V2V communications for vehicle safety applications in blind road spots:

the highway and urban scenario

Christos A. Spyropoulos

Supervisor: Katsaros Dimitrios

Co-Supervisor: Korakis Athanasios

Volos 2016

Επικοινωνία όχημα με όχημα για εφαρμογές οδικής ασφάλειας σε δρόμους με σημεία χωρίς ορατότητα: *σενάριο εθνικών οδών και αστικών περιοχών*

Χρήστος Α. Σπυρόπουλος

Επιβλέποντας: Κατσαρός Δημήτριος

Συνεπιβλέποντας: Κοράκης Αθανάσιος

Βόλος 2016

Words of Thanks

My sincere gratitude goes to my supervisor Mr. Katsaros Dimitrios for his guidance throughout this work. I would like to express my deeply appreciation for the overall support and his willingness to help me with any problems I faced.

Many thanks to my friends for their support and for all the unforgettable moments during our academic studies.

Of course, my special thanks go to my family for their unconditional support and their motivation. Their trust brought me here today.

Christos A. Spyropoulos

To my Family

Περίληψη

Οι εφαρμογές στα δίκτυα επικοινωνίας οχημάτων βασίζονται σε συνεργατική ογήδηση των «έξυπνων» οχημάτων προκειμένου να παρέχουν μια πιο ασφαλή και ποιοτικά καλύτερη οδηγική εμεπιρία στον οδηγό και στους άλλους επιβάτες. Κάποιες από αυτές χρησιμοποιούνται για να ενημερώσουν τους επιβάτες για σημεία ενδιαφέροντος κατά τη διάρκεια ενός μεγάλου ταξιδιού ενώ άλλες είναι σχεδιασμένες για να αποφεύγουν σημεία κίνησης στους δρόμους. Μεταξύ της πληθώρας των διαθέσιμων εφαρμογών σε V2V (όχημα-με-όχημα) επικοινωνία, αυτές που ασχολούνται με την οδηγική ασφάλεια είναι οι πιο ελκυστικές καθώς κυριολέκτικά μπορουν να σώσουν ζωές. Το πιο ενδιαφέρον σενάριο στην κατηγορία εφαρμογών οδιγικής ασφάλειας περιλαμβάνουν περιπτώσεις στις οποίες κινούμενοι κόμβοι χρειαζεται να ανταλλάξουν πληροφορία σε πραγματικό χρόνο προκειμένου να αποφύγουν ατυχήματα όταν η ορατότητά τους παρεμποδίζεται από εμπόδια στο δρόμο. Σε αυτή την εργασία, αναπτύσουμε μια καινοτόμα εφαρμογή που καθορίζει την επικοινωνία όχημα-με-όχημα για να κάνει τα οχήματα να «δουν» τον επικείμενο κίνδυνο σε τυφλά σημεία του δρόμου, τόσο για αστικές περιοχές όσο και για εθνικές οδούς. Επιπλέον παρέχουμε μια εκτίμηση της προτεινόμενης λύσης μας χρησιμοποιώντας σενάρια προσομοίωσης για να αναδείξουμε τις δυνατότητες που έχει για την εφαρμογή της σε πραγνατικές περιπτώσεις. Τέλος, αναλύουμε τους συμβιβασμούς του προτεινόμενου αλγορίθμου και καταλήγουμε με τα θετικά και αρνητικά του.

Abstract

Applications in Vehicle-to-Vehicle comminication networks rely on cooperative driving of “smart” vehicles in order to provide a safer and qualitatively better driving experience to the driver and the other passengers. Some of them are used to let the passengers know about points of interest during a long distance trip while other applications are designed to avoid road conjestions. Among the plethora of available applications in V2V communications networks, the ones that deal with road safety are the most appealing since they can litteraly save lives. The most interesting scenario in safety applications includes the case where mobile nodes need to exchange information in real-time in order to avoid traffic collisions when their visibility is hindered by road obstacles. In this work, we develop a novel application that defines the V2V communication to make vehicles “see” the upcoming dangers in blind road spots, both in urban areas and in highway scenarios. Moreover, we provide an assessment of our proposed solution using simulated scenarios to highlight its potentials to apply in real-life situations. Finally, we analyze its tradeoffs of the proposed algorithm and conclude for its virtues and shortcomings.

Contents

Figure List

Table List

1. Introduction

Intelligent **T**ransportation **S**ystems, widely known as ITS, are applications of advanced information and communication technology that aim to provide innovative transport and traffic management services in order to foster safety and better coordination in transport. Of course, ITS support in this way a more environmental impact of the overall transportation. The term ITS itself, may refer to all modes of transport; for example it may be used for road transportation, mobility and traffic management as well as for users or even inter-vehicle communications.

In the last few years **V**ehicular **A**d **H**oc **N**etwork**s**, or VANETs, have gained tremendous popularity since they are a key component of the Intelligent Transportation Systems. They are created by applying the principles of mobile ad hoc networks and are a subcase of mobile ad hoc networks (MANETs). However, they differ from MANETs since their end-to-end connectivity is not guaranteed and the vehicles (i.e. the nodes of the network) are highly mobile. Moreover, they can scale up to very large networks, but the probability that they split into parts is high.

VANETs use any wireless communication technology to generate the networks. They enhance communication between vehicles, creating vehicle-to-vehicle (V2V) communication, and between vehicles and infrastructures, offering vehicle-to-infrastructure communication. As mentioned above, the VANETs are at high risk of partitioning, as their topology changes quickly. This means that there will be possibly a lot of disconnections during those alterations in their network structure. This characteristic makes the designing of an efficient solution to disseminate data in a particular way between vehicles a very difficult challenge in the area of V2V communications.

However, V2V communications offer the opportunity for cooperative driving and enable safety applications. Vehicles can broadcast now, through beacons mostly, information regarding their position, their speed or any other information about other vehicles nearby. In this way, each node in a VANET is now able to know its surrounding environment in real time. The challenge now, is to find a way to process this kind of information, decide which to keep and then efficiently exchange it in order to help discover or even prevent accidents. The transmission protocols must take into account many parameters, the limited bandwidth of the communication channels or that the data should be distributed intelligently when dealing with road safety.

Those challenges have been very attractive in the scientific community and thus any solutions have been proposed that focus on safety purposes. For example, the work in [2] and [6] suggests new routes to the drivers to avoid any traffic congestions caused by accidents that happened on the road ahead. R. Sengupta et al [3] introduced the CCW prototype that warns the drivers about ongoing situations. Other proposed solutions, take into account the neighbouring cars, to enhance driving in dangerous weather conditions like fog [4], [5], while others warn about a road hazard [7]. From all the aforementioned, the protocols that are preventing accidents by exchanging messages with their neighbours are the most significant in nowadays V2V cooperative safety applications. We focus therefore in this work our attention on creating an algorithm suitable to disseminate information on-time in real time to prevent accidents in situations where the possible "threat" cannot be seen directly by the driver.

1.1 Motivations

During the last years great advances have been made in the field of vehicular communications. Cars, but also vehicles in general, are getting smarter due to the omnipresent connectivity and the design of intelligent applications. Among all available types of applications, those that stand out are these referring to the safety of the passengers. Through the exchange of real-time information, mainly via beacons, cooperative driving application improve both traffic safety and efficiency.

Modern cooperative safety applications rely on the fact that "smart" cars are constantly aware of their neighbouring environment. Wireless communications assist to exchange the information and let the drive make safer decisions this way even in emergency situations. To this end, many safety applications have already been proposed. Some of them recommend re-routing mechanisms to avoid traffic congestions [2], [6], while some others put on priority the warning about hazards by either transmitting repeatedly the warnings [7] or by periodically sending "acknowledge" messages [8].

However, designing an algorithm that can that informs the driver about vehicles that are crossing junctions and are not visible from other drivers due to any kind of road barriers is out of their scope. Even [9] is a great approach for avoiding collisions by making predictions and using GPS technology, it is only suitable only for urban areas. None of the aforementioned works manage to alert the drivers for dangers about crossroad junctions using beacons and information that includes only knowledge about the neighbourhood, suitable both for urban and suburban environments. Therefore, and for the first time in the literature, we present a solution for the above.

1.2 Contributions

Summarizing the above, this work aims to

- design a novel application for alerting the drivers, that are going to cross a road junction or to overtake in a highway, about passing threats
- develop an application suitable for V2V communications
- present both advantages and shortcomings of the aforementioned application

The rest of this work is organized as follows: Section 2 introduces the reader to the vehicular communications world; we present in this chapter already proposed solutions in the related area. In Section 3, we present our proposed solution. We give firstly a description of all the necessary background for the reader to get familiar with V2V communications. Furthermore, we present the tools used for simulation communication networks in this work and finally, in Section 5 we conclude our work and propose future tasks.

2. Related Work

Since the idea of smart vehicles first appeared, the great challenge was to make them communicate in order to offer a better and safer driving experience. Using wireless communications made it possible. Since then, several approaches, based on cooperative driving, for how to avoid collisions or any other traffic dangers have been proposed.

S. Dashtinezhad et al. [5] thought that in means of safety we have to consider situations where the weather makes driving risky and dangerous. Such weather conditions are for example heavy rain or fog. They came up with a solutions that provides a map of the neighbouring (or better said) vehicles, by gathering information about them via ad hoc wireless communication systems. However, it cannot prevent their intentions or provide any other information about dangerous situations.

The latter and also situations of unexpected road hazards, were the main concerns in the warning system that X. Yang proposed in [7]. The protocol assumed that the car in danger transmits the warnings in a 300 meter range to other cars. Congestion control policies defined the delivery of low emergency messages. In this case, the car maybe also facing mechanical failures, so it is already facing the danger.

Predicting a possible collision is something H. Tan [13] worked on. More specifically he used a system composed of a GPS and some motion sensors to predict the collision by exchanging messages of surrounding vehicles. Nevertheless, this solution does cover only cases where the areas under considerations is considerably small, something that does not reflect real-life scenarios. Sensors have also been used, brake sensor in particular, by R. Karlsson to statistically define whether to stop the vehicle or not [14]. However, this solution does not check for hidden dangers in crossroads. J. Jansson in [15] created a statistical decision making system to avoid collisions by using radar. Radars are not available on every vehicle to make this solution a competitor one.

