“Blocking Contagions in Vehicular Networks”

Thesis

Belikaidis Ioannis-Prodromos

“Blocking Contagions in Vehicular Networks”
“Αποτροπή Εξάπλωσης Ιώσεων σε Δίκτυα Οχημάτων”

A Diploma Thesis Presented
By

Belikaidis Ioannis - Prodromos
Μπελικαΐδης Ιωάννης-Πρόδρομος

JULY 2015

Department of Electrical and Computer Engineering
University of Thessaly

ΠΕΡΙΛΗΨΗ

Η παγκόσμια αύξηση του αριθμού των οχημάτων έχει οδηγήσει στην αναποτελεσματικότητα του συμβατικού κυκλοφοριακού συστήματος. Ο πολλαπλασιασμός των ατυχημάτων και η κυκλοφοριακή συμφόρηση οδηγούν καθημερινά στην απώλεια εκατομμυρίων ανθρώπων, χρημάτων και χρόνου. Το δίκτυο επικοινωνίας οχημάτων είναι μία πολλά υποσχόμενη προσέγγιση για την βελτίωση της οδικής ασφάλειας, τη διαχείριση της κυκλοφορίας καθώς και την πληροφόρηση των οδηγών και των επιβατών. Μερικές από τις λειτουργίες του δικτύου είναι για παράδειγμα η ενημέρωση για τις υπάρχουσες θέσεις στάθμευσης, η προϋδοποίηση των οδηγών για την κατάσταση του οδοστρώματος και η πληροφόρηση για διαθέσιμες ξενοδοχειακές ή τουριστικές εγκαταστάσεις που υπάρχουν στην εκάστοτε περιοχή κίνησης του οχήματος. Η πρόσφατη ραγδαία ανάπτυξη στον τομέα της πληροφόρησης και της επικοινωνίας επέτρεψε στα οχήματα να εξελιχθούν σε κινητές υπολογιστικές οντότητες που μπορούν να επηρεαστούν από κακόβουλες επιθέσεις π.χ. κακόβουλο λογισμικό, με τον ίδιο τρόπο που και ο προσωπικός μας υπολογιστής επηρεάζεται. Λαμβάνοντας υπόψη τα πολλαπλά οφέλη που αναμένονται από την οδική επικοινωνία και λόγω του τεράστιου αριθμού των οχημάτων (εκατοντάδες εκατομμύρια σε όλο τον κόσμο), είναι σαφές ότι αυτού του είδους οι επικοινωνίες είναι πιθανό να γίνουν η πιο σημαντική υλοποίηση των κινητών ad hoc δικτύων. Η κατάλληλη ενσωμάτωσή των εποχούμενων υπολογιστών και των συσκευών εντοπισμού θέσης, όπως δέκτες GPS σε συνδυασμό με τις δυνατότητες επικοινωνίας, δημιουργεί τεράστιες επιχειρηματικές ευκαιρίες, αλλά εγείρει και τεράστιες ερευνητικές προκλήσεις. Μία από αυτές τις προκλήσεις είναι η ασφάλεια. Η ασφάλεια αποτελεί κρίσιμο παράγοντα και μία σημαντική πρόκληση από την στιγμή που κάποιος εισβολέας μπορεί να προσπαθήσει να εισάγει ή να τροποποιήσει κρίσιμες πληροφορίες ζωτικής σημασίας, όπως παραδείγματος χάρη να μεταβάλλει την καθορισμένη πορεία ενός οχήματος για προσωπικό του όφελος, απόρροια του οποίου μπορεί να είναι, εκτός των άλλων, η πρόκλιση σοβαρών ατυχημάτων. Ομοίως, το σύστημα θα πρέπει να είναι σε θέση να επιβεβαιώσει την εγκυρότητα των οδηγών, αλλά την ίδια στιγμή, θα πρέπει να προστατεύει τα προσωπικά δεδομένα των οδηγών και των επιβατών. Αυτοί οι προβληματισμοί μπορεί να φαίνονται πανομοιότυποι με αυτούς που συναντάμε σε άλλα δίκτυα επικοινωνίας, αλλά δεν είναι. Ένας από τους απώτερους στόχους στο σχεδιασμό των εν λόγω δικτύων είναι να αντισταθούν στις διάφορες κακόβουλες παραβιάσεις και επιθέσεις ασφαλείας, που μπορούν να προκαλέσουν την κατάρρευση του δικτύου οχημάτων ή γενικότερα στην εξάλειψη όλων των παροχών που προκύπτουν από δίκτυα οχημάτων. Οι

οντότητες σε δίκτυο οχημάτων, δηλαδή οχήματα και μονάδες οδικής υποδομής (RSUs), θα είναι εξοπλισμένες με αισθητήρες και μονάδες ασύρματης επικοινωνίας, ενώ τόσο η όχημα-με-όχημα (Vehicle-to-Vehicle, V2V) όσο και η όχημα-με-υποδομή (Vehicle-to-Infrastructure, V2I) επικοινωνία θα ενεργοποιήσουν τις εφαρμογές ασφαλείας. Κόμβοι με εσφαλμένη συμπεριφορά ή ελαττωματικοί πρέπει να ανιχνεύονται και να εμποδίζονται προκειμένου να μην διαταράξουν την ομαλή λειτουργία του δικτύου. Στην παρούσα εργασία ερευνούμε τον αποκλεισμό της εξάπλωσης του κακόβουλου λογισμικού π.χ. μία μόλυνση, στο δίκτυο των οχημάτων υιοθετώντας το μοντέλο Susceptible-Infectious (SI) από την αντίστοιχη βιβλιογραφία των επιδημιολογικών μοντέλων. Η μελέτη προσομοίωσης που σχεδιάστηκε με τα χαρακτηριστικά του δικτύου των οχημάτων, δημιουργήθηκε για διάφορα σενάρια (αυτοκινητόδρομους και αστικά περιβάλλοντα). Τα αποτελέσματα αυτής της προσομοίωσης δείχνουν την αποτελεσματικότητα, την αποδοτικότητα και την καταλληλότητα της πρότασής μας για ένα πιο ασφαλές δίκτυο.

Η συγκεκριμένη εργασία οργανώνεται ως εξής: Στο κεφάλαιο 1 κάνουμε μία εισαγωγή στα διάφορα δίκτυα και αναλύουμε την επικοινωνία αλλά και τις εφαρμογές των δικτύων οχημάτων. Στο κεφάλαιο 2 συζητάμε τα προβλήματα ασφαλείας που επηρεάζουν αυτά τα δίκτυα και στην συνέχεια συναντάμε μία λεπτομερή περιγραφή του πως εξαπλώνετε μία ίωση στα δίκτυα που μελετάμε καθώς και του επιδημιολογικού μοντέλου που χρησιμοποιήσαμε. Στο κεφάλαιο 4 αναφερόμαστε στο προτεινόμενο πλαίσιο και τους μηχανισμούς άμυνας που υιοθετήθηκαν στα πειράματά μας. Οι αλγόριθμοι που χρησιμοποιήθηκαν στα προτεινόμενα αυτά σενάρια αναφέρονται επίσης, στο συγκεκριμένο κεφάλαιο. Στο κεφάλαιο 5 παρουσιάζονται τα εργαλεία προσομοίωσης και ο πειραματισμός που χρησιμοποιήθηκε για τα VANETs μας, καθώς και τα αποτελέσματα που προέκυψαν από την προσομοίωση.

ACKNOWLEDGMENTS

I am grateful to Dr Panayiotis Bozanis and Dr Athanasios Korakis for their condescension, trust and understanding throughout the duration of this Thesis.

Also I offer my sincerest gratitude to Dr Dimitrios Katsaros and Ph.D. student, Pavlos Basaras, who have supported me throughout my ΤΔΗΠΛΟΜΑΤΙΚΗhesis with patience, guidance and knowledge whilst allowing me the room to work in my own way.

A special thanks to my department's faculty, staff and fellow students for their valuable assistance whenever needed and for creating a pleasant and creative environment during my studies.

Last but not least, I wish to thank my family and friends for their unconditional support and encouragement all these years.

ABSTRACT

With the increase in the number of vehicles in the world, the transportation system has become inefficient. Increasing accidents and traffic jams are leading to loss of millions of lives, money, and time, year after year. Vehicular communication networking is a promising approach to facilitating road safety, traffic management, and infotainment dissemination for drivers and passengers. Vehicular network nodes, that is, vehicles and Road-Side infrastructure Units (RSUs) will be equipped with sensing, processing, and wireless communication modules and both Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication will enable safety applications. Recent advances in information and communications technologies led vehicles to evolve into mobile computational entities that can be affected from malicious attacks, e.g. malware, the same way as our personal computer could. As a result, security is a critical factor and a significant challenge to be met since an attacker may try to insert or modify life-critical information. One of the ultimate goals in the design of such networking is to resist various malicious abuses and security attacks. Misbehaving or faulty network nodes have to be detected and prevented from disrupting network operation. In this thesis we investigate on blocking the outspread of malware, e.g. a worm-like virus in vehicular networks by adopting the Susceptible-Infectious (SI) model from the literature of disease spreading. A simulation study designed to the Vehicular Network characteristics is created for various (highway and urban) scenarios. The simulation results show the effectiveness, efficiency and the suitability of our proposal for a more secure network.

TABLE OF CONTENTS

Page

MOTIVATION

Considering the tremendous benefits expected from vehicular communications and the huge number of vehicles (hundreds of millions worldwide), it is clear that vehicular communications are likely to become the most relevant realization of mobile ad hoc networks. The appropriate integration of on-board computers and positioning devices, such as GPS receivers along with communication capabilities, opens tremendous business opportunities, but also raises formidable research challenges. One of these challenges is security. Limited attention has been devoted so far to the security of vehicular networks. Yet, security is crucial. For example, it is essential to make sure that life-critical information cannot be inserted or modified by an attacker. Likewise, the system should be able to help establishing the liability of drivers but at the same time, it should protect as far as possible the privacy of the drivers and passengers. These concerns may look similar to those encountered in other communication networks, but they are not. Indeed, the size of the network, the speed of the vehicles, the relevance of their geographic position, the very sporadic connectivity between them, and the unavoidably slow deployment make the problem very novel and challenging.

CHAPTER 1

INTRODUCTION

Wireless Ad-hoc network [1] (Figure 1) is defined as a network which doesn't have a preexisting communication infrastructure. Network is created by some nodes which are available. In this type of network determination of which nodes to transfer data to which node is done dynamically, depending upon the connectivity of both devices. Ad-hoc network can use flooding data transfer. All devices are treated equally and therefore have the same status. The main use of wireless ad-hoc network is done by MANET. In MANET different participating node moves randomly in the created wireless Ad-hoc network.

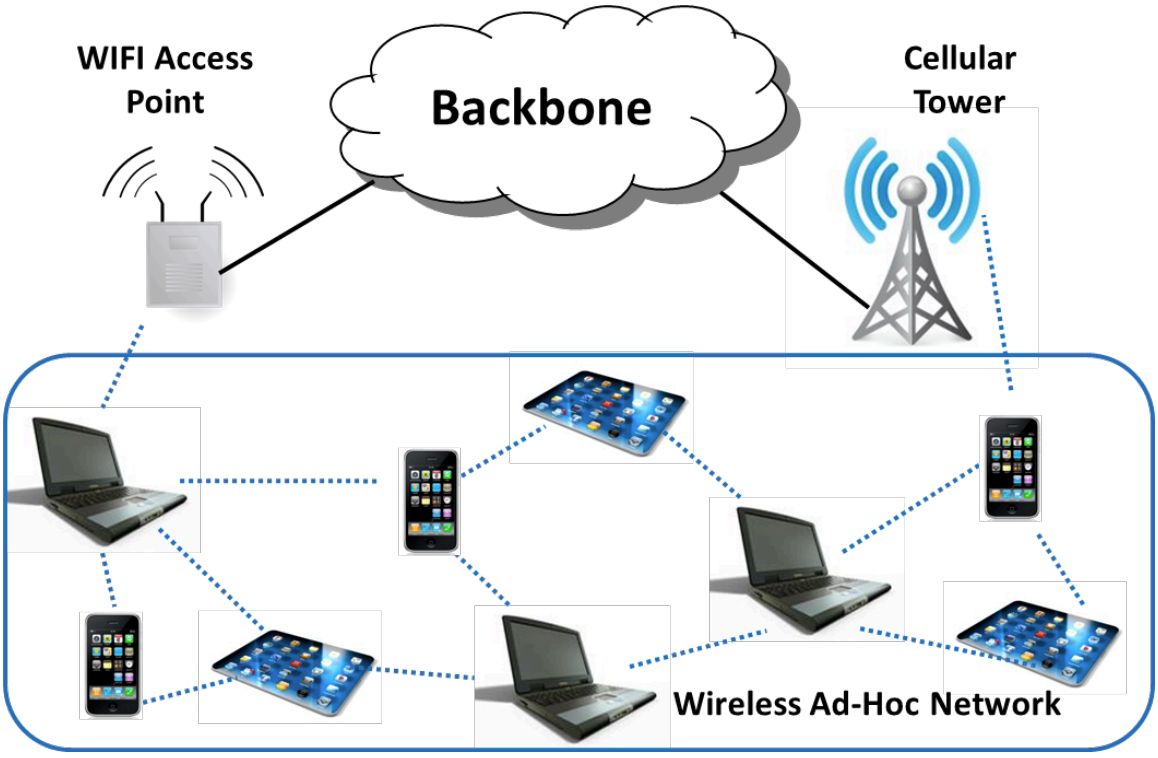


Figure 1 Wireless Ad-Hoc Network Architecture

MANET

MANET [5] stands for "Mobile Ad Hoc Network". A MANET is a type of ad hoc network that can change locations and configure itself on the fly. Because MANETS are mobile, they use wireless connections to connect to various networks. This can be a standard Wi-Fi connection, or another medium, such as a cellular or satellite transmission. Some MANETs are restricted to a local area of wireless devices (such as a group of laptop computers), while others may be connected to the Internet.

VANET

VANETs stands for “Vehicular Ad Hoc Network “. A VANET is a type of MANET and a part of ITS that allows vehicles to communicate with each other or with roadside equipment. While the vehicles may not have a direct Internet connection, the wireless roadside equipment may be connected to the Internet, allowing data from the vehicles to be sent over the Internet. For example, vehicle data may be used to measure traffic conditions or keep track of trucking fleets.

ITS

Stands for “Intelligent transportation systems”. A ITS is a type of networking system in which, each vehicle takes on the role of sender, receiver, and router to broadcast information to the vehicular network or transportation agency, which then uses the information to ensure safe, free-flow of traffic. For communication to occur between vehicles and Road Side Units (RSUs), vehicles must be equipped with some sort of radio interface or On Board Unit (OBU) that enables short-range wireless ad hoc networks to be formed. Vehicles must also be fitted with hardware that permits detailed position information such as Global Positioning System (GPS) or a Differential Global Positioning System (DGPS) receiver. Fixed RSUs, which are connected to the backbone network, must be in place to facilitate communication. The number and distribution of roadside units is dependent on the communication protocol is to be used. For example, some protocols require roadside units to be distributed evenly throughout the whole road network, some require roadside units only at intersections, while others

require roadside units only at region borders. Though it is safe to assume that infrastructure exists to some extent and vehicles have access to it intermittently, it is unrealistic to require that vehicles always have wireless access to roadside units.

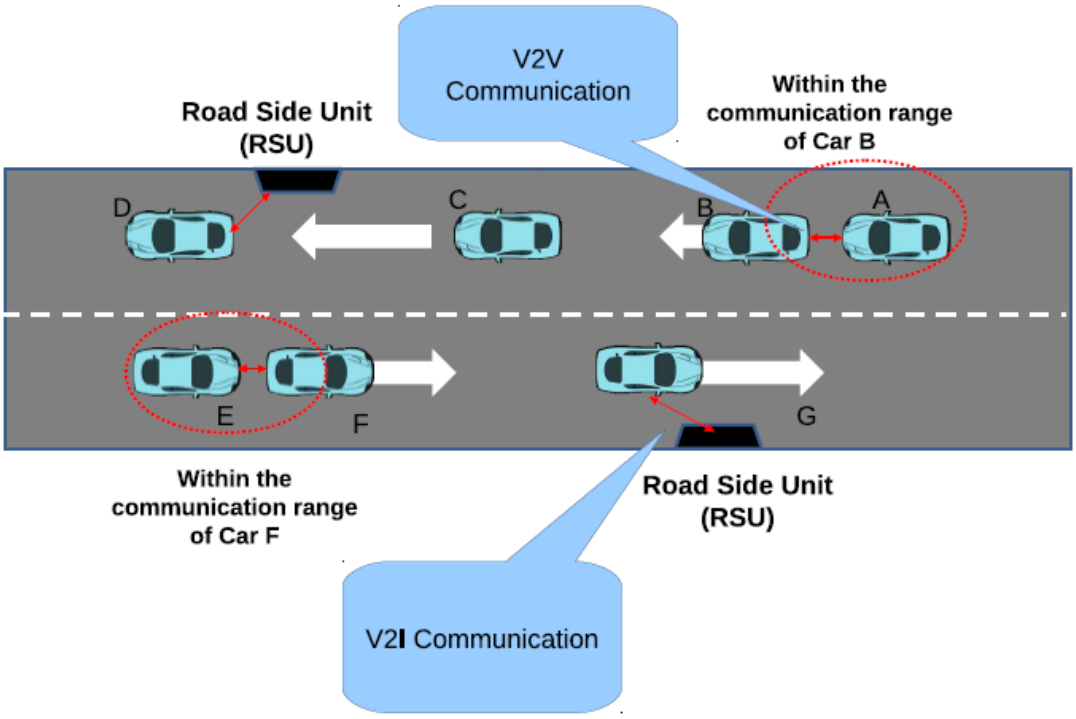


Figure 2 VANET communications.

Figures 2 depict the possible communication configurations in intelligent transportation systems. These include V2V and V2I communications. These communications rely on very accurate and up-to-date information about the surrounding environment, which, in turn, requires the use of accurate positioning systems and smart communication protocols for exchanging information.

