UNIVERSIDAD DE MURCIA

DEPARTAMENTO DE INGENIERÍA DE LA
INFORMACIÓN Y LAS COMUNICACIONES

Contributions to the Efficiency in Routing and Information Harvesting for Vehicular Networks

Contribuciones a la Eficiencia en el Encaminamiento y la Recogida de Información en las Redes Vehiculares

D. Juan Antonio Martínez Navarro

2015

Universidad de Murcia

Departamento de Ingeniería de la Información y las Comunicaciones

Contribuciones a la Eficiencia en el Encaminamiento y la Recogida de Información en las Redes Vehiculares

Tesis Doctoral

Presentada por:
Juan Antonio Martínez Navarro
Supervisada por:
Dr. Pedro Miguel Ruiz Martínez

Murcia, Diciembre de 2014

University of Murcia

Department of Information and Communications Engineering

Contributions to the Efficiency in Routing and Information Harvesting for Vehicular Networks

Ph.D. Thesis

Authored by:
Juan Antonio Martínez Navarro
Advisored by:
Dr. Pedro Miguel Ruiz Martínez

Murcia, December 2014

Agradecimientos

Me gustaría dedicar esta página para agradecer a toda la gente que, de una forma u otra, han ayudado y contribuido a que esta tesis sea una realidad.

En primer lugar a mis padres. Desde que tengo uso de razón siempre han estado a mi lado educándome, apoyándome y animándome en cada una de las etapas de mi vida por las que voy pasando. Por ello, no puedo más que estarles agradecido de por vida. También quiero agradecérselo a mis hermanos porque me han enseñado mucho desde bien pequeño.

Mi mujer *Susy*, es también un pilar importantísimo en mi vida. Gracias por aguantarme cada día, por todo tu cariño y por hacerme el hombre más feliz del mundo. A mi pequeñajo Juan Antonio, por esas sonrisas que pones en cuanto entro por la puerta. Hacen que uno se alegre y se le olvide absolutamente todo.

También quisiera dar las gracias a todos los compañeros de la sala dibulibu, por ese buen ambiente que hay. Las bromas, los consejos y sobre todo siempre listos para echar una mano cuando hiciera falta. Mención especial merecen Pedro J. y Alex que han estado conmigo desde que juntos comenzamos la carrera. No puedo olvidar a Fran, que más que un compañero ha sido casi un hermano por todas las cosas que hemos compartido gracias a la investigación.

Quisiera acordarme también de mis compañeros del proyecto *MARTA*, en especial Daniel Vigueras, Ángel Bartomeu y Víctor Cabrera, ya que ellos también han hecho posible que esta tesis fructifique.

En este punto, me gustaría deterneme para expresar mi más sincero agradecimiento a Pedro Miguel Ruiz. Desde el primer momento siempre ha tendido su mano para ayudarme en cuanto lo necesitase. Gracias por sus sabios consejos, y por esa ayuda inestimable cuando más lo he necesitado.

Por último, pero no por ello menos importante, quiero expresar mi más sincero agradecimiento a Antonio Skarmeta. Gracias por iniciarme en el mundo de la investigación y darme la oportunidad de trabajar y aprender tanto en este departamento.

There is nothing better for a man,
than that he should eat and drink,
and that he should make his soul
enjoy good in his labor.
Ecclesiastes, 2:24

No hay mayor felicidad para el hombre
que comer y beber,
y disfrutar en medio de sus fatigas.
Yo veo que también esto viene de la mano de Dios
Ecclesiastes, 2:24

Resumen

Son muchos los fenómenos que causan los accidentes en nuestras carreteras y que preocupan a las autoridades públicas de tráfico. Por un lado los excesos con el alcohol, unidos a velocidades vertiginosas provocan los accidentes más peligrosos y con mayor número de víctimas mortales. Sin embargo, por otro lado, hay otros fenómenos que también influyen en el aumento de dichos accidentes como son el uso del teléfono móvil al volante, las distracciones, las retenciones en hora punta, las grandes aglomeraciones de vehículos durante las operaciones salida y/o llegada en periodos vacacionales, o las frecuentes obras en las carreteras. Así que, como se puede observar son muy variados y de muy distinta índole los distintos motivos por los que los accidentes pueden ocurrir.

Tal cantidad de accidentes es una asignatura pendiente para nuestras autoridades públicas de tráfico. Por este motivo trabajan diariamente para reducir su gravedad y mortalidad así como para reducir la congestión tanto como sea posible haciendo posible un uso eficiente de nuestras carreteras.

Una de las tecnologías emergentes que puede asistir en la conducción y que puede ayudar por tanto a reducir los fenómenos anteriormente mencionados es el de las Redes ad-hoc vehiculares, Vehicular Ad-hoc Networks(VANETs). Éstas consisten, de forma aproximada, en una serie de vehículos equipados con dispositivos de red inalámbricos que les permite comunicarse entre ellos tan pronto entran dentro del rango de alcance. Gracias a esta comunicación vehículo-a-vehículo (V2V), en una región donde ha ocurrido recientemente un accidente, es posible difundir mensajes de emergencia a toda la zona definida por ejemplo mediante un radio de acción, previniendo de este modo a todos los vehículos cercanos de causar un daño mayor. Mas aún, se podría incluso conseguir que dichos vehículos reduzcan su velocidad de forma automática o que tomen desvíos alternativos aligerando el tráfico en la zona y permitiendo una atención más eficaz por parte de los medios sanitarios.

Existen, sin embargo, muchos desafíos a nivel técnico causados por las peculiaridades y características específicas de este tipo de redes que deben ser resueltos antes de poder realizar el despliegue de dicha tecnología.

Los protocolos de encaminamiento son la piedra angular de casi cualquier aplicación desarrollada para el entorno de las VANETs. Son, por tanto, de suma importancia ya que tienen la responsabilidad de entregar los mensajes a sus destinos asegurando la comunicación entre todos los nodos de la red, sin importar lo cercanos o lejanos que estén unos de otros. Así que, por ejemplo, si pensamos en una posible aplicación en la que dos vehículos compartan cierta información, o en una situación en la que un accidente de tráfico acaba de ocurrir y los vehículos involucrados directamente en el accidente quieren difundir un mensaje de emergencia a los vehículos cercanos que puedan verse afectados, en ambos casos son los protocolos de encaminamiento los responsables de que dicha transmisión sea exitosa.

A pesar de que ya se ha realizado mucho trabajo respecto a este tópico en otras redes ad-hoc como

son las redes móviles ad-hoc también llamadas MANETs y las redes de sensores también conocidas como WSNs, las peculiaridades de las VANETs hacen que dichas soluciones no sean practicables en estas últimas. Por otro lado, aunque también existen diversos protocolos de encaminamiento específicos para VANETs, algunas suposiciones tomadas por ellos como la selección del vecino que proporcione mayor avance hacia el destino incurren en una gran pérdida de paquetes. Por este motivo nosotros proponemos **Beacon-less Routing Algorithm for Vehicular Environments (BRAVE)**, una solución de encaminamiento oportunista que tiene soporte para redes tolerantes a retardos (DTN) y cuyas decisiones de encaminamiento son tomadas por los propios vecinos que proporcionan avance al mensaje que va a ser enviado, en vez de por el nodo emisor. De este modo, cuando los nodos vecinos reciben el mensaje de datos, ellos mismos responderán a dicho mensaje proponiéndose como siguiente nodo a reenviar dicho mensaje. Esta respuesta, no es aleatoria, sino que se realiza de forma ordenada siendo aquellos que proporcionan mayor alcance los primeros en responder. Esta solución ha sido evaluada a través de un conjunto simulaciones comparándola con los protocolos más reconocidos dentro de este ámbito y los resultados obtenidos muestran una mejora sustancial con respecto a las anteriores propuestas tanto en tasa de paquetes entregados como en el retardo medio extremo a extremo.

Todos estos protocolos de encaminamiento asumen la colaboración de todos los nodos para contribuir al fin de entregar el paquete a su destino. Sin embargo, existen usuarios maliciosos que, bien por afán de diversión o bien motivados por intereses económicos pueden tener un objetivo totalmente distinto al del resto de usuarios. Si el campo de investigación fuese otro distinto al de las VANETs, los actos de dichos usuarios tendrían consecuencias tales como la desconexión de los nodos, la eliminación de paquetes, o cualquier otro daño referido únicamente a la información transmitida por los nodos. Sin embargo, las acciones de usuarios maliciosos en las VANETs, además de las consecuencias a nivel de red que pueden deteriorar su buen funcionamiento y rendimiento, pueden tener un efecto completamente distinto ya que pueden afectar incluso a las vidas de los conductores y pasajeros de los vehículos.

Teniendo en cuenta las anteriores amenazas y ataques que pueden sufrir estas redes proponemos una mejora a nuestro protocolo de encaminamiento bajo el nombre de **S-BRAVE** reforzando la entrega de los datos en escenarios hostiles. Esta propuesta introduce el uso de diferentes mecanismos de seguridad como son el uso de los pares de claves pública y privada (PKI) que permite a los nodos firmar, validar la firma, cifrar y descifrar los paquetes que viajan a través de la red. Otra de las mejoras incluidas consiste en un mecanismo de intercambio de certificados que reduce la sobrecarga de la red. Por último, proponemos también el uso de una técnica llamada *Nodos de Guarda*, en inglés *Guard Nodes*, en la que los nodos vecinos a un nodo emisor vigilan la correcta transmisión y posterior reenvío de sus mensajes. En caso de que dicho reenvío no se produzca, los nodos de guarda reaccionarán proponiéndose ellos mismos como los siguientes a realizar el reenvío. Así se aseguran de que se lleva a cabo un nuevo reenvío, proporcionando de este modo un nuevo avance a dicho mensaje. Esta técnica ha obtenido un incremento notable en la tasa de paquetes entregados siendo una solución a tener en cuenta.

Aunque las redes vehiculares pueden operar de forma aislada sin necesidad de que haya presente una red de infraestructura que le de soporte, el número de posibilidades en cuanto a aplicaciones y servicios de valor añadido que pueden desarrollarse crece exponencialmente cuando dichas redes tienen la capacidad de conectarse a Internet. Las aplicaciones de tráfico pueden ser más rápidas y precisas

proporcionando incluso información en tiempo real a los conductores. Otras aplicaciones destinadas al entretenimiento de los pasajeros que comparten contenidos a través de la red o de reproducción de contenidos multimedia pueden ser desarrolladas, y por supuesto, todas aquellas relacionadas con la navegación de Internet: envío y recepción de correos electrónicos, recepción de noticias diarias también estarán disponibles tanto para conductores como para los pasajeros mejorando la calidad de los viajes y haciéndolos más confortables.

La conexión a Internet se puede llevar a cabo gracias a las distintas tecnologías inalámbricas que pueden ser embarcadas en los vehículos. Gracias al largo tiempo de vida de las baterías, los vehículos no tienen ningún tipo de restricción de consumo de energía. Por este motivo, pueden ser equipados con equipos más potentes, unidades de memoria más grandes así como diversas tecnologías inalámbricas como 802.11p, WiFi, Worldwide Interoperability for Microwave Access (WiMAX), General Packet Radio Service (GPRS), Universal Mobile Telecommunications System (UMTS), Long-Term Evolution (LTE) o la futura red móvil 5G. Además, los vehículos tienen a su disposición diferentes elementos desplegados en los entornos urbanos y autovías llamados Roadside Units(RSUs) así como puntos de acceso o routers WiFi proporcionados por compañías privadas para poder conectarse a Internet. La conexión a Internet a través de dichos dispositivos presenta también otro beneficio desde el punto de vista de los protocolos de encaminamiento ya que ésta puede utilizarse para derivar el tráfico destinado a vehículos lejanos reduciendo de este modo tanto la sobrecarga en la red vehicular como el retardo extremo a extremo.

Para aprovechar tales ventajas proporcionadas por las VANET híbridas (aquellas VANETs conectadas a Internet), hemos extendido nuestra anterior propuesta de encaminamiento BRAVE. Ahora, las decisiones de encaminamiento también consideran las interfaces de red disponibles en cada momento para el envío de los paquetes así como sus requisitos de calidad de servicio. Además, los nodos pueden hacer uso de las RSUs y puntos de acceso cercanos, incluso cuando no exista una conexión directa a los mismos. Todas estas características nuevas han sido evaluadas por medio de simulaciones mostrando una drástica reducción del retardo extremo a extremo en las comunicaciones vehiculares gracias al apoyo de la red de infraestructura.

Gracias a la conexión a las VANETs híbridas los usuarios pueden acceder a diferentes servicios y aplicaciones de Internet. Sin embargo, este acceso debe ser seguro para permitir que los usuarios registrados puedan acceder a los servicios subscritos, previniendo el acceso a aquellos que no lo están.

Los vehículos que accedan a estos servicios a través de las anteriormente mencionadas RSUs y puntos de acceso que actúan como pasarelas hacia Internet deben de autenticarse con cada una de ellas tan pronto como estén a su alcance. Este proceso es bastante pesado durando hasta un máximo de un par de segundos. Por este motivo se necesita una nueva estrategia de autenticación que ayude a reducir la duración de dicho proceso. Nuestra propuesta consiste en el uso de un esquema de **pre-autenticación** que permite hacer el proceso de autenticación más ágil. Nuestras evaluaciones muestran un importante beneficio al utilizar dicha técnica con una importante reducción de la sobrecarga de control de los mensajes involucrados en el proceso de autenticación. Esta mejora viene unida a un incremento en la tasa de paquetes entregados en más de un 40 %.

Finalmente, en esta tesis consideramos una aplicación diferente para las redes vehiculares. Apartando el tema de la seguridad en dichas redes, existen diferentes organismos y compañías interesadas en obtener información de diversa índole como la concentración de CO_2, la densidad de tráfico en cierta zona, la velocidad media, o cualquier otra información estadística de la que posteriormente se

puedan inferir comportamientos inadecuados al volante, zonas con un índice de contaminación muy alto, o incluso predicciones relacionadas con el mismo tema para prevenir futuros incidentes, ya sean de ámbito medio-ambiental o de tráfico en la ciudad.

Esta tarea podría llevarse a cabo situando diferentes sensores en posiciones determinadas para obtener tales medidas. Éstos, de forma periódica o a demanda realizarían la actividad de sensorización correspondiente y transmitirían dicha información hacia un nodo central encargado de realizar el análisis de dichos datos. Esta medida supondría por tanto un gasto en realizar dicho despliegue. Sin embargo, gracias al progreso de la tecnología de sensorización, así como su miniaturización, éstos pueden ser embarcados en los vehículos que pueden actuar como sensores móviles, evitando tal despliegue de sensores. A las redes de vehículos que proporcionan tal fin se las conoce como redes de sensores vehiculares Vehicular Sensor Network(VSN).

Aunque ya se han diseñado y desarrollado diferentes protocolos de recogida de información para este tipo de redes, éstos no tienen en cuenta la sobrecarga que causan en la red. A lo sumo ellos aprovechan los mensajes que ya hay circulando en la red para añadir sus valores reduciendo el número de mensajes pero aumentando su tamaño. Sin embargo, nuestra propuesta llamada **Compressed Sensing based Vehicular Data Harvesting(CS-VDH)** si que considera dicho tamaño. De hecho aplica una técnica de compresión llamada Compressed Sensing(CS) que es capaz de reducir enormemente el tamaño ocupado por los datos en el paquete permitiendo aún así una posterior reconstrucción muy precisa de los datos comprimidos. De hecho, en nuestras simulaciones hemos obtenido una notable disminución en el tamaño de los paquetes de datos así como un error de recuperación del 0.5 % de los datos.

En resumen, a lo largo de esta tesis centrada en el ámbito de las redes vehiculares, hemos contribuido a diferentes áreas como son las de encaminamiento, seguridad, control de acceso y a la recogida de información. Todas estas propuestas han sido evaluadas y comparadas en igualdad de condiciones con propuestas anteriores encontradas en la literatura y nuestras soluciones han proporcionado unos resultados de rendimiento que mejora notablemente dichas propuestas.

Abstract

Many are the phenomena that nowadays cause accidents in our roads and that worry our public traffic authorities. On the one hand, alcohol excesses, together with vertiginous speeds provoke the most dangerous accidents and the ones with the most mortal victims. Nevertheless, on the other hand, there also exist other phenomena which also influence the increase of road accidents like the use of the mobile phone at driving, distractions, traffic jams in rush hours, the enormous number of vehicles at the beginning and at the end of holidays or frequently road works in our roads. So, as we can see, the reason why road accidents can happen are very varied and of very different nature.

Such a high number of road accidents is a pending subject for our public traffic authorities. For this reason, they daily work to reduce their riskiness and mortality as well as reduce the congestion as much as possible making an efficient use of our roads

One of the emergent technologies which can assist drivers and therefore can help in the decrease of these tragic phenomena is that of Vehicular Ad-hoc Networks (VANETs). They roughly consists of a series of vehicles equipped with wireless network interfaces which allow them to communicate with each other as they enter inside their radio range. Thanks to this Vehicle-to-Vehicle (V2V) communication, warning messages can be spread over the region where a traffic accident happened, making nearby vehicles to reduce their speed or taking detours for instance. However, there exist a lot of technical challenges caused by the specific and particular characteristics of these networks that must be solved before the deployment of such a technology can be achieved.

Routing protocols are the building block for nearly every application developed under the scope of VANETs. They are of paramount importance because they are responsible for providing communication among all the nodes of the network, even when the communicating nodes are located far away ones from the others. So, if we think, for instance, of a possible application of two vehicles sharing certain data, or in a situation where a traffic accident has just happened and the colliding vehicle spreads a warning message to the vehicles which can get involved, a routing protocol is needed to hand over these messages to their destination.

However, despite a lot of work regarding this topic has been made under other ad-hoc networks like Mobile Ad-hoc Networks (MANETs) and Wireless Sensor Networks (WSNs), the particularities of VANETs made the proposals designed for Mobile Ad-hoc Networks(MANETs) and Wireless Sensor Networks(WSNs) not to be suitable for this latter one. On the other hand, although there are diverse VANET-specific routing protocols some assumptions like selecting the farthest node within the radio range that provides the most advance to the destination incurs in many packet losses. For this reason we propose **Beacon-less Routing Algorithm for Vehicular Environments (BRAVE)**, an opportunistic routing solution with support for Delay Tolerant Network (DTN) and whose routing decision is made by the neighbouring vehicles that provides advance for the message to be sent, instead

of by the current forwarding node. Thus, after overhearing the message, nodes propose themselves as the next forwarder answering first the nodes that provide more advance to the packet. This solution has been evaluated by means of simulation. Results show that it outperforms the other routing solutions presented in the literature in terms of packet delivery ratio and average end-to-end delay.

These protocols assume all nodes collaborate to obtain the goal of handing over packets to the destination. However, there are users who, well by fun or by economical interests may have a different objective. In other ad-hoc networks the consequences of their acts can mean the disconnection of a node, or the drop of several packets making them not to be delivered. Nevertheless, inVANETs these actions can cause victims because the messages that a malicious user can be dropping can be a warning message of a traffic accident avoiding nearby vehicles to react in time causing a more dangerous one. For this reason, security also takes a primary role in VANETs. There are other relevant attacks which can be performed into these networks like the injection of packets, the manipulation of the information of traversing packets and the likes which can deteriorate the behaviour of the network reducing its capacity and performance.

Considering the aforementioned threats we reinforce our previous routing solution which is presented by the name of **S-BRAVE**. This proposal introduces the use of public/private key pairs allowing nodes to sign, validate, cipher and decipher packets traversing the network. We propose a mechanism to exchange certificates reducing the overhead and finally employ a technique called guard nodes by which neighbouring nodes of a forwarding one overhear the correct transmission and posterior forwarding of messages, electing themselves as new forwarding nodes assuring the correct forwarding of this message. Our proposal has obtained a notable increase of the packet delivery ratio being a proposal to be aware of.

Although VANETs can operate standalone without requiring network infrastructure to be present, the number of possibilities of application and value-added services exponentially grows when they are connected to Internet. Traffic applications can be faster and more accurate providing even real-time information to drivers; entertaining applications like the streaming of multimedia contents can be consumed, peer-to-peer applications for sharing also different contents can be available; and everything related to surfing the Internet, sending and receiving e-mails, watch daily news will be available to both drivers and passengers improving the quality of their journey making them more comfortable.