J. Hillenbrand was also motivated by sensor-based collision mitigation and the prediction of uncertainties in V2V systems. However, the vision of accident-free driving, as they pointed out in their work, can only be achieved only via cooperative driving solutions.

Another solution more suitable for congestion detection and maybe rerouting was suggested by A. Lakas [8] in combination with shortest path theory. Here the vehicles send periodically messages to others and wait for their answer to get to a conclusion about the traffic. Neither this solution is suitable for collision avoidance as too many messages have to be exchanges in order to converge to a result and it mostly does not foresees collisions.

The proposal of C. Huang [16] was a rate-power control algorithm for broadcasting self-information messages. This would enable tracking of neighbours. However, even though this seems a feasible solution, the capacity of the bandwidth was exploit while sending the frequent messages. Biswas on the other hand presented in [6] a solution tailored for avoiding collisions on highways. The assumption though that all the vehicles know all other vehicles to warn them makes it unrealistic.

Granting all the decision making part of the process solely to the driver is also not preferred. R. Sengupta et al. in [3] used GPS and wireless communication to create a system that gets a 360-degree awareness of the surrounding environment of the vehicle. In this work the analysis of the situation is left to the driver while the system just informs about any ongoing situation.

A. Dogan has designed an intersection collision warning system using digital GPS location data and then broadcasts this information at a certain distance from the intersection using an ad-hoc wireless network [13]. This intersection collision warning system has been evaluated by a MATHLAB-based simulator which consists of vehicle traffic simulator and wireless simulator. However, it needs to exchange a lot of information between vehicles, like vehicle kinematic and dynamic information.

Clearly none of the above solutions is appropriate in cases where the moving vehicle cannot be seen by the driver of another vehicle, both approaching an intersection, due to a road barrier. This is exactly the gap we are trying to fill in the present work.

3. Our proposed solution

In this Section we provide the necessary background of V2V communications before me move on to our proposed solution. We describe in detail any information that the reader should be aware of in order to get familiar with the concept of cooperative driving.

3.1 Background Review

This subsection highlights all relevant information regarding the Ad Hoc Networks in general. We begin with an introduction to wireless Ad Hoc networks and once the reader is familiar with the basic concept we describe Mobile Ad Hoc networks. Later in this subsection we also present the simulation tools we used for the concept of our algorithmic solution.

3.1.1 Introduction to Wireless Ad Hoc Networks

A network is said to be Ad Hoc when it does not rely on a pre-existing infrastructure. In contrast to prior existing network architectures that needed some designated nodes (routers, hubs, switches, etc.) to forward the data, each node is here a member of routing by means of forwarding, or even flooding, any data intended for other nodes in the network; Figure 1 shows the evolution we see from older networks to ad hoc ones. The selection of the nodes that should forward this data/messages is highly dynamically and depends on the network connectivity each time.

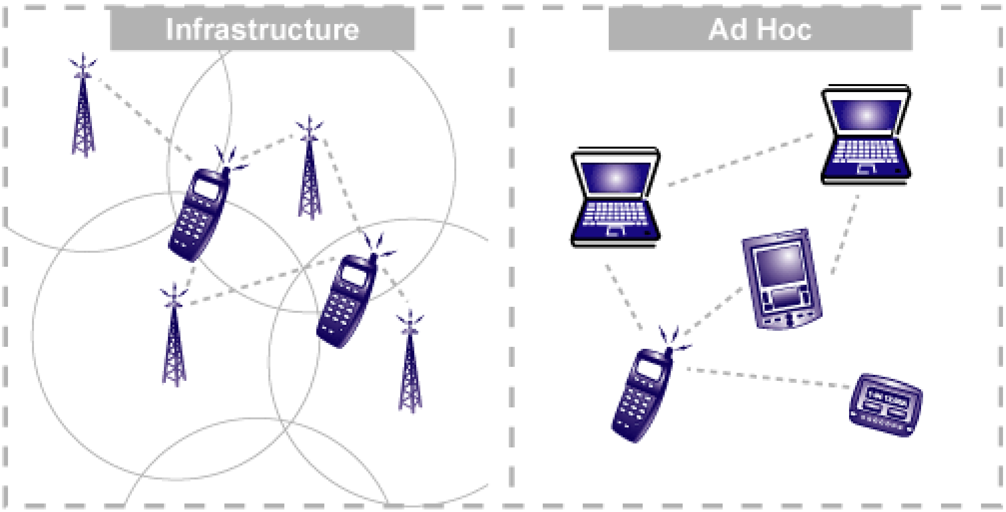

Figure 1 Infrastructure vs. Ad Hoc Scenario

Source: https://cdn.sparkfun.com/assets/learn_tutorials/5/3/4/network-infrastructure-vs-a.png

It is obvious now that a wireless ad-hoc network is a network where the communication links are wireless. In such a network, all devices within a distance are allowed to connect peer-wise without any need of a pre-existing centralized architecture or a single point of access (see Figure 1). All nodes contact each other here over wireless links. This, consequently, means that they also need to struggle with problems of interference or noise. [10]

Figure 2 Example of a Wireless Ad-Hoc Communication Network

Source: http://www.eusso.com/Models/Wireless/UGL2430-VPH/Diagram-1.jpg

Let's have a closer look at the characteristics of wireless ad hoc networks. First we have to point out that their topology is dynamic. This derives from the fact that some nodes leave the network, some others join over time and others are of course mobile. As mentioned above, each node of these networks volunteer to participate in the data forwarding part for other nodes of the network, playing both the role of a router and a host. This way, the network is controlled completely by its nodes.

Sometimes flooding techniques are used to disseminate the data over the network. However, this is not efficient as it overloads the network with useless workload and also exploits its capacity. Many routing protocols have been therefore proposed for a better way to spread the data in a multi-hop fashion without consuming unnecessary network resources.

On top of all of that, a minimum of configurations is necessary in case of ad hoc networks and their deployment is really fast. These two characteristics, along with their decentralized nature, highlight also the areas where ad hoc networks are preferred, like military conflicts or disaster situations both human and nature-forced.

Three most appealing types of wireless ad hoc networks are the following:

1. **M**obile **A**d **H**oc **N**etwork**s** (**MANETs**)
2. **W**ireless **S**ensor **N**etwork**s** (**WSNs**)
3. **W**ireless **M**esh **N**etwork**s** (**WMNs**)

- **Category 1: MANETs**

MANETs are continuously self-configuring, infrastructure-less networks of mobile devices connected in a wireless mode. MANETs are structured by a peer-to-peer and self-healing network. Each node in this type of network is moving independently and in any direction. The entire network may work autonomously but can also be connected to a greater Internet, containing a variety of transceivers. Due to their highly dynamic structure, the biggest challenge in MANETs is that each node has to maintain continuously the information for proper routing [17].

- **Category 2: WSNs**

WSNs are communication networks that contain autonomous sensors that are spread spatially [18]. Their main task is to monitor environmental (or physical) conditions like temperature, humidity, pressure, sound, etc. They undertake also to disseminate the information cooperatively via the network to an agreed single point. Initially WSNs were generated to assist military applications. Nowadays, such networks are mostly used for industrial purposes, in order for example to measure the health of the machines, as they support bidirectional communication. The architecture of a typical WSN is shown in Figure 3.

Figure 3 An example of a WSN

Source: http://www.altenergymag.com/articles/08.12.01/links/contents_clip_image006.jpg

- **Category 3: WMNs**

WMNs are communication networks whose nodes are forming reliable and redundant mesh topologies [19]. When one node fails, the rest of them can still "talk" directly or through several intermediate nodes. The main components of such networks are radio nodes like mesh routers, gateways and mesh clients. The first ones communicate with the gateways to send outgoing and receive incoming traffic; the latter, on the other hand, are usually cell phones, laptops and generally wireless devices. An example is shown in Figure 4.

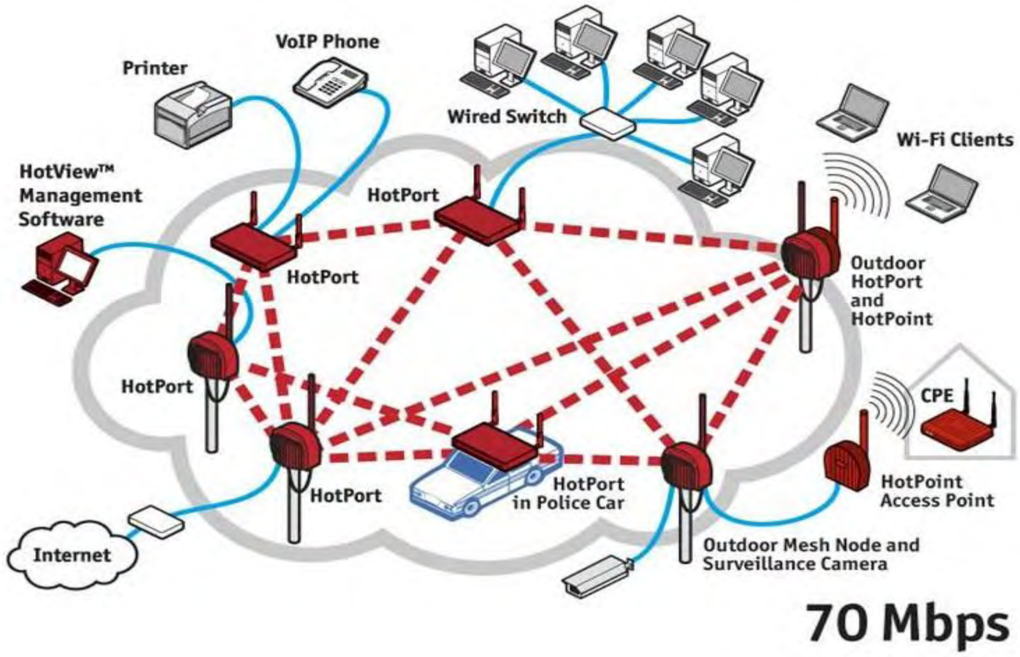

Figure 4 An example of a WMN

Source: http://www.busy-ant.com/images/network2.jpg

3.1.2 Introduction to Mobile Ad Hoc Networks (MANETs)

MANETs are a popular type of Wireless Ad Hoc Networks which consist of several wirelessly connected mobile devices, does not support any certain infrastructure but is self-configuring. All of its nodes are free to move in any preferable direction independently making the link connections very temporary as they are changing continuously. The entire network may work in an autonomous way but it can also be connected to a greater Internet, containing a number of transceivers. The extremely dynamic structure makes MANETs vulnerable to disconnections and delays. The challenge in MANETs' routing is even bigger when considering that each node has to forward data for any other node in the network, playing the role of a router. The continuous routing maintainance of the information is therefore a "heavy job". Through the years some of the the main challenges in MANETs were based on measures such as end-to-end packet delays, network throughput, the overhead, packet drop rate, and so on.

