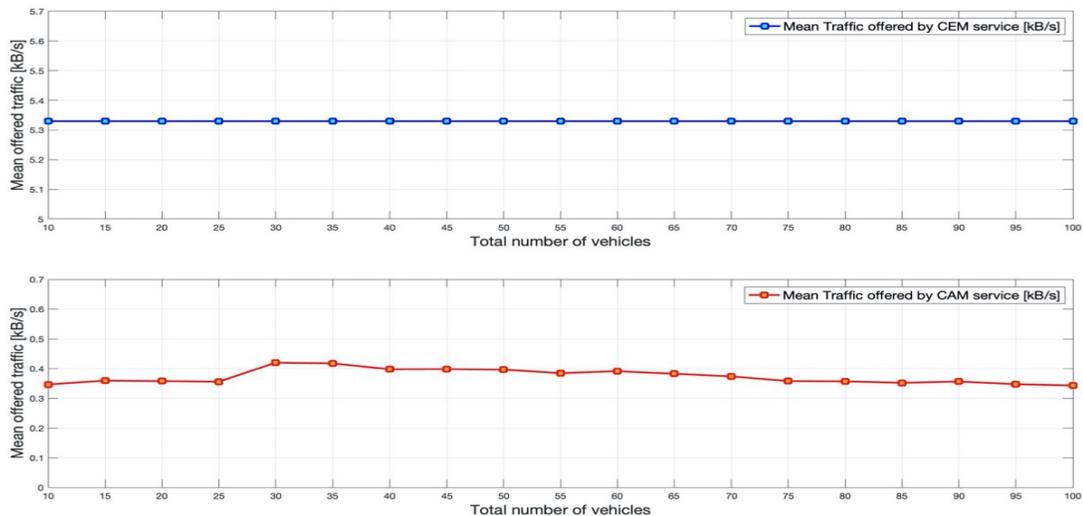


Figure 3.8 V2V simulation result: Mean traffic per vehicle offered by CAM and CEM services as function of total number of vehicles present in the scenario

Increasing the number of agents in the scenario, values retrieved for CAMs transmissions are almost constant but show little fluctuations, while for CEMs they are steady. These trends are more evident in Figure 3.9, where a zoom of two regions, one between 0 kB/s and 0.6 kB/s and another one between 5 kB/s and 5.7 kB/s, is shown. Therefore, each agent sends packets regardless number of vehicles present in its surrounding. More in detail, transmission of CAMs packets depends on many factors for which it could vary, even if only slightly. Instead, CEMs messages are sent at a fixed frequency, independently on number of vehicles in the scenario.

Another index which has a key impact on the performance of the network is the latency. It is often assumed that high performance comes from large bandwidth, which refers to the network capacity to carry traffic. Having a higher bandwidth means that simultaneous conversations are allowed, but it does not imply fast communications.



3.9 Zoom of Mean Traffic plot in regions [0 ÷ 0.7] kB/s and [5 ÷ 5.7] kB/s

Data latency means time duration between issuing a message from sender until it is received by receiver vehicles. Thus, the latency represents the elapsed time between the moment in which a message is generated by an agent and the one in which it is received by other vehicles in the network. This parameter has been evaluated both for transmission of CAMs and CEMs messages and it has been analyzed by a CSV logging mechanism available into ns-3. The addition of CE protocol to CA one can cause higher delay in transmission of CAMs messages due to the increasing of packets which must share the channel. The impact of CE protocol has been evaluated through two simulations lasting 100 s each, in which 10 vehicles are present in the same scenario illustrated in Figure 3.5. In the first case, only CAMs messages dissemination is allowed and no CEMs packets have been sent. In the second case, also CEMs messages transmission is permitted. Histograms representing the frequency of one-way delays for CAM messages in both cases are depicted in Figure 3.10 and Figure 3.11, respectively.

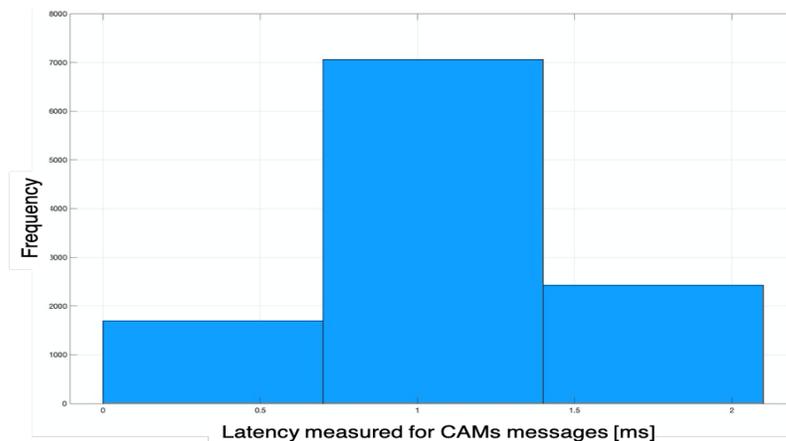


Figure 3.10 One-way delay measured for CAMs messages when CEMs dissemination is not allowed