The connection to the infrastructure network is possible thanks to the different wireless technologies that vehicles can be equipped with. Thanks to their long-life batteries they do not have any energy consumption requirements. So, they can be equipped with powerful CPUs, larger memory units and diverse wireless technologies like 802.11p, WiFi, WiMAX, GPRS, UMTS, LTE or the future 5G cellular technology. In addition they can take advantage of nearby Roadside Units (RSUs) deployed by public traffic authorities and free Access Points(APs) provided by private companies to get such access to the infrastructure. The infrastructure also presents other benefits regarding the point of view of routing protocols. It can be used to derive traffic aimed at far away vehicles reducing both the current overload of the ad-hoc network, reducing also the end-to-end delay in this sort of communications.

To take advantage of these features provided by **hybrid-VANETs**, we extend **BRAVE** in such a way that routing decisions are enriched considering the available wireless interfaces and their Quality of Service (QoS) features to provide the more appropriate route for a packet according to their QoS requirements. Nodes can also take advantage of nearby RSUs even when they do not have direct access. Our experiments show that this protocol reduces drastically the end-to-end delay in vehicular

communications thanks to the infrastructure support.

The access to the services provided by the infrastructure network must be secured so as to make registered users to gain access to their subscribed services, preventing also non-registered users to access the services. Vehicles accessing these services through the RSUs and APs acting as gateways towards the infrastructure network must be authenticated with each one as they are reachable. This process is very cumbersome lasting up to a couple of seconds. For this reason a new authentication strategy is needed to reduce the time wasted in this process. We propose the use of **pre-authentications** to make this authentication process more agile. Our evaluations show an important benefit of the use of this technique with an important reduction of the control messages to perform the authentication and an increase of the packet delivery ratio in more than a 40%.

Finally, in this thesis we consider a different application of VANETs. Putting the safety issue aside, different public organisms are interested in retrieving certain information like the CO_2 concentration, the traffic density, average speed and the likes from a determined region of an urban scenario. This is usual in big cities were this statistical information is a valuable tool to make predictions about driving behaviours and allow traffic authorities to have a global view of the traffic in the city.

This task could be achieved by placing different sensors in specific positions to obtain such measurements. However, thanks to the progress of the sensing technology and its miniaturization, vehicles are nowadays equipped with more and more sensors. So, vehicles can also take this task avoiding the effort of deploying the sensors. Such networks are usually called **Vehicular Sensor Network (VSN)**. Although there exist different harvesting protocols to perform this operation, they are not aware of the overload they cause in the network. At the most, they piggyback the information of different nodes to reduce the amount of messages, but increasing the corresponding payload of these packets. Our proposal called **Compressed Sensing based Vehicular Data Harvesting (CS-VDH)** applies a compressing technique called Compressed Sensing (CS) capable of reducing this payload by compressing the values in the packet, being also able to obtain later on an accurate reconstruction of the compressed data. The obtained results prove this technique to reduce the overload of the network by decreasing the size of the data packets providing also up to a 0.5% of recovery error.

In summary, along this thesis focused on the field of VANETs, we have contributed to the areas of routing, security and gathering information providing solutions that have obtained good results and which outperform existing proposals found in the literature.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

A-STAR Anchor-based Street and Traffic Aware Routing. 44, 47, 58–60

AAA Authentication, Authorization and Accounting. 28, 106–108, 110

ABR Associativity Based Routing. 18

AH Authentication Header. 28

AODV Ad-Hoc On Demand Distance Vector. 18, 35, 36, 40, 85

AP Access Point. 2, 5, 8, 28, 83, 85, 89

AVP Attibutes Value Pair. 30

BCS Bayesian Compressed Sensing. 127, 140

BOSS Beacon-less On Demand Strategy for Geographic Routing in Wireless Sensor Networks. 52

BRAVE Beacon-less Routing Algorithm for Vehicular Environments. 5, 8, 33, 35, 50–52, 54, 57–64, 66–68, 70, 74, 78–81, 83, 84, 91, 93, 95, 98, 103, 105, 112, 132, 135

BS Base Station. 86, 87

C2C-CC Car-to-Car Communication Consortium. 12, 13

CA Certification Authority. 64, 68–70

CALM Continuous Air-interface for Long & Medium range telecommunications. 14

CAR Connectivity-Aware Routing. 36, 37

CBF Contention-based forwarding. 40

CGSR Clustered Gateway Switch Routing. 17

COOPERS Co-operative Systems for Intelligent Road Safety. 13

CR Compass Routing. 22, 23

CRL Certificate Revocation List. 70

CS Compressed Sensing. 9, 122, 124–133, 136–143

CS-VDH Compressed Sensing based Vehicular Data Harvesting. 9, 128

CSA Common of Sub Areas. 57

CVIS Cooperative Vehicle Infrastructure Systems. 12

D-VADD Direction First Probe. 45

DB-VDG Delay-Bounded Vehicular Data Gathering. 124, 136, 137, 142

DoS Denial of Service. 27, 65

DSDV Destination-Sequenced Distance-Vector. 17

DSR Dynamic Source Routing. 18, 35

DSRC Dedicated Short Range Communications. 13, 54

DTN Delay Tolerant Network. 21, 34, 47, 91, 105, 121, 135

DVG Dependent Vehicular Group. 87

EAP Extensible Authentication Protocol. 28–30, 107, 108, 110

EDD Expected Disconnection Degree. 40

ESP Encapsulating Security Payload. 28

ETSI European Telecommunications Standards Institute. 13, 14

FC Fusion Center. 121–125, 127–130, 132, 134–136, 138, 139, 141–143

FCA Forwarder Coverage Area. 57

FIFO First In First Out. 46

FILO First In Last Out. 46

FSR The Fisheye State Routing. 17

FSS Forwarding Set Selection. 40

GeOpps Geographical Opportunistic Routing for Vehicular Networks. 26, 43, 58–61, 132

GFG Greedy-Face-Greedy algorithm. 24

gpcr Greedy Perimeter Coordinator Routing. 37, 38, 41, 42, 47, 58–60, 132

GPRS General Packet Radio Service. 8, 13, 90, 103

GPS Global Positioning System. 14, 22, 48

GPSR Greedy Perimeter Stateless Routing for Wireless Networks. 24, 35, 36, 41, 45, 132

GpsrJ+ Gpsr Junction+. 38, 41

GSR Geographic Source Routing. 41, 44, 47, 58–60

HVN Heterogeneous Vehicular Network. 86, 87

IEEE Institute of Electrical and Electronics Engineers. 13, 14

IETF Internet Engineering Task Force. 108

IKE Internet Key Exchange Protocol. 28–30

IP Internet Protocol. 28

IPSec Internet Protocol Security. 28, 29

IR Infra-Red. 13

ISO International Organization for Standardization. 13, 14

ITS Intelligent Transportation System. 11, 13, 14, 85

L-VADD Location First Probe. 45

LOUVRE Landmark Overlays for Urban Vehicular Routing Environments. 42

LS Location Service. 93–95

LTE Long-Term Evolution. 3, 83, 84, 90, 103, 105

MAC Media Access Control. 60, 61, 99

MANET Mobile Ad-hoc Network. 11, 14, 16, 17, 21, 31, 33, 35, 49, 89, 107, 108

MDDV Mobility-centric Data Dissemination Algorithm for Vehicular Networks. 45

METD Minimum Estimated Time Of Delivery. 43

MFR Most Forward within R. 22, 23

MIBR Mobile Infrastructure Based VANET Routing Protocol. 88

MoVe Motion Vector Scheme. 42

MPARP Mobility Pattern Aware Routing Protocol. 86, 87

MURU MUlti-hop Routing protocol for Urban vehicular ad hoc networks. 40

NC Nearest Closer. 23

NFL Neighbourhood Feedback Loop. 23

NFP Nearest with Forwarding Progress. 23

NGN Next Generation Network. 64

NoW Network on Wheels. 12, 36

NP Nearest Point. 43

NSA Number of Sub Areas. 57

OBU On-Board Unit. 1, 15, 83

OEM Original Equipment Manufacturer. 12

OLS Overlay Location Service. 85

OLSR Optimized Link State Routing. 17, 85

PaA PANA server. 30

PaC PANA client. 29, 30

PANA Protocol for Carrying Authentication for Network Access. 28–30, 107, 108, 110–112

PBR-DV Position-Based Routing with Distance-Vector recovery. 36, 66

PDR packet delivery ratio. 33, 59, 60, 62, 78, 79, 81, 91, 99, 101–104, 113, 116, 119

PKI Public Key Infrastructure. 64, 67–69, 107

QoS Quality of Service. 5, 91, 92

RHR Right Hand Rule. 36, 38

RoI Region of Interest. 90, 91, 94, 98, 99, 101, 102, 110, 112, 114, 123, 124, 128, 130, 132, 135, 139, 141

RSU Roadside Unit. 2, 5, 8, 9, 12, 14, 28, 83, 84, 90–95, 98, 99, 101–105

RVM Relevant Vector Machine. 140

SADV Static-Node Assisted Adaptive Routing protocol. 46

SAR Spatially Aware Routing. 41, 58–60, 132

SND Secure Neighbour Detection. 66

TO-GO TOpology-assisted Geo-Opportunistic Routing. 38

TORA Temporally Ordered Routing Algorithm. 18

TSK Transient Session Key. 108

TTL Time To Live. 60

UMTS Universal Mobile Telecommunications System. 3, 8, 13, 83, 84, 90–92, 97–99, 103, 105

V-Grid Vehicular Grid. 85

V2I Vehicle-to-Infrastructure. 11, 13, 85, 89, 90

V2V Vehicle-to-Vehicle. 2, 11, 13, 64, 85, 87, 89–91

VADD Vehicle-Assisted Data Delivery. 26, 45

VANET Vehicular Ad-hoc Network. 1–6, 8, 9, 11–16, 20, 21, 24, 26–28, 30, 31, 33–35, 37, 43, 46–50, 52–54, 57, 61–64, 66, 67, 69, 80, 83–85, 87–95, 97–99, 101, 103, 105–107, 109–111, 114, 119–121, 132, 135, 136

VITP Vehicular Information Transport Protocol. 124

VLS Vehicle Grid Location Service. 85

VNI Virtual Navigation Interface. 43

VSC Vehicular Safety Consortium. 13

VSN Vehicular Sensor Network. 6, 9, 15, 121–123, 125, 128, 130, 136, 140

WAVE Wireless Access in Vehicular Environments. 14, 63, 64

WiMAX Worldwide Interoperability for Microwave Access. 8, 83, 84, 86, 97, 103, 105

WRP Wireless Routing Protocol. 17

WSN Wireless Sensor Network. 5, 15, 64, 121, 122, 124, 125, 127, 130, 136

Chapter 1

Introduction

People usually use vehicles for diverse aspects of the daily life like: going to work or home, to make a holiday trip, or simply to go to a determined meeting point to go out with friends for instance. This massive use of vehicles cause a high vehicle density in our roads.

So many vehicles circulating increase the probability of a traffic accident to happen, and actually the amounts of victims caused by them are worrying for public traffic authorities. This is precisely the primary target of the development and deployment of VANETs in real life. By using wireless communications, vehicles can exchange traffic information which can make the act of driving safer allowing drivers to know about certain events on the road like traffic jams, traffic accidents, road works and the likes before being in the area where these events are happening.

Private industry provided a different point of view of the use of these networks. VANETs can also improve the quality at driving allowing both drivers and passengers to enjoy their journey with different entertaining applications.

In this initial chapter, we first explain what these networks are explaining their features and the challenges they present. We also overview the main objectives addressed by this thesis as well as the employed methodology. Finally we detail its organization and structure.

1.1. Vehicular ad-hoc networks

In the last decades, advances on computer architectures and communications have made these technologies available to nearly every one. Nowadays, everybody owns a PC, laptop or smart-phone which allows them to check the e-mail, navigate over the World Wide Web and/or play on video-games. These advances are also reflected in the vehicular industry. Vehicles are now equipped with more and more electronic systems to improve both security and quality at driving, Fig 1.1.

Communication is so important for vehicles that the scientific community has been actively investigating the field of Vehicular Ad-hoc Networks (VANETs). That is, a mobile ad-hoc network whose mobile nodes are vehicles circulating along different roads. These vehicles are able to communicate with others thanks to an equipment usually called On-Board Unit(OBU) which integrates at least a wireless interface. So, as a vehicle enters in the radio range of one another, they are able to exchange information.

The efforts employed in this field can be noticed because they are starting to be used in real

Figure 1.1: An example of the electronic equipment of a premium class vehicle.

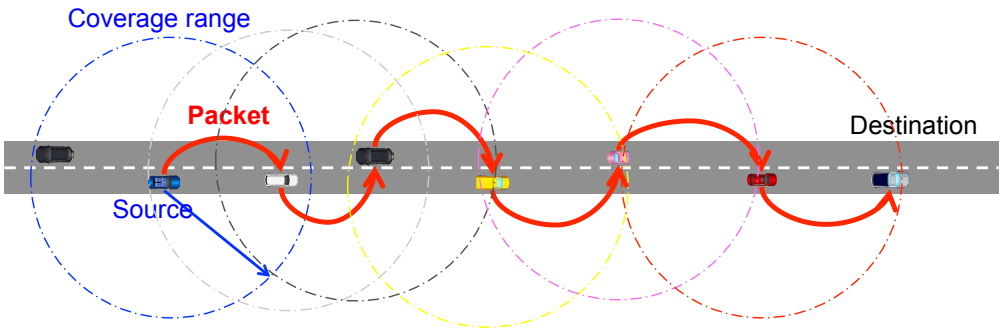


Figure 1.2: Multi-hop communication between vehicles.

life. Premium class vehicles, for instance, incorporate cellular interfaces with 3G/4G connectivity to provide multimedia streaming content as well as navigating over the World Wide Web while the vehicle is moving.

It is also possible to communicate nodes located farther than their radio range by using intermediate nodes as Fig. 1.2 shows. This sort of communication is called *multi-hop*, because packets travel hop-by-hop through intermediate nodes until they reach their destination. To do so, it is necessary to supply nodes with intelligence to make routing decisions. That is, handing over the packets to an intermediate node closer to the destination. This is the reason why the nodes of a VANET need to run a **routing protocol**.

There are other elements that make VANETs even more interesting. They are usually deployed by public traffic authorities whose main purpose is that of obtaining real-time information about traffic and/or informing drivers about certain events on the road like traffic congestion, traffic accidents, roadworks and so on. They are called Roadside Unit (RSU) and they can be used as gateways to the infrastructure. The same happens with any other free AP that also private commerces can offer. Such architecture in which vehicles can gain access to infrastructure networks is named hybrid VANET.

Since we have introduced a lot of elements which compose a VANET, let us depict them in Fig. 1.3 which sums up what a VANET is.

In blue, we can distinguish Vehicle-to-Vehicle (V2V) communication by using the aforementioned wireless interface. Besides, they can also communicate with APs and RSUs by using the same or a different wireless interface. Since these fixed elements are connected to the Internet, vehicles attached

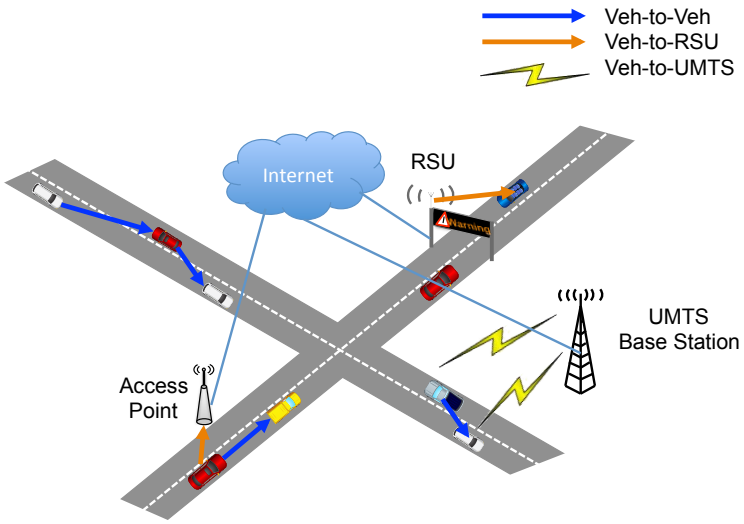


Figure 1.3: Hybrid VANET example.

to them can use them as gateways to the infrastructure network. Finally, by using a cellular 3G/4G interface like UMTS or LTE they can also gain direct access to the infrastructure.

VANETs have specific properties that make them challenging to the scientific community. Since mobile nodes are vehicles, their **velocity** can reach up to 36.1 m/s (about 130 km/h) in highways and up to 13.9 m/s (50 km/h) in urban environments. Besides, vehicles cannot move freely. Their **movements are constrained** by the streets and roads of the urban scenario. Finally, accelerations and decelerations happen continuously in an urban scenario due to the traffic signs and traffic lights causing **continuous partitions in the network**.

The emerging interest drawn by both public traffic authorities and private industry over VANETs is due to the concern about the numerous traffic accidents that usually happen in our roads with the consequent amount of lives taken away. Actually, the first target of VANETs is to make driving safer taking advantage of the information that vehicles can exchange, allowing a faster distribution of the different events that can occur (roadworks, traffic accidents, traffic jams, ...) allowing drivers to react faster to them.

Other interests were also put in these networks to offer different services like the access to real-time information, multimedia streaming services, peer-to-peer platforms to exchange multimedia content, and the likes making them very appealing.

1.2. Motivation

As we have just seen, not only public traffic authorities have put interest in the research and development of VANETs. Many private companies have found in them a market niche where they can develop interesting value-added services and applications which can make the journey more comfortable for both drivers and passengers. Hence, the scientific community has actively investigated this field to provide knowledge and useful tools for new developments and deployments in VANETs.

Within the topic of VANETs there are many areas where researchers have made a lot of contributions. The first and most important one is related to the transmission of information among nodes. Actually, communicating nodes, even when they are located far away ones from the others is primordial for nearly every application in VANETs. Achieving this task is responsibility of multi-hop routing protocols. Installed in every node of the network, allow them to make the appropriate routing decisions.

Although current VANET routing protocols have included different sort of information in its routing decisions like the direction of the current vehicle, its speed, information about the streets that vehicles follows and so on, they have not obtained the performance they were expected. Cabrera et al [2] provided a detailed study of the most common routing protocols in which they explain their main flaws contrasted with simulation-based experiments like the use of stale positions, beacon losses, or the transmission range assumption.

In Chapter 3 we tackle this issue, providing a detailed explanation of these drawbacks and defining a new routing protocol which make the most of the VANET features outperforming these ones.

The main purpose of VANETs deployment is that of improving safety on the road thanks to wireless technology among others. In other terms, by the use of VANETs it is expected to reduce dramatical traffic accidents, avoid traffic jams, improve quality at driving, and the likes. This is the reason why security in these networks gains more importance. A bad use of the information transmitted along the network could have a direct impact in the vehicles behaviour, and unlike other networks in VANETs, mobile networks are vehicles which transport people.

Raya and Hubaux [3] presented a study for securing VANETs. They characterized the attitude of an attacker in four dimensions: insider vs. outsider; malicious vs. rational; active vs. passive; and local vs. extended. Besides, they also presented a list of different attacks that could happen in these networks grouped according to its complexity. Among them we can mention some of the most commons ones like *denial of service* where the attacker jams the channel or injects massive packets to bring down the VANET; *sink hole* or *selective forwarding* where the attacker drops all the received messages or only several of them to impair the network; *bogus information* in which the attacker diffuses wrong information; a *sybil attack* used by attackers to receive information because they falsified their position.

In addition, standardization bodies have taken seriously security in their communication architectures and standards. In spite of this, security breaches can be discovered by malicious users who will exploit them in their benefit.

The problem is that these security attacks do not affect equally to all the developments. Depending on the exchange of messages needed to deliver the data packets, as well as the information inserted in the transmitted packets some development will be more resilient to these attacks than others.

Focusing on the routing layer of these networks. One of the assumptions taken in the design of routing protocols is that all the nodes of the network collaborate for a successful delivery of packets. However, real scenarios are not that way. As comented before, there are obscure interests as well as people with bad intentions whose target is another totally different from collaboration. Malicious users are able to manipulate information of traversing packets, inject information in them or use any other methods to obtain sensible information, impersonate, impair the network or cause a disaster at higher levels.

In Chapter 4 we address the issue of securing the routing layer. In particular we will focus or

efforts in our previous VANET routing protocol BRAVE.