MANETs can be further subdivided into more categories, namely the following:

1. **Vehicular Ad Hoc Networks (VANETs)**
   VANETs are communication networks that are used for communication between vehicles and roadside equipment.
2. **Smart Phone Ad Hoc Networks (SPANs)**
   SPANs create a peer-to-peer network without relying on cellular carrier networks, traditional infrastructure or any wireless Access Points. Conversely, they leverage existing hardware in commodity smart phones for this purpose (mainly via Bluetooth and WiFi). They are using multihop relays and an

architecture that is master-free so that any node can leave and join at any time without destroying the network. That is also the main difference from hub, WiFi-Direct and spoke networks.

3. **Internet based mobile Ad Hoc Networks (iMANETs)**
   iMANETs are ad hoc networks that link mobile nodes and fixed Internet-gateway nodes. A characteristic of them is that common ad hoc routing algorithms do not apply directly.
4. **Intelligent vehicular Ad Hoc Networks (InVANETs)**
   InVANETs are a kind of artificial intelligence that helps vehicles to behave in intelligent manners during V2V collisions, accidents, etc.

3.1.3 Introduction to Vehicular Ad Hoc Networks (VANETs)

Vehicular Ad Hoc Networks, also known as VANETs, are a key component of Intelligent Transportation Systems (ITS) and a special type of MANETs where the nodes represent vehicles or, much less, Roadside Units (RSUs). They are generated when applying the principles of MANETs to the domain of vehicles. [20]

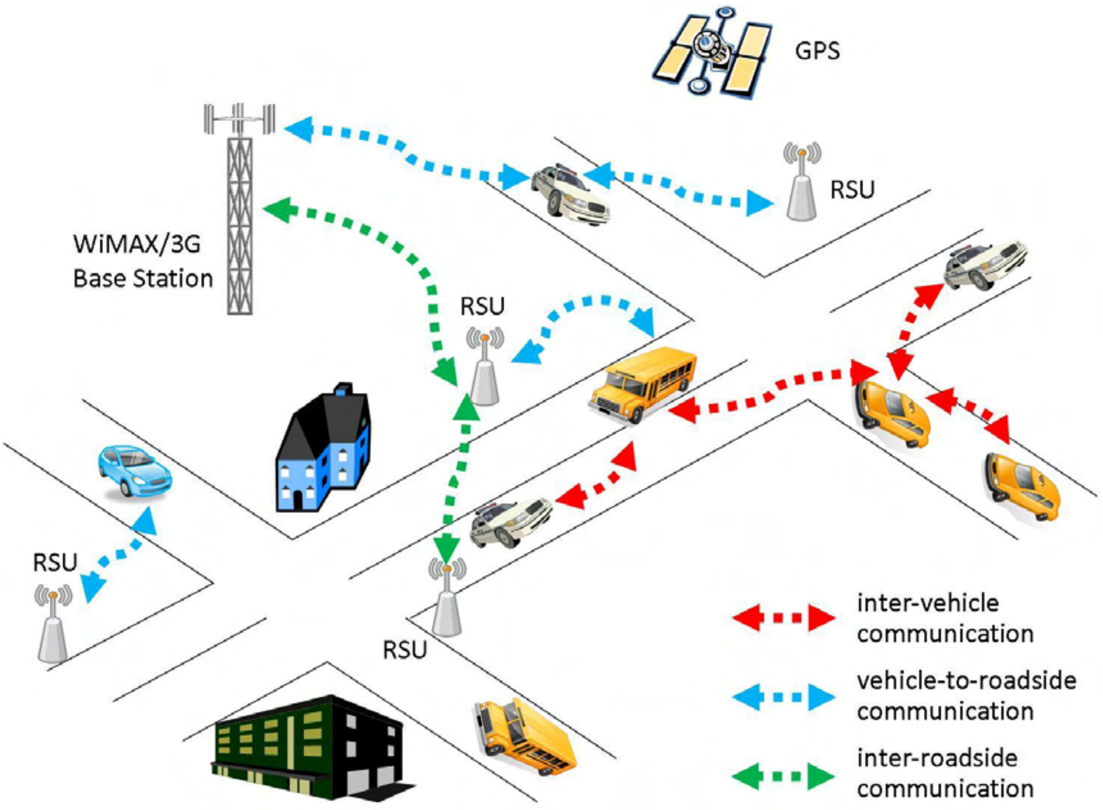

Figure 5 An example of a VANET

Source: http://adrianlatorre.com/projects/pfc/img/vanet_full.jpg

Since its initial introduction in cooperative driving for safety proposes VANETs have found a large variety of applications. VANETs are seen for example in:

- Electronic brake lights, allowing the driver or even an autonomous car to react to vehicles braking. The impressive case that also applies here is when the vehicle braking is obscured by other vehicles or road barriers.
- Traffic information systems. These are used to warn/inform the driver about obstacles and reports to the driver's satellite navigation system. The reports are incorporating the very latest information.
- Platooning, allowing vehicles to approach the leading vehicle up to some inches. This is achieved by getting wirelessly acceleration and steering information.

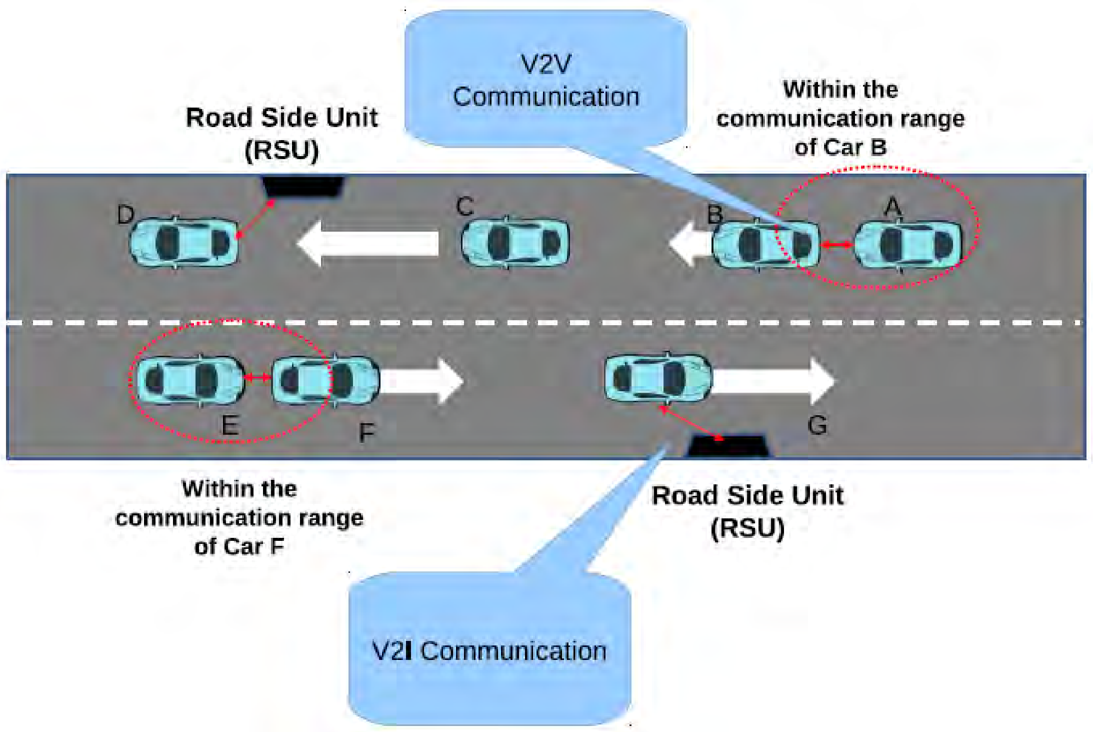

Figure 6 An example of V2V and V2I communication

Source: http://article.sapub.org/image/10.5923.j.jwnc.20130303.02_001.gif

It is obvious that modern communication systems are aiming to create such networks in order to send and receive meaningful data both between vehicles and between vehicles and infrastructures. The range of communication is limited to 1km. Vehicles inside this range "talk" directly with their neighbours; i.e. their one-hop neighbourhood.

In this direction, VANETs can be divided in two categories according to their communication ends type (e.g. see Figure 6):

1. Vehicle-to-Vehicle communications (V2V)
2. Vehicle-to-Infrastructure communications (V2I)

Vehicular Ad hoc Networks focus on a variety of mobility aspects, mainly the nodes' velocity, the nodes' density and also their movement patterns. In the first case, the velocity may range from zero to a speed over 150km/h. The no-movement represents an RSU or a traffic congestion. On the other hand, the high speed can only be assumed to appear on highways and in this case the communication duration is very limited because the communication range is of a few hundred meters. Considering for example diverging vehicles with a medium to high speed, one can imagine that the communication will only be possible for a few seconds inside a small range area and some additional Doppler effect may also occur.

Another property would be the node density. The density is different from type of road to another (i.e. highway vs. suburban road) and from time period to others (i.e. time where everyone goes to their work vs. time of day when most people sleep). To understand this concept, imagine a communications network with very few nodes. Now, imagine that a vehicle has to transmit a message immediately. This seems obviously almost impossible. Therefore, in a low density immediate messages are almost not possible to transmit, especially when no other vehicles are inside the communication range. The transmission may be possible after a while. In this case the vehicles have to cache their data and send them as soon as communication becomes available again. However, it is now possible that the message will be replicated several times. Contrary, very dense networks should choose the transmitter nodes so as to not create something like flooding situations and overloads.