V2I

The vehicle-to-infrastructure (V2I) communication is a technology that allows cars to communicate with infrastructure elements, like stop-lights, road sings or street lights. Vehicles can sent information to infrastructure and receive messages from infrastructure for example to improve road safety and infotainment.

I2V

The infrastructure-to-vehicle (I2V) communication configuration represents a single hop broadcast where the roadside unit (RSU) sends a broadcast message to all equipped vehicles in the vicinity. V2I communication configuration provides a high bandwidth link between vehicles and roadside units. The roadside units may be placed every kilometer, enabling high data rates to be maintained in heavy traffic.

V2V

Vehicle-to-vehicle (V2V) communications comprises a wireless network where automobiles send messages to each other with information about "what they're doing". This data would include speed, location, direction of travel, braking and loss of stability. Vehicle-to-vehicle technology uses dedicated short-range communications (DSRC), a standard set forth by bodies like FCC and ISO.

I2I

Infrastructure-to-infrastructure (I2I) communications refers to the connection between RSUs into a network that plays a coordination role by gathering local information on traffic and road conditions and then suggesting or imposing global certain behaviors on vehicles inside their area. For example due

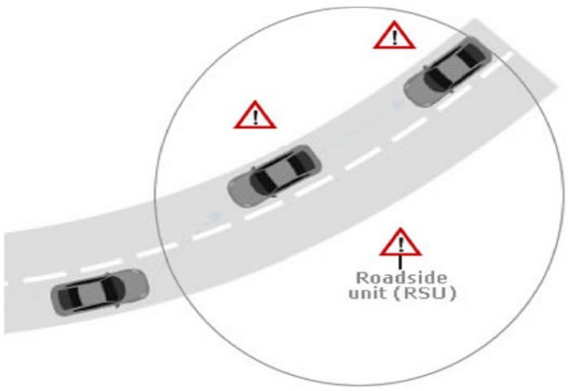
to greater congestion, infrastructure act to improve fuel efficiency and reduce emissions of individual vehicles, smoothing accelerations and decelerations.

APPLICATIONS OF VANETS

The primary goals of VANETs are to improve safety on the road. To achieve this, the vehicles act as sensors and exchange messages to different vehicles this messages include information like speed of vehicle, condition of road, Traffic density. This enables the drivers and authorities to react early to any dangerous situations like accidents and traffic jams. But the recent researches in the field of VANET have discovered many applications and technologies.

- Application for avoiding collision through distance calculation between two vehicles it can use sudden braking system.
- Application for detection of hazardous and dangerous driving conditions. This conditions can be damaged road, blocked road, if road is covered with snow or mud.
- Application for emergency call services after an accident occurs here the vehicle can automatically call to authority if an accident occurs.
- Applications for detecting rogue drivers which are disobeying traffic rules like crossing speed limit, talking in phone while driving, driving in the wrong side of the road.
- Application for Advanced Navigation Assistance (ANA) such a car park formation, real time vehicle congestion information, expected weather condition for driving, etc.,
- Internet connection services can be provided to vehicle added for travel comfort and improved productivity. This be done by data transfer between vehicle and road side unit.
- Application for advertisement of local/nearest service stations, nearest hotel, shops, mall.

Figures 3 and 4 show some of the applications that have been listed above.



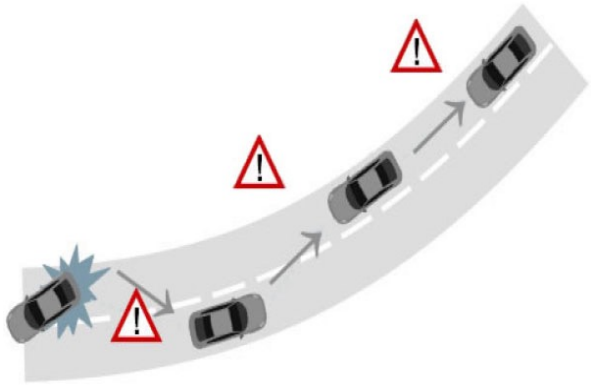


Figure 3 VANET application **Figure 4 VANET application.**

CHAPTER 2

SECURITY

The security of VANETs is crucial as their very existence relates to critical life threatening situations. It is imperative that vital information cannot be inserted or modified by a malicious entity. The system must be able to determine the liability of drivers while still maintaining their privacy. These problems are difficult to solve because of the network size, the speed of the vehicles, their relative geographic position, and the randomness of the connectivity between them. An advantage of vehicular networks over the more common ad hoc networks is that they provide ample computational and power resources. The reliability of a system where information is gathered and shared among entities in a VANET raises concerns about data authenticity. For example, a sender could misrepresent observations to gain advantage (e.g., a vehicle falsely reports that its desired road is jammed with traffic, thereby encouraging others to avoid this route and providing a less congested trip). More malicious reporters could impersonate other vehicles or roadside infrastructure to trigger safety hazards. Vehicles could reduce this threat by creating networks of trust and ignoring, or at least distrusting, information from untrusted senders.

Threats can be broadly categorized into three main groups, Availability, Authenticity and Confidentiality attacks.

AVAILABILITY

Denial of Service Attack: DoS attacks can be carried out by network insiders and outsiders and renders the network unavailable to authentic users by flooding and jamming with likely catastrophic results. Flooding the control channel with high volumes of artificially generated messages, the network's nodes, onboard units and roadside units cannot sufficiently process the surplus data.

Broadcast Tampering: An inside attacker may inject false safety messages into the network to cause damage, such as causing an accident by suppressing traffic warnings or manipulating the flow of traffic around a chosen route.

Malware: The introduction of malware, such as viruses or worms, into VANETs has the potential to cause serious disruption to its operation. Malware attacks are more likely to be carried out by a rogue insider rather than an outsider and may be introduced into the network when the onboard units and roadside units receive software and firmware updates.

Spamming: The presence of spam messages on VANETs elevates the risk of increased transmission latency. Spamming is made more difficult to control because of the absence of a basic infrastructure and centralized administration.

AUTHENTICITY

Masquerading: Masquerading attacks are easy to perform on VANETs as all that is required for an attacker to join the network is a functioning onboard unit. By posing as legitimate vehicles in the network, outsiders can conduct a variety of attacks such as forming black holes or producing false messages.

Replay Attack: In a replay attack the attacker re-injects previously received packets back into the network, poisoning a node's location table by replaying beacons. VANETs operating in the WAVE framework are protected from replay attacks but to continue protection an accurate source of time must be maintained as this is used to keep a cache of recently received messages, against which new messages can be compared.

Global Positioning System (GPS) Spoofing: The GPS satellite maintains a location table with the geographic location and identity of all vehicles on the network. An attacker can fool vehicles into thinking that they are in a different location by producing false readings in the GPS positioning system devices. This is possible through the use of a GPS satellite simulator to generate signals that are stronger than those generated by the genuine satellite.

Tunneling: An attacker exploits the momentary loss of positioning information when a vehicle enters a tunnel and before it receives the authentic positioning information the attacker injects false data into the onboard unit.

Position Faking: Authentic and accurate reporting of vehicle position information must be ensured. Vehicles are solely responsible for providing their location information and impersonation must be impossible. Unsecured communication can allow attackers to modify or falsify their own position information to other vehicles, create additional vehicle identifiers (also known as Sybil Attack) or block vehicles from receiving vital safety messages.

Message Tampering: A threat to authenticity can result from an attacker modifying the messages exchanged in vehicle-to-vehicle or vehicle-to-roadside unit

communication in order to falsify transaction application requests or to forge responses.

Message Suppression/Fabrication/Alteration: In this case an attacker either physically disables inter-vehicle communication or modifies the application to prevent it from sending to, or responding from application beacons.

Key and/or Certificate Replication: Closely related to broadcast tampering is key management and/or certificate replication where an attacker could undermine the system by duplicating a vehicle's identity across several other vehicles. The objective of such an attack would be to confuse authorities and prevent identification of vehicles in hit-and-run events.

CONFIDENTIALITY

Confidentiality of messages exchanged between the nodes of a vehicular network are particularly vulnerable with techniques such as the illegitimate collection of messages through eavesdropping and the gathering of location information available through the transmission of broadcast messages. In the case of eavesdropping, insider and/or outsider attackers can collect information about road users without their knowledge and use the information at a time when the user is unaware of the collection. Location privacy and anonymity are important issues for vehicle users. Location privacy involves protecting users by obscuring the user's exact location in space and time. By concealing a user's request so that it is indistinguishable from other users' requests, a degree of anonymity can be achieved.

MALICIOUS NODE DETECTION

As we mentioned there are several attacks and misbehaviors in VANETs which not only affect the driver's and vehicle's privacy but also compromise traffic safety and may lead to loss of life. Misbehavior can be generally referred to as any kind of abnormal behavior that is deviation from the average behavior of other vehicular nodes in the VANETs. In order to become a real technology that assures traffic safety VANETs require appropriate security techniques and mechanisms that will guarantee protection against various misbehaviors and malicious nodes that affects security of VANET. As a result malicious node detection and classification of misbehavior node detection techniques in VANETs is very important. In the littareture are presented

various efforts by researchers under **Node-Centric** and **Data-Centric** Misbehavior Detection Techniques and Figure 5 shows the hole taxonomy.

NODE CENTRIC

Node-Centric mechanisms require authentication mechanisms to reliably distinguish between different nodes. Many systems achieve this by assuming a trusted third party that issues credentials, which are then used to authenticate messages and the corresponding information, using a security mechanism like digital signatures. Node-centric mechanisms can further be divided into **behavioral** and **trust-based** mechanisms.

Behavioral mechanisms inspect a node's observable behavior (but not the information it is sending) and try to derive a metric that identifies how well a node behaves. For instance, a behavioral mechanism may inspect rates at which a neighboring node sends packets and decide whether a node significantly exceeds a "normal rate," which would then be considered as misbehavior.

On the other hand, **trust-based** mechanisms inspect the past and present behavior of a node and use this to derive a probability for future misbehavior. The assumption is that a node that behaved correctly in the past is more likely to behave correctly in the future. Essentially, this boils down to some form of reputation management scheme where correct behavior increases the reputation while misbehavior reduces it. These mechanisms are commonly used for reporting and local revocation of nodes in a VANET.

DATA CENTRIC

In contrast to those Node-Centric mechanisms, the second major category, namely Data-Centric misbehavior detection, subsumes all mechanisms that directly inspect the disseminated information to detect potential misbehavior. While Data-Centric mechanisms do not primarily care about the identities of individual nodes, they often still require some form of linking between messages to be able to reliably distinguish between different hosts. However, these mechanisms do not depend on the link-ability of messages, which makes them highly valuable for the detection of Sybil

attacks. Due to the strong privacy requirements in VANETs compared to other cyber-physical systems, which makes linkage between different messages more difficult, concerns for Sybil attacks are particularly relevant. In response to this, many VANET researchers have developed novel schemes to perform Data-Centric misbehavior detection. These can be divided further into **consistency** and **plausibility** mechanisms.

Of these two types, **consistency** mechanisms rely more strongly on protection against Sybil attacks. The purpose of consistency mechanisms is to compare measurements from different entities to detect and, where possible, resolve conflicts between these measurements. For instance, in a VANET, a single vehicle could report a severe traffic jam while other vehicles report free flow of traffic. A consistency-based mechanism would use such information to conclude that there is likely no traffic jam and that the single vehicle may have misbehaved or be faulty.

Finally, **plausibility** checking mechanisms are all mechanisms that have some implicit or explicit model of the real world and check whether incoming information is plausible within this model. For instance, in VANETs, speed reports of 700 km/h are not very plausible and may be filtered out. However, plausibility should be applied with caution in VANETs, as part of the focus of such networks is to detect outliers that indicate important, but rare, events, such as collisions between vehicles.

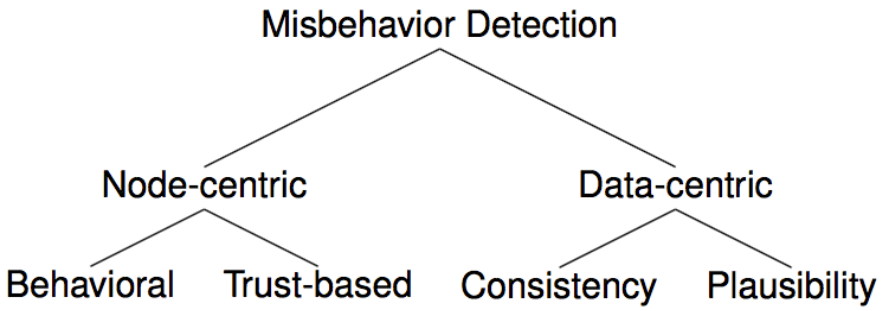


Figure 5 Taxonomy of misbehavior detection.

Security in VANETS is a big issue and there are many categories to be researched. In this thesis we focus on a particular category of attacks in VANETs, more specific malware attacks through viruses and worms propagation. Also misbehavior detection has a critical role in our proposal for a more secure Vehicular Network.

CHAPTER 3

In order to propose our mechanism and technics for a better blocking of malicious propagation in VANETs we need first to present what literature has to provide and the work that has been done from researchers all these years in a very significant and difficult be to explained matter.

MALWARE PROPAGATION

MALWARE

Worms unlike viruses, which attach parasitically to a normal program, are stand-alone automated malware which propagate thorough a network without any human intervention. In recent years they have emerged as one of the most prominent threats to the security of computer networks. A worm attack on VANET may interfere with critical applications such as engine control and safety warning systems hence resulting in serious congestion on the road networks and large-scale accidents. While there has been much research on the dynamics of worm spreading on the Internet there has been, to our knowledge, very few studies of worm epidemics in mobile ad hoc network in general and vehicular ad hoc networks in particular. Such studies are critical for assessing the risks associated with worm attacks on VANET, and devising effective countermeasures and techniques for their detection and mitigation.

PROPAGATION IN OUR SIMULATION

So far, malware on vehicular networks are studied as worm epidemics that self-propagate across vehicular nodes under the Susceptible-Infected model from the literature of the spreading of diseases in epidemiology. Generally, a worm's functions are:

- **Target discovery**: the way targets to propagate are discovered.
- **Carrier**: the infection mechanism used to propagate.

- **Activation**: how the worm starts its activity.
- **Payload:** the set of routines executed by the worm depending on the objective of the attack.

Our interest lies in the spread of the worm and thus we focus in the first two categories.

Target discovery is facilitated through beacon messages in V2V communications and thus an infected vehicle become aware of its neighbors within its range, i.e. potential victims. As far as the **carrier** mechanism is concerned, following the literature we utilize two approaches. First **broadcast carrier**, were a vehicle can infect all its neighbors at once, and second, **unicast carrier** were a worm can infect only one susceptible neighbor at a time. The carrier mechanism is also characterized by a second aspect, i.e. the number of transmissions (broadcast or unicast) required to complete the infection. This value depends on the length of the worm's code and on the way it is hidden in the messages. We translate this aspect to a second parameter, referred to as carrier latency and indicated as T in the following. The carrier latency is the amount of time a worm needs to self-propagate to all of its neighbors (in the broadcast case). We remark that T accounts for eventual protocol-related delays, due, e.g., to association or session establishment procedures, wireless channel contention or lost message retransmissions.

MODELING SI SPREADING FOR VANETS

Considering the worm epidemics from the viewpoint of the whole network, and borrowing the terminology from epidemiology, we will adopt a **Susceptible**, **Infected** (SI) model with Immunization. According to this model, **Susceptible** is a clean vehicle that has not been infected by any kind of malware and can become infected only by another vehicle that is currently infected and in close proximity, i.e. its neighbors. On the other hand **Infected** vehicles remain in this status without the ability to become recovered or susceptible again. This is due to the fact that we assume that the patch is not yet available in the area under attack.

$$S \rightarrow I$$

We start with the first infected vehicle as initial spreader, and its location at the time it was first infected as the origin of the worm infection. The population affected by the spread of the virus is formed by all the communication-enabled vehicles circulating in the geographical area of interest that suffer from the security flaw exploited by the worm. It is possible that a specific worm (or worm code) cannot infiltrate all vehicles running in the simulation. We thus characterize the population

that can be affected by the worm through a penetration rate parameter, indicated as P, i.e. the fraction of vehicles participating in the vehicular network and susceptible of being infected from the worm.

CHAPTER 4

BUILDING A SOLUTION

So far we have analyzed what a VANET environment is, the advantages, the disadvantages and also the important role of security in these networks. Recent advances in information and communications technologies led vehicles to evolve into mobile computational entities that can be affected from malicious attacks, fact that makes the protection of VANET more than necessary. If left unprotected against attacks, it can directly lead to the corruption of the vehicular network and possibly provoke big losses of time, fuel and thus money, or even lives.

An initial solution that could be used, is that when a vehicle is recognized, e.g. from an RSU as malicious/infected, this vehicle must be cut off from the rest of the networked environment. Moreover a large number (if not all) of its neighboring cars must also be omitted, since they have been in contact with the infected vehicle which may have caused infection to those vehicles, i.e. if the malware propagated to them. A solution like this one lead us to conclude that protocols based on information accumulated by vehicles cannot be employed since most or the entire network connectivity may be shut down due to potential infection. In order to be functional, these protocols need at least "some" connectivity between vehicles, i.e. necessary information to circulate inside the VANET proximity. In the current study, instead of cutting all the network links we use clustering techniques, which separate with certain criteria the neighbors of the verified malicious vehicles. We apply the K-means clustering algorithm, and split the neighborhood in K parts. The basis for the classification criteria are twofold for the highway scenario and threefold for an urban environment. First, we focus to those vehicles with the highest probability of being infected. High probability for infection refers to vehicles that have been in contact with the infected vehicle for longer time than new neighbors that just enter an infected cars vicinity. Second, we turn our attention to those vehicles that are highly connected and thus can affect a large number of other vehicles, i.e. influential vehicles. Finally, in an urban scenario, we apply a weight function based on a well studies heuristic from the literature of graphs, namely the PageRank algorithm, which gives weights to the road segments in the area under consideration. Thus the current categorization of the K parts is based on the weight of the next road segment that each vehicle is about to follow, i.e. its trajectory.