Despite VANETs can be totally meaningful on their own as isolated networks where vehicles exchange information about their positions, directions, speeds as well as broadcasting emergency information in certain areas to allow vehicles and drivers to avoid certain events on the road, their applications and services make a difference when they are connected to the infrastructure network which provides it access to the Internet.

Safety services are upgraded if vehicles upload real-time information of the road state. Centralised services can also offer real-time warning services with higher accuracy, and of course, there are a lot of companies interested in developing applications to diffuse real-time traffic news, stream multimedia content, connect to social networks and the likes making journeys more comfortable.

From the point of view of the routing layer, routing protocols should also benefit of this new ability. Actually, it supposes a great opportunity to improve both the performance of routing protocols in both packet delivery ratio and average end-to-end delay.

The possibilities are infinite: vehicles communicating with the Internet services, downloading multimedia content, using peer-to-peer platforms to share information; users subscripted to certain services which are aware of the location of the vehicles; public traffic authorities broadcasting real-time traffic information to drivers travelling over a determined area; and no more importantly taking advantage of the infrastructure network to communicate faster and more reliable vehicles locate far away ones from the others.

From the point of view of the routing layer, this connection also presents a challenge. Vehicles are now equipped with more than one network interface and the routing protocol must select the most suitable one depending on the QoS requirements of the packets. Moreover, routing protocols can take advantage of the infrastructure network to shorten the path between two very distant mobile nodes. We address this interesting task in Chapter 5.

From the point of view of application layer, the provision of these services and applications usually comes with an authentication process that ensures that only registered users gain access to them. This access can be achieved via RSU or AP if the vehicles cannot afford to be equipped with other kind of wireless technology like 2G/3G. However, since vehicles move at high speeds, their time attached to a RSU is limited. This implies, that whenever a vehicles reach a new RSU it has to restart the authentication process time and time again.

One of the drawbacks of the authentication process is precisely the time needed to complete the authentication process. It is a very heavy process taking from several hundreds of milliseconds to a couple of seconds. So, from all the time that the RSU/AP is available, a precious period of time is employed in continuous authentications. We deal with this issue in Chapter 6.

Finally, there is another application of VANETs which is emerging in these last few years. WSNs are usually used for gathering diverse kind of information usually sensed through small devices with low battery lifetime called sensors. Although in rural environments or military scenarios, the deployment of such devices are a good choice, in urban environments there is a cheaper way of doing it.

Vehicles are equipped with a lot of sensors so as to make sure that they are in good condition to be driven, or to show different weather conditions values. Like we have also commented, they have the ability to relay packets to other vehicles so, we can use them to measure the value of interest and the VANET to transmit this information. This new application of VANETs is called VSNs.

Adding new applications to VANETs suppose an increase in its overload, so this is the reason why

all the designs and developments are aware of the scalability of the network reducing at maximum the amount of packets exchanged by the nodes as well as their size. Consequently, VSNs proposals try to achieve the same goal. It is not sufficient to develop a protocol that is able to harvest the data, it must be efficient in the aforementioned terms. For this reason, in Chapter 7 we propose a harvesting protocol for these kind of network.

Next section presents a list of the objectives pursued in this thesis.

1.3. Objectives

Our target in this thesis is that of providing advance to the state of the art in the VANETs context. In particular to these three different topics: routing, security and gathering solutions. To achieve these goals concrete objectives must be established for each one of them.

Regarding routing in VANET we aim at:

- Analysing and comprehending the limitations of routing in VANETs.
- Designing an improved routing protocol.
- Implanting and evaluating the performance of routing protocols.

Security is a very general topic within VANETs, actually it covers all the layers of the VANET architecture. However, our objectives in this topic are twofold: strengthen routing protocols to be resilient to the security threads and attacks present in these networks, and propose a more efficient access control technique allowing vehicles to authenticate with gateways in a faster way.

For each of the first one we must first:

- Study and analyse the different security threats and attacks presents in VANETs
- Analyse the vulnerabilities that routing protocols have against these attacks.
- Develop a mechanism to strengthen routing when malicious users are present.

For the latter one:

- Study the different authentication mechanisms used when nodes access to the infrastructure network or to certain services.
- Analyse the main drawbacks of this process and provide a solution which alleviate them.

Finally, regarding the gathering protocols, our targets are the following:

- Analysing the current gathering protocols.
- Designing an improved routing protocol able to reduce the amount of transmitted messages as well as their size by compressing the carried information.

These goals are difficult to fulfil without a methodology which guide us with defined steps we have to follow.

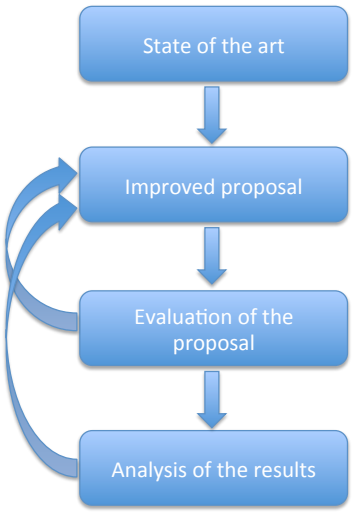


Figure 1.4: Followed methodology.

1.4. Methodology

The methodology employed to tackle the different problems addressed in this thesis follows the diagram of Fig. 1.4.

First of all, a great effort was made to analyse the related state of the art of the different involved technologies, acquiring this way, a precise definition the problem to be addressed.

Once this analysis is completed, obtaining the main drawbacks and flaws of the existent technology, a new improvement proposal is defined and analysed.

After validate it, it is time to assess its performance. This phase can be done by mathematical models, by means of simulations or moreover by real experiments. It is also time to define the metrics of interests for the problem (packet delivery ratio, control overhead, throughput, reliability, end-to-end delay, ...).

Since our problems involves hundreds of nodes, we evaluated the performance by means of simulations, concretely we have used the Network Simulator ns-2 (version 2.33)[1]. We have also employed another well-known tool called Sumo tool[2] to generate realistic traffic patterns. This evaluation not only consists in obtaining the performance of our proposal but also in compare it in equal conditions with the other proposed solutions found in the literature.

During the evaluation of the different proposals, new facts can allow us to tune or modify our proposal to obtain a better performance.

After the evaluation phase, we usually obtain a huge amount of data which must be processed. In our case, we obtain trace files describing all the events that happened during the simulation. These files are parsed so as to obtain the previously defined metrics for both our proposal and the other solutions.

This analysis allows us to extract a lot of kind of conclusions about the behaviour of the different simulated solutions. They can be used not only to prove the performance of the different simulated

[1]The network simulator ns-2. http://www.isi.edu/nsnam/ns/

[2]SUMO - Simulation of Urban MObility. http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/

proposals but also to improve our proposed solutions.

Next section enumerates the main contributions which are derived from this thesis.

1.5. Main contributions

In this section we briefly describe the main results obtained within the development of this thesis. They have been published in several international peer-reviewed conferences and journals. A list of such publications can be found in Section 8.3.

Beacon-less Routing Algorithm for Vehicular Environments (BRAVE). Very reliable and scalable routing protocol for VANETs which uses an opportunistic approach to forward packets using intermediate nodes. A sending node first broadcasts the data packet within its neighbourhood (one-hop neighbours). Afterwards, the neighbouring nodes which provide advance to the packet, i.e. the ones located closer to the destination than the current node, propose themselves as candidates answering to the originator node. This response is scheduled by a delay function allowing farther nodes to answer first providing the most possible advance for the selected packet. In case no neighbour provides advance to the destination, it stores the packet until it finds a new neighbour. By means of a simulation-based study, our approach shows a high packet delivery ratio compared with previous approaches in the literature even in sparse scenarios.

S-BRAVE. A routing proposal aimed at reinforcing the packet delivery ratio under hostile scenarios. This protocol adopts the technique of guard nodes to verify the correct forwarding by intermediate nodes. So, candidates to be the next forwarder but are not elected, after receiving the data packet do not remove it. They overhear the new selected forwarder to make sure it goes on with the transmission of the data packet. So, if they do not observe the retransmission of the packet by the selected node, they take its role transmitting it to the next hop. We evaluated the performance of this proposal by means of simulations and this technique increases the packet delivery ratio in presence of attackers or malicious users.

HYB-BRAVE. We extended our previous protocol aimed at vehicle-to-vehicle communication allowing it to deal not only with 802.11p, but with other different wireless technologies like GPRS, WiMAX, UMTS, etc. In addition, our design takes advantage of the RSUs and other APs to both communicate with nodes of the infrastructure as well as using them to shorten the path followed by data packets sent from the VANET and aimed at a far away vehicle of the same network. Our simulations proved the advantage of a hybrid network by increasing the delivery ratio and drastically decreasing the average delay per packet.

An efficient proposal for access control in VANETs. A protocol to make the authentication process more agile against gateways. It uses a mechanism called pre-authentication by which a node already authenticated with a gateway uses it to start a new authentication process. The difference with the traditional scheme is that now the node uses the current gateway to start the authentication with the new one. This way, when a node changes from its current gateway to the new one, it will not waste its time in the authentication process making the most of it to communicate with the infrastructure network.

Compressed Sensing based Vehicular Data Harvesting (CS-VDH). A harvesting protocol used in VSNs. This protocol uses an emergent compression technique developed in the field of the Theory of Information called Compressed Sensing (CS) whose main advantage is the possibility to compress information adding new elements to it without hardly increase the size of the compressed information and also permitting an accurate reconstruction of the information. Our simulations prove this benefit reducing the control overhead in up to 2000 messages, providing also a compression of even 4.5 elements to 1 with an accurate reconstruction of the information.

1.6. Organization of this thesis

The remainder of this thesis is organized as follows. Chapter 2 provides the reader with the needed background on vehicular ad-hoc networking to understand the rest of the document. This chapter highlights the most important issues which affect communication in ad-hoc networking.

We investigate the unicast routing protocols for VANETs in Chapter 3 analysing their features, advantages and disadvantages. We also propose a routing protocol which outperforms the other previous proposals in terms of packet delivery ratio.

Despite the good performance obtained by the protocol, it is not aware of non-collaborative and/or malicious nodes which try to impair the performance of the network. For this reason, we propose a mechanism to improve the packet delivery ratio in scenarios with malicious nodes in Chapter 4. The mechanism consists of using the neighbouring nodes that provide advance to the packet to monitor the correct forwarding of the packet.

Going back to the routing problem, in Chapter 5 we study the problem of connecting the VANET to the infrastructure via different wireless technologies as well as using the RSU deployed by public traffic authorities aimed at providing and obtaining real-time information about the traffic state. From the point of view of the routing layer, it presents several challenges like the use of different network interfaces, or the use of the infrastructure to shorten the path followed by packets between two distant nodes.

The services and applications provided in hybrid VANETs usually need an authentication process to identify the user. This process is very heavy taking up to a couple of seconds to be achieved. So, as a vehicle moves it will authenticate with each RSU it finds to access to these services. In Chapter 6 we propose a mechanism to accelerate this process reducing the time spent in continuous authentications.

Finally, in Chapter 7 we study another application of the VANET which is the use of the vehicles as a mobile sensor network to gather information. In this chapter we propose the use of a compression technique called Compressed Sensing (CS) to make the gathering process more lightweight.

Finally, Chapter 8 concludes this document, summarizes some of the open research issues that are to be continued in the line of this work, and lists the most relevant publications which support the results presented within this thesis.

Chapter 2

Background

After a first introduction to this thesis, in this chapter we provide the background needed to address this thesis.

We will begin with a historical view of VANETs explaining their main origins and their particular features which makes them different. The necessity of multi-hop communications will motivate the use of multi-hop routing protocols which are analysed next.

Security issues are also dealt providing the reader of the different security threads and attacks existent in VANET as well as some mechanisms to overcome them.

2.1. History of VANETs

Vehicular Ad-hoc Networking (VANET) is a form of wireless ad-hoc network to provide communications among vehicles and nearby roadside equipments. Actually, it is one of the most practical and direct applications of a more generic kind of ad-hoc network called MANETs. MANETs enable users to communicate without any physical infrastructure regardless of their geographical location, that is why it is sometimes referred as an infrastructure less network. They are self-organizing and adaptive allowing nodes to detect the presence of others and perform the necessary operations to facilitate communication and sharing of data.

However, Vehicular Ad-hoc Network (VANET) arise to cover a necessity of this era where the majority of the people uses a vehicle in their daily life, either his own car or the public transport service to go to work, to move between different towns or even inside the same town or city for many other reasons. This huge amount of vehicles circulating along the streets and roads has a dramatical side effect which is the increasing amount of traffic accidents which takes numerous lives away [4, 5]. Another side effect, which are correlated with the population density of the city is also the continuous traffic jams in rush hours with the consequent desperation of the drivers and passengers of the vehicles.

This motivated the use of automated highway systems and Intelligent Transportation System (ITS) to accelerate the development and use of intelligent integrated safety systems that use information and communication technologies so as to increase road safety and reduce the number of accidents in our roads.

VANETs are a cornerstone of the envisioned Intelligent Transportation System (ITS) because by enabling vehicles to communicate with each other via V2V, as well as Vehicle-to-Infrastructure (V2I)

by using roadside base stations also called Roadside Units (RSUs) or any other wireless interface contribute to make roads safer and more efficient.

The opportunities and areas of application of VANETs grew rapidly due to the interest awaken by vehicle manufacturers, telecommunication operators and other private companies actively supporting research and development in this field. Such applications go from emergency warning applications to others more entertaining like sharing multimedia content among vehicles.

2.1.1. Related projects and standardization efforts

According to Tsugawa [6, 7], probably the earliest research into inter-vehicle communication was conducted by the Association of Electronic Technology for Automobile Traffic and Driving (now the Japan Automobile Research Institute) in the early 1980s.

However, from the 1990s to the 2000 only American PATH [8] in 1994, and European Chaffeur [9] in 1997 investigated vehicular communications. In particular, the deployment of automated platooning systems by using the information transmitted among vehicles.

From 2000 on, this trend have drastically changed. Either in the U.S.A. or in the European Union many projects were funded with private and public resources to promote the investigation in such networks.

CarTalk2000 (www.cartalk2000.net). This European project was funded within the Information Society Technologies cluster of the EU's 5th Framework Programme for R&D. Its objective is focused on driver assistance systems allowing early reactions like braking if some vehicles in front have had to use emergency breaking.

Another application was the dissemination of critical information such as congestion, fog, ice or an accident in the immediate vicinity of the danger spot. The standardization among all the European Original Equipment Manufacturers(OEMs) was also another target of the project which has been addressed in the Car-to-Car Communication Consortium(C2C-CC).

FleetNet - Internet on the Road. The project was set up by a consortium of six companies and three universities: DaimlerChrysler AG, Fraunhofer Institut für offene Kommunikationssysteme (FOKUS), NEC Europe Ltd., Robert Bosch GmbH, Siemens AG, TEMIC Speech Dialog Systems GmbH, Universities of Hannover and Mannheim, and Technische Universität Hamburg-Harburg and Braunschweig.

The main objective of FleetNet was to develop platform for inter-vehicle communication systems and implement demonstrator applications to show the benefit of inter-vehicle communication systems. A study on business cases and market introduction strategies complemented the technical objectives and the project results were opened to appropriate international standardization bodies.

Network on Wheels (NoW) (www.network-on-wheels.de). This project went on the work carried out by Fleetnet. Its objective was to specify and standardise a communication system for transmission of sensor data and further information between vehicles. Not only safety applications were developed, but also non-safety ones by connecting vehicles to the Internet.

Cooperative Vehicle Infrastructure Systems (CVIS) (www.cvisproject.org). It started in 2006 with the main purpose of designing, developing and testing the technologies needed to

allow cars to communicate with each other and with the nearby roadside infrastructure. For such a task, they analysed the use of a wide range of technologies including cellular networks (GPRS, UMTS), WLAN, Dedicated Short Range Communications (DSRC), Infra-Red(IR).

Vehicles which take advantage of this new technology are able to obtain the quickest route to a destination thanks to real-time information provided by the own vehicles. Besides, they can also show to the drivers road warnings thanks to their displays and the communication to the infrastructure.

Another of the targets of this project is also the application and validation of the ISO CALM standards for continuous mobile communication providing feedback about them to the global standardisation bodies.

SAFESPOT is another research project co-funded by the European Commission Information Societ Technologies. It is aimed at designing cooperative systems for road safety based on V2V and V2I. To do so, the partners developed a Safety Margin Assistant that detects in advance potentially dangerous situations extending this way in both space and time, drivers' awareness of the surrounding environment.

Co-operative Systems for Intelligent Road Safety(COOPERS) (www.coopers-ip.eu). In this case, the target of this project is the use of the infrastructure network incorporating bidirectional infrastructure-vehicle links as an open standardised wireless communication technology. The information exchanged by both vehicles and the infrastructure will improve the safety of the road.

SEVECOM. This project aims at vehicular networks but from the point of view of security in communications. It is one of the firsts in analysing the different threats and attacks that can be performed to VANETs. Through this analysis the partners provided a path that the vehicular architecture must follow to limit the range of these attacks.

COMeSafety. The COMeSafety (www.comesafety.org) initiative was started to support the preparation of cooperative vehicle safety systems based on wireless communications. To do so, it provides an open integrating platform for both the exchange of information and the presentation of results. They also complement these efforts with the dissemination of this information in electronic newsletters and publications at major conferences and journals.

The list grows up to nearly a hundred projects. A vast list of them can be looked up in [10].

In Spain, a great effort was done by the State to promote the research in this context too. Actually there where two research projects worth remarkable, **m:via** and **Marta** with the collaboration 15 companies and 13 public research organisms. This collaboration was reflected not only by theoretical contributions to the scientific community but also with real experiments allowing researchers to check the validity of their proposals with real vehicles in a controlled segment of an urban scenario.

The emerging interests gained by VANETs is also reflected by the work carried out by established work groups of major standardizations bodies like International Organization for Standardization(ISO), European Telecommunications Standards Institute(ETSI), Institute of Electrical and Electronics Engineers(IEEE), and consortia such as Vehicular Safety Consortium(VSC) and C2C-CC to coordinate both public organisms and private industry defining different standards for ITS.

The IEEE defined standards for vehicular communications Wireless Access in Vehicular Environments (WAVE) [11, 12]. They were based on the IEEE 802.11 standard for wireless local and metropolitan area networks. Their radio range was desired to reach approximately 1000 meters working in the frequency band of $5.850 - 5.925\ GHz$.

The ISO proposed a set of standards referred as Continuous Air-interface for Long & Medium range telecommunications (CALM) [13] which assumes that vehicles are equipped with more than one wireless technology. This standard propose an architectural communication framework which intercepts the data received by any of the network interfaces and is able transmit/re-transmit also information through the most appropriate one depending on the quality requirements of the information to be sent. In addition, the ETSI based its proposed architecture of an ITS station [14] (personal device, vehicle, RSU or central node) on the work carried out by the ISO.

2.2. Features of VANETs

Although VANETs can be considered a particular kind of MANETs, they have different particularities that make them different to other ad-hoc networks [15]. As a consequence, direct application of previous proposals from other environments are not usually suitable for this new one. In the following paragraphs we describe these characteristics:

Energy consumption and storage requirements. Vehicles count with a long life battery supply. In other ad-hoc networks this was a very strict requirement making the development of protocols to be aware of it. Such a requirement is not present in these networks, so vehicles can be equipped with more powerful CPUs, include devices like Global Positioning System (GPS) receivers and navigation systems, and count with different wireless technologies with larger coverage range.

Predictable mobility. Unlike other ad-hoc networks, vehicles mobility is restricted by the roadways. This information is usually available from positioning systems and map-based technologies such as GPS. So, given a certain vehicle position and velocity parameters like its direction, acceleration, speed and the likes its future position can be estimated.

High mobility. Vehicles circulate obeying traffic rules and traffic signs in highways with two or more lanes per direction which limits their speed to 33.3 m/s (about 120 km/h) in most of the countries. By contrast, in urban scenarios, the speed is limited to a maximum of 13.9 m/s (about 50 km/h). Both speeds are higher than in any other mobile ad-hoc networks.

Highly dynamic topology. Due to the high speed of vehicles, the variability of the links between neighbouring vehicles is higher than in other ad-hoc network. As a consequence the topology of the network varies at a high rhythm. In ideal conditions (radio range of 1000 m) , two cars circulating at a speed of 25 m/s (90 km/h) and driving in opposite directions will have a link during 40 s.