Finally, movement patterns are equally important when investigating VANETs. Sometimes they can easily be predicted; for example if the road has only one direction. In other cases, it is very hard to guess the pattern, for example roads with multiple intersections. This is the case that is highly important when dealing with safety applications and is of our interest.

Below you can see the majority of the applications in ITS:

1. **Safety applications**

   They are one of the most important ones. They are trying to reduce the risk of traffic collisions mostly by warning the driver for possible accidents. They can additionally be incorporated incident management applications that inform drivers about on headed traffic, road barriers or traffic jam due to traffic collisions and accidents.

   Some very popular applications we face in real life scenarios are road sign notifications send to the driver to inform him/her about the upcoming signs on the road so that he/she can be prepared. Other safety applications are designed to warn about curve speed; those warn the driver about the size of the curve and speed and road limitations. Finally, one of the most important safety applications is the one that is used by vehicles in order to slow down to adjust the distance from the vehicle ahead.

2. **Efficiency applications**

   They are designed to manage traffic flow. In order to do that they also monitor road and vehicular conditions in nearby areas, information that is in time broadcasted. This category is even more important when coupled with safety applications to inform the driver about road situations.

3. **Comfort applications**

They are created for entertaining the passengers. They can for example provide information about nearest points of interest (tourist exhibitions, restaurants or bars) or advertisements. Games with other surrounding vehicles can also be found in this category.

Wireless Communication Technologies in VANETs

As mentioned above, VANETs are communication networks that rely on wireless communication channels. WLAN, or otherwise said the IEEE 802.11p wireless standard is an approved amendment to the IEEE 802.11 standard that adds Wireless Access in Vehicular Environments (also known as WAVE), a vehicular communication system. It defines enhancements to 802.11 required to support Intelligent Transportation Systems (ITS) applications.

IEEE 802.11p operates in the licenced ITS 5.9GHz (5.85-5.925 GHz) band and involves data exchange between high-speed vehicles and V2X communications (vehicle-to-roadside infrastructure). Moreover, WAVE offers stability, data is spread in high speed and immediate mode, and it manages to maintain the security of the transmitted messages.

A great advantage that the IEEE 802.11p amendment offers is that it defines a way to exchange information via a link, without any need to wait for the association and authentication procedures to complete before the messages can be exchanged. This is highly important when considering that the communication link between vehicles and the roadside infrastructure may exist only for a very short time slot. However, since we do not wait anymore for any authentication mechanisms to take place in the communication part, the authentication along with the data security must now be handled by higher network layers. [21]

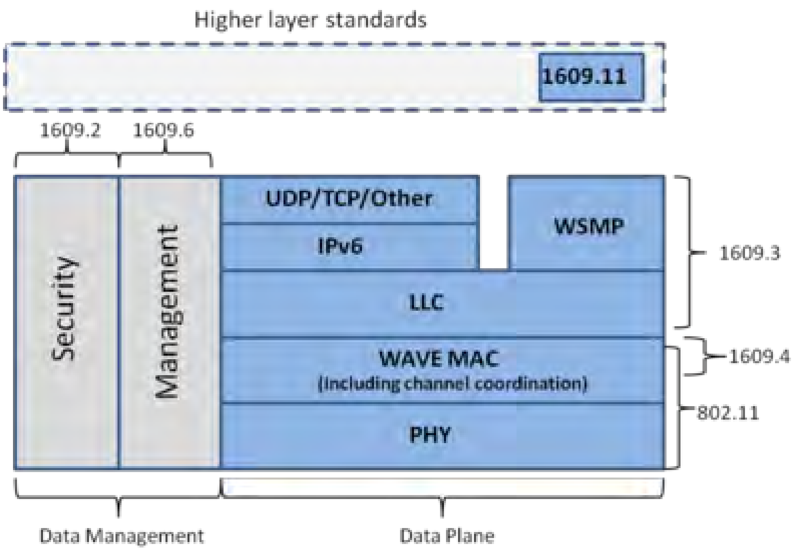

Figure 7 The 802.11p protocol

Source: https://upload.wikimedia.org/wikipedia/commons/thumb/0/01/PilaWAVE.png/400px-PilaWAVE.png

3.2 Simulation tools

Before we can move on to our proposed solution description we need to describe the testing environment of our work. In order to design, develop and finally test our V2V communication structure for avoiding collisions on crossroads we used different simulators and tools: OMNET++, SUMO and VEINS namely. Below, you can read in details about each of these simulation tools and their capabilities.

3.2.1 OMNeT++

OMNET++, the Objective Modular Network Testbed in C++, is a (C++)-based extensible, object oriented, modular component-based simulation library and framework which mainly is designed to build network simulations. In other words, it is a discrete event simulator for communication networks, wired and wireless, but also queuing, on-chip networks and other. Instead of containing hardwired and explicit support for computer networks or similar areas it offers an infrastructure for writing these type of simulations [22].

It is one of the most popular network simulation platforms both in the scientific community and in industry even though it is not a standalone network simulator; it is not a simulator but offers the necessary tools to design simulations.

Its architecture consists of components, the so called modules. OMNeT++ provides a component architecture for modelling distributed hardware systems and multiprocessors and an attractive architecture that makes it a suitable tool. What developers really appreciate about OMNET++ is the high reusability of models that comes for free. On top of that, it offers an extensive GUI support and the kernel is easily embedded into the user's applications and so are the models. Its components are written in C++ language and form then larger components and models. These use NED, a high level language, and gates for messaging. For the coding environment, OMNET++ uses an Eclipse-based IDE, a host for other tools and a graphical runtime environment. Finally, it offers extensibility in means of database integration, network emulation and real-time simulations.

The above modules can have parameters. If so, either NED files or an initialization file of OMNET++ are used to initialize them. In the latter file, named omnetpp.ini, besides the variables it contains also other model parameters and decryptions for the simulation execution and the total number of iterations. Model frameworks of OMNET++, are developed as independent projects and offer Internet protocols, support for sensor networks, domain-specific functionality and wireless ad-hoc networks or even photonic ones; and the list goes on!

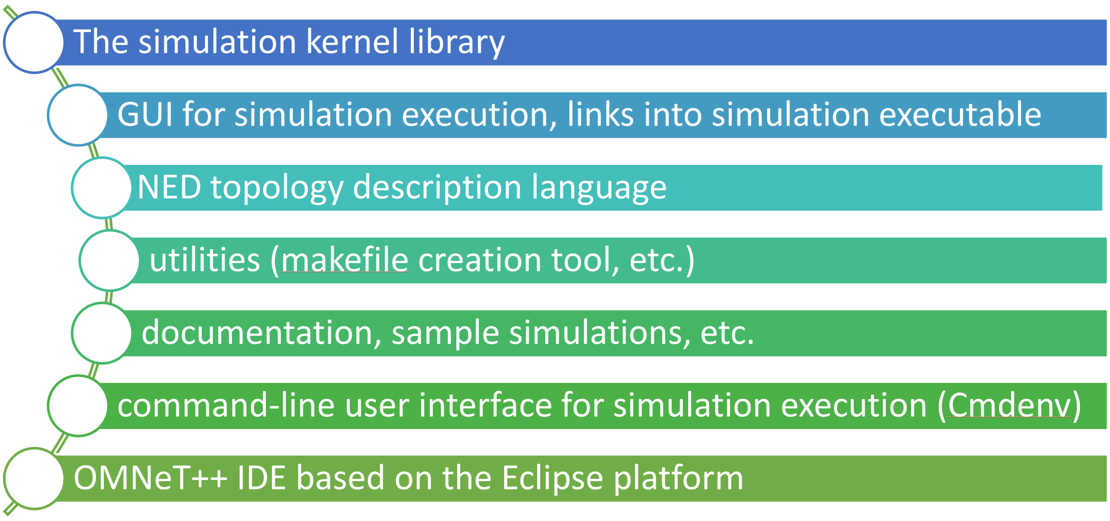

Figure 8 The components of OMNET++

The main simulation frameworks of OMNeT++ are:

- **INET Framework,** which contains models for the Internet stack (e.g. TCP, UDP, IPv4 etc.), wired and wireless link layer protocols such as Ethernet and IEEE 802.11, support for mobility, MANET protocols and many other protocols and modules. INET framework is considered the standard protocol model library of OMNeT++.
- **INETMANET,** which is kept up to date with INET Framework in order to offer more experimental properties and protocols, especially for MANETs.
- **MiXiM,** which is created for mobile and fixed wireless networks like wireless sensor networks and vehicular ad hoc networks. It offered detailed models of radio wave propagation, interference estimation, radio transceiver power consumption and wireless MAC protocols [12].
- **Castalia,** a simulator for networks of low-power devices like Body Area Networks (BAN) providing realistic simulation parameters.