PRELIMINARY WORK

Most of the so far proposed studies, focus in disseminating a patch, i.e. cure [5][6][8][9][10][11], to the infected vehicles in order to dispose the malware, and to susceptible vehicle-nodes in order to immunize them. The propagation of the cure can occur in different ways, e.g. through V2V communications or through 3/4G networks. Nonetheless a cure for a "new" virus may be dispatched to the area under attack with significant delays and moreover spreading the cure through vehicle communications can also be significantly delayed due the very nature of vehicular networks. In our work we focus on preventing the outspread of malware before the patch arrives in the area under consideration by cutting a portion of "important connections" in the vehicular network.

INTRUSION DETECTION MECHANISM

First we need to discuss how to detect potential infection in a vehicular network. In the literature of malware propagation in MANETS a number of proposed algorithms are devised in order to "break down" packets exchanged between nodes to detect potentially infected ones. However we cannot "trust" vehicles for such an important task for various reasons. First, an infected vehicle can tweak the result of the algorithm and say that a "clean" vehicle is infected (or that an infected car is clean), mislead the near vehicles, and thus harm the network in a cascade of such events. Even if detection was realized through vehicles, we cannot expect all vehicles to be capable of detecting viruses in exchanged messages. This fact implies potentially unprotected group of vehicles, which coupled with the nature of a vehicular network renders such an approach as a dangerous one. To this end we entrust the detection of malware in exchanged messages from vehicles to the RSUs which are scattered throughout the simulation map, in each evaluated case scenario (highway and urban environments).

In the current framework we utilize the literature of wireless sensor networks (WSNs) concerning Intrusion Detection Systems (IDS) [7] in order to deduct with a certain probability if a vehicle has sent malware messages. There are two important classes of IDSs: rule-based and anomaly-based.

Rule-based: can detect well-known attacks with great accuracy, but are unable to detect new attacks.

Anomaly-based: can detect both well-known and new attacks but they have more false positives and false negative alarms.

Note that we do not implement IDS mechanisms from the literature (it is beyond the purposes of the current study). Instead, probabilities are used for false positives-negatives which are processed in the RSUs.

CONFIGURING COMMUNICATION FOR THE PARTICIPATING ENTITIES (VEHICLES & RSUS)

As in typical VANETs, in our simulation vehicles periodically broadcast beacon/heartbeat messages in order to become aware of their surrounding vehicles, i.e. their vicinity. Nonetheless, the continuously changing topology of vehicular nodes due to the road network structure or the difference in mobility patterns between any two such nodes renders the vicinity of vehicles a continuously changing set of nodes. Thus it is assumed that a specific vehicle is no longer in proximity, if for example no beacons were received in two periods of beacons exchanges.

The current "neighbor list" is not the only information extracted from beacon messages. Generally, typical information included in beacons are the sending vehicle's current velocity, position from its GPS system (i.e. coordinates) or via proximity sensors, direction of movement and destination.

FIELDS OF EXCHANGED MESSAGES

In order to build on the proposed defense mechanism a number of fields must be added in the exchanged messages in order to decide the most important or most vulnerable nodes in the vicinity of each vehicle. To this end we add two fields in the exchanged packets. First, each node sends the number of nodes (and their IDs) which compose its current neighbor set, i.e., its degree. This choice follows from the literature of influential spreaders in complex networks, where nodes with high degree, i.e. many neighbors, can influence (propagate to) a large number of other nodes. Intuitively cutting links to those nodes will substantially hinder the outspread of malicious messages. Second, the duration that any two vehicles are connected also plays an important role. For example if a vehicle is detected as infected, the neighbors which have been in contact with the infected source for longer period have greater probability to be infected than those with little contact time, i.e. relatively new neighbors. With the above consideration we focus on these two metrics with different objectives, i.e. classify a vehicle's neighbors with respect to their connectivity, or with respect to their

contact duration. To summarize the accumulated information for the exchanged messages accumulated by each vehicle from its neighbors is:

- List of neighbor IDs
- List of neighbor degrees
- List of neighbor contact durations
- List of neighbor velocities
- List of neighbor positions
- List of neighbor direction of movement

The above information will hold the basis for building our proposed defense system.

RSU PLACEMENT & COMMUNICATION PHASES

As we already mentioned, the means to deduct which vehicles are infected are the RSUs.

Highway scenario

As mentioned the detection of malicious vehicles will take place in the RSUs. In a highway scenario the distance between any two consecutive RSUs is important. If the communication range between them overlaps, then we assume that all infected vehicles can be detected instantly since the exchanged messages can be "heard" from the RSUs at any position. Nonetheless, such a set up implies a large number of RSU placements, which is both unrealistic and expensive. In our case study we assume that RSUs are placed in distance of *m* meters between them, e.g. 1000*m*, and thus having a *void space* (where no RSU can hear the exchanged messages), assuming DSRC of 300m range of communications. Vehicles become aware of the area controlled by an RSU by periodic messages (RSU presence) broadcasted by the corresponding RSU. Thus void spaces are noted by the absence of such messages. In our experimentation we assume that infected vehicles act only in void spaces. Figure 6 illustrates the set up used in our experimentation. The 400m space is the area where malicious vehicles can act since they cannot "hear" RSU presence messages from the RSUs.

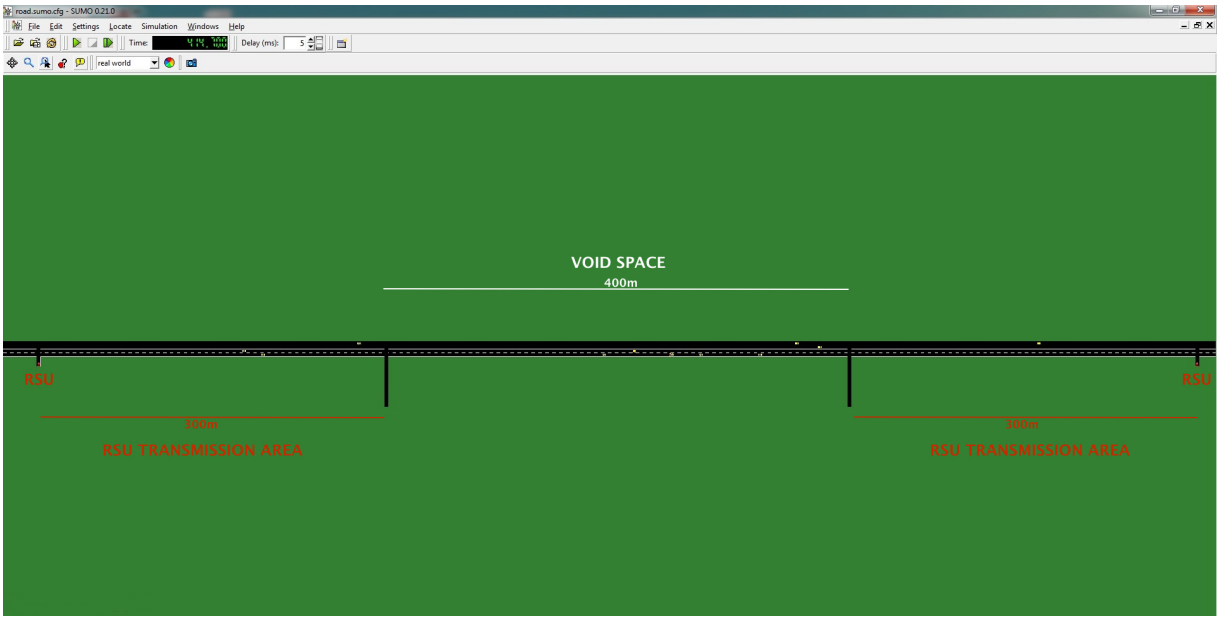


Figure 6 Highway spaces

Upon hearing an RSU presence message, i.e. entering the range of communication of the corresponding RSU, a vehicle responds to the RSU according to the protocol's specifications. This is where the RSU analyzes the message from the vehicle and deducts whether the vehicle is infected or not. If the vehicle is "clean" then the simulation flows normally. However, if the vehicle is found infected a number of actions need to take place. Note that RSUs upload the infected vehicle's id in a common database shared by all RSUs. Thus, infected vehicles are known even in different locations than the current whereabouts of the infected vehicle.

Urban scenario

The challenges met in an urban scenario are significantly more than those in the highway case. Here we have to account for cases of buildings which may interfere with the communication, a specific road topology (defined by the road network) and various directions and destinations for the participating vehicles. Thus placing RSUs can be a very challenging scenario in this particular case. However optimal placement of RSUs with respect to the obstacles (buildings) and road structure is beyond the scope of the current study and thus we employ a simple placement mechanism.

CUTTING LINKS, INFECTED & POTENTIALLY INFECTED NODES

Once a vehicle is identified as infected, its id is blacklisted and uploaded to the common database. In order to protect vehicles from the infected mobile node, the RSU periodically broadcast the list of infected ids (the black list, *BL*) and instructs healthy vehicles to cut any communication between the ids mentioned in the list. Thus the infected nodes are isolated from the rest of the environment. As a second precaution measure we need to account for the neighbors of the infected vehicle. Since they have been in contact with an infected source, some (or even all) vehicles may have been infected. In our case study we do not take prompt action to delete all neighbors of the infected source, since as we mentioned such drastic action may result in the failure of protocols which require ”some” connectivity to obtain information for the traffic/road condition. To this end we utilize a second list of vehicle ids, namely *potentially infected list (PIL),* which holds a portion of the infected vehicle's vicinity. Similarly to the BL, vehicles are also instructed to drop any packets received for those vehicles until further instructions, i.e. PIL is also periodically broadcasted.

POTENTIALLY INFECTED LIST (PIL)

PIL as we mentioned is an equally important list of vehicle ids, and in particular links that possibly need to be removed from the network. Here we apply the K-means algorithm in order to categorize an infected vehicle's neighbors based on the specific attributes mentioned, i.e. number of connections, contact duration and for urban cities the weighted road segments for a vehicle's trajectories. By considering high degree nodes we put in quarantine nodes, who if infected, can have a large impact in the near vicinity, since they can contact a large number of nodes. On the other hand by being cautious about nodes which have been in contact with an infected source for longer periods, we also protect the network from nodes which are potentially infected with higher probability. Finally, through weighted road segments, we give categorize the road map in segments with specific weights, i.e. important and less important road segments, and thus we prioritize on protecting vehicles that will follow strong weighted roads. Note that by putting nodes in the PIL (temporal list), we cut a vehicle's connections temporarily, until it is checked for infection. Upon the first reception of a message by the RSU, from a vehicle in PIL, it is judged for its state, i.e. infected or clean. If not infected it is removed from PIL and thus its connectivity is restored. If judged infected, the previously mentioned procedure is repeated, i.e. the vehicle is added to the BL and so on.

K-MEANS CLUSTERING ALGORITHM

K-means clustering is a method of vector quantization, originally from signal processing, which is popular for cluster analysis in data mining. K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Coronoid cells.

Given a set of observations (x_1, x_2, …, x_n), (in our study the degree or connection time of the neighbors) where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k (≤ n) sets S = {S_1, S_2, …, S_k} so as to minimize the within-cluster sum of squares (WCSS). In other words, its objective is to find:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

Where μ_i is the mean of points in S_i.

DEGREE & TIME CLUSTERING SPECIFICATIONS

In summary, we explained that we use the general algorithm of k-means clustering in order to cut a vehicles neighbors into K clusters from the vectors of data RSUs have collect.

For the use of **Degree Clustering** only the neighbors' degree vector will be used for partition processing. The first cluster has the neighbors with the higher degree in the vicinity of the vehicle that is recognized as infected. The other cluster has the neighbors with the lowest degree. As we already said, a higher degree means that the vehicle is connected with a large amount of vehicles, making it an Influential Spreader. The RSUs consider the first cluster as the community with the largest probability of infection and thus this cluster is chosen to be inserted into the PIL.

Repeatedly, from the K-means algorithm with k=2 using only the vector's data for the contact duration of the neighbors this time (i.e. **Time Clustering**), we have: one cluster with the highest duration and one with the lowest. We consider that the bigger the contact duration (i.e. two vehicles where communicating earliest in the past) the larger the probability of the vehicle being infected. With that in mind the first cluster is also chosen to be inserted into the PIL.

In cases of degrees or contact duration values that are very close to each other and only a small proportion stands out, we have inserted a method to compute again with k-means in order to gives us only a percentage of the links that need to be cut. For example if degree vector has the data {4, 62, 53, 51, 60, 54, 59} the K-means clustering will compute that cluster1 ={62, 53, 51, 60, 54, 59} and cluster2= {4}. Cutting the first cluster as we have explained will result to a deletion of a big vehicular network part, which may cause greater problems to communications. Thus, the second clustering will spit the new data (from cluster1) {62, 53, 51, 60, 54, 59} into cluster1={60, 62, 59} and cluster2={53, 51, 54} resulting in a better solution and a good condition of the network after the deletion of links. Finally as you can understand this method is triggered every time the cluster1 is bigger than 50% of our initial data.

URBAN SCENARIO ONE MORE CLASSIFICATION

In the highway scenario the topology of the road has little importance, i.e. we only have to consider the two directions. However, for an urban environment, which includes a number of intersection and thus a number of potential directions we have to include one more characteristic, i.e. the weight of the road segment that a vehicle is about to follow. On possible way is to measure the traffic on each road segment (through statistics) and thus assign to road segments with higher traffic, higher weight in order to obtain the ranking for the road segments. However real traces of traffic mobility are hard to find.

In our evaluation we consider a network graph G(V,E) were V is the number of nodes, i.e., intersections in the road map, and E depicts the road segments connecting those nodes. Thus for a road network we obtain a graph structure. In order to obtain the importance of each road segment we need to find the importance of its two adjacent nodes. PageRank is a widely used method in order to decide the importance of a node for identifying important webpages or broadly speaking influential node-entities in a graph-like connected environment.

In Figure 7 we illustrate an example of the Erlangen city of Germany illustrated as a graph.

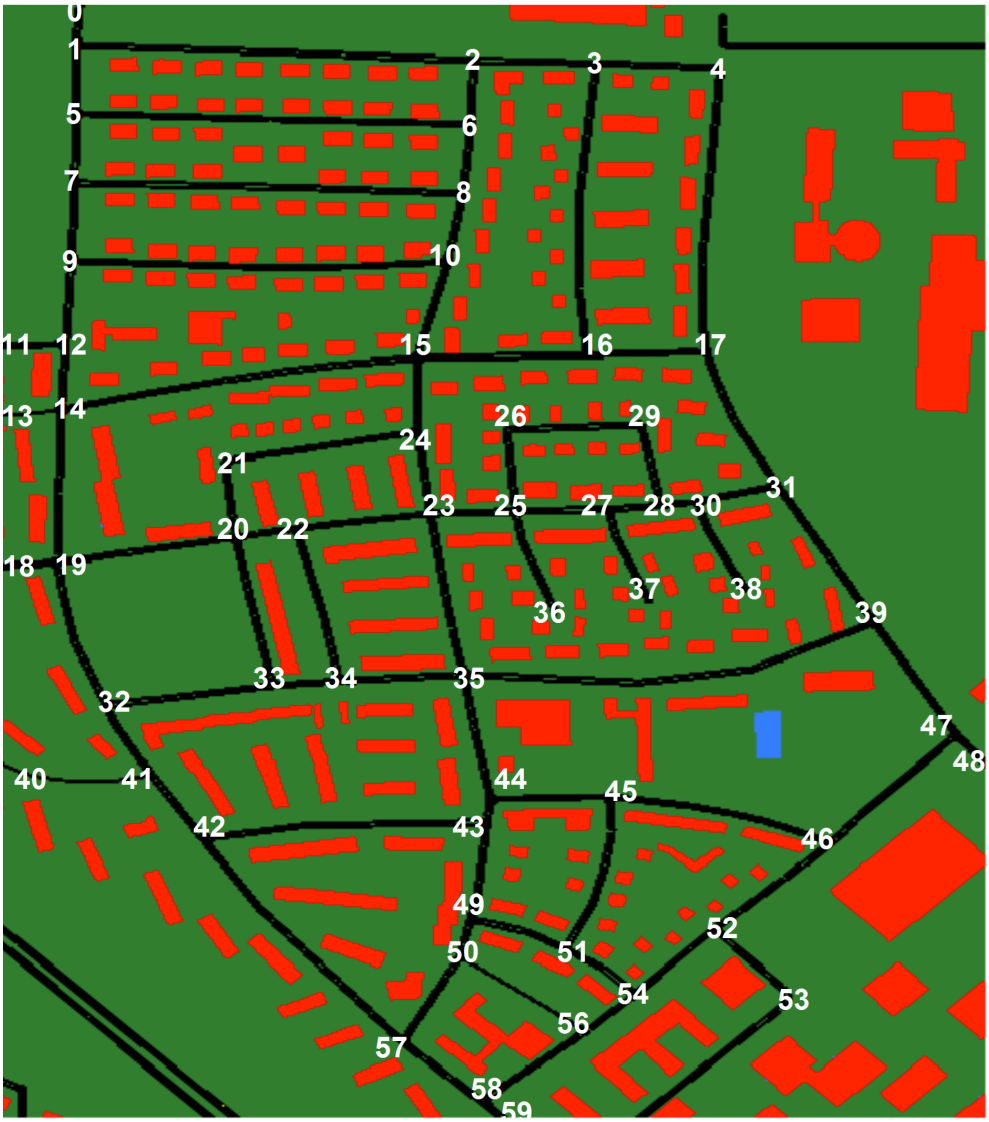


Figure 7 Simulation Map as a graph

PAGERANK ALGORITHM

PageRank [13] is a link analysis algorithm and it assigns a numerical weighting to each element of a hyperlinked set of documents, such as the World Wide Web, with the purpose of "measuring" its relative importance within the set (Figure 8). The algorithm may be applied to any collection of entities with reciprocal quotations and references. The numerical weight that it assigns to any given element E is referred to as the PageRank of E and denoted by *PR(E)*.