Various communication environments. VANETs are usually evaluated in two kind of scenarios. Highway and urban scenarios. In highways, the restriction of the mobility of the vehicles is harder because they can only follow the highway in one direction or the opposite. By contrast, in cities, the numerous streets, traffic signs, intersections and the likes make the assessment these networks more complicated.

Frequently disconnected network. The different vehicle densities depending on the period of the day, the importance of the streets, and the effects of both junctions and traffic lights seriously affect the behaviour of the VANET. This cause continuous network partitions, splitting groups of vehicles, merging another ones, or isolating others. In addition, in lowly dense scenarios, the whole network can be disconnected preventing certain nodes to communicate with others.

Interaction with on-board sensors. Nodes are assumed to be equipped with on-board sensors to provide information such as the geographic position via a GPS receiver or the acceleration via an accelerometer. This information is very useful for the development of new protocols so it must be available for them.

The last of these features have awakened a different purpose for VANETs. The use of sensors has been widely employed by WSNs to monitor wide natural areas of restricted or complex access, environmental control, or to serve other military purposes. Thanks to them, and their wireless communication, they are able to relay the sensed information to a pre-established destination which will take the responsibility of analysing them.

This strategy can also be used in urban scenarios obtaining the same result. Nonetheless, vehicles are now equipped with these sensors too. So, instead of making an effort and a expense to deploy such sensors. Vehicles are able to make such a task. This is how Vehicular Sensor Networks (VSNs) emerged.

These networks also require multi-hop communication to transmit the sensed data to a pre-established destination and to exchange messages which also increase the overload of the network, so new techniques must be employed to make it scalable providing an interesting research niche studied by the scientific community.

These features also determine the design and development of routing protocols aimed at VANETs. They must use localized information, because obtaining global information implies a great overload due to the high variability of the links among nodes. This localized information, like the position of the neighbouring nodes is primordial because they are the first step in transmitting the information, having vague or imprecise information about them will end in a very poor performance.

Every VANET specific protocol must also scale properly due to the large size of these networks. Actually, there are certain period of times where VANETs count with a huge amount of vehicles. This usually happens at rush hours in cities and the road network close to them.

2.3. Multi-hop routing protocols

One of the contributions of this thesis is focused on the routing protocols for VANETs. As we commented in Chapter 1, VANETs consist of a number of vehicles equipped with an OBU with wireless capabilities. Such vehicles, making use of these devices can communicate with another neighbouring vehicles as soon as they are within their radio range, see Fig. 2.1.

The objective of routing protocols is to extend this communication even when these vehicles are farther than the radio range. To do so, they make use of intermediate vehicles to relay the transmitted messages until it reach the destination.

So, by using a routing protocol, a node after identifying the destination of a received packet by looking at its header, makes a decision about what to do with the packet. This decision is called

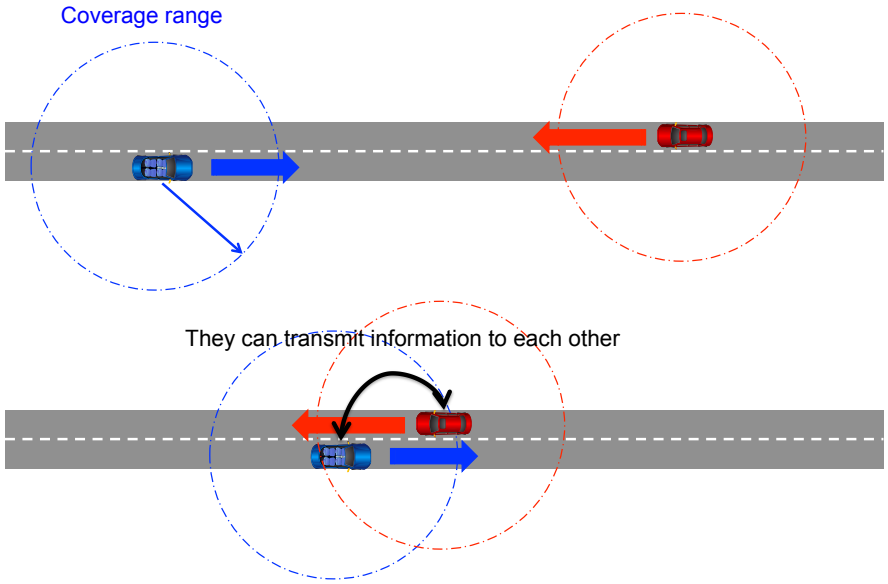


Figure 2.1: An example of two vehicles communicating when they are within the radio range of each other.

routing decision. If the destination of the packet is not the node itself, its routing decision should be to forward the packet to another node. Nevertheless, this decision cannot be made randomly. The node must know something about its surrounding environment so as to make an appropriate decision and that is precisely the challenge of routing protocols (Fig. 1.2).

Routing protocols can be classified according to the sort of destination. For instance, in the previous example, we assumed that the destination of the packet is a single node. The kind of routing associated to this sort of destination is called **unicast routing**. However, there exist other kind of destinations:

Multicast routing. Let us imagine, that several nodes without wondering about their location are subscribed to certain information, for instance a daily news summary of a certain on-line version of a newspaper. Although the source of the information is only one, the information must reach the users. So we have a single packet with multiple destinations.

Broadcast routing. There exist other situations where a certain information sent by a node must be spread to the whole network. Broadcast routing protocols are responsible for such a task flooding the network hop by hop.

Geocast routing. In ad-hoc networks there exists another kind of destination. Instead of defining a node or a group of nodes as a destination, the selected destination is a geographical region of interest. This case is very useful to announce certain information like roadworks, traffic jams and such to the vehicles circulating inside the aforementioned geographical region of a VANET.

We have already stated that VANETs are one of the most useful applications of MANETs. Actually, they share many similarities like self-organization, self-management, low bandwidth and short radio transmission range. For this reason, most of the proposed solutions for MANETs were brought

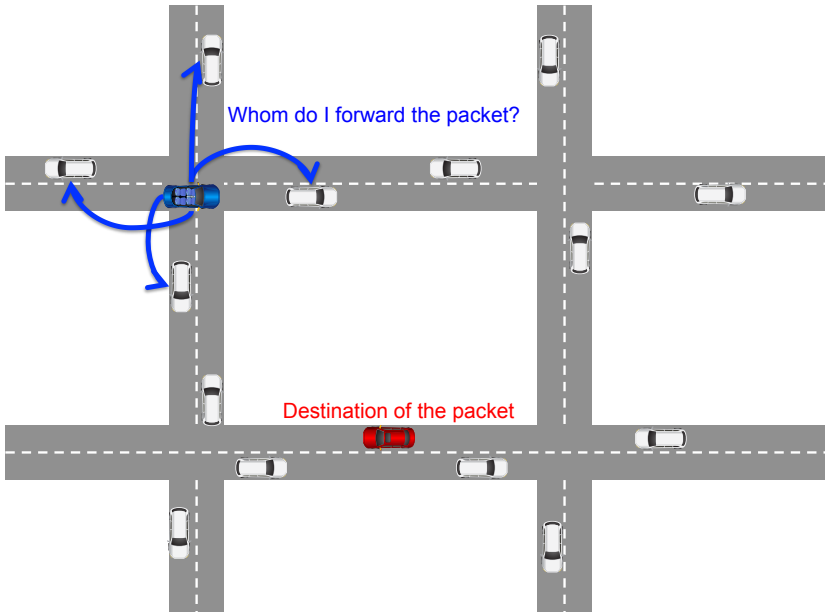


Figure 2.2: Intelligence of routing protocols. How to forward the packet?

to this new research area expecting to obtain the same result and this what happened to the research of the routing topic too.

According to many comparative analysis and surveys like ones by Panda, Feeney, Royer et al., Abolhasan et al. or Kumar et al. [1, 16–19], MANET routing protocols are usually classified in two different groups: proactive, reactive, and hybrid routing protocols. Fig. 2.3 presents a diagram with the main MANET routing protocols grouped according to these three aforementioned groups.

Proactive routing protocols: These protocols attempt to maintain up-to-date routing information about every pair of nodes in the network. To do so, they periodically propagate routing updates at fixed intervals. Among the main ones we can highlight: Destination-Sequenced Distance-Vector(DSDV) , Clustered Gateway Switch Routing(CGSR), Wireless Routing Protocol(WRP), Optimized Link State Routing(OLSR) and The Fisheye State Routing(FSR). In addition, in Table 2.1 we provide a list of the main proactive routing protocols highlighting the following features:

- Routing structure: It can be flat where all the nodes of the network have the same role, or hierarchical where certain nodes takes more responsibility. The latter kind of strategy is used when nodes are grouped by clusters and a leader of each cluster is elected.
- Number of tables: Proactive routing protocols, maintain the network information in such structures. However, depending on the implementation the requirements are totally different, allowing ones to work with only one table while others require more.
- Frequency of updates: Nodes must periodically exchange information about the state of the network in order to maintain it up-to-date. This periodicity varies depending on the protocol.

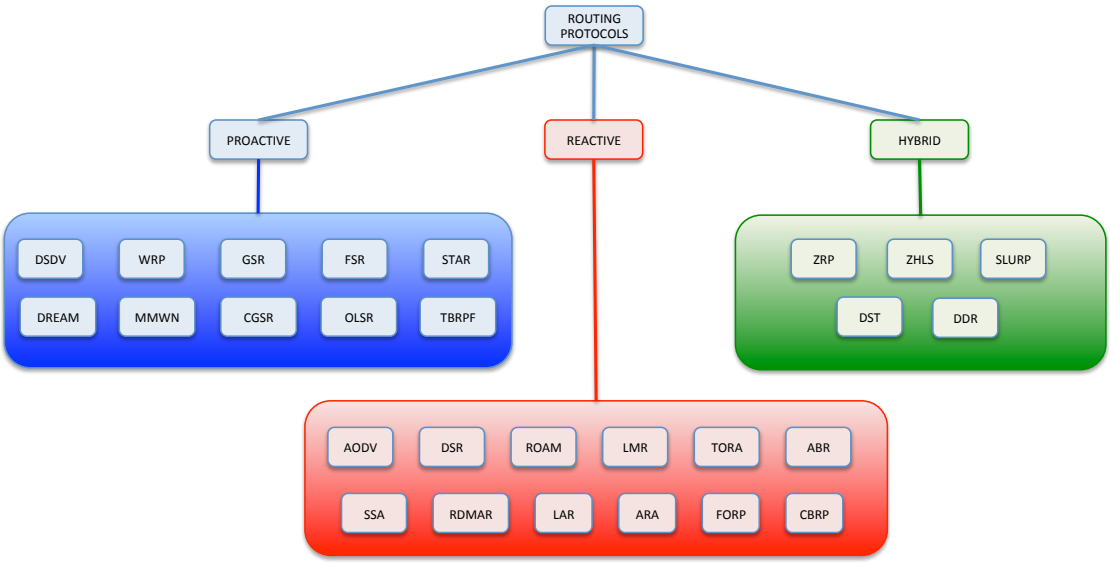


Figure 2.3: Taxonomy of MANET routing protocols.

- Use of Hello messages: These ones are also periodically broadcast by nodes to announce its presence in the network.
- Critical nodes: These are special nodes with a pretty important role in the protocol.
- Characteristic feature: Below this column we mention the key feature of each protocol.

Reactive routing protocols: These ones follow a totally different strategy. Instead of maintaining such as costly information during the whole time, nodes start a discovery process to know the position of the destination when there exists a demand for it. In other words, when a node intends to transmit certain information to a destination node, it triggers a discovery process to know its location building a path towards it. The most representative reactive protocols are: Dynamic Source Routing (DSR), Ad-Hoc On Demand Distance Vector (AODV), Temporally Ordered Routing Algorithm(TORA) and Associativity Based Routing(ABR). As we have done with the proactive protocols again we present a table (Table 2.2 summarizing their main features:

- Routing structure: The same as in proactive routing protocols. It can be flat where all the nodes of the network have the same role, or hierarchical where certain nodes takes more responsibility. The latter kind of strategy is used when nodes are grouped by clusters and a leader of each cluster is elected.
- Multiple routes: Storing multiple routes towards allows nodes to find a cached route towards a destination without triggering a discovery process.
- Beacons: Beacons are periodic messages issued by nodes to advertise their presence to the neighbouring nodes.
- Route metric method: Although these protocols are based on a query stage to discover the destination. They differ in how they rely on them, by adding extra information about their lifetime, stability, signal strength and the likes.

Protocol	RS	Number of tables	Frequency of updates	HM	Critical nodes	Characteristic feature
DSDV	F	2	Periodic and as required	Yes	No	Loop free
WRP	F	4	Periodic	Yes	No	Loop freedom using predecessor info
GSR	F	3 and a list [1]	Periodic and as local[2]	No	No	Localised updates
FSR	F	Same as GSR	Periodic and local [2]	Yes	No	Controlled frequency of updates
STAR	H	1 and 5 lists	Conditional[3]	No	No	Employes LORA and/or ORA. Minimize CO
DREAM	F	1	Mobility based	No	No	Controlled rate of updates by mobility and distance
MMWN	H	Maintains a database	Conditional[3]	No	Yes, LM	LORA and minimized CO
CGSR	H	2	Periodic	No	Yes, Clusterhead	Clusterheads exchange routing information
HSR	H	2 (link-state table and location management)[4]	Periodic, within each subnet	No	Yes, Clusterhead	Low CO and Hierarchical structure
OLSR	F	3 (Routing, neighbour and topology table)	Periodic	Yes	No	Reduces CO using MPR
TBRPF	F	1 Table, 4 lists	Periodic and differential	Yes	Yes, Parent node	Broadcasting topology updates over a spanning tree

RS = routing structure; HM = hello message; H = hierarchical; F = flat; CO = control overhead; LORA = least overhead routing approach; ORA = optimum routing approach; LM = location manager

Table 2.1: Characteristics of proactive protocols [1].

[a]GSR also has a list of all available neighbours.
[b]In GSR and FSR link-state is periodically exchanged with neighbouring nodes.
[c]In conditional update methods, the updates occur if a particular event occurs.
[d]Number of link-state tables may vary according to the number of logical levels.

- Route maintained in: Routes are stored in route tables or route caches. The second ones are limited so a replacement policy must be defined to, for instance maintain those more recently used.
- Route reconfiguration strategy: When a certain link of the route gets broken, the message cannot go on. In this case, a reconfiguration strategy to discover the destination is needed.

Hybrid routing protocols: These protocols are both proactive and reactive in nature with the objective of increase the scalability. To do so, nodes with close proximity proactively maintain routes to near by nodes, while for farther nodes they use a route discovery strategy. Again, in Table 2.3 we summarize their main features:

- Routing structure: The same as in proactive routing protocols. It can be flat where all the nodes of the network have the same role, or hierarchical where certain nodes takes more responsibility. The latter kind of strategy is used when nodes are grouped by clusters and a leader of each cluster is elected.
- Multiple routes: Storing multiple routes towards allows nodes to find a cached route towards a destination without triggering a discovery process.
- Beacons: Beacons are periodic messages issued by nodes to advertise their presence to the neighbouring nodes.
- Route metric method: Although these protocols are based on a query stage to discover the destination. They differ in how they rely on them, by adding extra information about their lifetime, stability, signal strength and the likes.

Protocol	RS	Multiple routes	Beacons	Route metric method	Route maintained in	Route reconfiguration strategy
AODV	F	No	Yes, hello messages	Freshest & SP	RT	Erase route then SN or local route repair
DSR	F	Yes	No	SP, or next available in RC	RC	Erase route the SN
ROAM	F	Yes	No	SP	RT	Erase route & [1]
LMR	F	Yes	No	SP, or next available	RT	Link reversal & Route repair
TORA	F	Yes	No	SP, or next available	RT	Link reversal & Route repair
ABR	F	No	Yes	Strongest Associativity & SP &[2]	RT	LBQ
SSA	F	No	Yes	Strongest signal strength & stability	RT	Erase route then SN
RDMAR	F	No	No	Shortest relative distance or SP	RT	Erase route then SN
LAR	F	Yes	No	SP	RC	Erase route then SN
ARA	F	Yes	No	SP	RT	Use alternate route or back track until a route is found
FORP	F	No	No	RET & stability	RT	A Flow_HANDOFF used to use alternate route
CBRP	H	No	No	First available route (first fit)	RT at cluster head	Erase route then SN & local route repair

RS = routing structure; H = hierarchical; F = flat; RT = route table; RC = route cache; RET = route expiration time; SP = shortest path; SN = source notification; LBQ = localised broadcast query.

Table 2.2: Characteristics of reactive protocols [1].

[a]Start a diffusing search if a successor is available, else send a query with infinite metric.
[b]Route relaying load and cumulative forwarding delay.

- Route maintained in: Routes are stored in route tables or route caches. The second ones are limited so a replacement policy must be defined to, for instance maintain those more recently used.
- Route reconfiguration strategy: When a certain link of the route gets broken, the message cannot go on. In this case, a reconfiguration strategy to discover the destination is needed.

Protocol	RS	Multiple routes	Bc	Route metric method	Route maintained in	Route reconfiguration strategy
ZRP	F	No	Yes	SP	Intrazone and interzone table	Route repair at point of failure and SN[1]
ZHLS	H	Yes, if more than one virtual link exists	No	SP, or next available virtual link	Intrazone and interzone table	Location request[2]
SLURP	H	Yes, depending on if a leading node is found by MFR	No	MFR for interzone forwarding. DSR for intrazone routing.	Location cache and a node_list	SN, then location discovery
DST	H	Yes, if available	No	Forwarding using the tree neighbours and the bridges using shuttling	Route tables	Holding time[3] or shuttling
DDR	H	Yes, it alternate gateway nodes are available	Yes	Stable routing	Intrazone and interzone table	SN, then source initiates a new path discovery

RS = routing structure; H = hierarchical; F = flat; RT = route table; RC = route cache; RET = route expiration time; SP = shortest path; SN = source notification; LBQ = localised broadcast query.

Table 2.3: Characteristics of hybrid protocols [1].

[a]The source may or may not be notified.
[b]A location request will be sent if the zone ID of a node changes.
[c]Packets are held for a short period of time during which the nodes attempts to route the packet directly to the destination.

The application of these protocols was not successfully at all. They obtained a very poor performance in terms of packet delivery ratio and control overhead due to the highly dynamic nature of VANETs. So, these strategy were cast away in favour to a new one called **geographic routing**

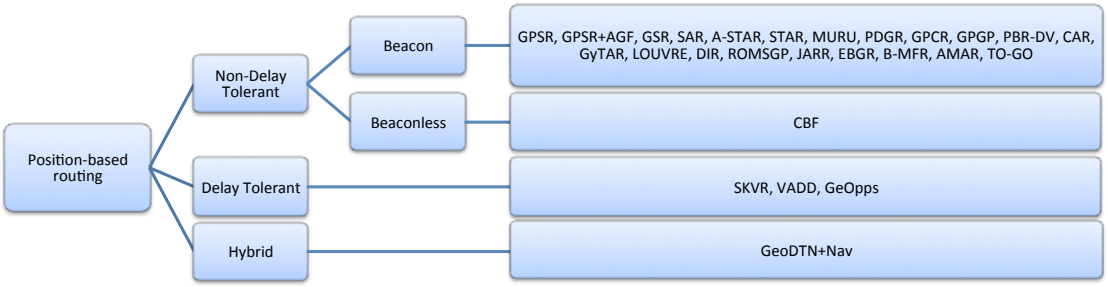


Figure 2.4: Taxonomy of VANET routing protocols.

or **position-based routing** whose routing decisions are based on the geographic coordinates of the nodes. This strategy will be explained in the next section.

Thanks to geographic routing, a new generation of VANET routing protocols were proposed which improved previous ones in packet delivery ratio, end-to-end delay and network overloading. In the literature, authors like Luo and Hubaux [20], Chennikara et al. [21], Li et al. [22], Lin et al. [23], Sharef et al. [24] or one of our contribution to this research area [25] provided surveys relating and comparing them. In Fig. 2.4, we present a diagram of the main position-based protocols according to the most recent one, in 2014 by Sharef et al.

This taxonomy classify the protocols according to their **Delay Tolerant Network (DTN)** support. This new ability arose precisely to deal with lowly dense networks and sparse scenarios. Up to now, routing protocols, transmit the packets to the next forwarder following a pre-established path determined by intermediate nodes which acted as relays for these packets.