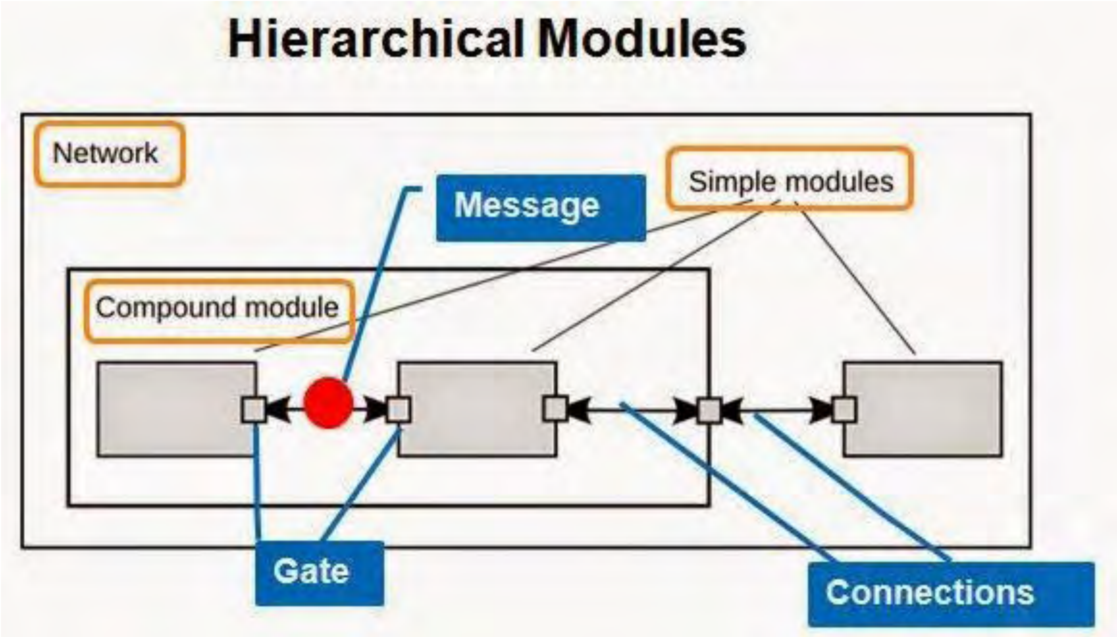

Figure 9 Hierarchical Modules, OMNET++

Source: http://4.bp.blogspot.com/-SBAA1o7fbS0/VRPF01o_GYI/AAAAAAAAAQs/tlnn7zynWWs/s1600/H%2Bmodules.JPG

3.2.2 Simulation of Urban Mobility – SUMO

SUMO [12] is an open-source[1], highly portable traffic simulation suite that allows modelling of traffic systems such as vehicles and public transport along with pedestrians. The main goal here is to simulate the movement of a given traffic on a specific network. In order to do that SUMO supports a simulation of each vehicle so that it can act independently in the network field, forming its own route; its position is updated every time unit. The outcome is presented graphically using an OpenGL-bases graphical user interface.

SUMO is a suitable tool for route finding, network import, and visualization and emission calculation. It is designed to manage large road networks. More specifically, what makes sumo so attractive in V2V communication systems development is that it answers a variety of applications. For example, SUMO allows performance evaluation of various aspects ranging from modern algorithms and traffic lights to weekly time planning, eco-aware routing based on pollutant emission and investigations on network-wide influences of autonomous route choice. Moreover, it has been answering research questions in the field of vehicle route choice with many new methods. The most important characteristic that makes it an appealing solution in V2X communications research is that it provides realistic vehicle traces and evaluates applications in an one-line loop with a network simulator.

[1] Licensed under General Public License (GNU)

Creating networks with SUMO

As mentioned above, SUMO's main feature is that it supports creation of road networks, including also the surrounding environment, i.e. streets and buildings. To describe such a network, the .nod.xml file is used. It is a description of the network nodes and junctions. However, to describe their connections, i.e. the edges of the network, users should use the .edg.xml file. Other formats are also supported. For example, one can add OpenDrive or OpenStreetMaps as well, since it will be converts to the property sumo description files. The output is a network description file in .net.xml format.

Figure XXX depicts an overview of how a road network of the SUMO environment looks like. As one can observer, streets are basically a set of roads and can be represented by their position, shape and even the speed limit. The crossroads of the field, are called conjunctions. Traffic light can also been added to regulate the traffic. However, in every other case the vehicle coming from the right is of priority here. Vehicles, are considered to be in SUMO typical cars, busses motorcycles, bicycles or even taxis!

Now, that we have the vehicles and their road field, we need to describe also the routes that should be followed. To represent the traffic demand, we need to assign the starting point, the endpoint and a velocity value for the vehicle. Finally, impressive is also the fact that the scenarios have a more realistic feeling since accelerations and imperfections of actions can also be part of a scenario. [23]

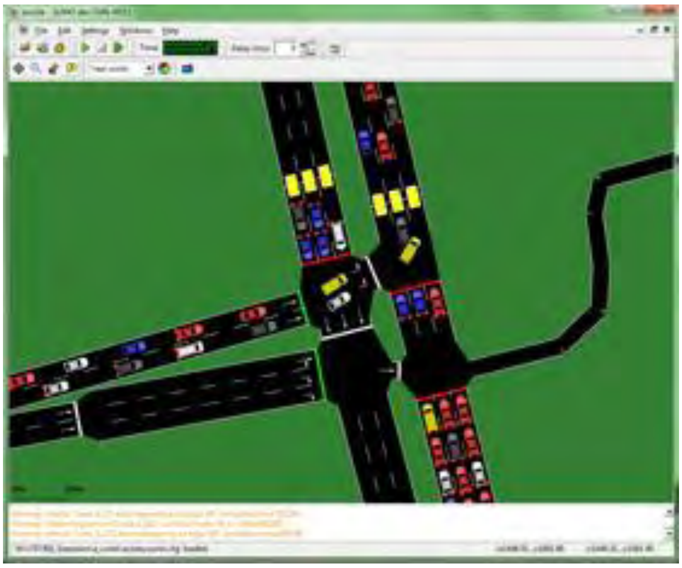

Figure 10 The SUMO simulation environment

Source: http://www.dlr.de/ts/en/Portaldata/16/Resources/Projekte/SUMO/SUMO-Benutzeroberflaeche.jpg

3.2.3 Vehicles In Network Simulations – Veins

Veins is an open-source framework suitable for running vehicular network simulations. It is twisted together with OMNET++ and SUMO (see descriptions in subsection 3.2.1 OMNeT++ and subsection 3.2.2 Simulation of Urban Mobility – SUMO for details). It extends those to offer a complete suite for Inter-Vehicle Communication (IVC).

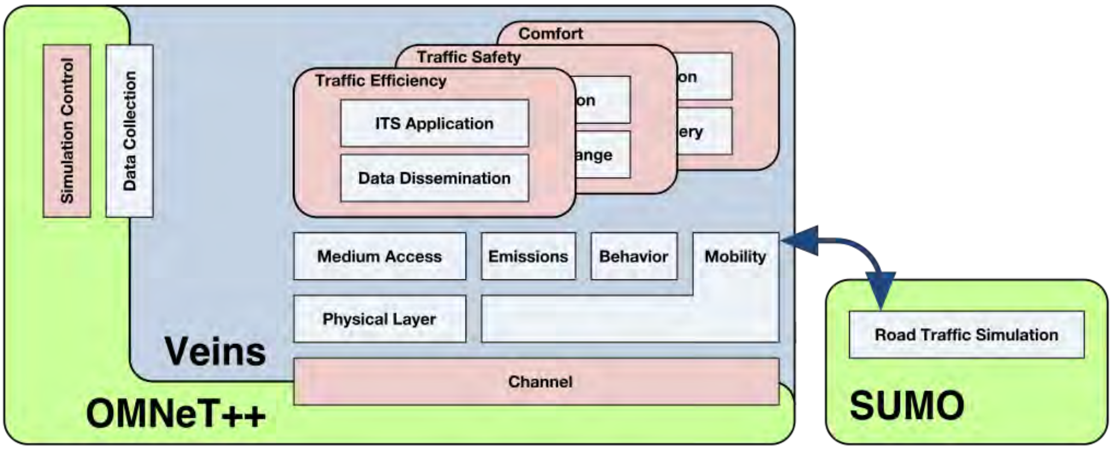

Figure 11 Veins's Architecture

Source: http://veins.car2x.org/documentation/veins-arch.png

As Figure 11 shows, SUMO and OMNET++ need to run in parallel and need to connect via TCP sockets in order to run IVC simulations. Each one of the components is there to serve a specific task. SUMO performs the road traffic simulations and OMNET++ coupled with the physical layer modelling toolkit MiXiM, perform the simulations. Both simulators are bi-directionally coupled but the actual simulations are performed online. The Traffic Control Interface (TraCI) is a standardized protocol for the above described communication. This way described, movements of the vehicles in SUMO are reflected on OMNET++.

Traffic Control Interface - TraCI

The Traffic Control Interface (TraCI) is a standardized protocol used to support bidirectional communication among traffic and network simulators. The simulators need to interconnect through a TCP client-server based architecture in order to communicate. In vehicular communication systems TraCI plays the role of the server and the simulators act like clients. Every network simulation, starts a new instance of SUMO. The client forwards commands to be executes server-side to retrieve information about road and traffic conditions. The whole system relies on a request-response mechanism and thus the server sends responses to each incoming request. For example, it sends information about the vehicles' positions in each simulation step to achieve the aforementioned movement reflection in the network simulator. It can also demand a change in the module's status. SUMO provides information about the road network environment like routes and vehicle properties. In the same way, the network simulator can also change a state in SUMO via TraCI. At the end, the client is responsible to stop the connection. All the above functionalities are offered in Veins through TraCIMobility module.

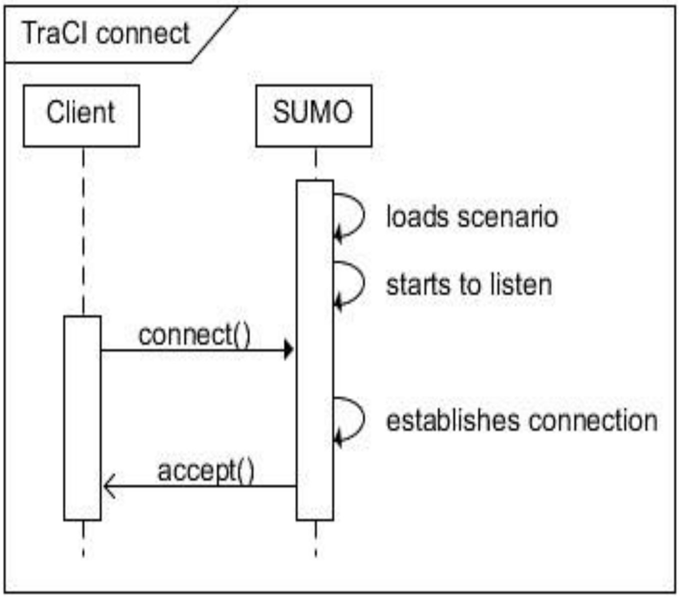

Figure 12 How a connection is established between the client Network Simulator and the server SUMO-traffic simulator using TraCI