A PageRank results from a mathematical algorithm based on the map, created by all road intersection as nodes and the roads themselves as edges. In Figure 7 we can see all these intersections (ordered from 0 to 59) in our Urban Scenario map.

The PageRank algorithm is mentioned in detail in reference [12], but here we mention the general formula. To compute PageRank we need to define several variables.

- Binary link variable $\mathbf{L_{ij}}$. if page j joins to page i, then $\mathbf{L_{ij}=1}$, otherwise it is zero.
- Total number of pages in our consideration , **N**.
- Number of outbound link:

$$c_j=\sum_{i=1}^{N} L_{ij}$$

- A parameter **d=0.85** is a positive constant.

Google page rank is defined as recursive formula:

$$P_i=(1-d)+d\sum_{i=1}^{N}\left(L_{ij}/c_j\right)P_j$$

For initial values of page rank $\mathbf{P_j}$, we can use number of outbound links, that is $\mathbf{P_j}=C_j$. In the related reference can be found more information and details about the use of the PageRank algorithm.

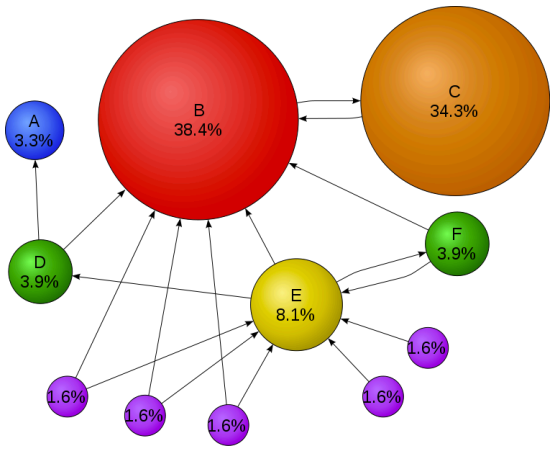


Figure 8 Mathematical PageRank for a simple network, expressed as percentages.

ASSIGNING WEIGHTS TO ROAD SEGMENTS IN URBAN SCENARIO

Now that we obtained the PageRank for each node (i.e. intersection), we define the weight of a road segment as the product of the PageRank scores of its adjacent nodes. The final proposed attribute will be used with the same logic as degree and time in the K-means algorithm. Thus we consider on more classification based on what is the next road segment that a vehicle will follow. Vehicles which are to follow a road segment of high score will be included in the PIL since they can potentially do more damage.

CHAPTER 5

SIMULATION TOOLS

Simulation is an important tool used for study and evaluation of complex systems. Simulation of networks and protocols enables development and study of the suggested protocols prior to deployment. One of the broadly used simulation tools in academy is a very powerful open source network simulator OMNeT++. In order to allow the most accurate modeling of vehicular movements mobility of vehicles hybrid simulation framework is required which is composed of a network simulator OMNeT++, a road traffic simulator SUMO which is well-established in the domain of traffic engineering and the appropriate framework that combines those two simulators, called VEINS.

SUMO

SUMO (Simulation of Urban Mobility) is an open source microscopic traffic simulator licensed under General Public License (GNU) and developed by Institute of Transportation Systems at the German Aerospace Center [10] using C++ standard. It allows users to create a road network of their preferences containing buildings and streets or to import a road network from different format (e.g. OpenStreetMap) and convert it into a SUMO network. Also each vehicle can be modeled explicitly, in order to move individually through the network and has their own route updating the position of each vehicle every time step, which gives SUMO the feature of time-discrete vehicle movement. This traffic simulator also provides an OpenGL graphical user interface.

Traffic simulation in SUMO can be conducted in two ways as described below and the overview of the simulation process is given in Figure 9.

OMNeT++

OMNeT++ is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators. "Network" is meant in a

broader sense that includes wired and wireless communication networks, on-chip networks, queuing networks, and so on. Domain-specific functionality such as support for sensor networks, wireless ad-hoc networks, Internet protocols, performance modeling, photonic networks, etc., is provided by model frameworks, developed as independent projects. OMNeT++ offers an Eclipse-based IDE, a graphical runtime environment, and a host of other tools. There are extensions for real-time simulation, network emulation, database integration, System C integration, and several other functions.

VEINS

Veins is an open source framework for vehicular network simulations consisting of SUMO and OMNeT++ simulators to tender a complete suite for Inter Vehicle communication (IVC). It was designed by Transportation and Traffic Science community. Veins is part of MiXiM framework of OMNeT++ adding support for IEEE 802.11p and IEEE 1609 family - WAVE technology. Thus, this framework handles Wave Short Messages (WSM) and provides beaconing WAVE services, access categories for QoS (Quality of Service) and multi channels operations.

The bidirectional communication between OMNeT++ network simulator and SUMO road network simulation is shown in Figure 10.

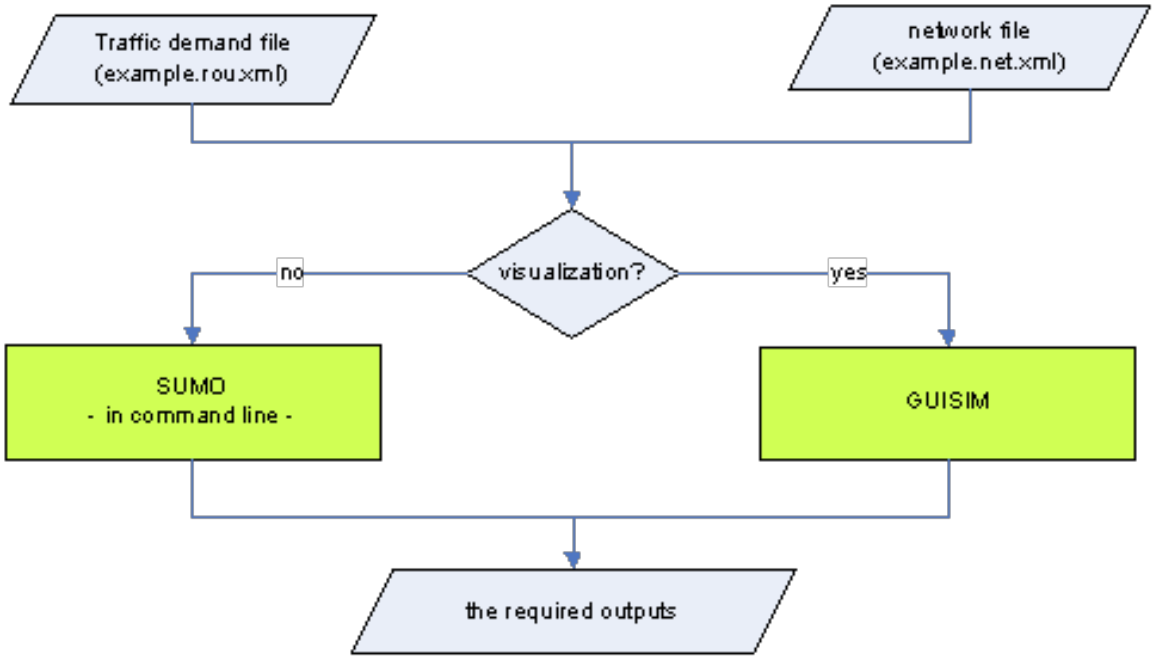


Figure 9 Traffic simulation process for SUMO.

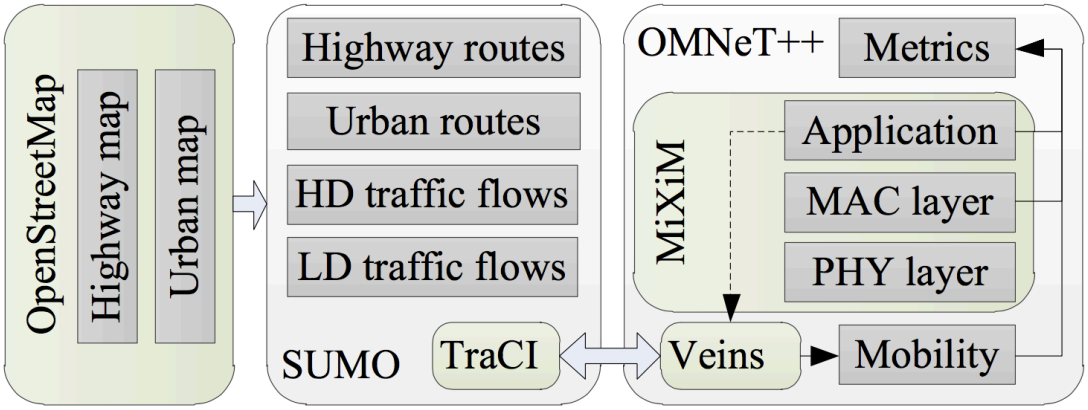


Figure 10 Communications between SUMO and VEINS and their elements.

SIMULATION SETUP

COMPETITOR LISTS

- **Unprotected**: RSUs detect a malicious vehicle that enters their transmission area and block it without inserting any of its neighbors in the PIL.
- **Degree**: RSUs detect the malicious vehicle and also cut the links of the highest degree nodes through PIL.
- **Time**: Same as Degree but this algorithm cuts the highest contact duration times between vehicles.
- **PageRank**: For this algorithm we cut the vehicles that their destination has the highest road weight that we have calculate for out Urban scenario.

GENERAL EXPERIMENTATION SETTINGS

- Standard Dedicated Short Range Communications (DSRC) are used.
- The simulation will start the malicious diffusion when the roads are fully populated and run for s seconds.
- One initial spreader at the center of map either at 5000m in highway or in the center of the city.
- The malicious diffusion will follow the SIR model, as implemented in "Worm-Epidemics in a Large Scale Vehicular Network".
- Penetration Rate will start from 100% and that mean that all vehicles in the simulation can be infected if not protected by the worm that propagate from the initial spreader.
- Beacon intervals are set to 1 sec, for example 5 messages will be transmitted and received in about 5 seconds.
- Carrier Latency T is a contentious number of messages that a vehicle must receive from an infected source (i.e. infected vehicle) in order for a susceptible car (i.e "clean") to become infected. For example for T=4, 4 messages are needed to be received from a vehicle to become infected.

EXPERIMENTATION SETTINGS FOR HIGHWAY

Map Settings. For the highway scenario we have created in sumo simulator a straight road 10 kilometers long with two lanes in every direction. The RSUs are placed from the beginning of our road and in every 1000 meters. Transmissions will not be obstructed by any means in this type of environment.

Transmissions Settings. The RSU placement gives us a number of 11 RSUs throughout our map. Standard Dedicated Short Range Communications (DSRC) that are used in our simulation provide about 300m of wireless transmissions. With these information in mind, the void space between two RSUs is 400m and 4000m in total distance that the worm or virus can propagate. The remaining 6000m are combination of the RSU's protected communication areas.

Vehicles Settings. Vehicles can only enter the road from the down-left or top-right direction and the lanes will be chosen randomly. They are inserted in the simulation as clean cars (i.e. Susceptible and thus can be infected) and given random speeds from a random number generation function. Typical speeds in a highway are ranging from around 80 and up to 120Km/h, that's why the velocities imparted are between 22 and 33 m/s. With this method we avoid an unrealistic vehicular environment that could exist if we have given standard speeds to our vehicles like

usually SUMO does. This fixed velocity distribution would create determined vehicles movement and thus the neighborhood forming would be the same every timestamp. Therefore, the setting we try to introduce, create a topology that reflects as much as possible a real environment that we encounter in our every day situation.

Network Densities. In order to have a big range of different types of network, we introduce the concept of vehicle (or network) densities. These values varying from sparse to dense network gives us a scenarios from a more "open" road to traffic jams (but velocities never approach zero). Specifically we have work with rates that measure the vehicles per lane per hour and are from 300 up to 1300 (Veh/lane/hour).

The different speeds and vehicle densities create a large range of realistic environments that VANETs encounter in real life situations.

Message Exchange Settings. Messaging is done with broadcast and unicast transmissions between participating entities. In V2V communications a multi-destination distribution method is needed. Broadcast is used in these types of connections, which means transmitting the same data to all possible destinations (i.e. vehicles). V2I communications on the other hand use unicast transmissions, so as the nearby RSU to receive the report of the each vehicle, infected or not. I2V communications and especially BL and PIL need to be sent in every car inside the transmission area of the RSUs.

EXPERIMENTATION SETTINGS FOR CITY ROADS

Map Settings. For the urban scenario we have created with the help of the OpenStreetMap and SUMO the city of Erlangen in Germany. In this simulation we chose to test our proposed implementation in a part of the city that is like a grid with many intersections with 1 square kilometer area. Choosing an urban environment like this one means that building will exist at the side of the roads, near intersections and logically everywhere in the map. These characteristics present limit to wireless communication. Fragmentation in transmissions areas will exist due to the LoS and NLoS (Line of Sight & Non Line of Sight) [14] zones that would be created from the nearby buildings. The RSU placement and transmission areas are presented in Figure 11 with the yellow dots representing the RSUs and the blue lines their communication fields , making the ordinary black road the void spaces that infection can propagate.

Vehicles Settings. Same as in highway scenario velocities will randomly be chose , but this time from 6 to 15m/s (i.e. typical speeds of 20 up to 55 Km/h). This time not only the lanes will randomly be picked, but also destination and routs won't be predefined. Arbitrary movement will create the realistic environment that we seek, with loops or small trips inside our simulation map. For example think about a person

cannot find any parking space and in order to find one has to circle the city block or try to find parking at the street nearby.

Network Densities. Small vehicle speeds and many intersections will create big traffic jams and suppress movement. That is why, we only need densities from 900 up to 2100 Veh/hour.

Messages Settings. Broadcast and unicast transmissions will be exactly the same as in our highway scenario.

Route Settings. For our routes in the city of Erlangen we used a random function that creates many different possible journeys for the vehicles to follow. This function can be found under the name "randomtrips" in the SUMO simulation environment and provide the best solution for a random vehicular topology for our experimentations.

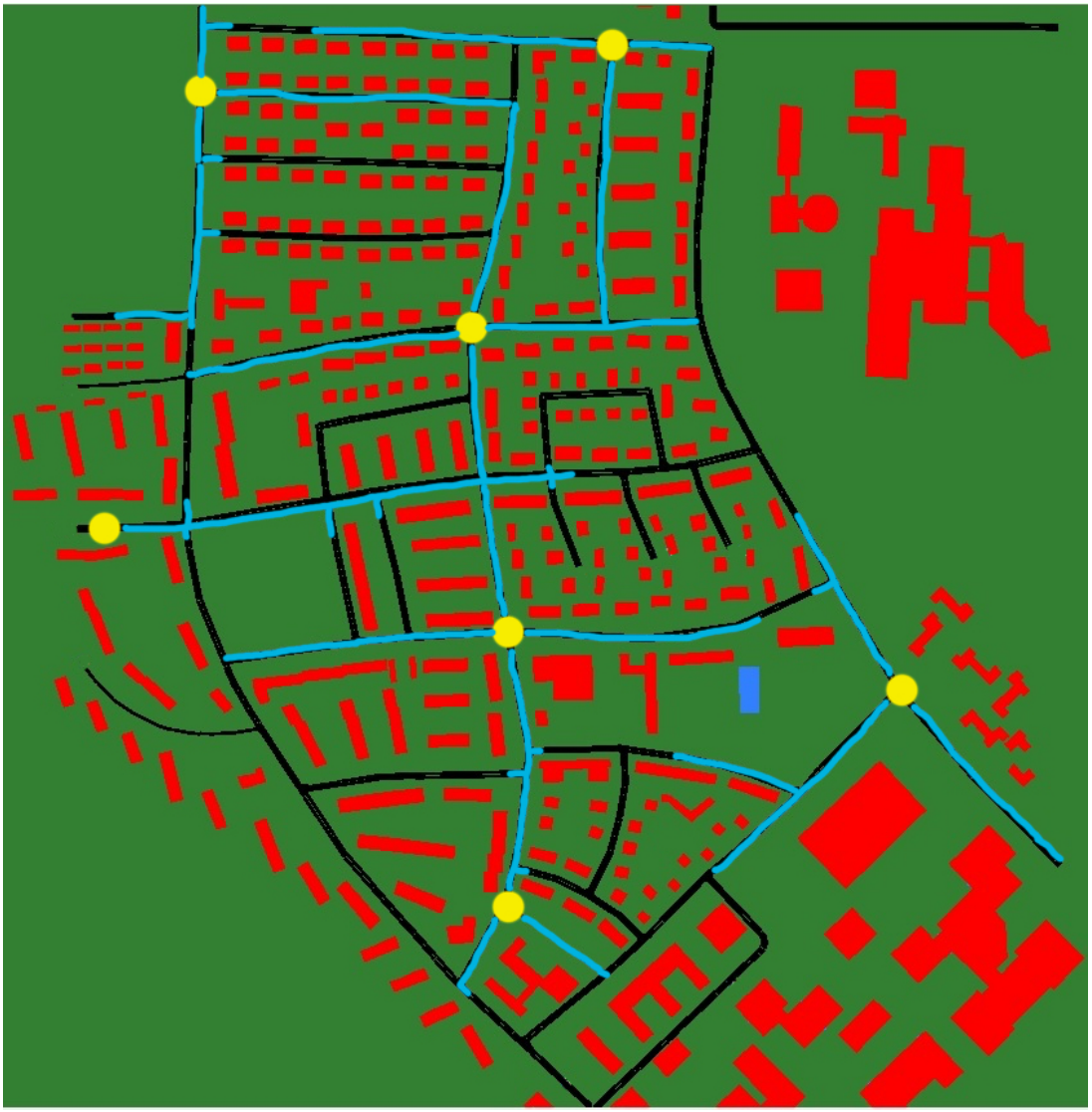


Figure 11. RSUs placement and their transmission are

EXPERIMENTATION RESULTS

As we have already explained, there is only one way an infected vehicle can propagate the virus, which happens when its current position is inside the void space between two RSUs. Getting infected is another story and there are two ways of becoming malicious. The first one is when a clean (Susceptible) car is inside the void space i.e. before gets inside the RSU transmission area. In the second scenario, the susceptible vehicle is already inside the protected area, it can hear the malicious packets coming from the infected vehicle (the malicious vehicle is inside the void space), however RSU cannot hear the infected vehicles since it is out of its range. A case scenario, which happens either when a vehicle is at the right side of the RSU (vehicle A in figure 11) or at the left side of the RSU (vehicle B in figure 11). For example car B is inside the void space but also in the (yellow) transmission area of vehicle B. In this case if vehicle C is infected the virus can propagate to the vehicle B and infect it as well. The second scenario can produce a multi hop spreading between the RSUs and the propagation of the worm to multiple void spaces