Due to the highly dynamic nature of VANETs and their heterogeneous density, nodes often reached a point where there was not exist a neighbouring node candidate to be the next forwarder. To overcome this situation, routing protocols followed different recovery strategies where, for instance, packets went backwards to find a better route to the destination or they were dropped directly.

A different strategy emerged based on the ***store-carry-and-forward paradigm***. Instead of transmitting the packet backwards or dropping it, the node acquired the ability of transporting it by itself like a ferry. This way, when the current node finds a promising neighbouring vehicle it will forward the message to it.

The other feature used in this classification is the use of beacons. These are periodic messages broadcast to the close vicinity (one-hop neighbouring nodes) providing location information of the node, its speed, direction and the likes. Although there are protocols which do not use them, standardization bodies have included these messages as primordial in their standards defining their periodicity, structure and content. This is one of the reason why there are so many protocols using them.

2.4. Geographic or position-based routing

Both proactive and reactive routing protocols aimed at MANETs, require the list of nodes which build the path from the source node to the destination. While proactive protocols, by periodic

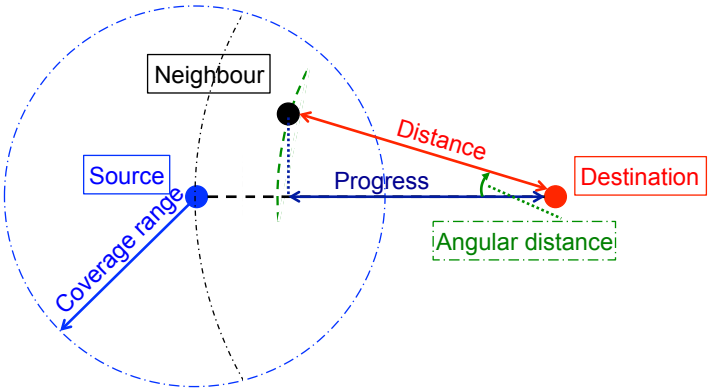


Figure 2.5: Next hop selection criteria.

messages, keep this information inside the nodes, increasing this way the overload of the network, reactive ones use a discovery stage to build a path, storing in each hop the identifier of the next one to reach the destination.

This is precisely the greatest advantage of geographic routing. Since its routing decisions are based on the geographic location, building a path towards the destination node is not required. It only needs to know the current node location, and the destination and neighbours position. Thanks to GPS receivers which are usually integrated in vehicles, it is easy to obtain this information.

2.4.1. MFR, greedy and compass routing

According to Rührup [26], the first approaches for geographic routing were developed in the 1980s for packet radio networks by Takagi et al. and Hou et al. respectively [27, 28] and wired networks by Finn [29]. These approaches describe rules used by the forwarding nodes to select a neighbour as a next step. Since routing decisions are locally optimal, these approaches are termed *greedy forwarding*.

The key aspect of geographic is the next-hop selection criterion which is based on the distance of the neighbours to the destination or the so-called *progress* they can provide to the packets. In Fig. 2.5 the three different criteria found in the literature are exemplified.

In this image there are three nodes: source *(s)*, neighbour *(n)* and destination *(d)*, located at determined positions of the scenario. For the neighbouring node (in black), we calculate the three next-hop selection criteria:

- In blue, we can see the concept of progress defined by Takagi et al. [27] for Most Forward within R(MFR) routing stratey. It is the projection of the location of the neighbour on the source-destination line. That is $\mid n'd \mid = sd * \frac{sd}{\mid st \mid}$.
- The distance to the destination ($\mid nd \mid$) or advance, in red, was defined by Finn [29] and it corresponds to the distance gain towards the destination which is $\mid st \min - \min xt \min$.
- Finally, the angular distance or separation ($\angle nsd$), in green, was proposed by Kranakis et al. [30] for their Compass Routing(CR) protocol.

Let us show the difference between these next hop criteria by using another figure (Fig. 2.6).

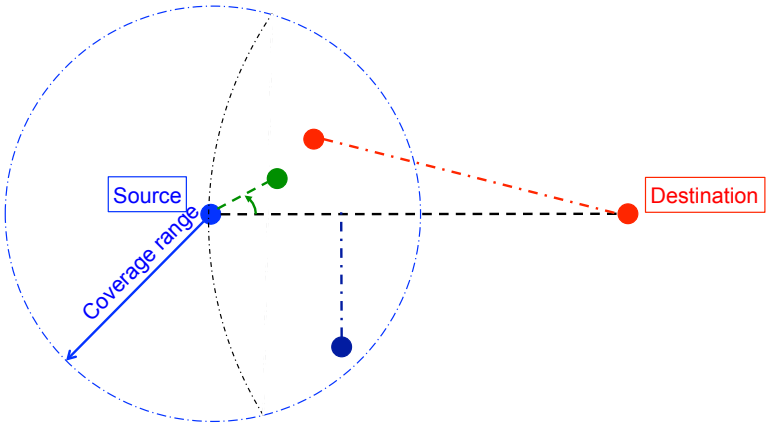


Figure 2.6: Next hop selection criteria.

In this case, neighbours providing positive advance to the destination are represented by different colours: red, green and blue. For the same source node, depending on the aforementioned next hop criteria the selected node is one or another. We have identified them according to the previous figure, so, MFR will choose the blue one; the greedy proposal from Finn will choose the red one; and finally CR protocol will choose the green one because it has the less angle.

In addition to these criteria which try to find a neighbour such as it provides the most gain towards the destination, other criteria were aware of energy requirements. So, instead of trying to reach the most promising neighbour as the one which provides the biggest advance, they select the closest neighbour with positive advance saving the energy at transmitting the packets. They are Nearest with Forwarding Progress(NFP) developed by Hou and Li citehou1986transmission, and Nearest Closer(NC) by Stojmenovic and Xu [31].

Geographic routing has a drawback. Since the routing decisions are made locally by the forwarding node, i.e., taking into account the neighbouring nodes within its coverage range. It often fails in local minimum situations where the forwarding node does not find any neighbours providing advance to the destination, Fig. 2.7. Therefore, to overcome this situation routing protocols define a recovery strategy.

Different alternatives were designed to overcome the local minimum problem. Stojmenovic and Xu [32] propose GEDIR, a method which go backwards under this situation. When a node does not find a neighbour which provides advance to the packet it sent it backwards. Thus, the receiving node excludes this forwarder from its candidates selecting another one expecting not to take the same route. They also enhanced the first geographic routings proposed above by including 2-hop information in their routing decisions.

He et al. [33] integrated in its proposed solution SPEED, a mechanism called Neighbourhood Feedback Loop(NFL) which allows neighbours to provide information about their average send delay by broadcasting *on-demand backpressure beacons*. This information is used to select the next forwarder with a delay lower than a determined threshold avoiding congested and overloaded areas.

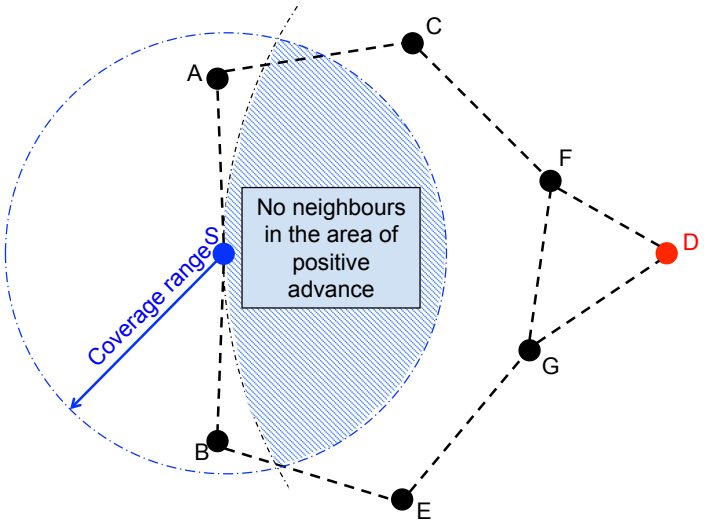


Figure 2.7: Node S fails in local minimum.

2.4.2. Recovery strategies

As we have already seen in different images, ad-hoc networks can be represented by graphs where whose vertices are the nodes geographically located and the edges are the links between them.

Planar graph routing, also known as face routing is a geographic routing strategy that is able to overcome the local minimum problem of greedy forwarding. Local minima exist at the border of void regions where nodes do not find a neighbour closer to the destination than itself. Planar graph routing is based on a packet can be routed along a sequence of faces in the graph to solve the local minimum. To do so, nodes in such a situation apply the left-hand or right-hand rule (Fig. 2.8). This rule is well known in maze problems: one can find the way out of a maze by having his right hand always in touch of the wall while walking. Applying this technique in a graph suppose to find a next forwarding node in (counter-)clockwise order after the current one. As result, the packet traverse the encountered face until it finds a next promising neighbour.

The only requirement for a successful application of this rule, is that the graph must be planar. Unplanarized graphs contain crossing links which cause detours or routing loops (Fig. 2.9)., fortunately the issue of making a graph planar has been studied in the literature by Gabriel et al., Toussaint, Bose et al. and Gao et al. among others [34–37].

Kranakis et al. [30] and Bose et al. [36] proposed the first algorithms that traverse a sequence of adjacent faces, Compass Routing II and Face-2 respectively.

While the first one traverse the face in order to determine the edge that intersects the $s-d-line$ and is closest to the target, in order to know when to change the face. The latter avoids the complete traversals and performs the face change before crossing the $s-d-line$. On each traversal, a node u checks whether the edge to the next node (u, u') intersects the s-d-line as Fig. 2.10 shows.

Bose et al. proposed the Greedy-Face-Greedy algorithm(GFG), a combination of the efficient greedy forwarding and face routing on a planar sub-graph to recover from local minima. A variant of this algorithm is known as Greedy Perimeter Stateless Routing for Wireless Networks (GPSR) [38], one of the firsts and well-known VANET specific routing protocols.

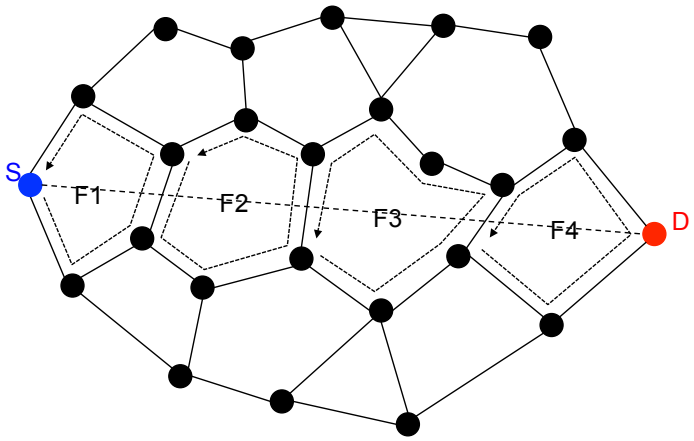


Figure 2.8: Graph traversed rounding the different and adjacent faces encountered by the packet in its way towards the destination.

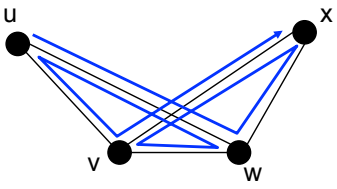


Figure 2.9: Crossing links causing a detour (starting from node u).

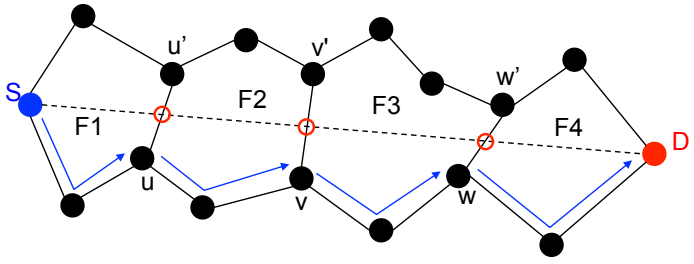


Figure 2.10: Face2 showing where face changing takes place at nodes u, v and w.

Although the planar routing strategy is able to overcome void regions, this technique is not very effective under highly dynamic networks such as VANETs. This nature innate to VANETs was used to provide a different repair strategy.

If the information is **delay tolerant**, i.e. it does not matter when the information must arrive to the destination, VANET nodes can store the packets waiting to find a neighbouring node that provide advance to these packets. This technique is employed by Geographical Opportunistic Routing for Vehicular Networks (GeOpps) or Vehicle-Assisted Data Delivery (VADD) providing another advantage, since the information is not transmitted, more transmissions can be performed in such networks [39].

2.5. Security in VANETs

One of the main concerns of public traffic authorities is the numerous traffic accidents that happen on our roads and the consequent victims. This is precisely the main motivation for the study of VANETs, making a good use of wireless communications to prevent these dramatic incidents making the act driving safer.

Since the first objective of VANETs which is saving lives is so important. All the protocols, services and applications developed must also be secure to assure a good behaviour of the network preventing malicious users to impair them. This is the reason why security is a traversal research field that covers every single layer of the network architecture.

Parno and Perrig [40], in 2005, envisioned the future technology used for vehicle manufacturing employing wireless communications with a radio range of at least one kilometre. Among the different aspects of these vehicular networks, they identified the main challenges these vehicular network would have to cope with like the bootstrap with only a few vehicles equipped with this technology, its high-speed mobility, or security related aspects like authentication, message integrity or privacy.

In order to come early to the security issue, they outlined a classification of the different adversaries these networks would have to face, as well as different kind of attacks that could affect their behaviour. This study was resumed later on by Raya et al. [3, 41] and Lin et al. [42].

Malicious users or attackers have a different motivation to the rest of the common users. It can go from obtaining a certain benefit like private or secret information from the nodes of the network to impair or harm these nodes due to economical interests or only for fun, for instance.

Although a complete relation of the different kind of attackers and attacks are thoroughly explained in Chapter 4, next we depict some of the most common ones and their effects on the network.

One of the most common is sink-hole or black-hole. In this network, every message that arrives to a malicious node it is dropped without forwarding it. A derived attack from this one is called selective forwarding where the node only forwards certain messages, dropping others.

In Fig. 2.11 *Vehicle A* broadcasts a message to warn the following vehicles about a traffic accident. Since the malicious user in *Vehicle B* does not forward the message, the other vehicles are not warned which could incur in a more dangerous accident with more involved vehicles.

Another attack consists in manipulating the information of packets to be forwarded with the consequent damage. This time, in Fig. 2.12, *Vehicle B* is a malicious node which manipulates the information about the area where the accident occurred causing the same effect.

Razzaque et al. [43] proposed a series of security requirements that vehicular communications must satisfy so as to cope with the different threats and attackers that could be present in these networks:

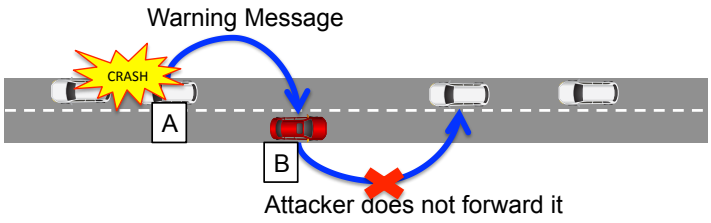


Figure 2.11: Sink-hole attack.

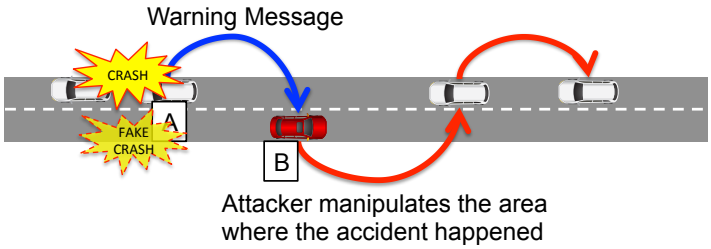


Figure 2.12: Message manipulation attack.

- Authentication: This is the most important requirement to prevent malicious nodes to transmit fake information. Thus, nodes will only process packets whose forwarder is known.
- Verification of data consistency: Message exchanges are frequently in vehicular communications. So, for instance, if after a request immediately goes a reply. This consistency must also be guaranteed avoiding attackers that despite being authenticated, they send fake data.
- Message integrity: Attackers can also alter the content the packets they forward causing troubles to the next hops. For this reason we need techniques to maintain the integrity of the messages.
- Availability: There are attacks like Denial of Service(DoS) by jamming which are able to bring down the VANET. Protection against this sort of attack must be provided by using some other techniques.
- Non-repudiation: Attackers can cause great damage to people due to their attacks. In case this happens, nodes cannot refuse the transmission of a message.
- Privacy: People are worried more and more about their privacy. Unauthorised nodes must not be able to guess the identity of other nodes.
- Traceability and revocation: By contrast, it is also necessary to allow authorities to track nodes and even to disable the equipment of malicious or abusing users.

Despite these requirements, there are many security threats that can affect the routing layer of VANETs. Some of them are inherent to the nature of the wireless medium while others affect in a different way to the routing protocols depending on their implementation. For this reason, it is important to focus on the security aspects at designing a routing protocol.

2.5.1. Authentication protocols for access control

In this thesis, our contribution regarding the security in VANETs is twofold: on the one hand we are evaluating security in the routing layer, and on the other hand, we address the problem of access control when mobile nodes (vehicles) attach to RSUs or APs directly or indirectly by using intermediate nodes to gain access to the services they have subscribed in the infrastructure. In this section we provide some background related to the latter one.

Nowadays the amount of services available in the network is immense. They are usually accessed thanks to different technologies provided by telecommunication operators which also need to manage and control their users. This is guaranteed thanks to an authenticated access to the network service assisted by the well known Authentication, Authorization and Accounting (AAA) infrastructures.

However, due to the diverse deployed technologies and that the access control is a cumbersome process the operators face the issue of providing an authentication mechanism able to deal with the different technologies employed to access the aforementioned services. A flexible way to make this authentication process is by the use of Extensible Authentication Protocol (EAP) [44], which allows the use of the different authentication technologies through the so-called Extensible Authentication Protocol (EAP) methods. They are run between the EAP peer, usually the mobile node, and an EAP server located within the Authentication, Authorization and Accounting (AAA) infrastructure, through an EAP authenticator, a RSU or AP acting as a gateway between both wireless and wired network which simply forward the EAP between them.

In the literature, we can find two widely used protocols able to transport EAP over multi-hop scenarios. PANA [45] and Internet Key Exchange Protocol(IKE). The latter had several vulnerabilities and problems which were solved in its second version IKEv2 [46].

Internet Key Exchange Protocol (IKE)

IKE is a protocol aimed at making a secure key exchange between two entities called *initiator* and *responder*, which desire to establish a secure communication through Internet Protocol Security (IPSec) [47]. IPSec provides integrity and authenticity to the exchanged messages by including its Authentication Header(AH) header, and it provides also confidentiality by its Encapsulating Security Payload(ESP) header. To do so, IPSec has two different operation modes: the *transport mode* in which these headers are applied directly to the Internet Protocol(IP) packet to be protected, and the *tunnel mode* in which the original packet is firstly encapsulated within another IP packet, being the outer one the place where the aforementioned IPSec headers are applied.

In IKEv2 each request message has an associated answer. Each pair of messages is known as a message exchange. In Fig. 2.13 we present the IKEv2 message flow. The first exchange entitled IKE_SA_INIT is used to establish a security association[1] at IKE level which allows the secure transmission of the following messages. This exchange is used to negotiate the cryptographic algorithms which will be used next and to generate a session key called *SKEYSEED*, result of the Diffie-Hellman exchange [48] which involves the exchange pseud-random numbers between the two parties. Since the used of one key during a long period of time makes it vulnerable, it is a common practice to derive new keys. This mechanism is used by IKEv2 when the IKE_SA use the SKEYSEED to derive more keys as it need them.

[1] The establishment of shared security attributes between two network entities to support secure communication.

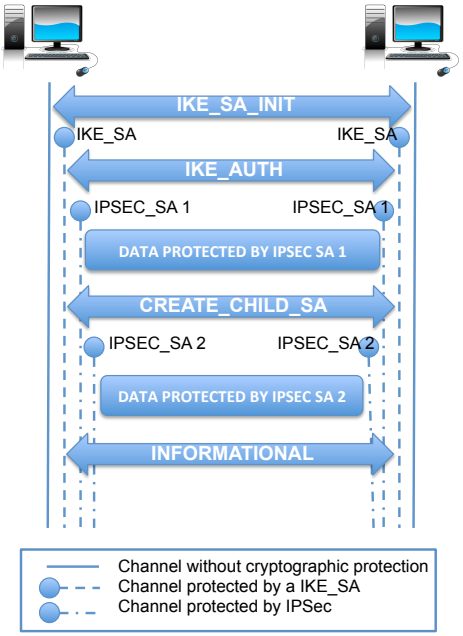


Figure 2.13: IKEv2 exchange sequence.