Source**:** http://sourceforge.net/apps/mediawiki/sumo/index.php?title=TraCI

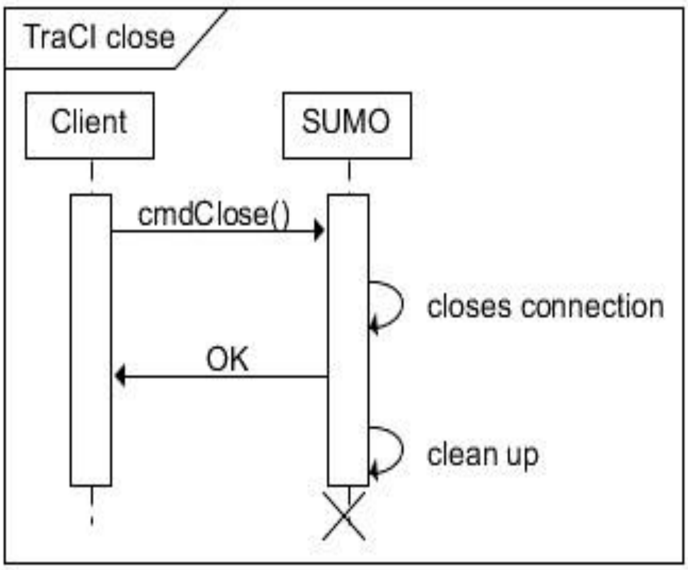

Figure 13 How a connection closes between the client Network Simulator and the server SUMO -traffic simulator using TraCI

Source: *http://sourceforge.net/apps/mediawiki/sumo/index.php?title=TraCI*

3.3 The BLCA algorithm

After providing the necessary background in details on *Vehicular to Vehicular communication networks* we are now ready to describe in this section the proposed application, namely The BLCA Road Safety Application, and how it excels in accident avoidance.

The **BLCA** Road Safety Application, standing for ***BL**indspot **C**ollision **A**voidance,* aims to reduce accidents that are caused in cases where the visibility of the driver is hindered by other vehicles causing blind spots. The basic idea behind our application is to use those vehicles which create the blind spots to inform other drivers whether they are about to get involved in a collision.

Before we can move on to more details, it is for the best interest of the readers that we present here some examples that show our cases. Initially let as mention that BLCA applies both in case of a highway and for urban scenarios. Of course the necessary adjustments need to be made.

Case 1: Urban Scenario

In this subsection we provide the definition of the problem that happens in an urban area. Then we describe extensively how BLCA resolves this issue.

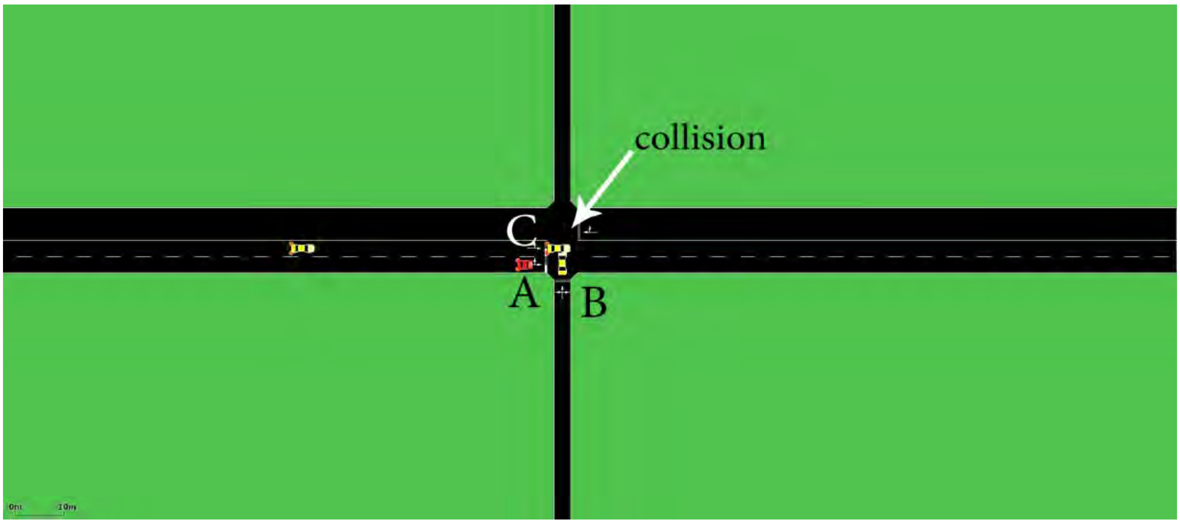

Figure 14 A case of road collision (SUMO environment)

To better understand the background of the urban scenario, imagine that the above road is placed in a city. It consists of a big main road with two lanes in each direction and a smaller road. Together they form a crossroad. Since this crossroad is placed in a city we assume that the maximum velocity of the vehicles cannot exceed the 30m/s which is a lot grater of the actual speed limit in a city (13.9m/s).

As we notice in the Figure 14 above the driver in vehicle A decided while he was moving in a priority road to stop and let driver in vehicle B cross the junction. In this scenario, driver in vehicle B has limited visibility of the priority two lane road, especially of the lane vehicle C is moving on, due to the position of A on the road. It is certain that these cars follow a collision course. Therefore, under normal circumstances, taking also into account a possible driver's distraction, and without V2V communication there would be a great possibility that CarB and CarC would crash.

Using V2V communication we try to solve the aforementioned problematic situation. More specifically we want to exchange messages between those vehicles that alert drivers about the oncoming collision. Figure 15 shows exactly the preferable scenario.

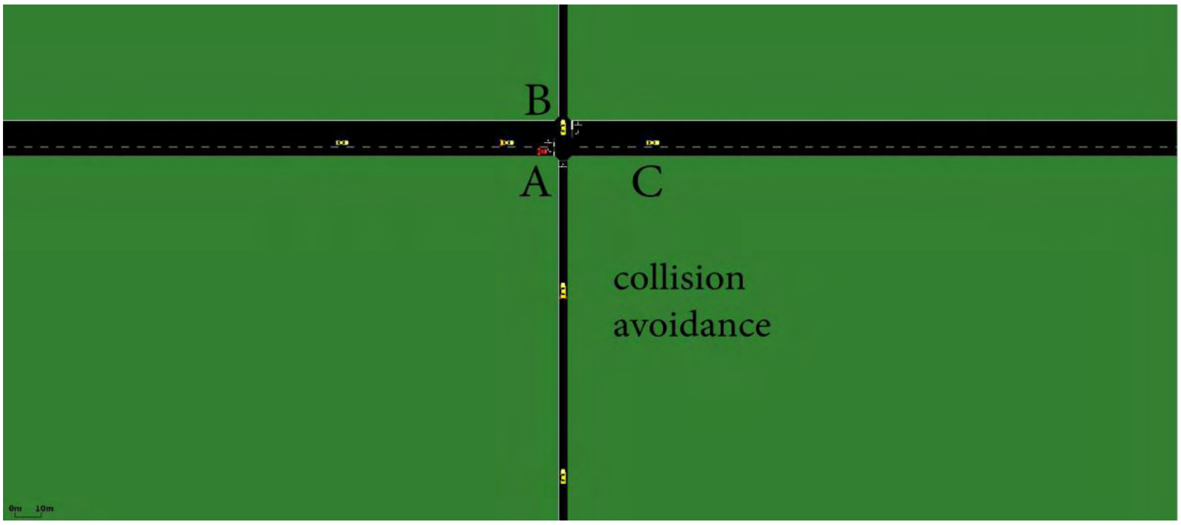

Figure 15 A case where road collision is avoided

At this point we will describe our proposed solution for collision avoidance in this case. First of all we need to mention that we assume that the cars are heading toward each other and they have a velocity greater than zero and less than 40m/s. The latter velocity is too high if we consider that we are dealing with a city area. Moreover the speed of each car differentiate a lot through time as a result of urban traffic. The most important thing of BLCA is that it takes advantage of the roads obstacle (carA) instead of ignoring it. More specifically carA is used to warn vehicles that are in a collision course.

A pseudo-code for this approach is as follows:

<u>*Vehicle A/ Obstacle*</u>

1. *VehA Send request asking other cars for current position and speed.*
2. *VehA receive the coords and calculate estimated position in 1 second*
3. *if estimated position of two vehicle is within a small range (8m)*
    a. *send a warning*
    b. *goto 1 with 0.1s time step*
4. *else*
    a. *goto 1 with 0.1s time step*

<u>*Other Cars*</u>

1. *Send coords for current position and speed*
2. *if receive Warning with your id*
    a. *adjust speed and respect priority*
3. *else*
    a. *ignore message*

Using this approach we accomplice to find a solution to this scenario. The first major difficulty was to set the beacon interval to communicate with vehicles. A value about 1s even if in the first look seems to be a decent choice has a significant problem. Speed of the cars in a city tends to change frequently, and a 1s interval could cause false knowledge of the exact coordination of a vehicle. If we put to equation that drivers need time to respond after a warning, the fault would be great. For those reasons the beacon interval has to be as low as 0.1s. In greater beacon intervals in case of stopped vehicle which accelerate to cross the junction the calculations will be incorrect and few meters are critical in those situations. So, for safety reasons we use so low beacon intervals.

Furthermore, SUMO by default is accident free, so to study the above case scenarios certain default rules had to change. First vehicles must be impatient to cross the junction while there are other vehicles moving on them. They must not respect other vehicle speed and reaction time when crossing the junction. All the above must happen to simulate a driver with obstructed view and with limited knowledge of the crossing lane's state. All that are parameters of human fault. To accomplish those challenges we took advantage of TRaCI interface. In case of buildings obstructing the view we make carA to calculate and estimate future position of the involved vehicles because it has the best possible position to communicate with both oncoming vehicles. Beacon from vehicles moving in nearby roads that might be on range discarded. Default parameters of OMNeT++ in the physical and mac layer of IEEE 802.11p, provide a theoretical transmission range about 500m which is more than enough. This scenario could be implemented with the same logic when these is a case of a red light violation.

Case 2: Highway Scenario

In this subsection we define and describe the problem that appears in a highway area. Then we explain in details how BLCA resolves this issue.