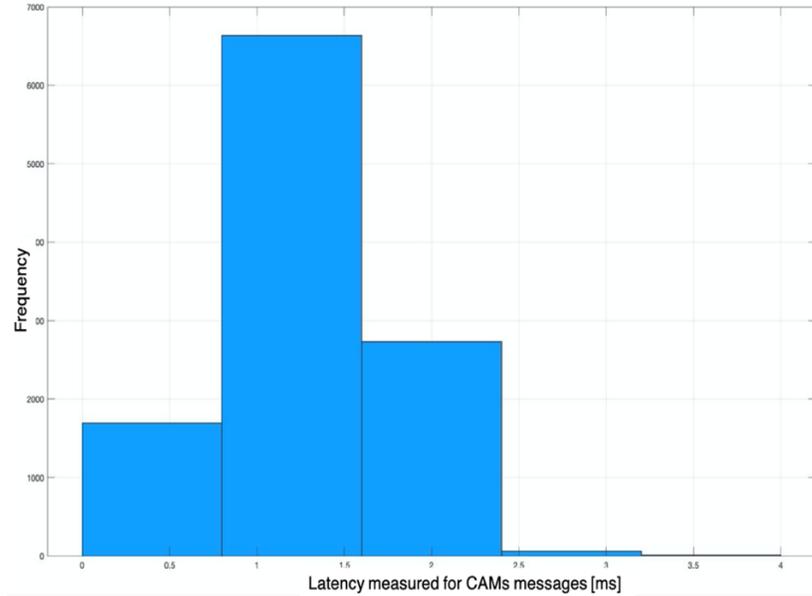


Figure 3.11 One-way delay measured for CAMs messages when CEMs dissemination is allowed

It can be observed that a delay ranging from 0.7 ms and 1.4 ms has higher probability to be experienced in the case in which only CAMs packets are sent. Instead, in case in which CE protocol is allowed too, a delay between 0.8 ms and 1.6 ms is the most likely. On average, latencies of 1.07 ms and 1.11 ms are measured in first and second case, respectively. As expected, the addition of CEMs packets transmission causes the latency for CAMS to increase. Nevertheless, this augmentation is almost negligible.

Also, latency measured for CEMs messages has been inspected and depicted in Figure 3.12. It can be observed that a delay ranging from 1.5 ms and 2.5 ms has the higher probability to be experienced. On average, it is equivalent to 1.73 ms. Therefore, CEMs messages requires more time to be delivered due to their higher dimension.

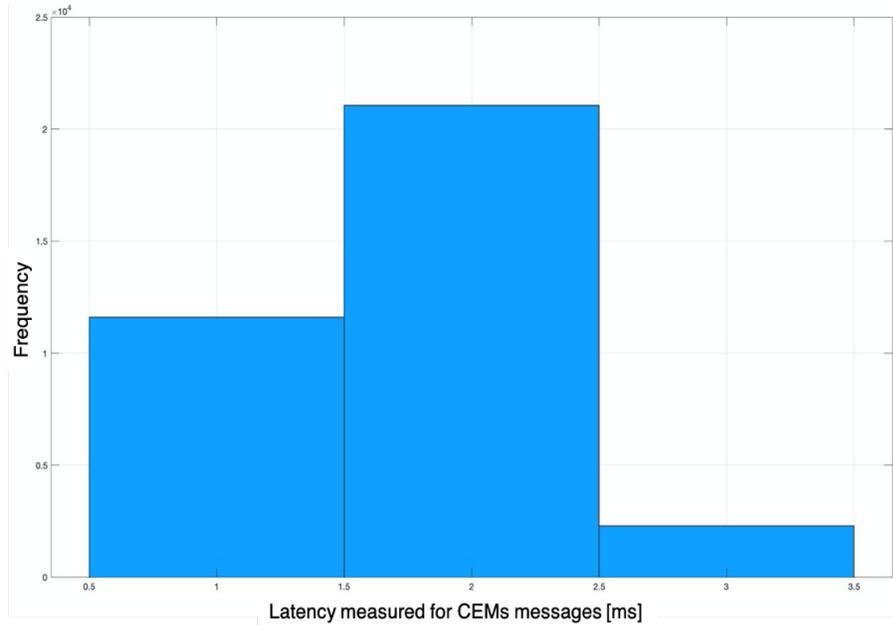


Figure 3.12 One-way delay measured for CEMs messages

The evaluation of latency in case in which more than 10 vehicles are present in the simulated scenario has been addressed. More in detail, ten simulations have been done, each one with several agents involved ranging from 10 to 100, with a step of 10 in between. The level of detail with which data have been collected has a granularity of 1 ms because CAMs can be described with no more than this precision.