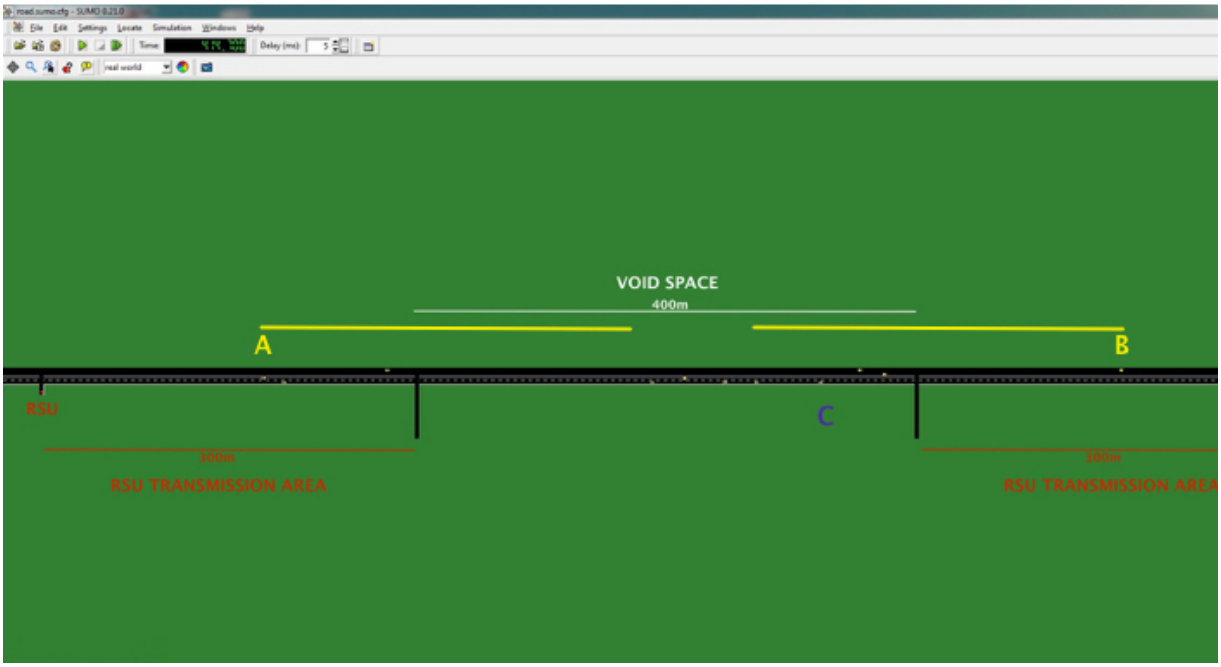


Figure 12. Vehicles positions inside RSUs transmission areas can that be infected that can be infected from vehicles inside void spaces

EXPERIMENTATION RESULTS FOR HIGHWAY SCENARIO

IMPACT OF CARRIER LATENCY

For our experimentation we have chose to test our scenario in Carrier Latency from T=2 up to 20. At T=20 the infection did not propagate to any other vehicles and only the initial spreader was infected. In this situation the initial infected vehicle did not have the opportunity to propagate its infection inside the void space due to the slow carrier latency before its entrance to transmission area and the identification as infected from the RSU. Values from T= 12 up to 18 illustrate that the virus impact was also very small due to the slow latency carrier such as for T=20. The first value, T=2 in Figure 13 shows that the proposed mechanisms did not work very well, blocking less that 10% (for the Time K-means Clustering). This was due to the fast worm propagation, making all the vehicles in the vicinity of an infected vehicle, infected as well. We noticed that vehicles become all infected at once inside the free area (void space) and thus clustering these groups cannot produce good results. Only a solution for cutting all the links would maybe have a better result, but this implies that the network will be destroyed as we have already explain in previous chapters. Best results are displayed for the values 4 to 8. T equals to 4, 6, 8 and 10 show us that for a more moderate in propagation speed worm, our proposed mechanism will work at its best. Measuring up to 50% less infection in our Vehicular network, indicates a well–accomplished goal. In cases of a worm spreading at these carrier latencies ensures a more smooth dissemination of the malware from vehicle to vehicle, creating neighborhoods where not all cars will be infected and hence a good clustering should provide better solution.

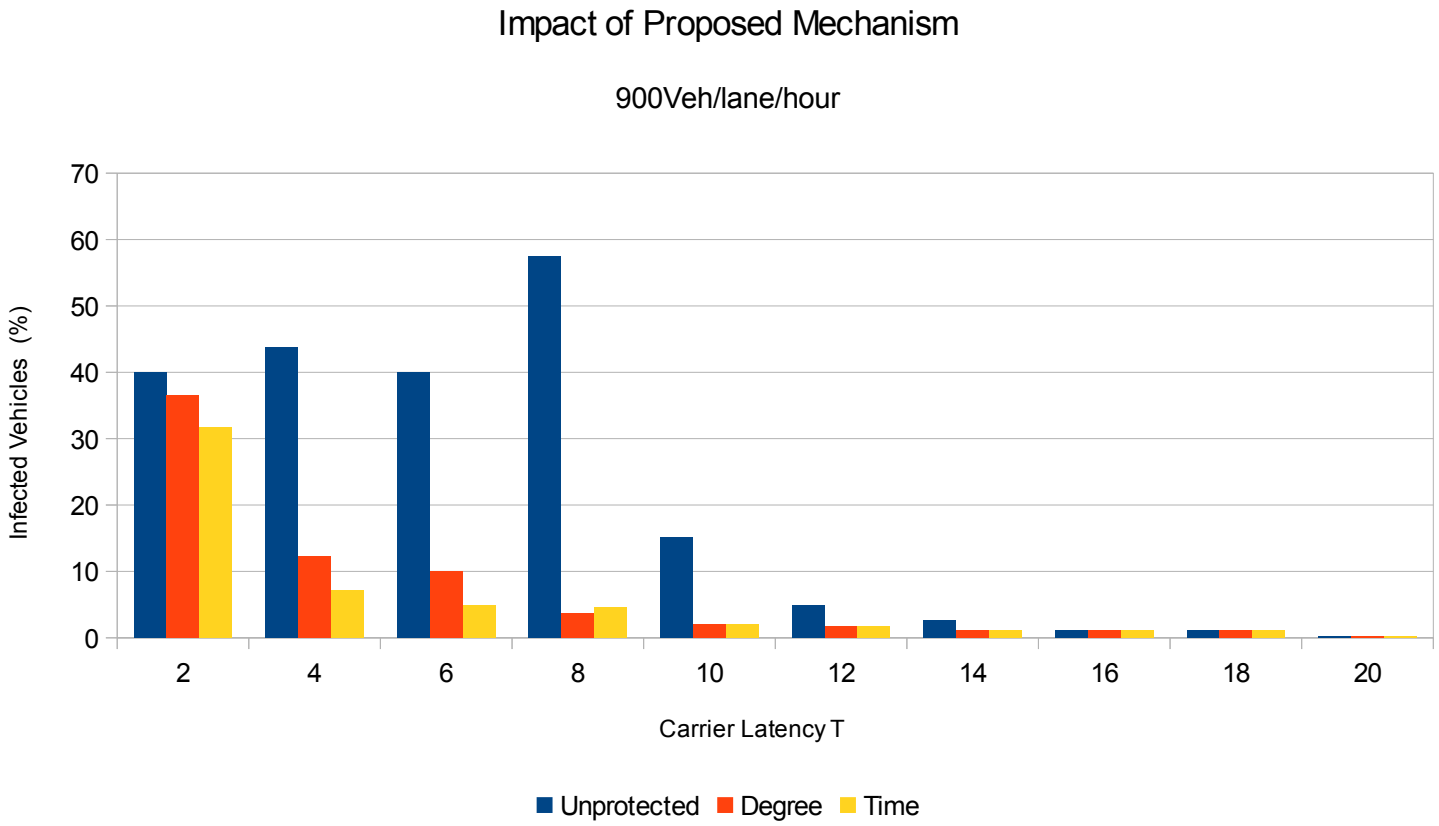


Figure 13. Carrier latency -to- Infected vehicles ratio for the Highway scenario

IMPACT OF DENSITY PER HOUR PER LANE

We tested our propositions for different network densities from sparse (300 Veh/lane/h) to dense (1300 Veh/lane/h) as we already mentioned. Figure 14 illustrates again a remarkable result of our Time clustering algorithm. The worm achieves to infect only a fraction of the Vehicular Network and never manage to pass the 10% mark of infected vehicles. It was our initial idea that by calculating the connect duration of the vehicle neighbors, will give us good possibility of cutting only the truly infected nodes and with annihilation of false positives judgments. This belief brought the outcome that we have expected and Time clustering algorithm performs very well in blocking the outspread of the infection through cutting vehicle links that have been in contact for longer duration with the infected vehicle. At the dense (1300 veh/lane/h) scenario Time clustering outperforms Degree for these reasons.

Note that owing to the density and the speed distribution of the vehicles we cannot anticipate linear infection propagation in the observed results. This is due to the fact that by introducing more vehicles in the simulation we have different distributions in the obtained speeds thus a different scenario in both density and speed distribution. We have observed in our simulation that different speed distributions at the time the infections occurs can affect greatly the impact of the proposed algorithms and thus this

justifies the obtain results. Therefore we examine independent every value of densities (from 300 to 1300 vehicles per lane per hour) or carrier latency T (from 2 to 20).

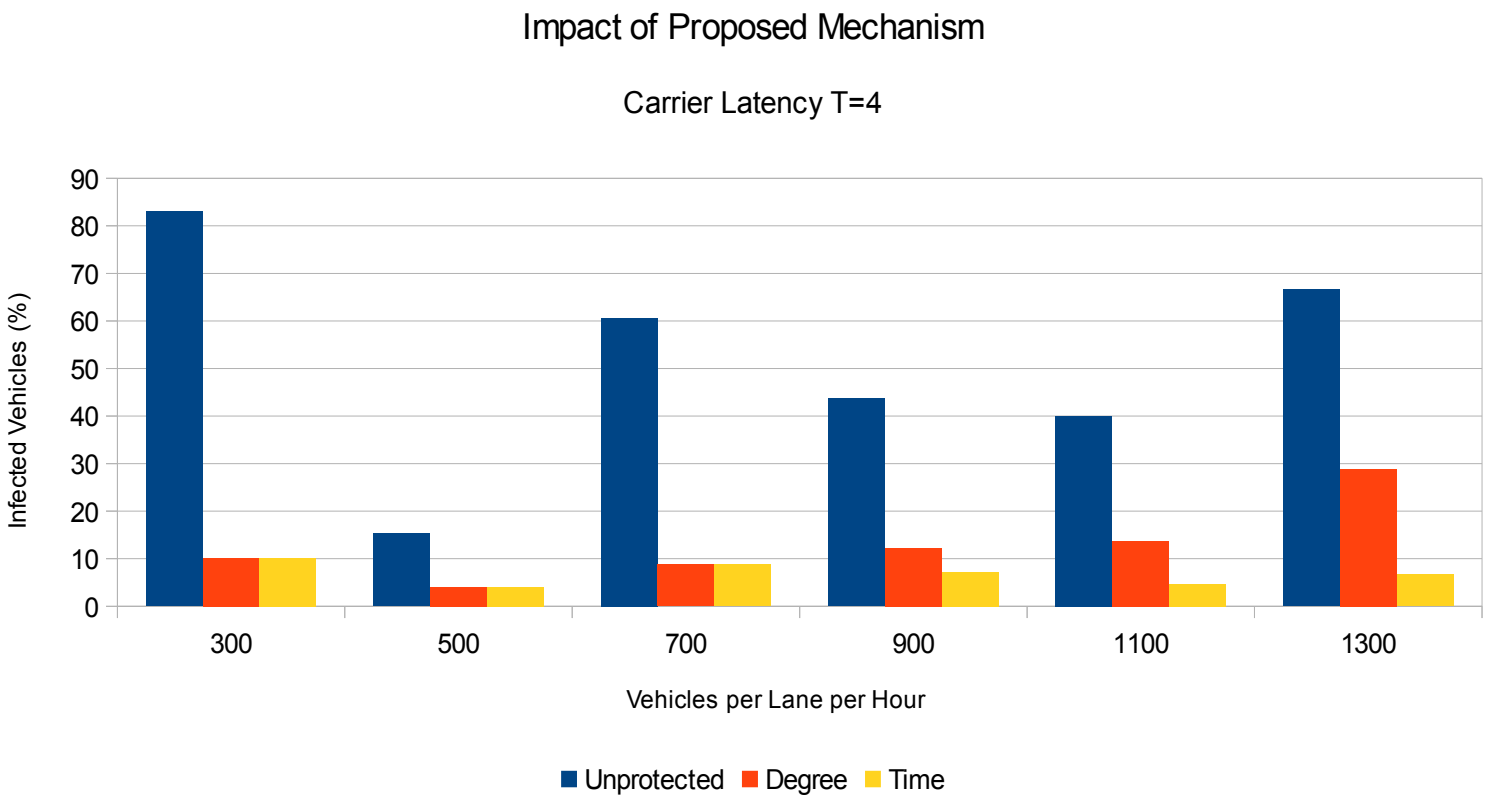


Figure 14. Vehicles per lane per hour –to- Infected vehicles ratio for the Highway scenario

The next three plots (Figures 15, 16 and 17) present the infections occurred over time for the same experiments. These Figures present each infection at each distinct time and for each density. From the infections over time we can understand and study the sustainability of the virus inside our VANET up until no more infections take place. Comparing the first two plots (Degree and Time mechanisms) we notice not only that Time clustering algorithm managed to better protect our network but also has block the infection faster in time. When there are many vehicles in the simulation thus there are more potentially victims and the virus will be present in the network for a larger periods of time, which is illustrated in the case of 1300 veh/lane/hour. The paradox we see through our simulation is for the lowest density i.e. 300 veh/lane/hour. We observed that the virus did not propagate to a large number of vehicles as we see at

figure 17, however it lasted the longer in the network. This is due to the fact that there were not any neighbors to infect, although there were occasions where the propagation of the virus passed to the next void space as explained in Figure 12. This incident allows the maintenance of the infection for such a long period and the small number of vehicles preserve the infection at low rates due to the fact that there are not many vehicles to get infected. Also 700, 900 and 1100 densities shows almost identical results in their infection spreading time.

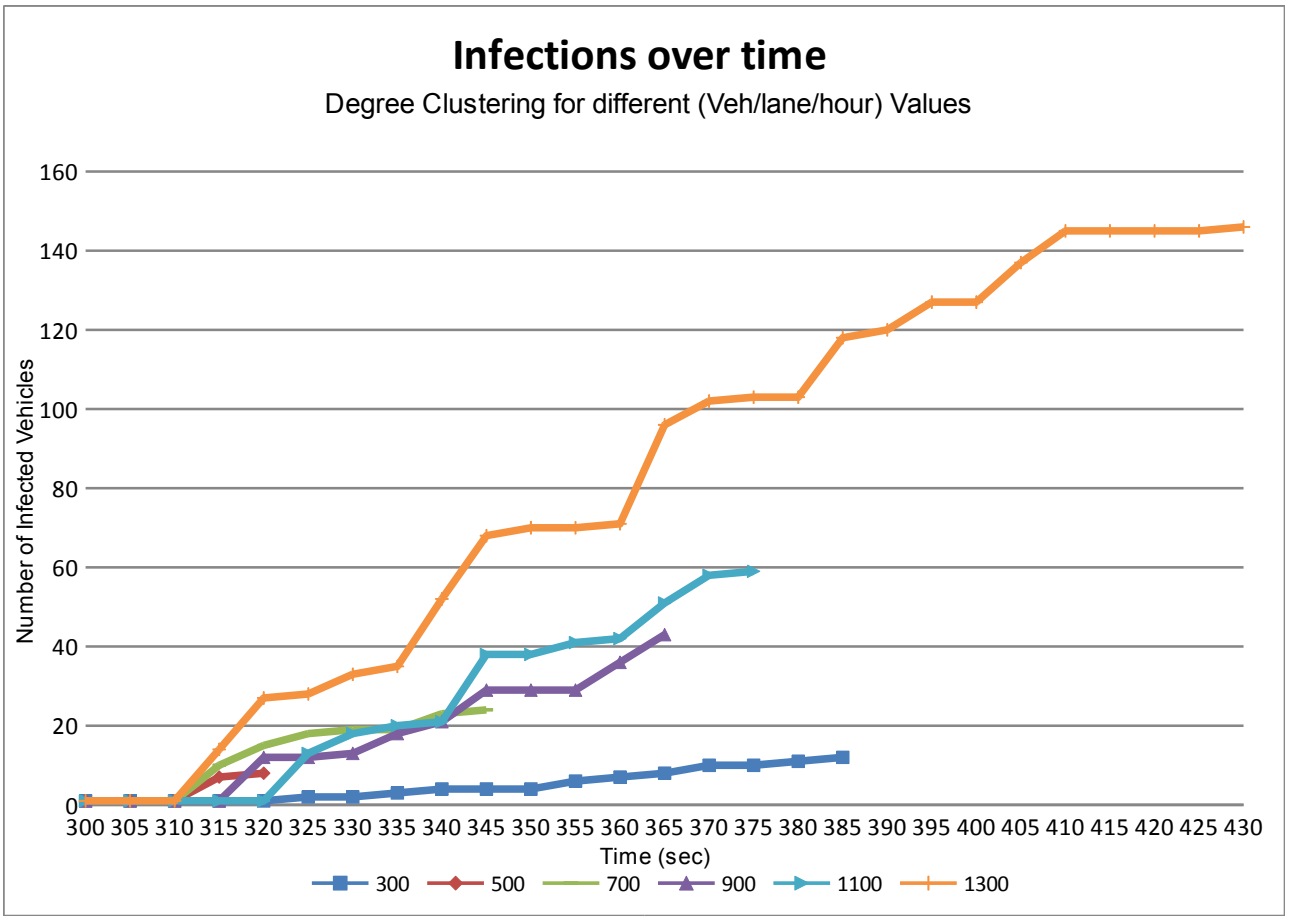


Figure 15. Time –to- Number of Infected Vehicles Under the Degree Clustering Algorithm for the Highway scenario