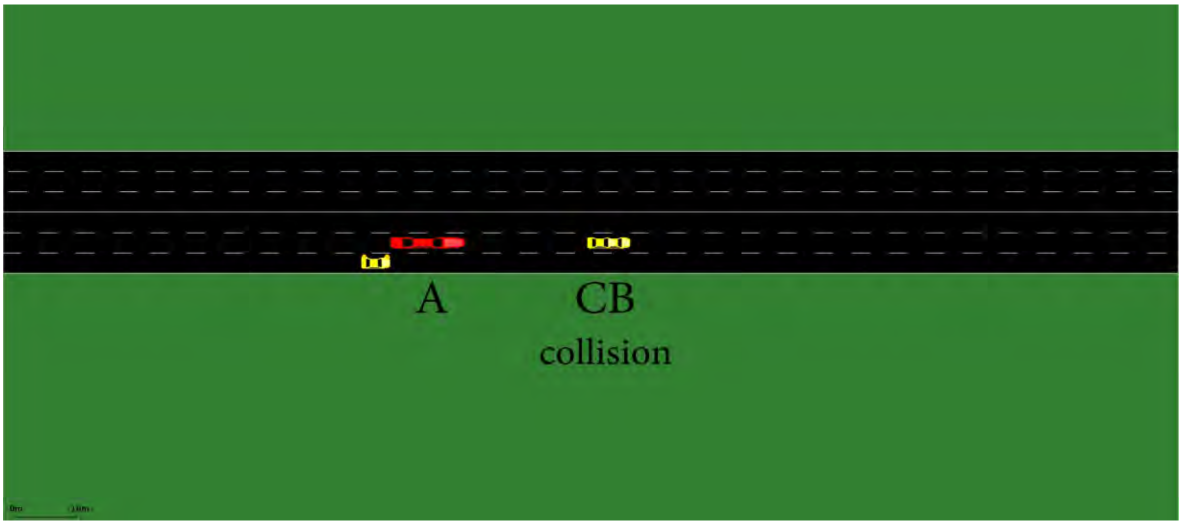

Figure 16 The highway scenario (SUMO environment)

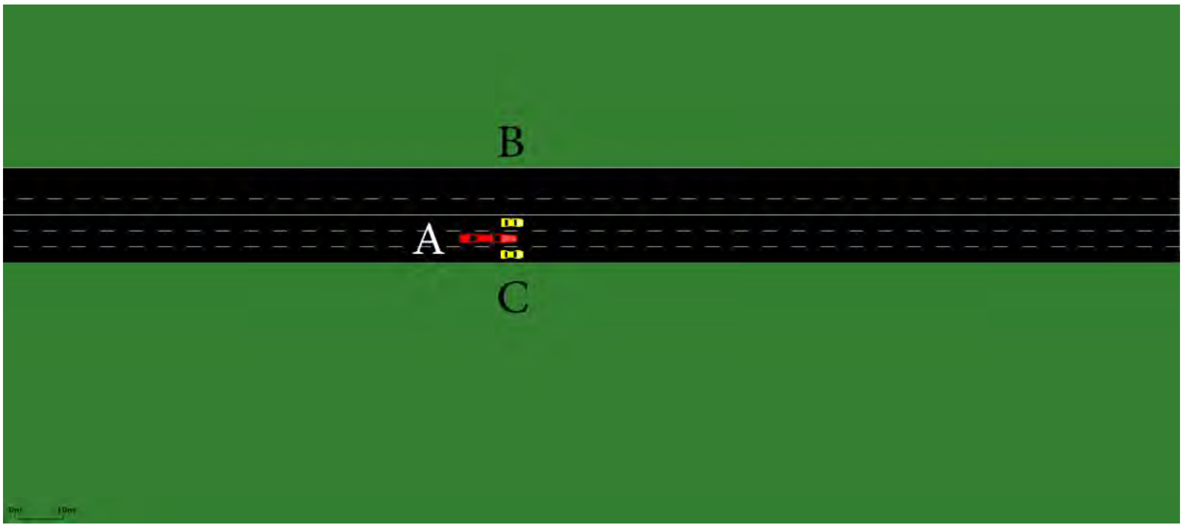

Figure 17 The highway scenario (SUMO environment) (2)

To get a closer look at the background of the highway scenario, imagine that the above road is a highway road. This means that it consists of a big main road with three lanes in each direction. Since this is placed in highway we assume that the maximum velocity of the vehicles exceed in many cases 36m/s which is the limit and reach greater speeds. On the other hand vehicles velocity cannot be less than 14m/s in this type of road.

As we notice in the Figure 16 and Figure 17 above the driver in vehicle A which is a large vehicle moves at the middle lane of the road. Driver in vehicle B decides not to change lane and overtake legally the large vehicle and continues his course in the rightmost lane. Simultaneously Driver in vehicle C legally overtakes the large vehicle. It comes to a point that cars B and C want to change lane and go to the middle one. In this scenario, drivers in vehicles B, C have limited visibility due to large vehicle. If these cars change lane at the same time is certain that they will collide. Therefore, under normal circumstances, taking also into account a possible driver's distraction, and without V2V communication there would be a great possibility that CarB and CarC would crash.

Using V2V communication we try to solve the aforementioned problematic situation. More specifically we want to exchange messages between those vehicles that alert drivers about the oncoming collision. Figure 18 shows exactly the preferable scenario.

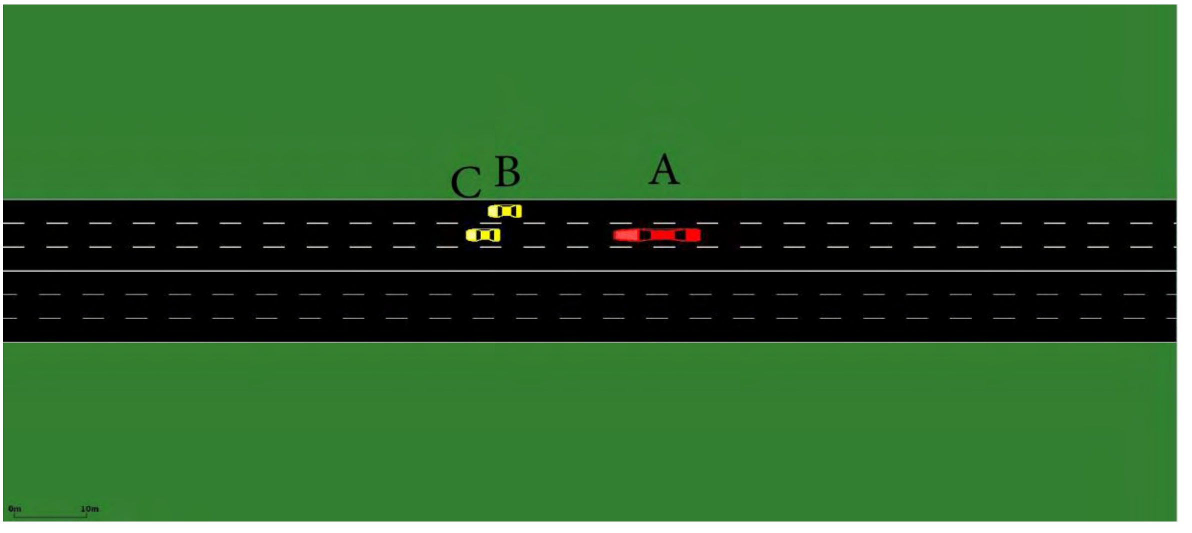

Figure 18 Collision avoidance example in highways

At this point we will describe our proposed solution for collision avoidance in the highway case.

First of all we need to mention that we assume that the cars are heading in the same way and they have a velocity greater than 13m/s and less than 80m/s. The latter velocity is too high if we consider that the speed limit is 36m/s. Moreover the speed of each car differentiate a little through time as a result of the large velocity cars in motorways have. The most important thing of BLCA is that it takes advantage of the roads obstacle (carA) instead of ignoring it. More specifically, carA is used to warn vehicles that are in a collision course.

A pseudo-code for this approach is as follows:

Vehicle A/ Obstacle

1. *VehA Send request asking other cars for current position and speed.*
2. *VehA receive the coords and calculate estimated position in 1.5 second*
3. *if estimated position of two vehicle is within a small range(10m)*
   a. *send warning*
   b. *goto 1 with 1s step*
4. *else*
   a. *goto 1 with 1s step*

Other Cars

1. *Send coords for current position and speed*
2. *if receive Warning with your id*
   a. *adjust speed and respect priority*
3. *else ignore message*

Using this approach we accomplice to find a solution to this scenario. The first major difficulty was to set the beacon interval to communicate with vehicles. A value about 0.1s even if in the first look seems to be a decent choice as we seen above in case 1, has a significant problem. Speed of the cars in a highway tends to change less frequently than in a city, and a 0.1s interval could create much unneeded information of vehicle positions. For those reasons the beacon interval has to be about 1s. We have the tools in this scenario to predict future position of vehicles more easily. A change of speed even about 10m/s is little in ratio of 50m/s speeds. In greater beacon intervals the results are not reliable. So for safety reasons we use so those beacon intervals.

Furthermore, SUMO by default is accident free, so to study the above case scenarios certain default rules had to change. First vehicles must not respect other drivers. They must not respect other vehicle speed and reaction time when changing lanes. Vehicles must do overtakes from the rightmost lane. All the above must happen to simulate a driver with obstructed view and with limited knowledge of the surrounding vehicles. All that are parameters of human fault. To accomplish those challenges we took advantage of TRaCI interface. CarA calculates and estimates future position of the involved vehicles because it has the best possible position to communicate with both overtaking vehicles. Default parameters of OMNeT++ in the physical and mac layer of IEEE 802.11p, provide a theoretical transmission range about 500m which is more than enough.

4. Evaluation of the Results

In this section we provide the results as well as their analysis. We describe the system we used for the experimentation phase and the different measures of evaluation.

4.1 The system of experimentation

Our proposed solution allows vehicles to communicate in a way that makes the avoid collisions in case of blind spots. In order to simulate this case we tested our algorithm on Windows. This system is of 6GB Ram and has a disk capacity of 100GB. An 8-core Intel CPU was running the calculations. We also used Omnet++ revision 4.6 Veins version 4.4 and SUMO version 0.25.0. During all the experiments there was no significant interference from other workloads.

4.2 The experimental settings

As mentioned above, our proposal is the first one in the literature that is suitable both for highways and for urban areas. Therefore, there are no competitors in order to compare efficiency results.