Results plotted in Figure 3.13 show the mean values of the end-to-end delays over all vehicles on x axis, with the total number of vehicles in the simulation on y axis. The first point is that CEMs are transmitted with a higher latency. Indeed, having CEMs packets a larger dimension, their transmission takes longer. Another aspect to underline is that the delay between packets generation and their reception increases, both for CAMs and CEMs, adding more and more vehicles in the simulation.

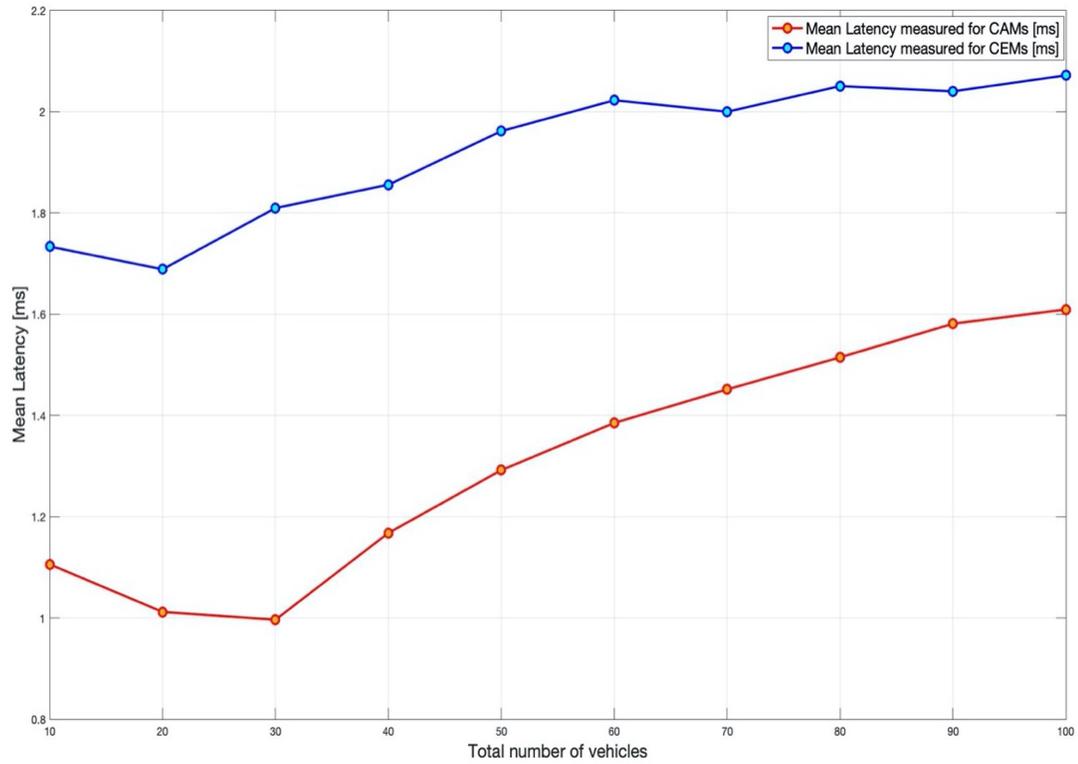


Figure 3.13 Mean Latency over total number of vehicles present in the simulated scenario

This trend is due to the higher wireless channel contention when large number of vehicles, and so large number of packets, must be transmitted.

CHAPTER FOUR

CONCLUSIONS

4.1 Collaborative agents number choice

A question that may arise from the analysis made in section 3.1 could regards the reason why nine collaborative agents have been used to validate GNSS solution for the positioning of the target agent, and how results may vary by changing this number.

Along the overall target agent route, cooperative integrity solution undergoes fluctuations around the different trajectory sections' mean values, as shown in Figure 3.1. Mean and standard deviation of the integrity value over the entire path and over each segment of it can be evaluated increasing the total number of aiding agents from 1 to 9. To better figure out the way in which evaluated quantities change, their trends have been represented in Figure 4.1 and 4.2 respectively.