After this first exchange, another one called IKE_AUTH takes place. This one is already protected by the previous security association and it is employed to negotiate the IPSec operation mode (transport or tunnel mode) to be used by the following messages. Both parties also agree about the kind of authentication that must be used: pre-established keys IKE-PSK, private-public key, certificates or even the use of the EAP protocol. It is worth noting that the use of the latter one is used with some restrictions in this protocol. Although EAP is a mutual authentication protocol, EAP over IKEv2 only authenticates the initiator.

Once these exchanges have been done as well as the first IPSec association, through a CREATE_CHILD_SA exchange, new security associations (CHILD_SA) can be derived from this. This exchange can be started by both *initiator* and *responder* but only after both IKE_SA_INIT and IKE_AUTH have taken place.

Finally, the INFORMATIONAL exchange can be used for event notification, configuration as well as assisting to the security association removal. This exchange can be used only after initial exchanges and it is secured by the IKE_SA security association.

Protocol for Carrying Authentication for Network Access (PANA)

The main idea behind PANA is that of defining a protocol independent on the link layer for the transportation of authentication methods in the network access service. We previously commented that EAP was a protocol designed to deal with different authentication methods. So, this is the protocol carried by PANA to achieve the authentication process.

The model adopted by PANA for the network access comprises the following entities:

PANA client(PaC): It is the client side of the protocol, usually the mobile node. It is responsible for providing the credentials so as to prove its identity for the network access authorization. It

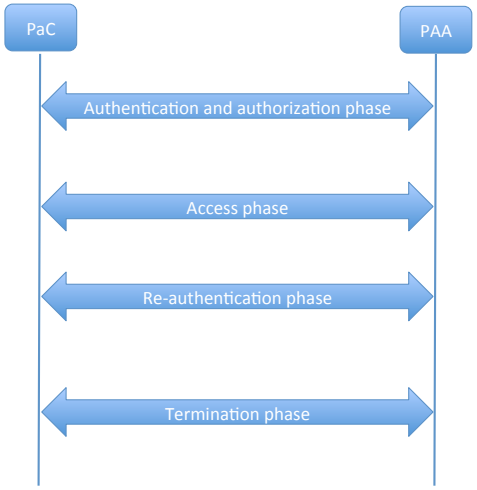


Figure 2.14: Phases of PANA.

also matches the EAP peer of the EAP model.

PANA server(PaA): It is responsible for verifying the PaC's credentials. Regarding the EAP model, again both PaA and PaA correspond to the same device. The EAP server can also be colocated in the same device.

Authentication Server: In case the EAP server is not colocated in the same device as the PaA. For this situation, the PaA must consult a backend authentication server implementing the EAP server functionality.

Regarding the message flow between PaC and PaA. The protocol messaging consists of a series of requests and answers. Each message can carries a series of attributes using the Attibutes Value Pair(AVP) form. In Fig. 2.14 we can see the phases this authentication protocol.

Once the PaC is successfully authenticated by the *authentication and authorization phase*, it enters the next phase called *access phase* where it can send and receive IP traffic by using the infrastructure. During this phase, both PaC and PaA can check the liveness of the secure session by sending notification messages.

In case they want to update the lifetime of the secure session, they can perform a re-authentication. To do so, the *re-authentication phase* is defined.

Finally, when PaC or PaA choose to discontinue the access service. They can send an explicit disconnect message starting the *termination phase*.

Although both protocols IKEv2 and PANA uses different kind of messages, they share the same goal which is to establish a secure channel allowing the user to gain access to the desired service, preventing malicious users to overhear the exchanged security attributes.

2.6. Conclusions

This chapter has reviewed the most important concepts that must be taken into account before reading the remainder of this thesis. We started describing Vehicular Ad-hoc Networks (VANETs),

explaining their features, particularities and applications. We have also presented its historical development from the point of view of multi-hop communications.

We took advantage of such a situation to introduce multi-hop routing protocols starting with the first proposed solutions taken from the generic MANETs. Their flaws and drawbacks motivated the use of a different routing strategy called geographic routing which is a key aspect to succeed in the definition of routing protocols for such highly dynamic networks.

We ended this chapter showing the expectation these networks have caused worldwide enumerating different projects and standards provided by the main standardization bodies.

The first problem tackled in this thesis is related to the routing strategies for VANETs. As we have seen in this chapter, although the use of geographic routing suppose an important advance in the development of routing protocols there are still many obstacles to overcome in order to develop a suitable routing protocol for these networks. In Chapter 3 we address this issue.

Chapter 3

Multi-hop routing in vehicular ad-hoc networks

Ad-hoc networks allow nodes to send messages to other nodes even if they are farther than their radio range. This is possible thanks to intermediate nodes located in the way to the destination which forward these messages to the next hop. To do so, multi-hop routing protocols are needed. They try to find the appropriate neighbours more suitable to deliver the messages to the destination.

This topic is widely studied for a great variety of ad-hoc networks environments. Since VANETs are a particular case of ad-hoc networks it is not casual that we find a lot of different proposed solutions to successfully achieve this objective.

VANETs have certain particularities that make them different to MANETs. Concretely, the high speed of nodes, their motion restriction as well as the effect of traffic signs increases the variability of the links among nodes. This makes generic MANET routing protocol not quite suitable for such a task. Actually, geographic routing, a technique based on nodes location to make routing decisions outperforms traditional routing protocols for MANETs.

In this chapter, we analyse the different VANET routing protocols found in the literature and classify them depending on the elements used in their routing decisions. Our in depth analysis allowed us to figure out their main drawbacks and design flaws opening up a new way to develop our proposed routing protocol Beacon-less Routing Algorithm for Vehicular Environments (BRAVE).

One of the key aspects of BRAVE is its opportunistic scheme to select the next forwarding node. Instead of being the current one the one which makes the decision of selecting the next hop, neighbours are the ones that make this decision selecting themselves as the best forwarders after overhearing the data packet. Another important features of BRAVE is its recovery strategy which employs a store-carry-and-forward paradigm in case a node reaches a local optimum.

We have assessed the performance of BRAVE by a series of simulations comparing it against several of the most well-known alternatives found in the literature. These experiments show that BRAVE outperforms existing solutions in terms of PDR obtaining also a good trade-off between delivery ratio and end-to-end delay thanks to its store-carry-and-forward recovery strategy.

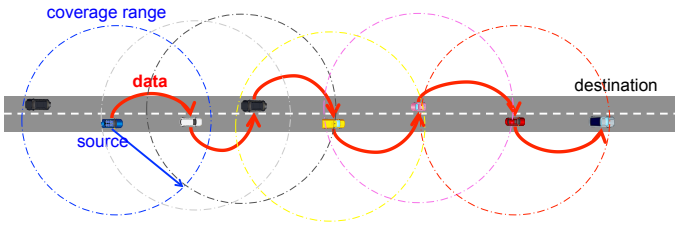


Figure 3.1: Multi-hop routing. Sending data using vehicles as forwarders.

3.1. Introduction and motivation

In Chapter 1 we already introduced and motivated the use of VANETs. The necessity of providing communication among the nodes of the network, even when they are farther than their radio range also allowed us to introduce multi-hop routing protocols, Fig. 3.1. Actually, in Chapter 2 we presented an historical view of the evolution of such specific VANETs routing protocols explaining also *geographic routing*, the appropriate routing strategy that these routing protocols have adopted so as to deal with such highly dynamic environments that VANETs are.

In the last few years, we have seen a vast number of VANET routing solutions come up as we can see in one of our contribution [49] and a survey written by Li and Wang [50]. Most of these protocols, despite using the geographic routing strategy, are still not able to make the most of VANETs. This is because of, although in ideal conditions their behaviour is correct, some assumptions like selecting the farthest node within the radio range that provides the most advance to the destination incurs in many packet losses.

A complete analysis of the performance of these routing solutions was made by Cabrera et al. [51]. They state that existing solutions experience a series of problems that negatively affect their performance:

- Selection of next hops based on their progress may end up in failed transmissions and retransmissions.
- The use of beacon messages induce forwarding inefficiencies including cycles caused by stale information.
- Some of these protocols fail to deliver messages because they are not able to handle disconnected topologies.
- Finally, for trajectory-based routing schemes, objective functions may make data packets to fall into a local optimum until the message is dropped.

Regarding the repair or recovery strategies used when nodes fall into local optima, *perimeter routing*, which was introduced in Chapter 2, is not suitable for VANETs because of their highly dynamic nature. For this reason, VANET routing protocols propose different repair or recovery strategies to overcome them.

One of the most commonly used repair strategy is that of using the *store-carry-and-forward* paradigm. It takes advantage of the high speed of nodes to make them to transport the messages until they find an appropriate neighbouring node. This strategy can be used in networks whose packets are not delay-sensitive, also called Delay Tolerant Networks (DTNs).

Considering the above problems of existing solutions we present a novel beacon-less routing scheme for VANETs called BRAVE. This protocol uses an opportunistic forwarding scheme to avoid the issues commented above. That is, the next forwarder for the data message is reactively selected among those neighbours that have successfully received the message. In addition, the protocol is able to operate in a store-carry-and-forward paradigm to deal with uneven network densities and disconnected topologies. Moreover, the proposed solution is fully localized (only needs information provided by neighbours) which guarantees ultimate scalability with respect to the number of vehicles in the network.

The remainder of this chapter is organized as follows: In Section 3.2 we describe the main routing protocols found in the literature. Next, in Section 3.3, we consider one of the most common assumptions of greedy routing protocols, the transmission range assumption, which makes them not to obtain the performance simulated under ideal conditions. Different routing design alternatives are dealt in Section 3.4. Our proposal is explained in Section 3.5 which is evaluated in Section 3.6. Finally in Section 3.7 we will comment the benefits obtained in this article as well as our next research steps related to this work.

3.2. Related work

Before giving the details of our proposed solution, we are going review the main routing protocols for VANETs developed up to date. Since VANETs have gained a lot of scientific interest in the last few years, a lot of contributions have been developed for this specific topic.

First contributions came by the intention of applying some generic MANET routing protocols in VANETs, for example AODV [52] or DSR [53]. However, those protocols exhibited a low performance. Therefore, a next step was to optimize some of these protocols for VANETs like PRAODV and PRAODV-M [54], obtaining a slightly improvement of the packet delivery ratio compared to AODV and DSR. The problem lied in the operating mode of these protocols. They have to build and maintain a path from the source node to the destination, and due to the highly dynamic nature of VANETs it is a costly task and hardly unfeasible.

Later contributions introduced the use of the geographic routing strategy in VANET scenarios which proved to be more scalable and effective than traditional protocols. During the last few years a lot of protocols have been developed following this strategy as described in some surveys like our contribution [25], Li and Wang [50] or and more recently by Sharef et al. [24]. This latter one provides a complete review of the different routing protocols covering also broadcast and multicast protocols which we consider out of the scope of this chapter.

The taxonomy that we present in the following is done according to the information that routing protocols consider to make their routing decisions. In particular, we classified routing protocols in four groups: basic solutions, map-based protocols, protocols based on trajectories and traffic information-based protocols.

3.2.1. Basic solutions

This category comprises those routing protocols that work only with control messages among neighbours. These messages usually contain the vehicle identification, its position and its velocity vector among others.

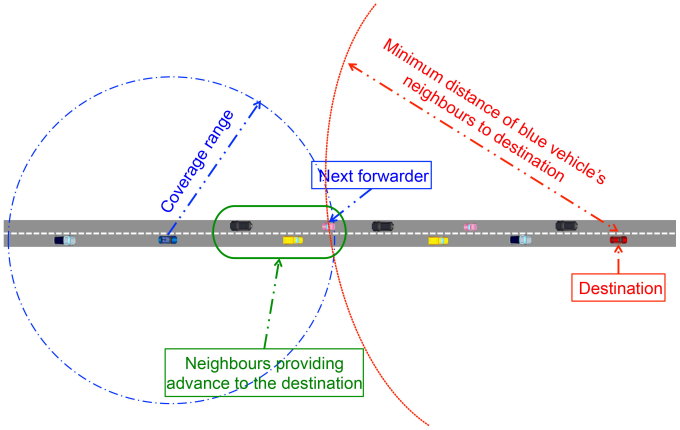


Figure 3.2: GPSR - Greedy mode.

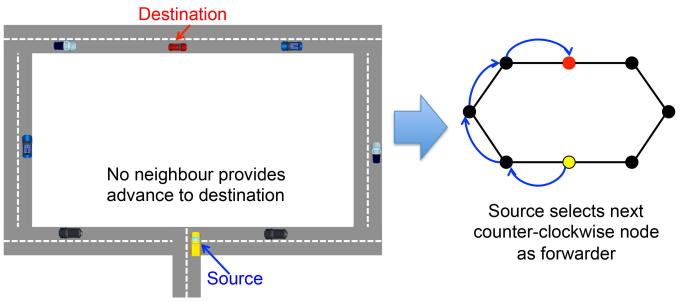


Figure 3.3: GPSR - Perimeter mode.

Karp and Kung [38] proposed **GPSR**, a position based routing protocol aimed at mobile networks consisting in two operation modes:

- **Greedy mode** where the current node selects the next forwarder among its neighbours considering their distance to the destination. The one whose distance to the destination is the closest one is the most promising forwarder and therefore the selected one, see Fig. 3.3.

- **Perimeter mode** is a recovery strategy used when a node reaches a point where it is unable to find a new neighbour closer to the destination to forward the message towards the destination. That is, there is no neighbour whose distance to the destination is less than the current one. At this point GPSR declares the current node the local maximum to the destination and it applies the Right Hand Rule (RHR) to overcome this situation. This rule consists in, firstly, composing a planarized graph using the neighbouring nodes as the vertices and the links between nodes as the edges of the graph and secondly, in selecting among these vertices the next forwarder. The RHR states the next edge to be traversed is the one that is sequentially counter-clockwise with respect to current node x from edge (x, y) as shown in Fig. 3.3.

Both modes require the current node to know the position of its neighbours. This is done by a simple beaconing process where nodes periodically announces their position within these periodic messages called beacons.

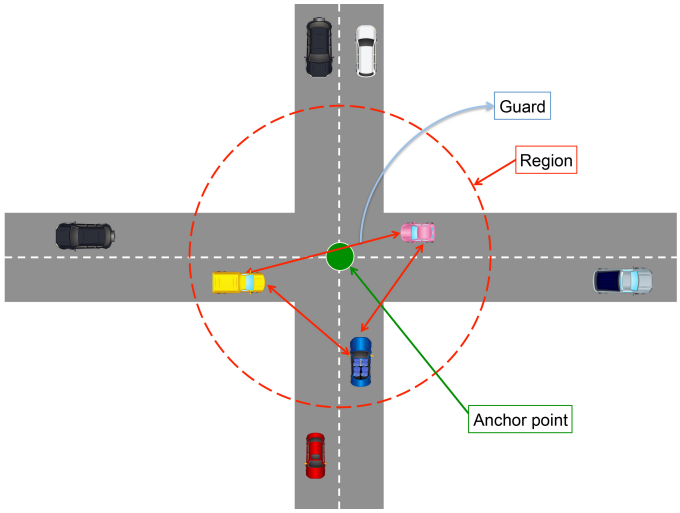


Figure 3.4: Vehicles exchange a guard about an anchor point.

Schnaufer et al. [55], inside the mark of the NoW project, designed and developed a protocol called **Position-Based Routing with Distance-Vector recovery (PBR-DV)** aimed at urban scenarios. It combines the position-based greedy routing with AODV style recovery. In the recovery stage, the node stuck in a local optimum broadcasts a request message. The receiving nodes not located closer to the destination than the node which started the recovery, re-broadcast the packet and stores the identifier of the sender in their routing table. Otherwise, i.e., when the message arrives to a node providing distance progress, it sends a route reply packet with its own position to the node which started the recovery stage sending the request message.

Another basic routing protocol is **Connectivity-Aware Routing (CAR)** developed by Naumov and Gross [56]. It is a position based routing protocol which comprises four stages: destination location and path discovery; data packet forwarding along the found path; path maintenance; and error recovery. The first stage is achieved by broadcasting a route request message along the network to find the destination. Once it is reached, it answers composing a reverse path to the source.

Since CAR is not aware of the roads of the urban scenario it uses an interesting technique to detect the junctions or intersections of streets. Nodes whose neighbours have non-parallel velocity vectors are identify themselves as being located near a junction (*anchor nodes*). Actually, they are the ones which introduce their position in the header of reply messages. So, when the source receives the reply it will geographically route the information to the destination following the anchors of the received answer.

The high dynamicity of VANETs, makes nodes in junctions not to remain in that position for a long time. Nevertheless, CAR needs to have the position of the junctions alive somehow to make messages to turn on these junctions. So, Naumov and Gross conceived the concept of **guards**, see Fig. 3.4. It basically represents temporary state information tied to a specific area instead of corresponding to a vehicle. Thus, nodes located within these areas are responsible for keeping this information alive. To do so, they include their position as well as a radius to compose a circular area around the anchor nodes' position, and unlike normal beacons, these ones are forwarded by nodes located inside the area indicated in the beacons sent by anchor nodes.

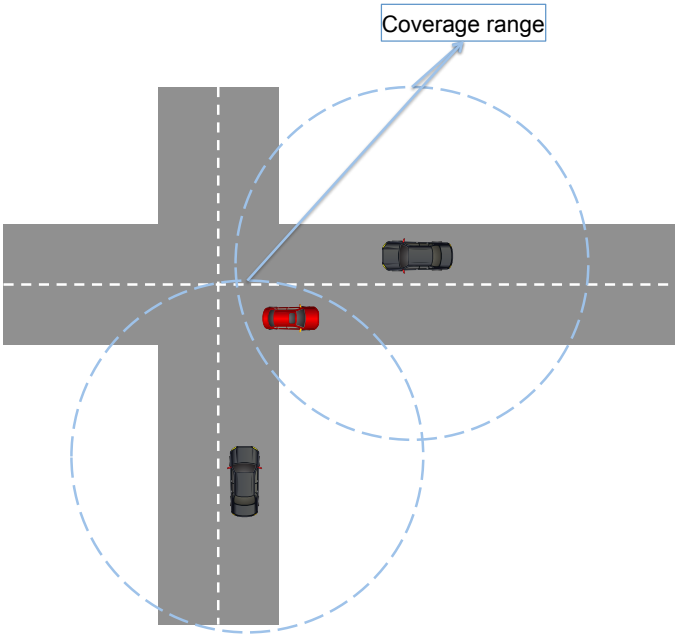


Figure 3.5: Coordinator discovery process.

Despite all the efforts employed in this protocol there are some flaws that jeopardizes its performance. Its first stage, i.e. the destination discovery, is very expensive in terms of number of transmissions. In addition, messages must include a whole list of anchor nodes to make the source to reach the destination. Finally, regarding guards, it is not possible to guarantee the presence of vehicles in junctions acting as anchor nodes and preserving the junction information.

Another protocol that follows this trend is **Greedy Perimeter Coordinator Routing (gpcr)**, proposed by Locher et al. [57]. Like in CAR, routing decisions are made by nodes in junctions called **coordinators** because its the only place where packets can change their direction taking other ways. Nevertheless, junctions are detected differently. gpcr defines two mechanisms for detecting junctions.

The first one requires extra information inside beacons including not only information about the sending node, but also the identification of its current neighbours. Therefore, if a node receives two beacons from different neighbours that are in the same radio range, and these neighbours do not include each other, this means that there is an obstacle between them, a building for instance, and the receiving node is able to surround it, see fig 3.5. This usually happens in an intersection but it can also happen in a curve.

The second one does not need any extra information into the beacons because it calculates a correlation coefficient with respect to the position of neighbours.

On the other hand, nodes located within a segment of the street, send the data according to a restricted greedy approach. Since coordinators are the ones with decision capability, they announce themselves indicating such a role by activating a bit in their beacons. So, when a node is going to forward a packet, it checks if one of its neighbours is a coordinator. In this case, it forwards the packet to such coordinator. Otherwise, it selects the farthest neighbouring node following a line defined by the position of the previous forwarder and its own location as figure 3.6 shows.