4.3 The performance measures

1. Average beacon rate. This measure represents the average number of beacons that are send each time step. As time step we used seconds.
2. The Vehicles velocity. This represent how fast a vehicle moves on the field.
3. The road traffic density. This represents how many cars appear on the road for a specific area.

4.4 The results

4.4.1 Impact of beacon rate

In this section we present the obtained results. We try to capture here all the advantages that algorithm has to offer in the V2V safety application area. First we present the results to show how our solution behaves for different settings.

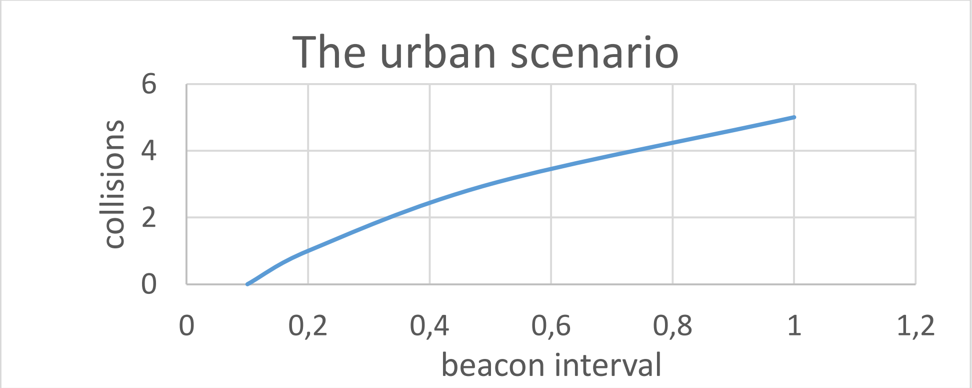

Figure 19 Collisions vs. beacon (urban scenario)

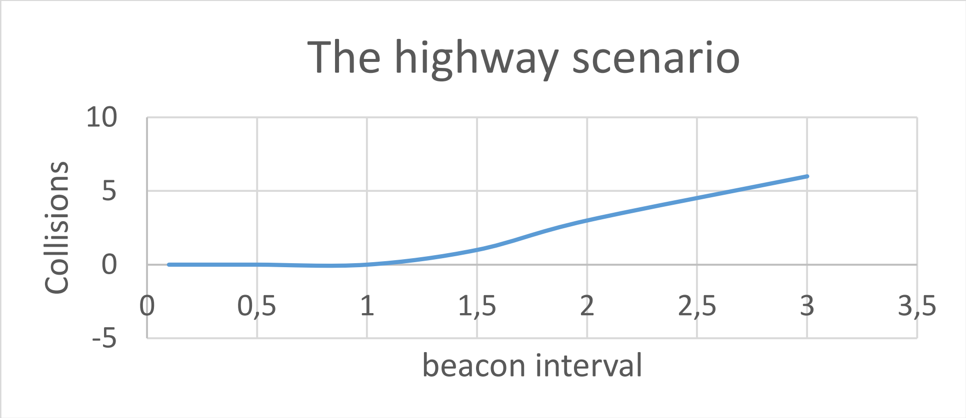

Figure 20 Collisions vs. beacon (highway scenario)

It is obvious from Figure *19* and Figure *20* that the collisions between the cars are growing as beacon interval grows. However in case 2, the is no difference between beacon intervals 0.1 – 1

This is a very important aspect, because it shows both for highways and for urban scenario that we have to choose the right beacon interval with not causing unnecessary overload or collisions.

4.4.2 Impact of the vehicle's velocity

A very interesting outcome of the experimental phase for the urban area case, is that even though the normal considered velocity of a car moving on the streets of a city would be 14m/s, we tried our proposed application with more extreme speeds and it maintained the collision avoidance. More specifically we increased gradually the velocity up to 40m/s. In all situations, we observed that the vehicle-obstacle managed to calculate fast enough the oncoming accident and spread the information efficiently and on time.

Experiment Number	Velocity	Crash or no crash?
1	10 m/s	No crash
2	15 m/s	No crash
3	20 m/s	No crash
4	25 m/s	No crash
5	30 m/s	No crash
6	35 m/s	No crash
7	40 m/s	No crash

Table 1 Crash or No crash?

Table 1 depicts exactly the case described above. We tested the application by increasing in each experimental round the velocity by 5m/s. One can see that no crash occurred during all experiments. This is a highly important outcome for our safety application.

4.4.3 Impact of road traffic density

A very important measurement for the efficiency of our proposed solution is the density of the road traffic. This means we take into account the number of the cars moving in a specific area at a specific time in the experimentation phase.

We noticed that if the beacon intervals are very small, as in the urban scenario and in case of extreme road congestion, then there might occur some data loss. This is however expected, since too many cars are exchanging even much more messages and some of the cars might not receive all data. Although, that scenario does not lead to an increased accident number because the vehicles are almost not moving. It is a situation of course to examine further.

If the cars are moving on a highway, in case of very high road traffic we assume that the speed of the vehicles is decreasing and that the beacon intervals are large, and therefore there might be a wrong judgment of vehicles' future positions. This can be solved by changing dynamically the beacon rate.

5. Conclusion & Future Work

In the last years the advances in communication systems along with the omnipresent connectivity has led to smarter and more intelligent vehicles. Since the initial introduction of V2V communications a lot of applications for cooperative driving have been generated among a plethora of available applications that have been designed those that are the most appealing ones are the ones dealing with driving and road safety.

Smart vehicles are getting more and more aware of their surrounding neighbourhood. This helps the drivers avoid traffic situations like road congestion or accidents. The importance of the latter one is now more than ever imperative. Sometimes the vision of the driver is very restricted especially in crossroads due to unforeseen barriers. (E.g. other large sized cars placed in the corner or a large vehicle commuting in the middle lane of a highway). This is exactly the gap we are trying to fill this work.

In this work, motivated by the latter unsuitability of all existing solutions to avoid accidents both in highways and in urban streets when vision is hindered by road obstacles, we designed a novel road safety application, namely the BLCA application for this purpose. We addressed the challenges in the design, modelling and simulation of V2V communication network applications, which are highly ad hoc especially in case of a highway. For the implementation and testing of our algorithm we used popular network simulation tools such as OMNET++, SUMO and Veins. We assessed its performance for two distinct cases, the highway and a smaller urban street.

We tested the performance in means of different performance measures; among them average beacon rate and velocity of the vehicles. We recognized its virtues and suitability for each case of road.

As a future work, we plan to make some optimizations by tackling different aspects of our algorithm. For example, we want to add more than one obstacles on roads or use more than one "informer" node to warn the drivers in order to make our system highly robust. To this end we also plan to examine the above cases experimentally. Finally, we will also study the case of V2I communications in order to involve infrastructures in our study case.

References

[1] https://en.wikipedia.org/wiki/Intelligent_transportation_system

[2] http://www.csd.uwo.ca/faculty/beau/PAPERS/itsc-13.pdf

[3] R. Sengupta, S. Rezaei, S. E. Shladover, D. Cody, S. Dickey, and H. Krishnan, "Cooperative collision warning systems: Concept definition and experimental implementation," Journal of Intelligent Transportation Systems, vol. 11, no. 3, pp. 143–155, 2007.

[4] S. Dashtinezhad, T. Nadeem, B. Dorohonceanu, C. Borcea, P. Kang, and L. Iftode, "TrafficView: a driver assistant device for traffic monitoring based on car-tocar communication," in Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th, 2004, vol. 5, pp. 2946–2950.

[5] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "TrafficView: traffic data dissemination using car-tocar communication," ACM SIGMOBILE Mobile Computing and Communications Review, vol. 8, no. 3, pp. 6–19, 2004.

[6] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-tovehicle wireless communication protocols for enhancing highway traffic safety," Communications Magazine, IEEE, vol. 44, no. 1, pp. 74–82, 2006.

[7] X. Yang, L. Liu, N. H. Vaidya, and F. Zhao, "A vehicle-to-vehicle communication protocol for cooperative collision warning," in Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on, 2004, pp. 114–123.

[8] A. Lakas and M. Cheqfah, "Detection and dissipation of road traffic congestion using vehicular communication," in Microwave Symposium (MMS), 2009 Mediterrannean, 2009, pp. 1–6.

[9] http://ieeexplore.ieee.org/document/1707356/

[10] http://en.wikipedia.org/wiki/Wireless_ad_hoc_network

[11] http://en.wikipedia.org/wiki/Wireless_mesh_network

[12] Krajzewicz D.; Erdmann J.; Behrisch M.; Bieker L.; "Recent Development and Applications of SUMO – Simulation of Urban MObility", International Journal on Advances in Systems and measurements, vol.5, no.3&4, 2012. Available: http://sumo-sim.org/pdf/sysmea_v5_n34_2012_4.pdf

[13] H. S. Tan and J. Huang, "DGPS-based vehicle-tovehicle cooperative collision warning: Engineering feasibility viewpoints," Intelligent Transportation Systems, IEEE Transactions on, vol. 7, no. 4, pp. 415– 428, 2006.

[14] R. Karlsson, J. Jansson, and F. Gustafsson, "Modelbased statistical tracking and decision making for collision avoidance application," in American Control Conference, 2004. Proceedings of the 2004, 2004, vol. 4, pp. 3435–3440.

[15] J. Jansson and F. Gustafsson, "A framework and automotive application of collision avoidance decision making," Automatica, vol. 44, no. 9, pp. 2347–2351, 2008.

[16] C. L. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan, "Adaptive intervehicle communication control for cooperative safety systems", Network, IEEE, vol. 24, no. 1, pp. 6–13, 2010.

[17] https://en.wikipedia.org/wiki/Mobile_ad_hoc_network

[18] http://en.wikipedia.org/wiki/Wireless_sensor_network

[19] http://en.wikipedia.org/wiki/Wireless_mesh_network

[20] https://en.wikipedia.org/wiki/Vehicular_ad_hoc_network
[21] https://en.wikipedia.org/wiki/IEEE_802.11p
[22] http://docslide.us/documents/omnet-and-mixim-framework-introduction-1-what-is-omnet-omnet-is.html
[23] http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/
[24] https://omnetpp.org/documentation