In Figure 4.1 it can be observed that increasing the number of collaborative agents, the cooperative integrity mean value has an overall decreasing trend over each segment and over the entire trajectory. Choosing nine aiding agents, it has been assumed to be in the worst condition for retrieving the cooperative integrity parameter. Therefore, GNSS solution for the target agent has been verified under more conservative hypotheses.

In Figure 4.2, it can be pointed out that increasing the number of collaborative agents, the cooperative integrity standard deviation monotonically decreases over each segment and over the entire trajectory.

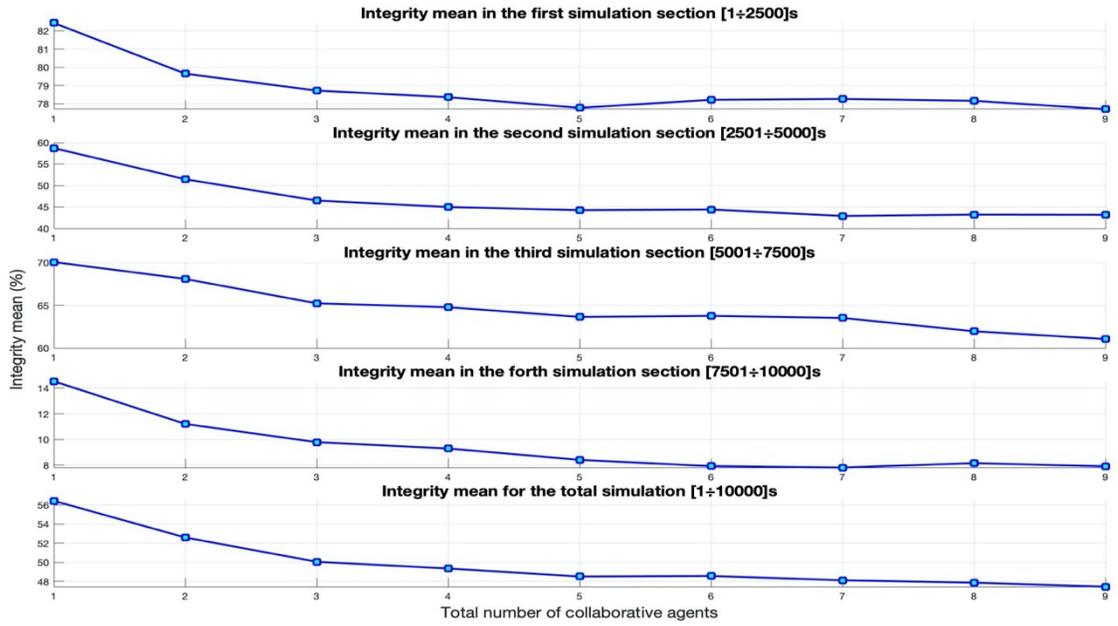


Figure 4.1 Integrity Mean Trend compared to total number of aiding agents

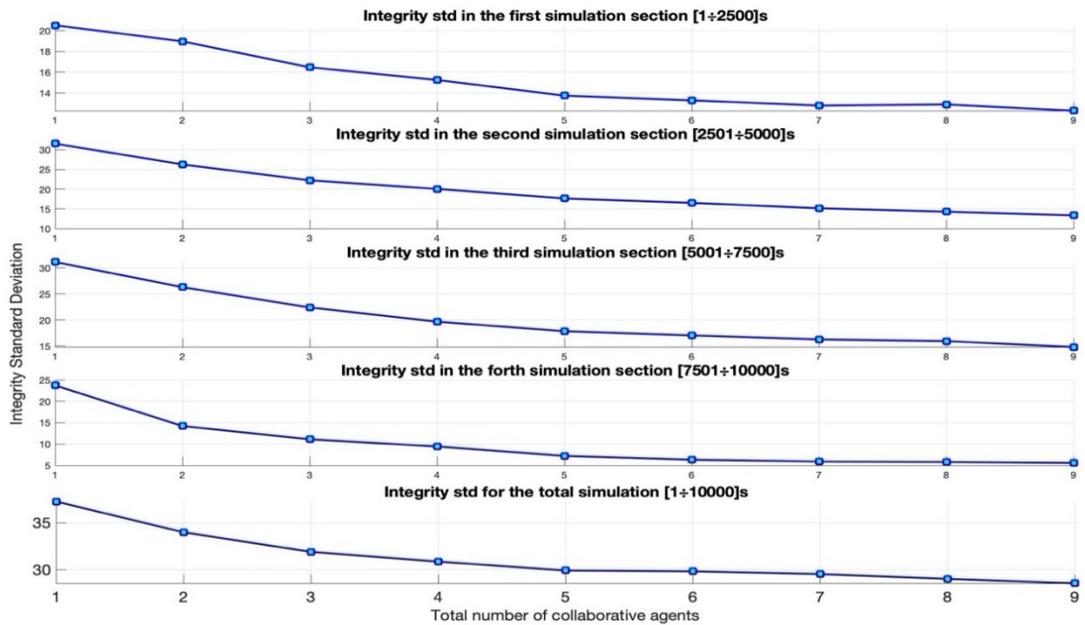


Figure 4.2 Integrity Std Trend compared to total number of aiding agents

Results obtained by this analysis are summarized in Table 4.1 and Table 4.2 respectively. Therefore, choosing a fleet of 10 vehicles, of which one is the target agent, allows to reduce integrity parameter fluctuations. The above analysis suggests that, by further increasing the total number of cooperative vehicles, the integrity estimate may be more and more accurate by inducing the variance of the cooperative integrity parameter to decrease. However, the computational cost for integrity monitoring related to the augmentation of vehicles fleet should be considered.

Table 4.1 Integrity Mean values over the entire trajectory and over each section of it varying total number of collaborative agents.

Number of aiding agents	1 st section Mean (%)	2 nd section Mean (%)	3 rd section Mean (%)	4 th section Mean (%)	Overall route Mean (%)
1	82.431	58.704	70.079	14.534	56.437
2	79.656	51.442	68.101	11.211	52.603
3	78.731	46.479	65.237	9.791	50.056
4	78.368	44.980	64.787	9.299	49.358
5	77.787	44.239	63.643	8.414	48.521
6	78.224	44.385	63.769	7.926	48.576
7	78.273	42.891	63.525	7.818	48.127
8	78.169	43.208	61.957	8.162	47.874
9	77.709	43.162	61.051	7.912	47.459

Table 4.2 Integrity Standard Deviation (Std) values over the entire trajectory and over each section of it varying total number of collaborative agents.

Number of aiding agents	1 st section Std	2 nd section Std	3 rd section Std	4 th section Std	Overall route Std
1	20.523	31.512	31.097	23.727	37.254
2	18.975	26.251	26.279	14.206	33.987