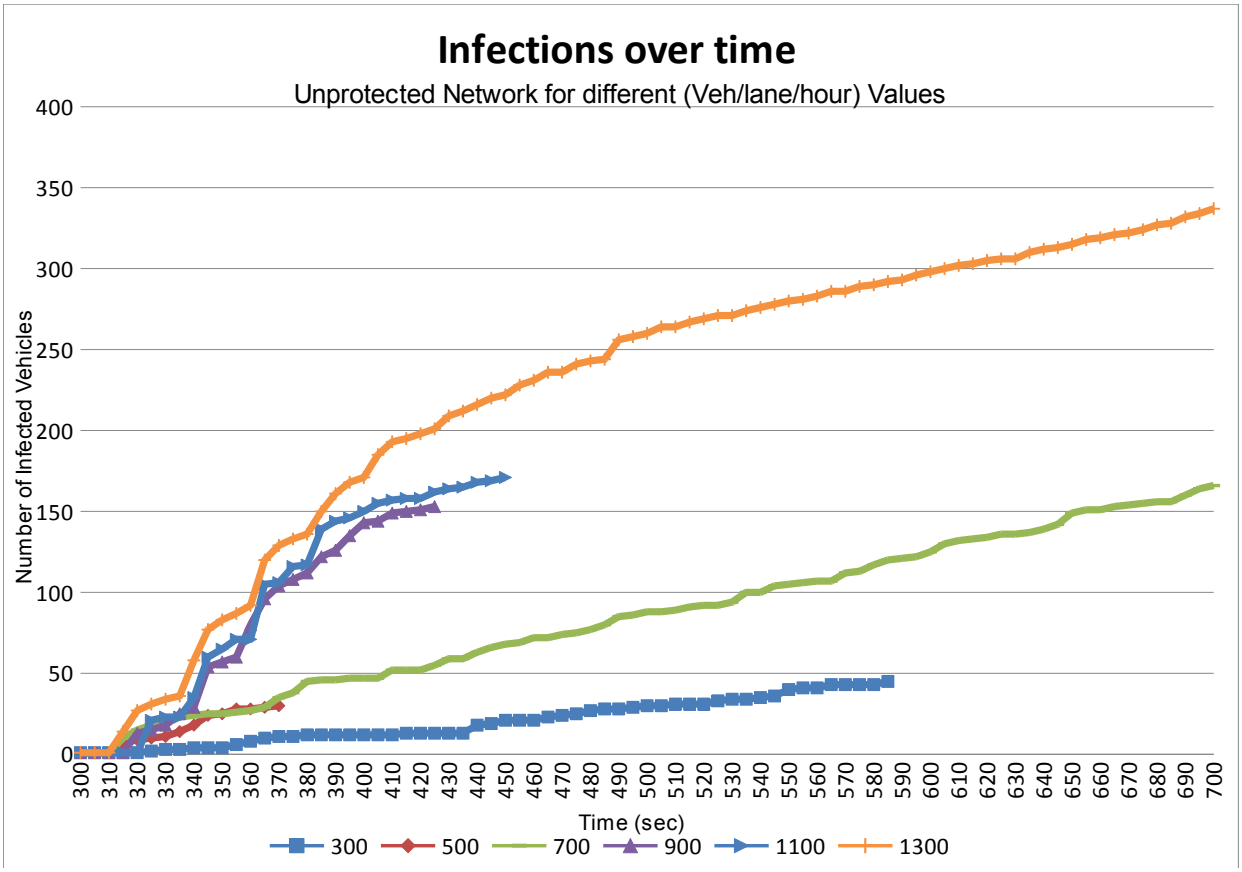


Figure 16. Time –to- Number of Infected Vehicles Under an Unprotected Network for the Highway scenario

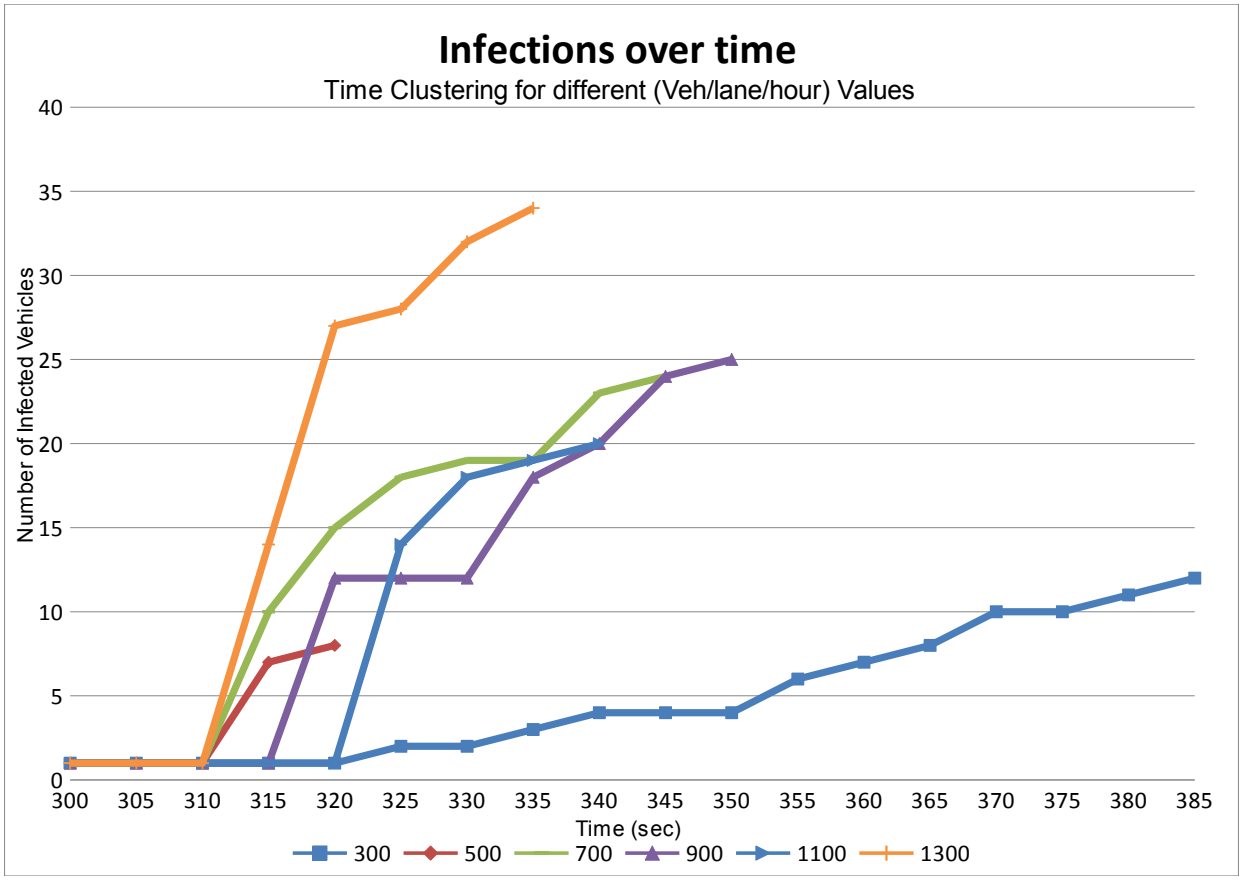


Figure 17. Time –to- Number of Infected Vehicles Under the Time Clustering Algorithm for the Highway scenario

IMPACT OF CONNECTIVITY OF THE VEHICULAR NETWORK

Figures 18, 19 and 20 indicate the connections that were cut (ratio from all the neighbors of the infected car) during the usage of the proposed mechanisms, Degree and Time clustering. Infection Events are the number of incidents where a vehicle is detected as infected from RSUs (through the malicious node detection mechanism) and blocked from the rest of the vehicular network (i.e. inserted to the black list, BL). Ratio of CUT/TOTAL connections shows the percentage of the links that where cut from the current network, i.e. the neighbors that each algorithm compute as probably infected and inserted into the PIL (Potentially Infected List). These outcomes are for the same 900 Veh/lane/h density and for different carrier latencies for 4, 6 and 8. Our intention was to obtain some connectivity of the VANET, in order to prevent the network destruction in cases of completely cut all the node links of the vicinity. Time clustering method allowed more connections to remain in the network, coupled with the fact that it has also better results from the degree clustering and thus makes it the default choice of our solution.
Note: Blank spots in our graphs like in figure 18 for the infection event 42, mean that there were not any neighbors and thus no links were cut.

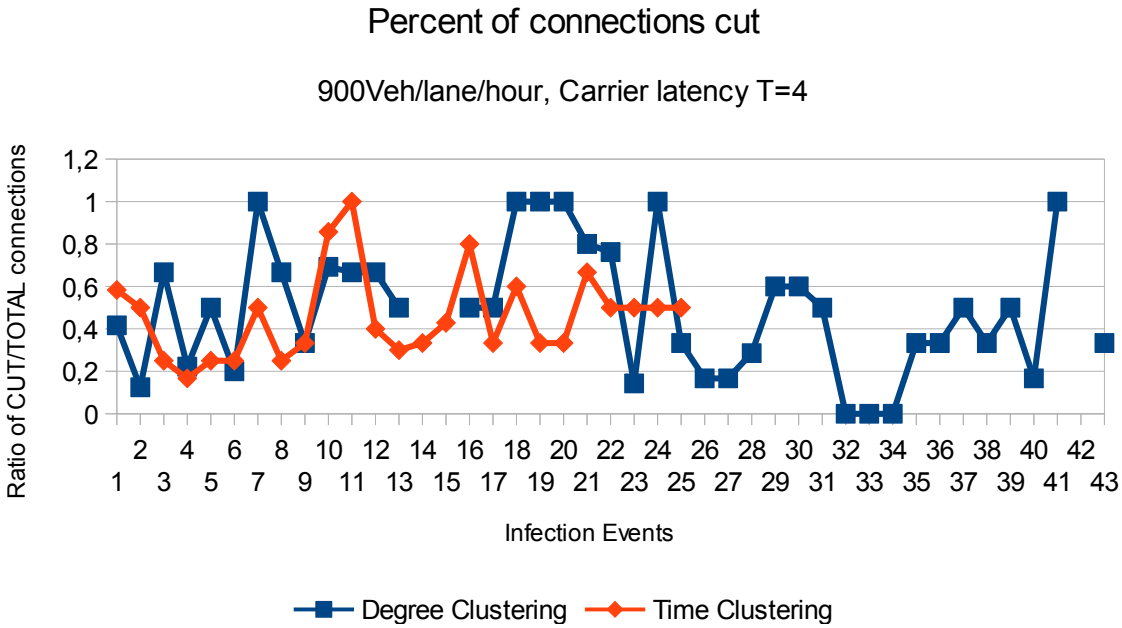


Figure 18. Infection Events –to- Ratio of CUT/TOTAL connections for the Highway scenario

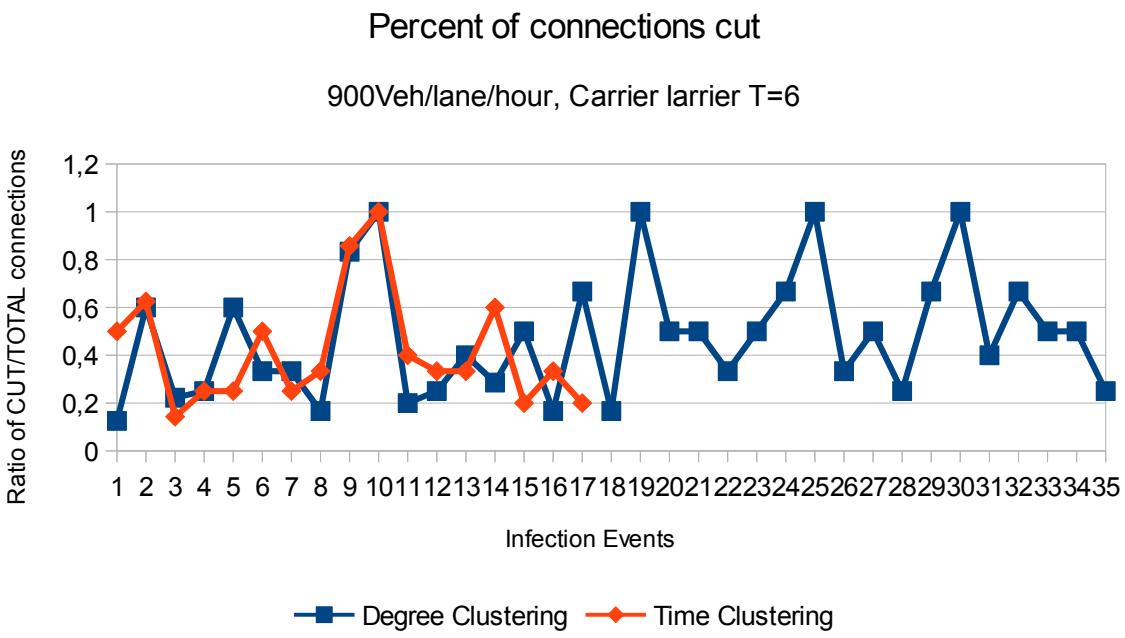


Figure 19. Infection Events –to- Ratio of CUT/TOTAL connections for the Highway scenario

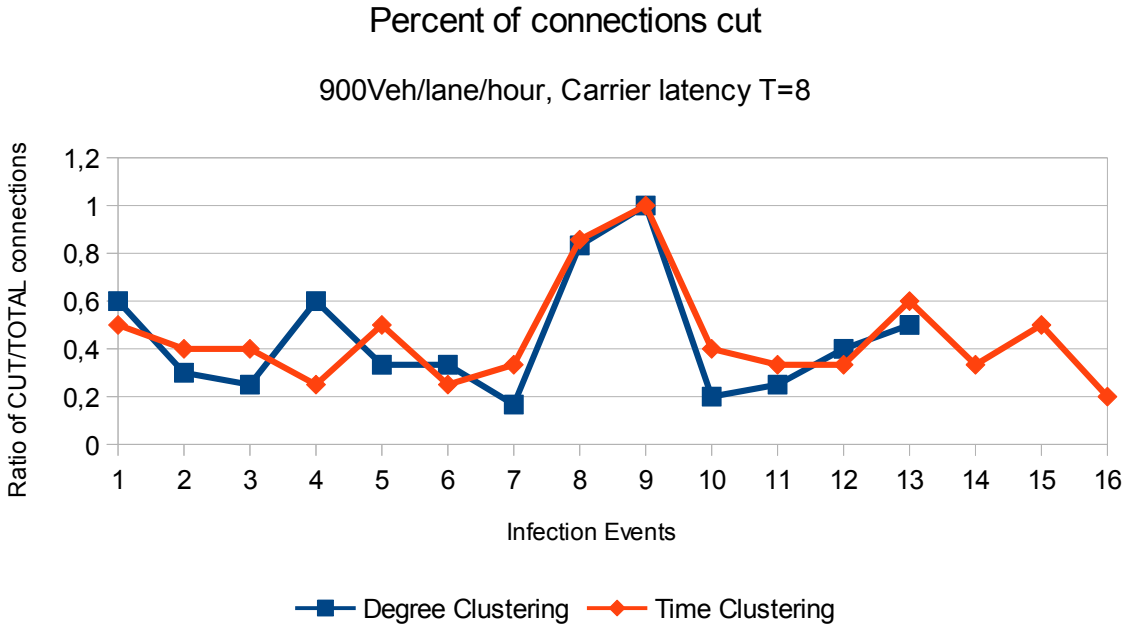


Figure 20. Infection Events –to- Ratio of CUT/TOTAL connections for the Highway scenario

SPEED DISTRIBUTION OF VEHICLES

Lastly the graph below (Figure 21) illustrates the different speeds (random) that the first 200 vehicles had in our experimentation. This speed distribution is responsible for creating a more random and more realistic environment for our simulation.