Another remarkable contribution of gpcr is the repair strategy applied when a node gets stuck in

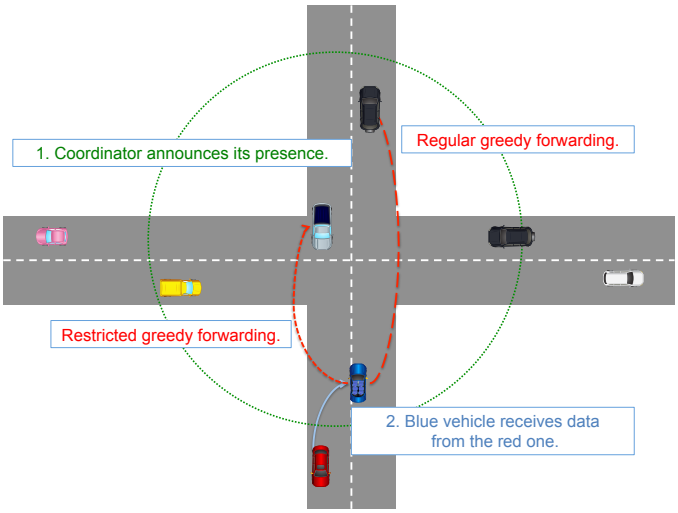


Figure 3.6: Example of restricted greedy forwarding.

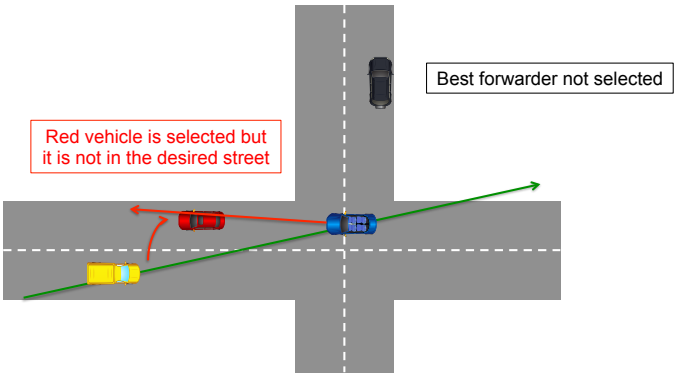


Figure 3.7: Recovery strategy flaw selecting a forwarder.

a local optimum. In such case, the node at the local optimum sends the message back in the opposite direction until it reaches a coordinator. When the message arrives to a coordinator, it selects the next forwarder among the neighbours located in the street that is the next one counter-clock wise from the street packet has arrived on as the RHR indicates.

More recently, Lee et al. [58] presented a protocol called **TOpology-assisted Geo-Opportunistic Routing (TO-GO)**. This employs a two-hop beacon strategy which is extracted from Gpsr Junction+ (GpsrJ+), a routing solution which is explained in the next category. It also uses the Right Hand Rule (RHR) in the perimeter mode when a node gets stuck in a local optimum. However, the most relevant novelty is the use of opportunistic forwarding to make the geographic routing more reliable to the different phenomena of the signal propagation. This idea was already conceived by Füßler et al. [59] in their Contention-based forwarding (CBF) in 2003. They also introduce the concept of Forwarding Set Selection (FSS). Likewise, the nodes inside this set will set a timer based on the relative distance to the target. The closer the distance, the sooner the timer goes off and the sooner the packet gets forwarded.

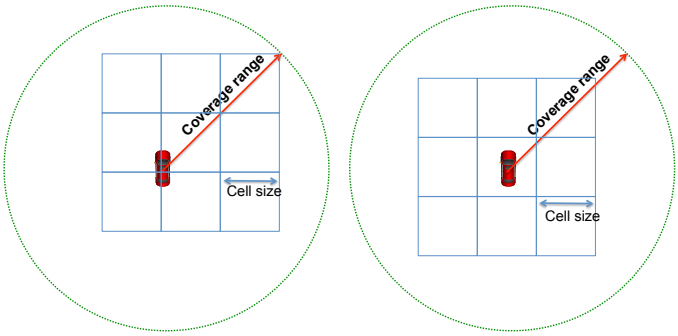


Figure 3.8: Calculation of cell size of the grid.

3.2.2. Map-based solutions

Several years ago, only premium class vehicles had installed a navigation system to guide the driver towards its destination through the different roads in its way. Recently, this features is becoming a commodity that can be installed in nearly every vehicle. For this reason, introducing a street map as another element to make routing decisions turned out to be very a useful mechanism providing new routing solutions as we describe next.

The first protocol analysed in this category is called **MUlti-hop Routing protocol for Urban vehicular ad hoc networks (MURU)**, developed by Mo et al. [60]. It inherits the discovery process from AODV which is enhanced by using the street map to restrict the trajectory of the route request messages. They also analysed the robustness of paths between source and destination, and they found out that is a concave function whose maximum value is obtained when the hop distance is not too small or too large. For this reason, they introduced a new metric called Expected Disconnection Degree (EDD) of the links among nodes which predicts the link breakage probability of each hop. MURU always finds a path from a source node to its destination with the smallest EDD.

GVGrid, by Sun et al. [61], takes advantage of the street map to build a grid dividing the map into cells. The size of these cells is calculated by the own nodes taking into account the coverage range and also guaranteeing the communication between neighbouring nodes. For this reason its authors set the size of the grid side to $sqrt(2) * radius/4$ as shown in Fig. 3.8. These cells are used in the following way: In the process of discovering a route towards the destination of a packet, only one node per cell will be included in the route request message.

Unlike other proposed solutions, the selection of the next forwarder is not based on beacons. Sun et al. claim that these beacons cause a lot of collisions and hinder the overall communication over dense networks. However, standardization organizations have gone in the opposite direction adopting the use of these beacons as a key element of vehicular communications. GVGrid also uses a route request-reply scheme to discover the position of the destination as well as a route maintenance process. These features make GVGrid to be aimed at crowded urban scenarios. Actually it is not able to deal with sparse regions, network partitions or highways scenarios.

Lee et al. [62] proposed an improvement to gpcr called **GpsrJ+**. The differences between gpcr and this proposal are the following: (i) decisions about which road segment to turn does not need to be made by junction nodes; (ii) it uses a two-hop beacon strategy to improve the routing decisions; and (iii) GpsrJ+ does not need an expensive planarization strategy since it uses the natural planar

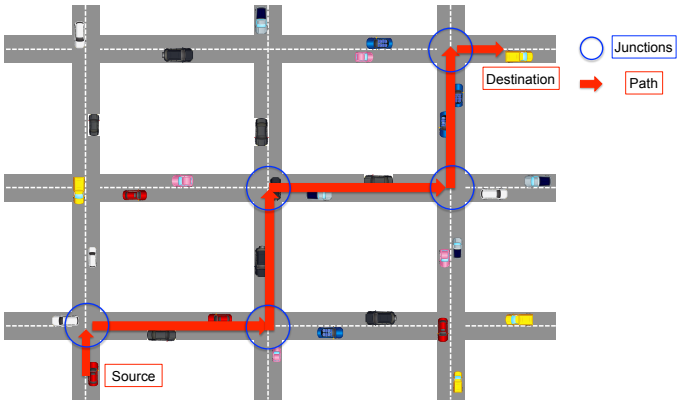


Figure 3.9: Example of junctions and paths obtained by Dijkstra's algorithm.

feature of urban maps. Thus, they were able to reduce the hop count used in the perimeter mode by as much as 200% compared to GPSR, allowing geographic routing schemes to return to the greedy mode faster.

Geographic Source Routing (GSR), by Lochert et al. [63], also assumes the aid of a street map in urban environments, but in a totally different way. A sending node uses Dijkstra's shortest path algorithm to obtain the route to the destination provided by the list of junctions. This is the path that the message follows to reach the destination, so the information related to these junctions must travel with the packet somehow along its journey to know in which junction the message must turn to a different street. For this purpose, the authors propose two different choices: the first one includes all the junctions in the header of the packet; the second option is to recalculate the list of junctions at each junction with the consequent additional computational cost at each node (Fig. 3.9).

Despite the use of this new feature, GSR fails to deliver packets under low dense scenarios where it is usual to have road segments where density of cars is so low that nodes cannot progress the packet towards its next junction. For these cases, when a local optimum is reached, GSR applies the *Right Hand Rule* expecting to find a suitable neighbouring node to bring the message. Otherwise it drops the packet.

Tian et al. [64] improve GSR giving birth to a new protocol called **Spatially Aware Routing (SAR)**. Like GSR, it uses a street map to calculate the route to the destination. However, it introduces several strategies to deal with local optima, i.e., when the node cannot find any neighbours closer to the next junction. Three different strategies were proposed: storing the packet in a buffer and trying to forward it periodically; applying standard greedy routing (route the packet towards the destination instead of the next junction); and finally, recompute the list of junctions. The first strategy allows the vehicle to transport the message itself instead of transmitting it to a neighbour. This way, over sparse networks, the packet will receive a new chance to be delivered to the destination. Their results show that this strategy increases the packet delivery ratio in up to a 20% obviously increasing the delay to reach the destination too.

The information about urban maps is used by Lee et al. [65] in their routing protocol called **GeoCross** to improve the graph planarization used by the perimeter routing mode when a node gets stuck in a local optimum. They propose employing the graph comprised by streets and junctions, instead of using the location of the nodes and their links among them. This technique was already

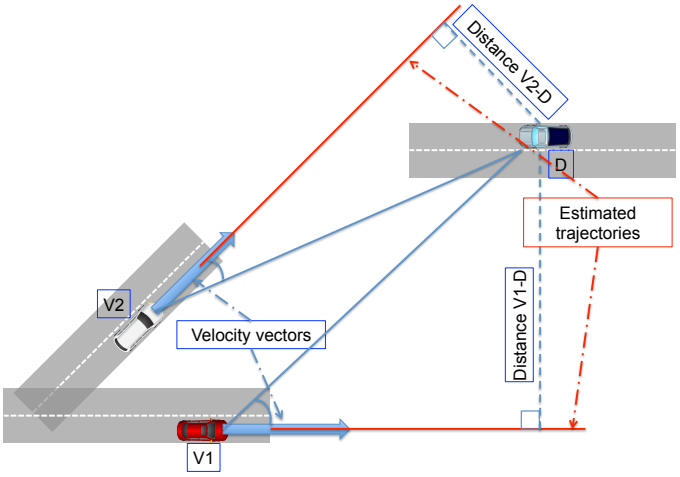


Figure 3.10: Trajectories and distances calculated by MoVe.

introduced by Lochert et al. in gpcr [57]. This information is used in another way by **Landmark Overlays for Urban Vehicular Routing Environments (LOUVRE)** (Lee et al. [66]). It uses the junctions of the streets to build an overlay network on top of an urban topology. Thus, junctions will participate in the overlay if and only if the traffic density of the underlying network guarantees the multi-hop vehicular routing between the two overlay nodes.

3.2.3. Based on trajectories

At the beginning of this taxonomy we reviewed basic solutions which take advantage of sensors such as odometers and speedometers to obtain the current velocity vector of a vehicle. This information is used to detect junctions on the streets allowing routing protocols to turn on these intersections. The following protocols go a step further using, not only velocity vectors, but also calculating the trajectories of the nodes so as to select the neighbouring nodes which will be closer to the destination in the near future.

MoVe, proposed by Lebrun et al. [67], is one of the aforementioned protocols which takes advantage of vehicles trajectories. It uses a *HELLO-RESPONSE* exchange for detecting neighbouring nodes. When a sending node periodically issues a HELLO message. A neighbouring node receiving this message answers with a RESPONSE message which includes its mobility information. If the neighbour is predicted to be closer to the destination than the current node, see Fig. 3.10, it forwards the message to this neighbour. This process is repeated until the message reaches the destination.

Although the use of these trajectories seems to be a good approach to reach the destination, the mobility of vehicles is restricted by the streets which can modify their direction getting away from the destination. This flaw is solved by the protocol we describe next.

Leontiadis and Mascolo propose**GeOpps** [68], a protocol that does not only employ the knowledge about the routes of the vehicles but also the information of the positions provided by navigation systems to calculate the Nearest Point (NP) of one node to the destination of the message. This calculation is more complex than in protocols based on trajectories because it must check for the whole planned route the nearest point to the destination.

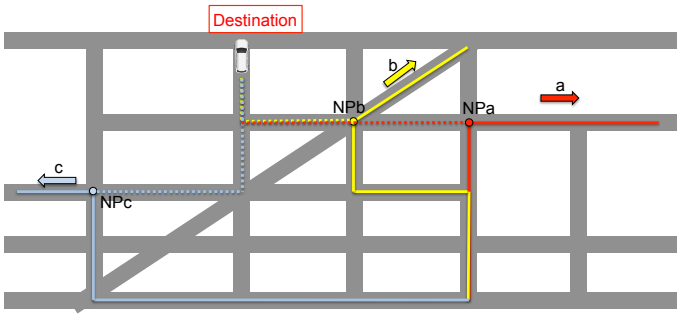


Figure 3.11: Calculation of NPs.

In Fig. 3.11 we show an example of calculation of the NP to the destination for two different nodes.

The operation of the protocol is the following: Vehicles include the destinations of the packets they have stored in its beacons which is periodically broadcast to one-hop neighbouring nodes. When they receive them, they estimate the Minimum Estimated Time Of Delivery (METD) to the destination sending it to the enquiring node. This estimation is calculated by the following formula:

$$METD = ETA\,to\,NP + ETA\,from\,NP\ to\ Destination \tag{3.1}$$

where ETA means the Estimated Time of Arrival of the vehicle to the destination. After receiving the estimation, it will decide to forward the packet to the neighbour if the neighbour METD is lower than for the current node, keeping it otherwise.

One pillar that supports GeOpps is the use of navigation systems to access the whole planned routes towards a specific destination. However, drivers often take the route planned by navigation systems as a reference to reach the destination varying this route as they get close to the destination. In addition, sometimes roads in cities change their driving direction making these navigation systems to use outdated information making them to recalculate the route. All in all, vehicles real routes may differ from the planned ones which will hinder the performance of GeOpps in terms of its delay and also its delivery ratio.

The last routing protocol mentioned in this section is also the most recent one. In 2010, Cheng et al. [69] presented **GeoDTN+Nav**, a VANET routing protocol which integrates both the efficient position-based routing for connected partitions and delay tolerant forwarding for routing between partitions. They also introduce the Virtual Navigation Interface (VNI) which provides generalized navigation information even when vehicles are not equipped with navigation systems. VNI is independent from GeoDTN+Nav and can be used by other routing protocols serving different purposes.

3.2.4. Traffic information

The routing protocols reviewed so far were incrementally including new information in its routing decisions like neighbours positions, velocity vectors, trajectories and street maps. The last group we are describing in this classification takes into account traffic information which is pretty useful because now routing decisions are made based on traffic density, a very visual property to be aware of because the denser the street the more likely the packet can make it hop-by-hop to the end of the street. Protocols like **GyTAR** [70], **A-STAR** [71], **VADD** [72], **MDDV** [73] or **SADV** [74], exploit this

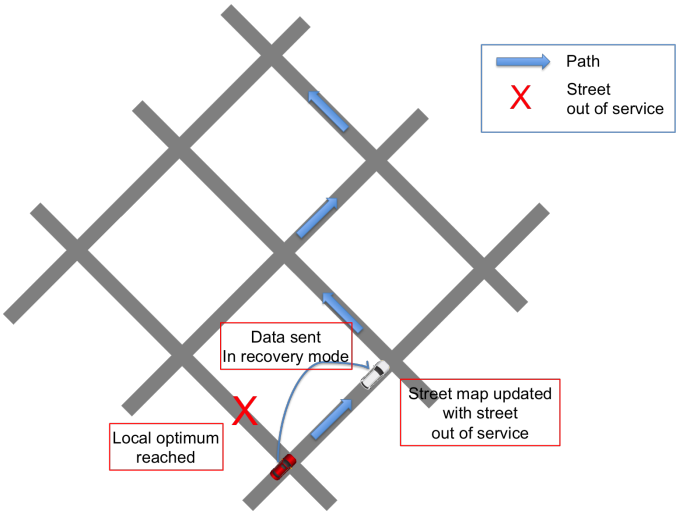


Figure 3.12: A-star recovery strategy.

feature as we will see in the following paragraphs.

Seet et al. [75] propose an anchor-based routing protocol with street awareness called **Anchor-based Street and Traffic Aware Routing (A-STAR)**. This protocol is based on the anchor-nodes like GSR. That is, a sending node computes a route to the destination based on the junctions of the street map. However, it introduces weights on the streets based on their traffic density which will affect the routing decision.

Two different techniques are proposed so as to obtain these weights on the streets. The first one is to obtain what they call a *statistically rated map*. They realize that in metropolitan areas there are streets with different number of lanes and they accommodate different amount of traffic. They also noticed that there exists a relation between the number of bus lines circulating on each streets and the traffic flow along them. So this map basically includes weights inversely proportional to the number of bus lines in each street. This way, a sending node can obtain the route with the less weight by applying *Dijkstra's least-weight path algorithm*. The second one provides a more updated information through a *dynamically rated map*. Although this information is more realistic, the cost of obtaining this information requires sensor deployment over the scenario to obtain such information.

The other interesting contribution of A-STAR is its recovery strategy. The *Right Hand Rule* does not provide a good performance in urban scenarios. So, instead of using it, when a node gets stuck in a local optimum, it calculates a new anchor path avoiding temporally the area where the local optimum happened which is marked as *out of service*. In addition, this information is piggybacked in the recovered packets allowing nodes which receive these recovered packets to update their maps avoiding these out of service areas.

Fig. 3.12 shows an example of this recovery strategy. The *red vehicle* reaches a local optimum, so it marks the street of the left as *out of service*, as it starts the recovery stage, it include this information into the packet to be forwarded. This way, the *white vehicle* will also mark the aforementioned street as out of service avoiding its use and new local optima in that area.

The drawback of this repair strategy is that when only one path is available for a node, after marking the area as *out of service*, there will not be any other available paths, so the node will drop

the packet.

GyTAR, proposed by Jerbi et al. [70], takes advantage of traffic information in junctions. Instead of composing a route to the destination, packets travel from one junction to another and they are scored considering their traffic density and their distance to the destination.

Between two junctions, the selected strategy is that of forwarding the packets to the destination in an improved greedy mode where the next forwarder is selected by considering their future predicted positions. So, the closest to the destination is the selected next hop. Regarding the repair strategy used when a node gets stuck in a local optimum, GyTAR bases this strategy on the idea of "carry and forward". That it, the node stores the packet until it finds a new suitable neighbour to forward the packet.

Zhao and Cao [72] propose several **VADD** protocols with the idea of store, carry and forward packets. Routing decisions are dynamically made at junctions where each street is evaluated regarding the delay it introduces to data if they were selected. In the end, the protocol aims at obtaining the path with the minimum packet delivery delay. For this purpose these protocols consider the road's length, current speed, the maximum allowed speed, the mean traffic density, the delay introduced by next possible roads and also the probability of choosing them. The estimation of the delays for all the streets suppose solving an algorithm of $\Theta(n^3)$ complexity, being n the number of junctions. So, in order to reduce its complexity, a boundary area is defined around the current position of the forwarding node, limiting this way the number of streets to analyse.

VADD introduces three packet modes: intersection, straightforward and destination.

Data forwarding in *straightforward mode* is easy because within a street the packet can only take two opposite directions defined by the street. VADD uses GPSR to forward the packet aimed at the next junction. In *destination mode*, the coordinates of the destination are the target of the packet using GPSR too for its forwarding. Intersection mode is more complex than the previous two ones. This is the reason why the authors propose three alternatives to achieve this task: (i) Location First Probe (L-VADD) where the closest node to that direction is the more promising one to forward the packet. This scheme present routing loops. (ii) Direction First Probe (D-VADD) avoids these loops by selecting among the nodes moving towards the selected direction the one to forward the packet but they introduce a longer packet delivery delay compared to the previous scheme. (iii) The authors propose a trade-off choice called Hybrid Probe (H-VADD) where the first option is to use L-VADD and if a loop happens they change to D-VADD reducing this way the packet delivery delay.

Parametrization is the one of the most drawbacks of this protocol. The determination of the boundary area to achieve a good trade-off between computational complexity and accuracy can be a hard task. In addition, the authors claim that their hybrid approach obtains the best performance, however it is not clear how to achieve it due to the difficulty of detecting cycles.