3	16.483	22.227	22.377	11.111	31.909
4	15.262	20.055	19.663	9.424	30.873
5	13.752	17.638	17.826	7.224	29.927
6	13.300	16.532	17.019	6.336	29.837
7	12.799	15.158	16.261	5.889	29.549
8	12.904	14.290	15.927	5.789	29.046
9	12.299	13.369	14.795	5.562	28.568

Indeed, the benefit related to the reduction of integrity parameter variance could not be justified by the increasing computational cost. Thus, final decision has been to use datasets relative to 10 cooperative agents as bound for CIM application.

As future work, a trade-off between the “reliability” of position estimate integrity, and cost of its computation could be established.

4.2 Cooperative Integrity Monitoring: Possible Implementations

The proposed CIM method is based on GNSS cooperative measurements and on the concept of covariance intersection to retrieve final integrity parameter. It can be implemented on fleet of vehicles that are in a certain range of action in which they are able to cooperate among themselves. However, this range could be additionally limited by radio-range used at communication level in the network.

It is important to observe that if all measurements retrieved by GNSS were altered in a similar way for all collaborative agents, the integrity parameter would result “confirmatory”, no adding useful information to GNSS solution. Instead, cooperative integrity monitoring parameter reacts if, for some reason, the position estimate has been altered with respect to what estimated through local GNSS observations and verified by the network of agents. Possible real implementations in which CIM method will react are given in the following:

- A sensor integrated to GNSS generates a bias on the final estimate given by an overall integrated navigation system
- A cyber-attack provides false position estimates to application layer

APPENDIX
COPYRIGHT PERMISSION LETTER

Sara Golisciani, Via De Lauziers 46, San Giorgio a Cremano, NA 80046

[06/30/2021]

Dear Alex Minetto:

I am completing a master's thesis at Oakland University entitled "Collaborative Positioning in low latency vehicular networks: Cooperative Integrity Monitoring." I would like your permission to reprint in my thesis excerpts from the following:

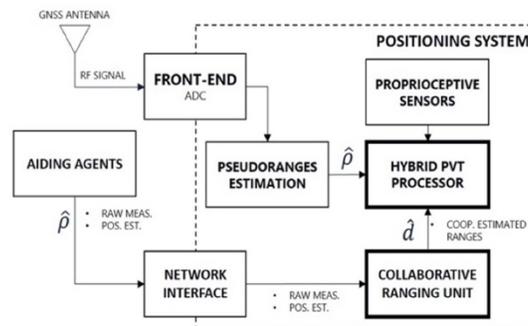


Figure 1.5 High-level block scheme of a networked positioning system including collaborative ranging module