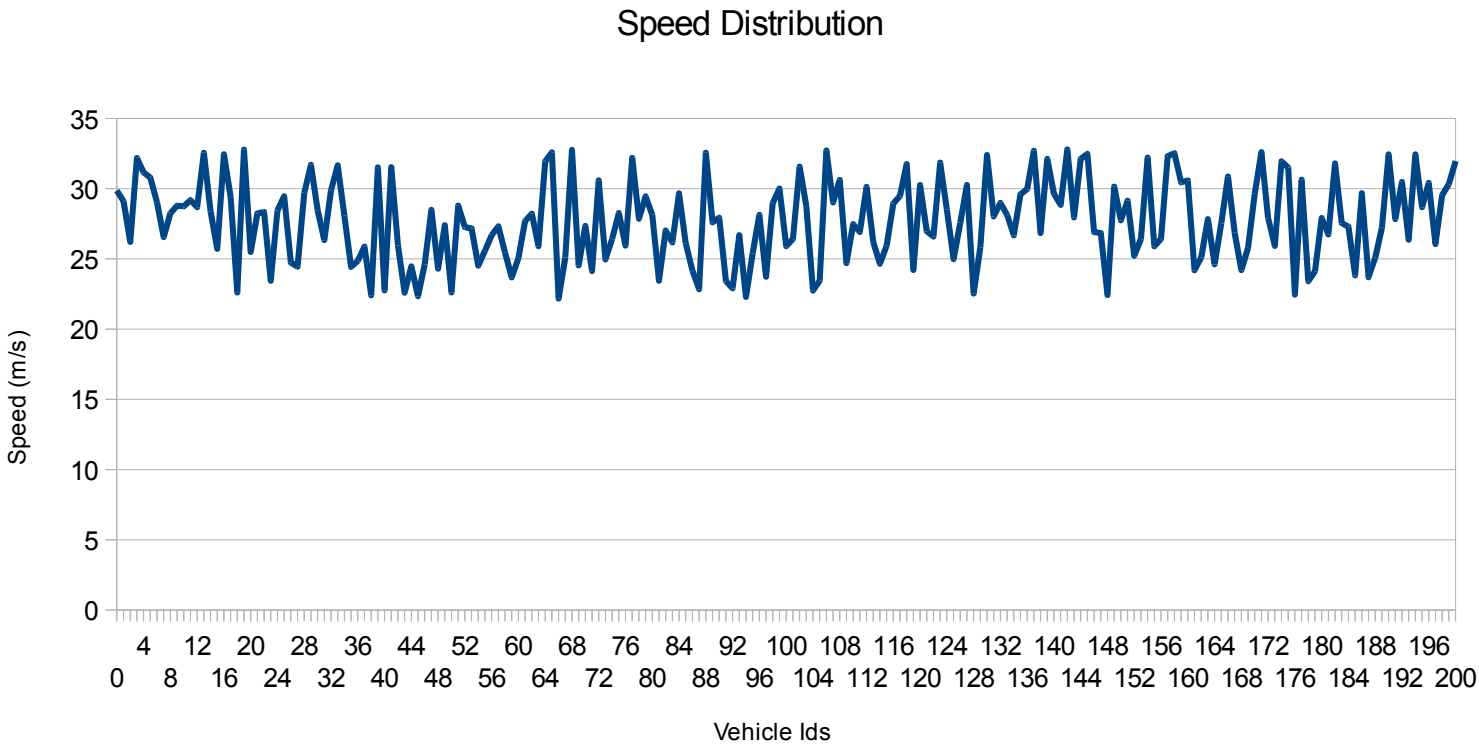


Figure 21 Vehicle Ids –to- Speed for the Highway scenario

EXPERIMENTATION RESULTS FOR URBAN SCENARIO

IMPACT OF CARRIER LATENCY

Likewise in the highway scenario, in Figure 22 we investigate the impact of the malicious propagation for different Carrier Latency (T) values through our proposed mechanisms. This time we test our scenario under the density of 1200vehicles/hour and T ranging from 2 up to 10. In the Urban scenario of T equal to 8 and 10 remind us the cases of the highway scenario when T was equal to 12 up to 20 with equivalent comments. At T=10 we have zero spreading capability i.e. only the initial spreader was in our simulation, which means that infected vehicles could not sustain a connection over this period of time in order to infect other vehicles or that the RSUs recognized it as malicious and thus could not propagate the virus. From our experimentations we can understand that in an urban scenario cars cannot communicate with each other for long periods of times due to obstacles (i.e. buildings). Such consideration results vehicles to frequently disconnect from each other although are inside the transmission area from one another. Lastly we can say that our proposed mechanisms Degree, Time and PageRank performed relatively close to each other due to the fact that the generation of the routes was random. With that in mind, in our future work we will try to obtain realistic traces in order to achieve a better performance analysis for the proposed defense mechanisms. Here, we also notice a different behavior for T=2 and the virus was dealt better than in the highway scenario (where we had a very aggressive propagation of the infection). For example we observe that from the unprotected case of about 70% infection, all the protected cases (Degree, Time and PageRank) manage to stay below 30%. This is due to the fact that in the Urban scenario we have include the cases of buildings, making it more complicated. Meaning that when the carrier is at its fastest case T=2 it can be handled effectively by the proposed defense mechanisms.

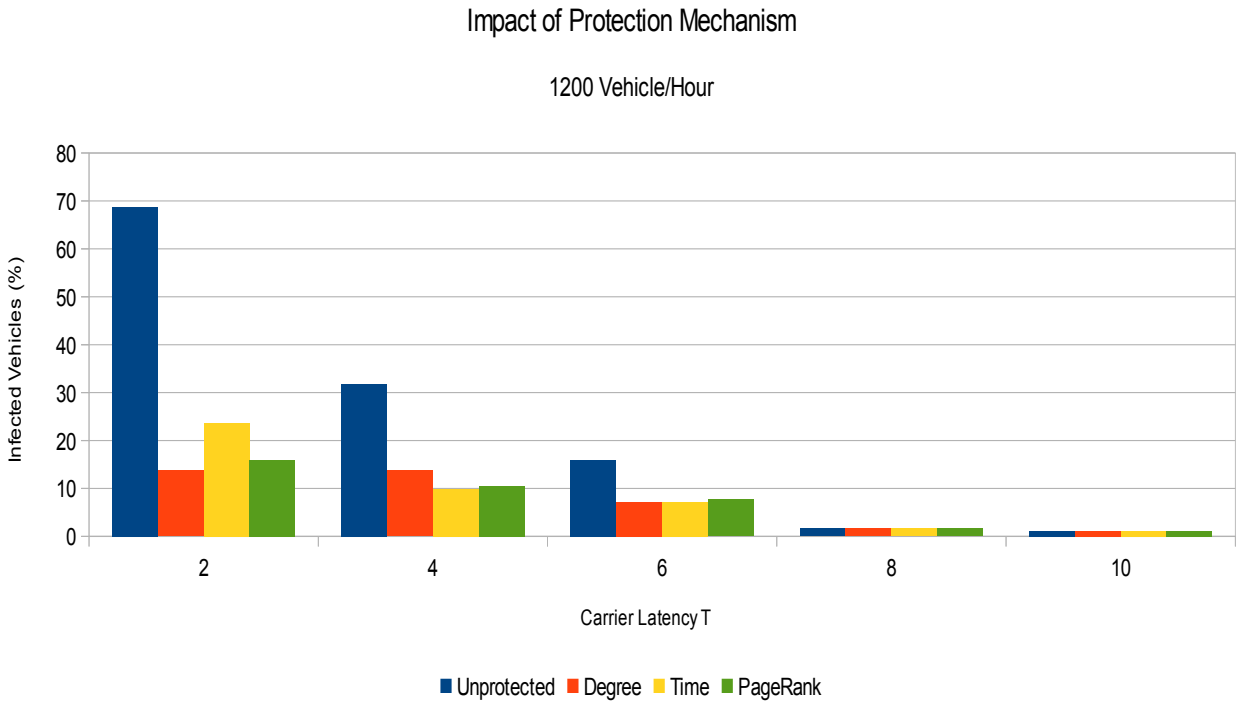


Figure 22. Carrier latency –to- Infected vehicles for the Urban scenario

IMPACT OF DENSITY PER HOUR PER LANE

Once again we introduce the impact of network density for a Carrier Latency of T=4 in Figure 23. For this experimentation Time clustering algorithm does not display good results and as we have explained above this is due to the limited time connections between cars. Los and NLoS (that is Line of Sight and Non Line of Sight) areas are ruining the functionality of this algorithm, which makes difficult to predict the best possible candidates to cut through the clustering mechanism. This phenomenon allow the Degree clustering algorithm to better perform because it does not need any time burier for its computations, which is why we assume that succeeds to bring a better overall result. Also PageRank seems to perform better than its competitors when we increase the density of the vehicles as we cut the vehicles that they have destinations to road segments with high weights.

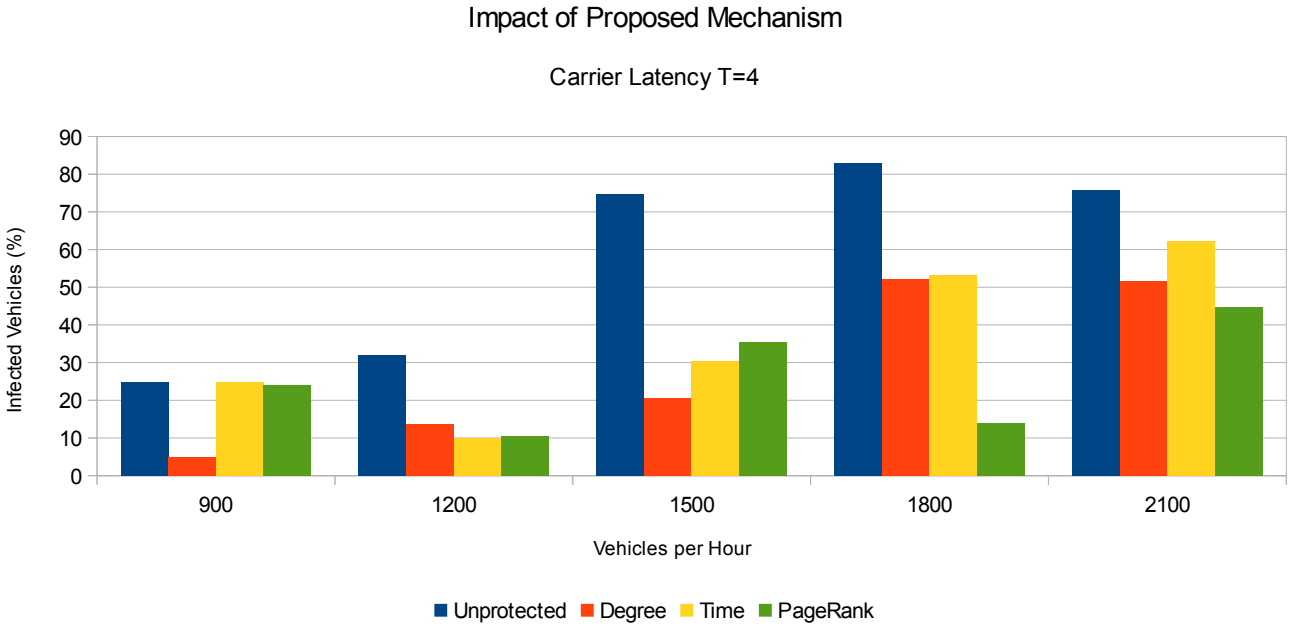


Figure 23 Vehicles per hour –to- Infected vehicles for the Urban scenario

Figures 24 to 27 demonstrate the number of infected vehicles over time, i.e. when every infection took place in our simulation environment. The conclusions we can draw form these plots are that for all the different algorithmic environments, the worm sustain its propagation for almost the same time in every scenario and the changes we can notice are only the outcome (number of infections) of our proposed defense mechanisms. In this situation we do not observe the paradox from the highway scenario. In particular, we notice that in higher densities the infection takes longer time to be vanquished from the network and also the increase in time lead to the increase of infected vehicles.

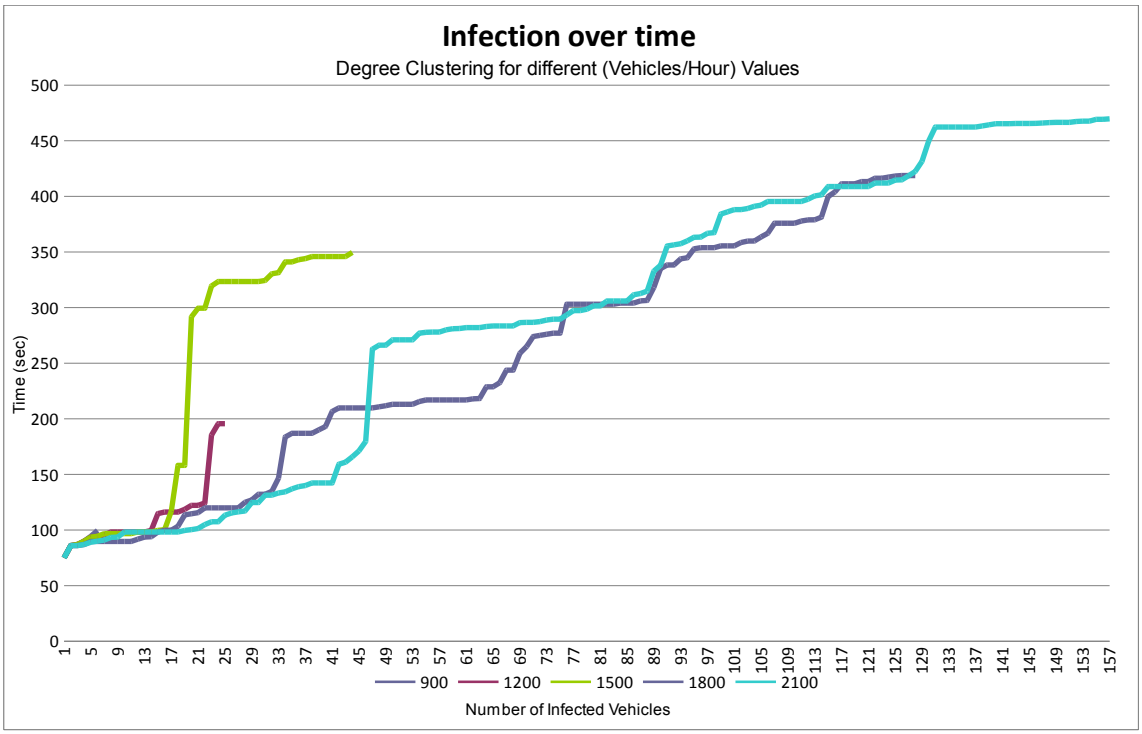


Figure 24. Number of infected vehicles –to- Time under the Degree clustering algorithm for the Urban scenario

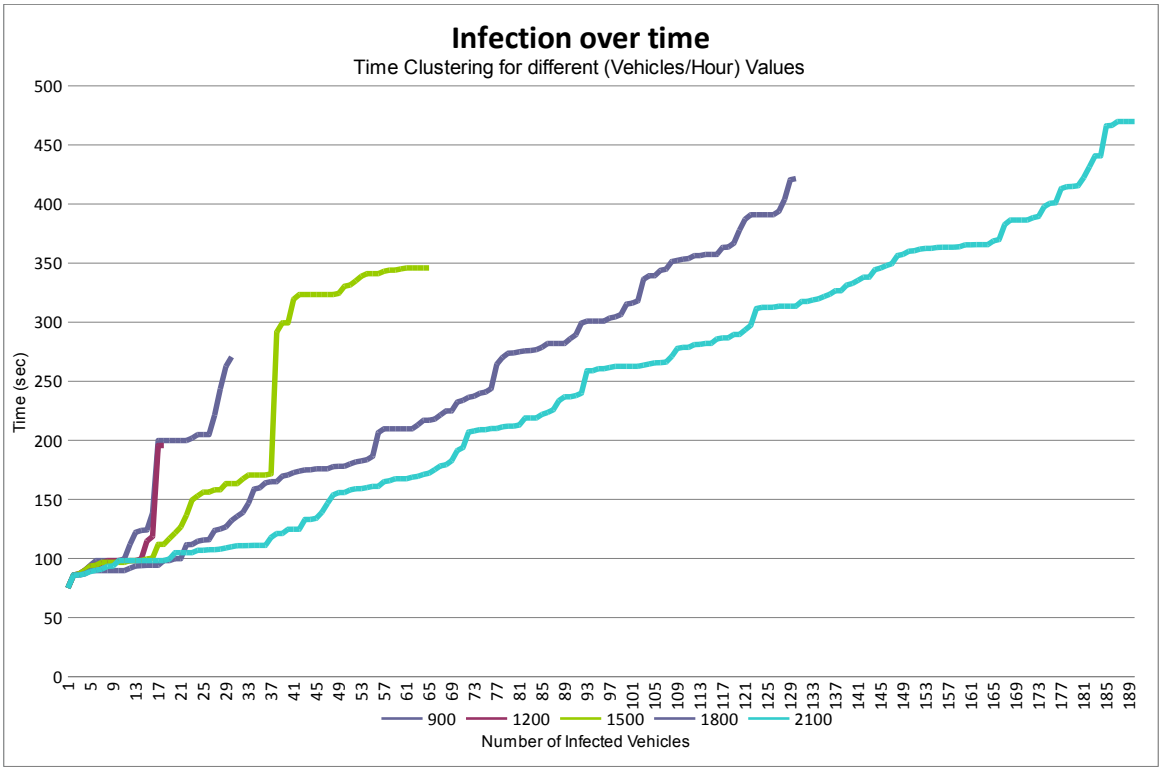


Figure 25 Number of infected vehicles –to- Time under the Time Clustering algorithm for the Urban scenario

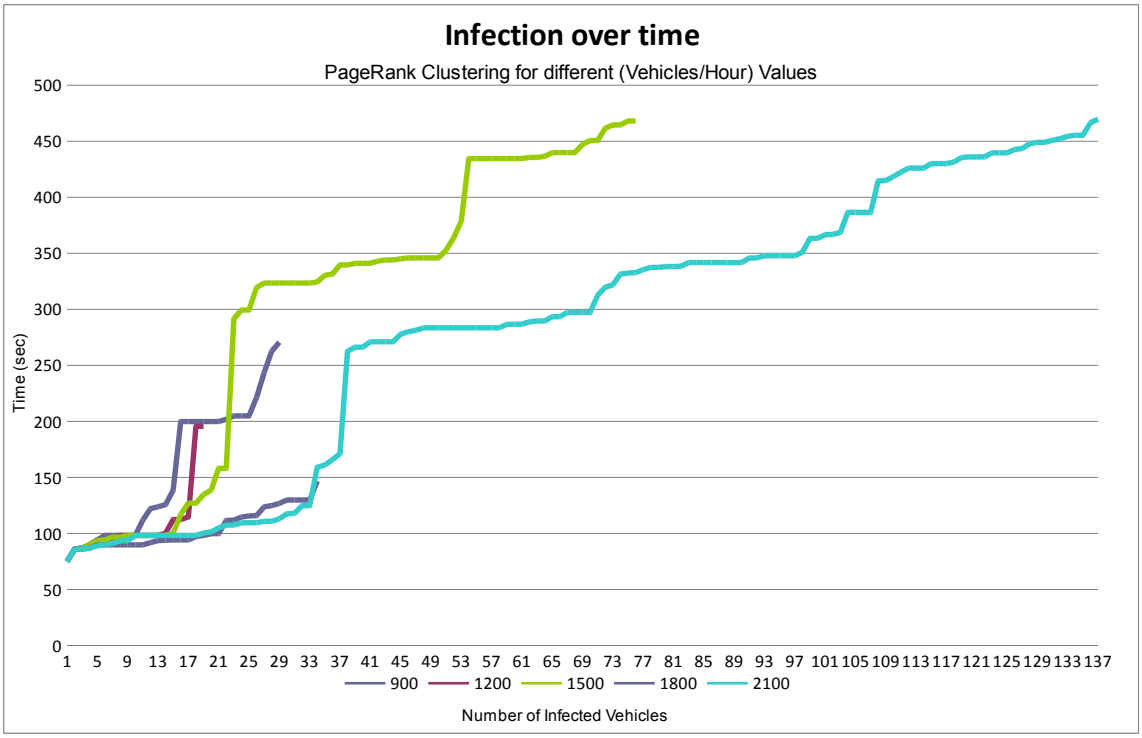


Figure 26. Number of infected vehicles –to- Time under the PageRank Clustering algorithm for the Urban scenario