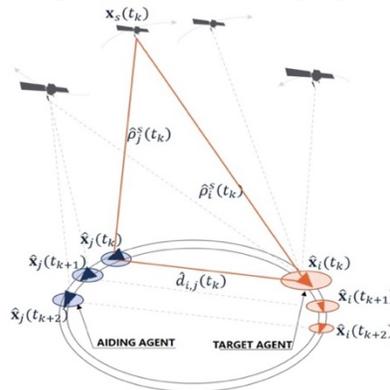


Figure 1.6 Pseudo-range and inter-vehicle range estimation

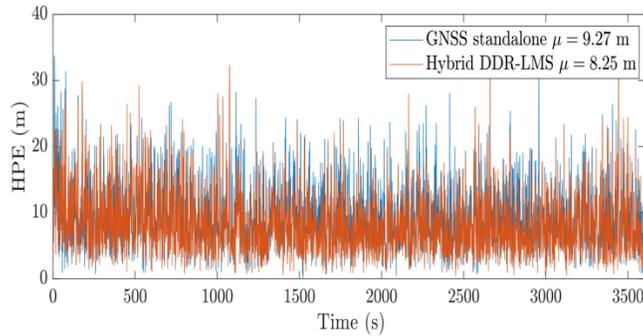


Figure 1.7 Absolute Horizontal Position Error

The excerpts to be reproduced are the Figure 1.5,1.6 and 1.7 used in your article on “GNSS-only Collaborative Positioning Among Connected Vehicles” that I am intending to mention in the introduction of my thesis. The requested permission extends to any future revisions and editions of my thesis, including non-exclusive world rights in all languages. These rights will in no way restrict republication of the material in any other form by you or by others authorized by you. Your signing of this letter will also confirm that you own (or your company owns) the copyright to the above-described material.

If these arrangements meet with your approval, please sign this letter where indicated below and return it to my email address saragolisciani@oakland.edu. Thank you very much.

Sincerely,
Sara Golisciani

PERMISSION GRANTED FOR THE
USE REQUESTED ABOVE:

[Alex Minetto]

Date: 06/30/2021

Sara Golisciani, Via De Lauzieres 45, San Giorgio a Cremano, NA 80046

[07/06/2021]

Dear Marco Malinverno:

I am completing a master's thesis at Oakland University entitled "Collaborative Positioning in low latency vehicular networks: Cooperative Integrity Monitoring." I would like your permission to reprint in my thesis excerpts from the following:



Figure 2.4 Area Speed Advisory application scenario

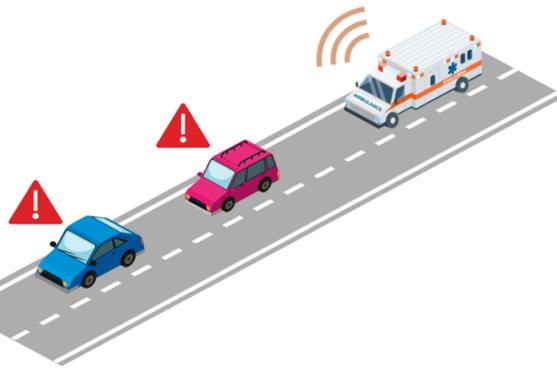


Figure 2.4 Emergency Vehicle Alert application scenario

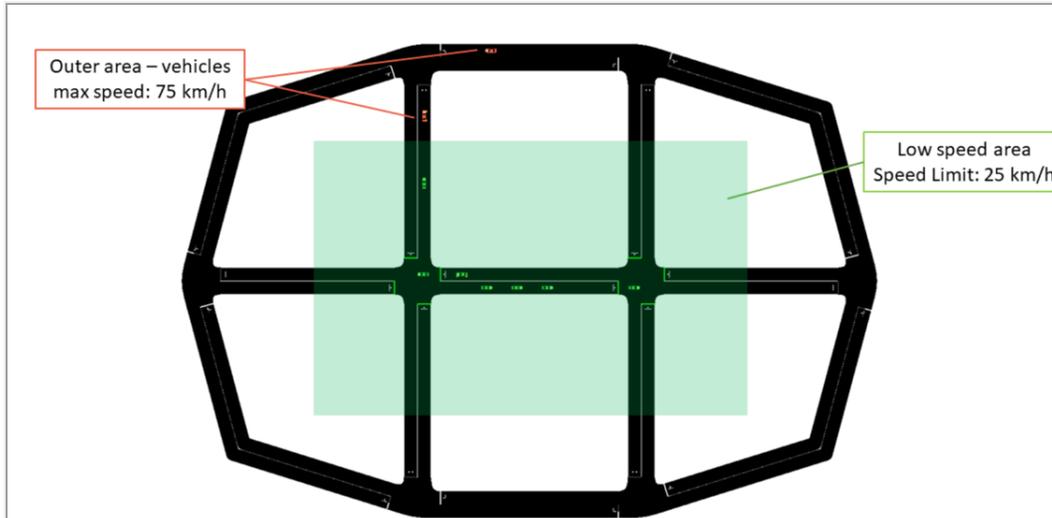


Figure 2.3 Speed areas' subdivision for Area Speed Advisory application

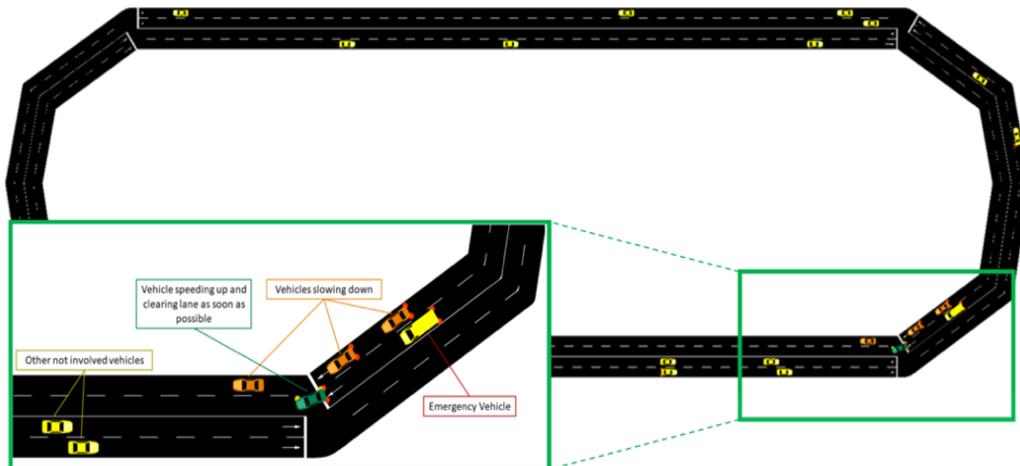


Figure 2.5 Road segment and actions logic in Emergency Vehicle Alert application

The excerpts to be reproduced are the Figure 2.2 and 2.4 used in your work on “Safety Applications and Measurement Tools for Connected Vehicles” and Figure 2.3 and 2.5 used in your article “A Multi-stack Simulation Framework for Vehicular Applications Testing”, that I am intending to mention in the part relative to the Methodology of my Thesis. The requested permission extends to any future revisions and editions of my thesis, including non-exclusive world rights in all languages. These rights will in no way restrict republication of the material in any other form by you or by others authorized by you. Your signing of this letter will also confirm that you own (or your company owns) the copyright to the above-described material.

If these arrangements meet with your approval, please sign this letter where indicated below and return it to my email address saragolisciani@oakland.edu. Thank you very much.

Sincerely,
Sara Golisciani



PERMISSION GRANTED FOR THE
USE REQUESTED ABOVE:



[Marco Malinverno]

Date: 07/06/2021

Permission Copyright Letters

REFERENCES

- [1] N. Zhu, J. Marais, D. Bétaille and M. Berbineau, "GNSS Position Integrity in Urban Environments: A Review of Literature," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2762-2778, Sept. 2018, doi: 10.1109/TITS.2017.2766768.
- [2] N. Alam and A. G. Dempster, "Cooperative Positioning for Vehicular Networks: Facts and Future," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1708-1717, Dec. 2013, doi: 10.1109/TITS.2013.2266339.
- [3] K. Sjoberg, P. Andres, T. Buburuzan and A. Brakemeier, "Cooperative Intelligent Transport Systems in Europe: Current Deployment Status and Outlook," in *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 89-97, June 2017, doi: 10.1109/MVT.2017.2670018.
- [4] Minetto A., Nardin A., & Dervis, F. (2019, July). "GNSS-only collaborative positioning among connected vehicles." In *Proceedings of the 1st ACM MobiHoc Workshop on Technologies, Models, and Protocols for Cooperative Connected Cars* (pp. 37-42).
- [5] N. Alam, A. Kealy, and A. G. Dempster, "Cooperative Inertial Navigation for GNSS-Challenged Vehicular Environments," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1370-1379, Sept. 2013, doi: 10.1109/TITS.2013.2261063.
- [6] N. Alam, A. Tabatabaei Balaei and A. G. Dempster, "A DSRC Doppler-Based Cooperative Positioning Enhancement for Vehicular Networks with GPS Availability," in *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4462-4470, Nov. 2011, doi: 10.1109/TVT.2011.2168249.
- [7] P. B. Ober, "Integrity Prediction and Monitoring of Navigation Systems", vol. 1. Leiden, The Netherlands: Integricom, Feb. 2003.
- [8] J. Xiong, J. W. Cheong, A. G. Dempster and Z. Xiong, "An Integrity Monitoring Method for Multi-sensor Collaborative Navigation," 2020 *IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2020, pp. 461-468, doi: 10.1109/PLANS46316.2020.9109996.
- [9] *International Standards and Recommended Practices, Annex 10 to Convention on International Civil Aviation*, Int. Civil Aviation Org., Montreal, QC, Canada, Jul. 2006.