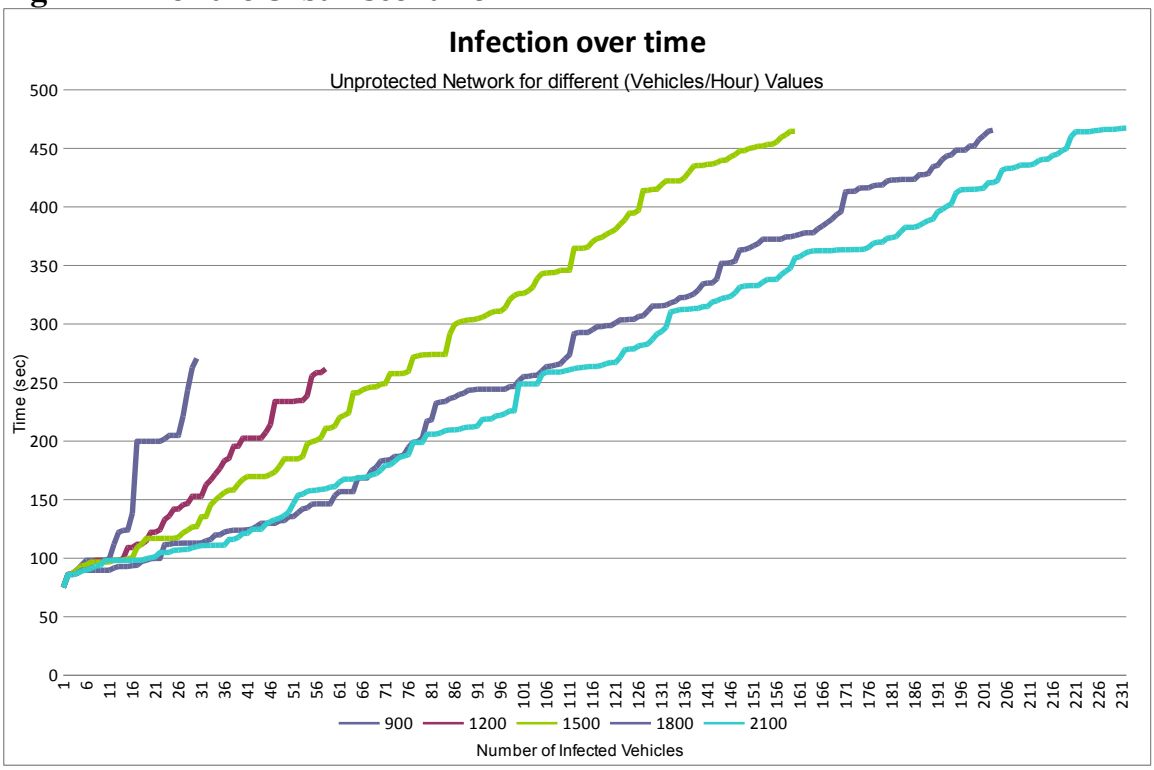


Figure 27 Number of infected vehicles –to- Time under the Unprotected Network for the Urban scenario

IMPACT OF CONNECTIVITY OF THE VEHICULAR NETWORK

Moving to the last part of our experimentation (in a similar fashion to Figures 18, 19 and 20) we present the results in Figures 28, 29. Again note that the blank spots in our graphs are when a vehicle has no neighbors, i.e. no other cars are in the vicinity and none links where cut. Here we experimented under the 1200 veh/h network density and for T=4 and 6 carrier latencies. We once more notice that PageRank clustering did a better job preserving a larger number of connected vehicles and as we mentioned before, provide in average better results for blocking the infection. From these graphs we were unable to reach the same conclusions as we did for the highway scenario. This is due to the fact than in urban scenario we confront a more challenging situation with multiple disconnections in V2V and V2I communications as a result of obstacles existence that we have mentioned earlier. The proposed mechanisms display similar results but remarkable is the performance of the PageRank algorithm although we experiment in a random route distribution.

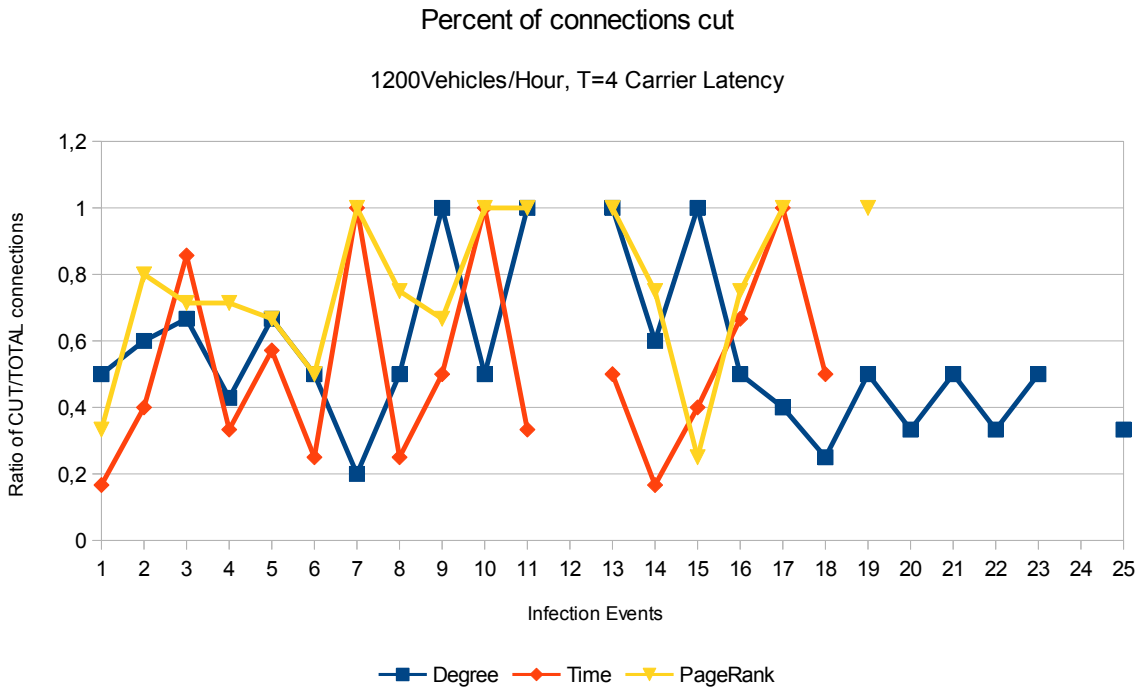


Figure 28. Infection events –to- Ratio of CUT/TOTAL connections for the Urban scenario

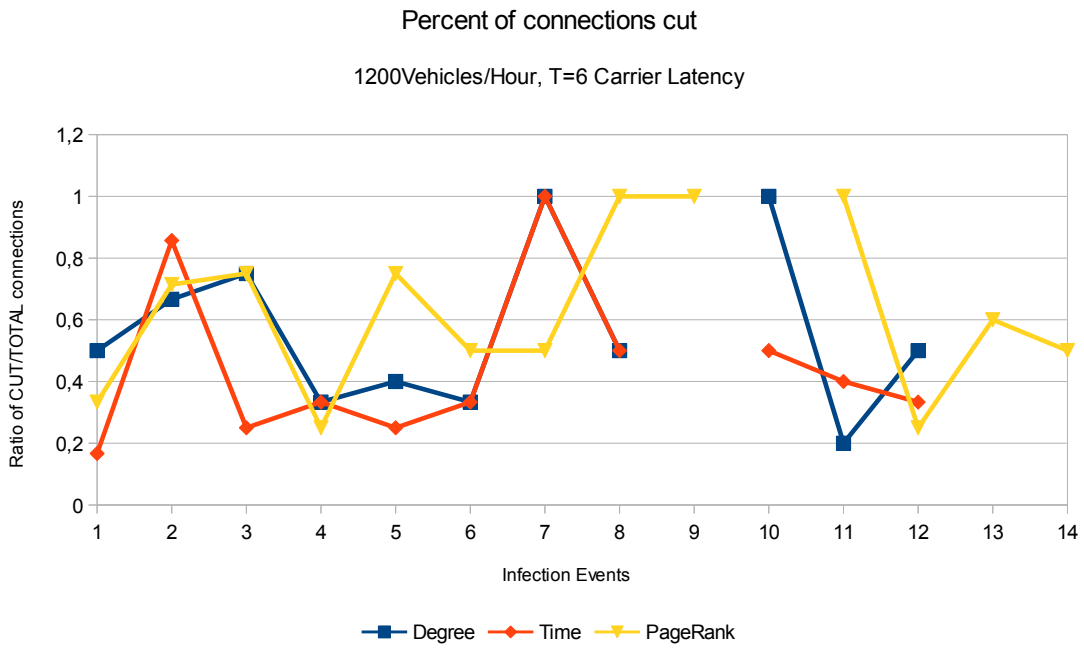


Figure 29. Infection events –to- Ratio of CUT/TOTAL connections for the Urban scenario

SPEED DISTRIBUTION OF VEHICLES

The velocities of the vehicles in our experimentation are shown in the Figure 29 and they are ranging from 8 up to 15 m/s.

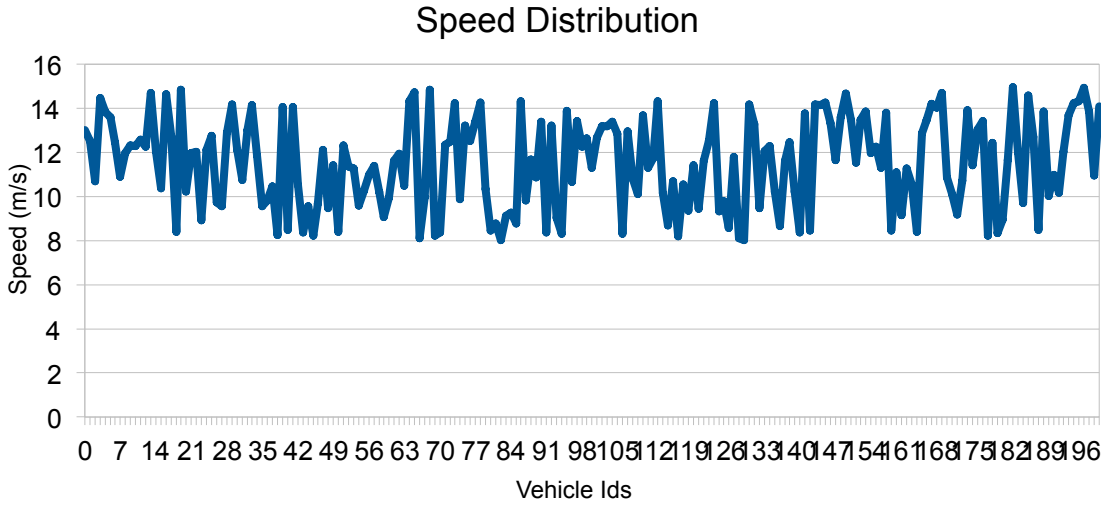


Figure 30. Vehicle Ids –to- Speed for the Urban scenario

CONCLUSION

We presented a study for blocking the spread of virus in vehicular networks. Our proposition was based in three approaches, first is the Degree Clustering by cutting neighbor nodes with higher degree. Second was Time clustering, i.e. cutting neighbor nodes with the highest connection duration with the infected source. And finally the PageRank clustering algorithm in which we cut the neighbor nodes with the higher road weight destinations. We observed that fast worms, (i.e. those with low carrier latency) infect the network topology rapidly and slow worms have more difficulties to spread and sustain.

The three models were evaluated under diverse environments such as different carrier latencies, network densities (Vehicles per hour), random speed distribution and in two scenarios Highway and Urban. In the highway scenario we find that Time clustering with the K-means algorithm implements the bests results in both duration of the virus lasted in the network and infect percent. These scenarios create different and very special challenges to the communication system and indicate the way for future implementations. In the urban scenario is a more complex environment and thus we highlight the promising results of the PageRank algorithm. The Urban scenario is more interesting due to the special elements that characterize it.

For our future work we need to simulate a more realistic Urban scenario with traffic from real data for the vehicle routes, in order to better analyze our PageRank clustering algorithm implementation. This simulation should be composed with data acquired from real time mobility of vehicles for known cities. We believe that such an experiment will result to a better and more profound understanding of all the proposed defense algorithms implemented in this Thesis.

REFERENCES

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

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

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

[4] http://en.wikipedia.org/wiki/Mobile_ad_hoc_network

[5] Syed A. Khayam and Hayder Radha "Analyzing the Spread of Active Worms over VANET " 2004.

[6] Lin Cheng and Rahul Shakya "VANET worm spreading from traffic modeling" 2010.

[7] Leandros A. Maglaras "A Novel Distributed Intrusion Detection System for Vehicular Ad Hoc Networks"2015

[8] Maziar Nekovee "Modeling the Spread of Worm Epidemics in Vehicular Ad Hoc Networks" 2006.

[9] L Cheng and R Shakya "Worm Spreading and Patching in Inter-vehicle Communications" International Journal of Communication Networks and Information Security (IJCNIS) 2010.

[10] Oscar Trullols-Cruces ,Marco Fiore and Jose M. Barcelo-Ordinas "Worm Epidemics in a Large-scale Vehicular Network" 2012.

[11] Oscar Trullols-Cruces , Marco Fiore and Jose M. Barcelo-Ordinas "Understanding, Modeling and Taming Mobile Malware Epidemics in a Large-scale Vehicular Network" 2013.

[12] http://infolab.stanford.edu/~backrub/google.html

[13] https://en.wikipedia.org/wiki/PageRank

[14] https://en.wikipedia.org/wiki/Non-line-of-sight_propagation

APPENDIX A

EXPERIMENTATION CHALLENGES

VANETs are an instantiation of mobile ad hoc networks. MANETs have no fixed infrastructure and instead rely on ordinary nodes to perform routing of messages and network management functions. However, vehicular ad hoc networks behave in different ways than conventional MANETs. Driver behavior, mobility constraints, and high speeds create unique characteristics of VANETs. These characteristics have important implications for designing decisions in these networks. Thus, numerous challenges need to be addressed for V2V communications to be widely deployed.

NODE VELOCITY

One of the most important aspects of mobility in VANETs is the potential node velocity. Nodes either denote vehicles or RSUs in this case. Node velocity may range from zero for stationary RSUs or when vehicles are stuck in a traffic jam to sometimes over 120 km per hour on highways. In particular, these two extremes each pose a special challenge to the communication system. In case of very high node velocities, the mutual wireless communication window is very short due to a relatively small transmission range of several hundred meters. For example, if two cars driving in opposite directions with 90 km/h each, and if we assume a theoretical wireless transmission range of 300m, communication is only possible for 12 seconds. Moreover, the transceivers have to cope with physical phenomena like the Doppler effect. Reviews related to V2V communication have shown that routes discovered by topology-based routing protocols get invalid (due to changing topology and link failures at high speeds) even before they are fully established. High node velocities means frequent topological changes. However, slow movements usually means stable topology, but a very high vehicle density, which results in high interference, medium access problems, etc. For such reasons, very scalable communication solutions are required.

MOVEMENT PATERNS

VANETs are characterized by a potentially large number of nodes that are highly mobile, i.e. according to car's speed. This high mobility can be more or less important depending on road nature (small streets vs. highways). Vehicles do not move around arbitrarily, but use predefined roads, usually in two directions. Unpredictable changes in the direction of vehicles usually only occur at intersections of roads. We can distinguish two types of roads:

- **City roads**: Inside cities, the road density is relatively high. There are lots of smaller roads, but also bigger, arterial roads. Many intersections cut road segments into small pieces. Often, buildings right beside the roads limit wireless communication.
- **Highways**: Highways typically form a multi-lane road, which has very large segments and well-defined exits and on-ramps. High-speed traffic encountered here.

A node can quickly join or leave the network in a very short time leading to frequent network partitioning and topology changes. These movement scenarios pose special challenges particularly for the routing. Even on a highway, that gives smooth traffic in one direction, frequent fragmentation may encountered.

NODE DENSITY

Apart from speed and movement pattern, node density is the third key property of vehicular mobility. The number of other vehicles in mutual radio range may vary from zero to dozens or even hundreds. If we assume a traffic jam on a highway with 4 lanes, one vehicle at every 20 meters and a radio range of 300m, every node theoretically has 120 vehicles in his transmission range. In case of very low density, immediate message forwarding gets impossible. In this case, more sophisticated information dissemination is necessary, which can store and forward selected information, when vehicles encounter each other. In this case, the same vehicle may repeat the same message multiple times. In high-density situations, the opposite must be achieved. Here, only selected nodes should repeat a message, because otherwise this may lead to an overloaded channel.