- [10] Z. Chen, H. Zhao, S. Hu, S. Chao, and W. Feng, "Integrity Monitoring Algorithm for GNSS-based Cooperative Positioning Applications" in 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019), October 2019, doi: 10.33012/2019.16881
- [11] M. Joerger and B. Pervan, "Fault detection and exclusion using solution separation and chi-squared ARAIM," in IEEE Transactions on Aerospace and Electronic Systems, vol. 52, no. 2, pp. 726-742, April 2016, doi: 10.1109/TAES.2015.140589.
- [12] Y. C. Lee, "A Position Domain Relative RAIM Method," in IEEE Transactions on Aerospace and Electronic Systems, vol. 47, no. 1, pp. 85-97, January 2011, doi: 10.1109/TAES.2011.5705661.
- [13] J. Xiong, J. W. Cheong, Z. Xiong, A. G. Dempster, S. Tian and R. Wang, "Integrity for Multi-Sensor Cooperative Positioning" in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 2, pp. 792-807, Feb. 2021, doi: 10.1109/TITS.2019.2956936.
- [14] M. Malinverno, F. Raviglione, C. Casetti, C.F. Chiasserini, J. Manges-Bafalluy, and M. Requena-Esteso, "A Multi-stack Simulation Framework for Vehicular Applications Testing", in Proceedings of the 10th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet '20), Association for Computing Machinery, New York, NY, USA, 17–24, Nov. 2020. doi: 10.1145/3416014.3424603
- [15] M. Malinverno, "Safety Applications and Measurement Tools for Connected Vehicles, in Doctoral thesis Polito, Mar. 2021, doi: <http://hdl.handle.net/11583/2895395>
- [16] F. Wang, W. Zhuang, G. Yin, S. Liu, Y. Liu, H. Dong "Robust Inter-Vehicle Distance Measurement Using Cooperative Vehicle Localization", in Sensors, 2021; 21(6):2048, doi: <https://doi.org/10.3390/s21062048>
- [17] M. Loreti, "Teoria degli errori e fondamenti di statistica", Introduzione alla fisica sperimentale, Dec, 2006, doi: <http://wwwcdf.pd.infn.it/labo/INDEX.html>
- [18] A. Minetto, A. Gurrieri and F. Dovis, "A Cognitive Particle Filter for Collaborative DGNSS Positioning," in IEEE Access, vol. 8, pp. 194765-194779, 2020, doi: 10.1109/ACCESS.2020.3033626.
- [19] M. B. Hurley, "An information theoretic justification for covariance intersection and its generalization," Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997), 2002, pp. 505-511 vol.1, doi: 10.1109/ICIF.2002.1021196

- [20] B. Noack, M. Baum and U. D. Hanebeck, "Covariance intersection in nonlinear estimation based on pseudo-Gaussian densities," 14th International Conference on Information Fusion, 2011, pp. 1-8.
- [21] P. O. Arambel, C. Rago and R. K. Mehra, "Covariance intersection algorithm for distributed spacecraft state estimation," Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148), 2001, pp. 4398-4403 vol.6, doi: 10.1109/ACC.2001.945670.
- [22] D. Eberly, "The Area of Intersecting Ellipses", Geometric Tools, Redmond WA 98052, 2008, doi: <https://www.geometrictools.com/>
- [23] D. Eberly, "Intersection of Ellipses", Geometric Tools, Redmond WA 98052, 2000, doi: <https://www.geometrictools.com/>
- [24] G.B. Hughes, M. Chraibi, "Calculating ellipse overlap areas", Computing and Visualization in Science, June 2011, doi: 10.1007/s00791-013-0214-3
- [25] Wikipedia, "Bhattacharyya Distance", doi: https://it.xcv.wiki/wiki/Bhattacharyya_distance