Wu et al. [73] propose **Mobility-centric Data Dissemination Algorithm for Vehicular Networks (MDDV)**, an opportunistic protocol which introduces traffic information by analysing the number of lanes of the streets. Upon receiving a message, every node moving in the direction towards the destination assumes the role of forwarder for a given time. So, unlike other protocols where only one vehicle is responsible for forwarding a packet, in this proposal, a group of vehicles are responsible for this task. This scheme increases the number of transmissions in the network so a trade-off between the message overhead, i.e. the number of vehicles which will act as a forwarders, and the reliability of the same must be obtained.

The last protocol we are reviewing in this chapter is **Static-Node Assisted Adaptive Routing protocol (SADV)** proposed by Ding et al. [74] SADV, introduces a fixed static node in the street map, concretely at junctions. These static nodes are able to measure the vehicle density between junctions. To do so, they introduce a time-stamp at forwarding a packet when they find a good candidate. This message will be received in the other end of the junction by another static node allowing it to estimate the delay of the street. This way, they compose a delay matrix of the urban scenario which is used by these static nodes to make forwarding decisions.

Within the street, nodes use geographic routing to reach the nearest junction. SADV is delay tolerant and introduces different buffering strategies like: First In First Out (FIFO), First In Last Out (FILO), or least delay increase which aims at reducing the increase in the overall packet delivery delay.

The main drawback of these approaches is that the mechanism to gather traffic information in these protocols is not always clear, and even having this information, traffic prediction is difficult to calculate incurring in erroneous or imprecise routing decisions.

3.3. The transmission range assumption

The most of the reviewed protocols assume ideal transmission range in such a way that if the distance between two nodes is less than this transmission range, they can directly communicate. However, in real scenarios, different factors like fading, interference, collisions and the likes affect the propagation and decoding of wireless signal. In fact, the probability of reception decreases as the distance between transmitter and receiver increases. A complete study regarding these questions was made by Cabrera et al. [2].

The geographic-based routing protocols reviewed above use greedy heuristics to select the next forwarder for a given message. Thus, they choose the farthest neighbour as next hop, which will have a low probability of reception. This problem worsens as the density increases, because the probability of selecting a neighbour in the limit of the range transmission is high.

Solution. In order to deal with the transmission range assumption, we propose two schemes. The first one consists of a receiver-based next hop selection, i.e., the sender transmit the data message without pre-selecting a forwarder neighbour. The neighbours of the node will be the ones that make the decision [76]. The second one consists of being aware of the link status with the neighbours and make intelligent forwarding decisions according to this status information.

3.3.1. Density

The density of the streets is another argument be aware of. As commented above, VANETs are unbalanced network which can cause shortcomings to the routing protocols. When a source node calculates the path that a packet must traverse to reach its destination, it does not have the knowledge in real time of the traffic density in each street. If a pre-calculated route makes a packet to traverse a lowly dense street it is more likely the packet to reach a point where there is no promising neighbour to deliver the data packet.

Solution. To deal with these problem we propose to recalculate the route that must follow a packet in each forwarded. Therefore, if a forwarder being aware of its environment detects that there

is a better way to reach the destination, it will apply this decision to the message, improving the delivery ratio.

3.3.2. Use of stale information

Geographic routing protocols for VANETs highly depend on the knowledge of neighbours positions which is updated periodically via beacon messages. However, if some beacons get lost due to temporary transmission errors, some vehicles become unaware of the existence of nearby neighbours. Besides, positions get outdated because of the mobility of the vehicles provoking routing problems related to the interval at which beacons are issued and also related to the time that the information is considered useful (usually from one up to three times the beacon interval).

Beacon losses

If a vehicle no longer receives the beacon of one neighbour it will remove the information about such neighbour. However, due to temporary interferences, these beacons could not be received and as a result, neighbours think wrongly that it is no longer reachable.

This situation is a problem for geographic routing protocols which do not incorporate DTN-support, like GSR, gpcr or A-STAR among others. The reason is that if due to interference errors a vehicle does not receive a beacon from a neighbour it assumes that is no longer reachable. If this neighbours is the only suitable and it discards a packet that could be sent to this neighbour in the following beacon intervals.

Stale positions

Due to the aforementioned beacon losses, routing protocols work with outdated information and therefore stale positions.

On the one hand, these stale positions could cause packet losses if the selected neighbour has sent its beacon in the limit of the transmission range, but due to its speed is now out of this range. In this case, the node will select it as a neighbour but the packet will not reach the neighbour being lost.

On the other hand, another drawback of this stale information is that it may cause a temporary loop between two vehicles if the geographic routing protocol also follows the store-carry-forward paradigm. For instance, two vehicles move along a road with opposite directions. Vehicle A follows the direction to the destination an B the opposite but is nearer to the destination. If in a moment vehicle A does not receive the beacon sent by the vehicle B then it will assume that its more promising to reach the destination. However vehicle B is now farther from the destination because of its speed. Thus, A will send the message to B because it thinks is the better node to reach the destination and B will again send the message to vehicle A due to the same reason, see Fig 3.13.

Solution. In order to solve the first problem related to stale position we propose to employ the store-carry-forward paradigm and also position estimations. For the second drawback, we propose that vehicles can piggyback their current speed vectors within the periodic beacons. Thus, nodes could make an estimation of the position of a neighbour in the next moment.

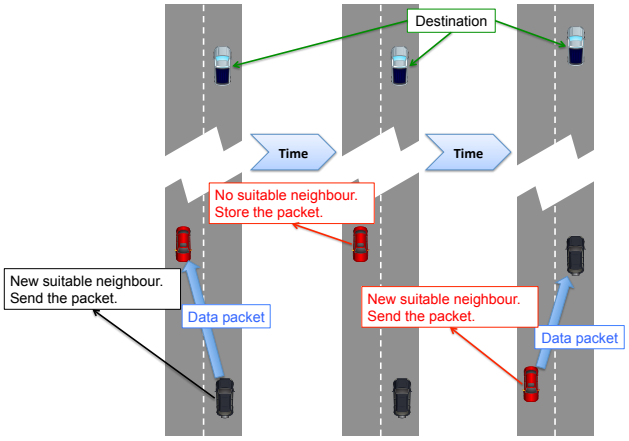


Figure 3.13: Temporary loop example.

3.4. VANET Routing Design Alternatives

VANETs have very distinctive properties such as constrained mobility, uneven network density, possibility to access to additional information (e.g. GPS coordinates, city maps, speed, etc.) and a potentially high number of vehicles. While the former constitutes a design challenge in terms of required scalability, the other features can be exploited by VANET-specific routing solutions to increase their performance. This means that VANET routing protocol designers have to face a number of design choices to make their protocols work under those conditions. We analyse below the most relevant design issues.

Achieving scalability. Given that VANETs can consist of a large number of vehicles, protocols that maintain routing tables or flood the network to find end-to-end paths are not adequate. Most recent routing solutions for VANETs have moved towards the use of localized algorithms. That is, a node makes routing decisions based solely on information locally available in the close vicinity of that node. Protocols with localized operation are highly desirable for VANETs because their control overhead can be greatly reduced by not requiring nodes to know the topology of other parts of the network. With this approach, routing protocols must be designed to work with partial (i.e. local) information while still achieving a good overall performance.

Trajectory pre-computation. Given that vehicles can only move along streets and roads, most VANET routing protocols pre-compute a desired trajectory for data packets to follow (e.g. list of streets to follow). The routing process then takes care of selecting forwarding cars hop-by-hop so that the message effectively travels along those streets. The advantage of this approach is that the overall path may still be valid regardless of the particular vehicles located at each street. However, it may happen that some streets in the trajectory cannot be followed due to the lack of vehicles. Moreover, traffic conditions may also have changed since the source node pre-computed the trajectory. Hence, in our proposed protocol we allow intermediate nodes to re-evaluate and re-compute a new trajectory when needed.

Neighbourhood discovery. Discovering neighbours is of paramount importance for selecting next hops in VANET routing protocols. It can be performed as part of the route establishment or using dedicated 1-hop control periodic messages called beacons. These beacons contain information about

the identifier, position and other relevant information of the node. However, the selection of a proper beaconing interval becomes really important to find a good trade-off between control overhead and updated neighbourhood information. Some protocols may use adaptive beaconing intervals depending on the mobility of the network. Another recent approach is the reactive discovery of neighbours in a per-packet basis as part of the data packets forwarding. These protocols are usually called beacon-less.

Identification of the destination. In traditional MANET routing protocols, routing is based on the identifier of the nodes. However, in VANETs most routing protocols usually route messages to a particular area or position. In these geographic routing solutions the source node needs to know the position of the destination. For this purpose, some routing protocols rely on broadcasting query messages including the identifier of the destination. When such a message is received by the destination it replies with a response message including its current position. Some other protocols just assume that location of the destination can be obtained by any other external protocol responsible for this discovery. Another alternative is that of using a distributed location service in which nodes update periodically their positions and any other traffic information like its speed, direction and so on matching this information with their identifiers. Data sources can query these location servers to obtain the position associated to a particular identifier.

Data forwarding. Although ad-hoc routing protocols usually create routing tables containing the next hop to reach the destination based on a given metric. This can be inefficient in highly dynamic scenarios due to the need to repair routes after link breaks. A more appealing solution for VANETs could be to route in a per-packet basis. This is very common in geographic routing solutions in which the data packet is routed according to the current neighbourhood of the forwarding node in the very moment of forwarding the message. Traditional geographic routing protocols usually select the neighbour which provides a greater advance towards the destination.

Dealing with network partitions. VANET networks are characterized by an uneven vehicle density. Even if traffic is dense, crossings, traffic lights and the likes produce frequent network partitions. This means that some data packets may eventually reach a vehicle which is not able to continue routing the message as planned. Some protocols just neglect this issue by assuming that there is always enough vehicle density. There are better alternatives to deal with those situations. For instance, the node detecting the situation may try to find a different path. Some protocols use void avoidance ideas from geographic routing over the street map. Another option is to store the message until a new forwarding opportunity (i.e. new neighbour) appears. This is interesting for delay-tolerant information. However, in very congested streets it is not very likely that a new neighbour is discovered within a reasonable amount of time. Probably the best option would be an adaptive scheme which varies the operation mode depending upon the network conditions and the application requirements.

Prediction of future events. Some salient features of VANETs such as the constrained mobility and the knowledge of odometry and position information allow vehicles to predict future positions. When that information is exchanged with neighbouring vehicles, it also allows routing protocols to make more informed decisions. While using prediction seems like a good approach, it must be carefully considered. Inaccurate information or predictions can seriously reduce the performance of the routing protocol.

Use of additional information. Vehicles have access to lots of information about their context. They are expected to be able to use navigation software and even to access external information services providing information about traffic densities and so on. Some of this information may provide

additional advantages to routing solutions. They help routing protocols at making informed decisions when selecting paths, neighbours, etc. Being able to take advantage of that information is a winning strategy for VANET routing protocol designers.

In the following subsections we explain the main problems of the aforementioned protocols and the reason why these problems affect their performance proposing solutions that have been taken into account in the developing of our proposed routing protocol BRAVE.

3.5. Our VANET specific routing protocol

In this section we propose Beacon-less Routing Algorithm for Vehicular Environments (BRAVE) [77], a fully-localized protocol specifically designed considering the design issues mentioned above. The overall idea behind its operation revolves around the ideas of spatial awareness and beacon-less geographic forwarding. By spatial awareness we refer to allowing intermediate nodes to change the initial plan (streets to follow) based on their local information and their view of the street map. This allows the protocol to avoid following trajectories which may become bad alternatives as the message travels to the destination. In addition, it avoids having to disseminate additional information (e.g. density of vehicles along streets) across the whole VANET, so that any possible data source can compute a good trajectory. Thus, with our proposed scheme additional information only needs to be disseminated to nearby areas.

Unlike many of the previous solutions, BRAVE performs hop-by-hop data forwarding along a selected street using an opportunistic next hop selection based on the idea of beacon-less geographic routing. Instead of making forwarding decisions based on positions of neighbours gathered by periodic beacons, BRAVE uses a reactive scheme. The current forwarder sends the data packet and the next forwarder is selected among those who already received the data message correctly. To give priority to those providing better improvement, we adjust the timing of the responses so that the best candidates (according to some metric) answer first and cancel responses from other nodes. This idea solves well-known issues in existing routing protocols that were reported in [51].

In the next subsections we elaborate on the detailed operation of each part of the protocol and how they work together to route data messages efficiently in VANETs.

3.5.1. Spatial Awareness or Additional information

The assumption that vehicles have access to a street map and possibly some additional information such as traffic densities, estimation of delays, etc. has become quite common in existing solutions. In most protocols the source vehicle generating the data message computes the shortest path from its current position to the position of the destination along the street map. To do that, they model the street map as a graph where edges represent streets and junctions are represented by vertices. Some solutions, use additional information just to add weights to edges before computing the shortest path. Then, some sort of geographic forwarding is used to send data along that path.

While this idea of following a trajectory allows the protocols to follow the established path without relying on specific vehicles, it may introduce some inefficiencies. In particular, if the information that the source node has is not accurate or the information changes while the data packet is being delivered, the protocol may end up having troubles to follow the initial trajectory. For instance, the

data packet may reach a street with no vehicles. Some protocols propose to re-evaluate the overall path at junctions, but there are still situations in which good forwarding alternatives are missed.

In BRAVE, the street map and any available additional information is used to select the next junction to be reached using hop-by-hop forwarding. The main novelty is that in our case the trajectory of the packet is not pre-computed by the source vehicle. Each intermediate vehicle re-computes the trajectory and decides to which next junction the data message is routed. So, changes in the direction the message should follow may happen at any intermediate vehicle with updated information, not only when vehicles or messages reach junctions.

To compute the next junction, the current node holding the data packet applies Dijkstra's shortest path algorithm to the (possibly weighted) street graph. After that, it stores the next two junctions towards the destination. The first one is used as the next intermediate destination for the geographic routing process. The second one is used to help in the process of deciding when to change from one street to another.

Another interesting aspect of BRAVE is the criteria to change the direction of the geographic routing from one junction to another. While most protocols mention that when the message reaches the junction the new geographic destination is the next junction in the pre-computed trajectory, this simple idea is not easy to implement in practice. What does it mean reaching the corner in terms of distance?. For instance, some protocols consider being in the corner as being within radio range of the geographic position of the junction. However, radio range is quite variable and fixing any value may work fine our pretty bad depending on which particular scenario we are facing.

In BRAVE, a node uses the above-mentioned two junctions to decide when to change to the new street. In particular, a vehicle establishes as the new next junction the second one when it has any neighbour whose distance to the second junction is smaller than the distance between the two junctions. That is, given a vehicle f currently routing the data message with first junction as position FJ and second junction as position SJ respectively. Then, the current node changes to route the message toward the second junction when it has at least a neighbour at position N such that $dist(N, SJ) \leq dist(FJ, SJ)$, where $dist(.,.)$ represents the distance (e.g. Euclidean) between the given two points.

We can see in Fig. 3.14 an example in which the distance between the neighbouring vehicle a and the second junction is less than the distance between the two junctions. As shown, this condition is sufficient to guarantee that the street change happens where required. Moreover, this solution works well without needing to rely on any external configuration parameter such as distance to junctions, radio range, etc.

As we have seen this concept of using two junctions to guide the geographic routing process allows following an overall trajectory without the overhead of carrying the whole trajectory in data packets. In the next subsection we discuss how the data forwarding proceeds towards the first junction using other vehicles as relays.

3.5.2. Data forwarding along streets

We have already analysed how BRAVE decides to route data messages along a different street. Now, we discuss how a message is routed within the current street.

As in most VANET routing schemes, we employ a variant of geographic routing so that vehicles forward the message to one of its neighbouring vehicles being closer to the next junction then itself.

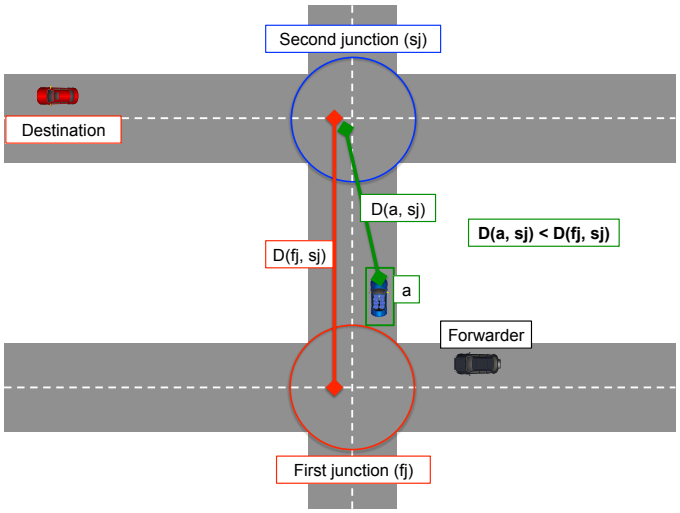


Figure 3.14: Illustration of the criteria to change between first and second junction.

However, as showed in [51] existing protocols have serious issues to effectively route data messages. In particular, their main issue is that they use periodic beacons to gather positions of neighbours, producing inconsistencies that may lead to stale information, forwarding loops, and so on. Another issue which is often neglected about the use of beacons is that even if information contained in beacons is fully accurate, as beacons messages are small, they usually have higher probabilities of correct reception than data messages. Thus, in realistic deployments it happens often-times that a vehicle receives beacons from a neighbour whose radio link is so weak that data messages cannot reach the next hop.

To avoid these issues, our data forwarding strategy is based on beacon-less routing. That is, rather than selecting next hops based on information provided by beacons, BRAVE selects next hops reactively among those nodes that successfully received the data message. The advantage is that this solution prevents failed forwardings due to neighbours for which beacons were received but for which the radio link is quite weak to successfully receive the data packet. This idea has been effectively proven in [76], under realistic wireless sensor network deployments were the Beacon-less On Demand Strategy for Geographic Routing in Wireless Sensor Networks (BOSS) was defined. We have taken it as a first step to develop our proposed solution. The most significant feature adopted in BRAVE is the addition of data-ferrying capabilities so that messages can be temporarily stored if there is not a good forwarding opportunity. This way, previously stored messages are considered when new neighbours are discovered.

To avoid the use of beacons, BRAVE uses four message types: *DATA, RESPONSE, SELECT* and *ACK*. Given that there may be multiple data sources in the VANET, each of these messages should be clearly identified as belonging to a particular message forwarding. So, they all include what we call a *message key*, which is just an unique value which is obtained by concatenating the identifier of the source node (SRC) and a sequence number (SEQ) which is set by the source when generating the message. Thus, a forwarding vehicle holding a data message which is being routed along a street broadcasts to its 1-hop neighbourhood the DATA message and waits for responses during a certain period of time $Tmax$. This DATA message contains the position of the current vehicle as well as the

position of the destination and the next two junctions.

Every neighbour receiving the DATA message and being closer to the next junction than the current forwarding node stores this DATA message and schedules a RESPONSE message. The time at which this response message is scheduled depends upon the goodness of this neighbour as next hop. For instance, if our routing metric is hop count, then it is scheduled so that closer the neighbour is to the next junction the sooner it sends back its RESPONSE message. In the next subsection we describe all the details about how to set that time and some improvements to the basic operation to prevent multiple responses.

After receiving a RESPONSE message, each vehicle (except the forwarding vehicle) cancels its timer and deletes the data packet if the destination and the key of that message are the same in its RESPONSE message. This way, we prevent confusion among different exchanges. When the forwarding vehicle receives a RESPONSE message, it checks that the message is aimed at him, and also compares the key of the received message with the expected one. If they are equals, the vehicle process that message. If a neighbour does not receive a RESPONSE message from other neighbour it will not cancel its timer sending its own RESPONSE to the forwarding node. Thus, the forwarding node will receive both RESPONSE messages. However the forwarding node will attend the first RESPONSE message ignoring the rest of them. When a node sends a RESPONSE message, it starts a timer to wait to be selected as next forwarder. If this timer expires, the node is no longer a candidate to be a forwarder for this message so it deletes the data message resetting its state (idle).

Once the forwarding vehicle has received the RESPONSE message, it broadcasts a SELECT message that indicates which neighbour is selected as the next forwarder and the *message key* of the corresponding DATA message. Each neighbour receiving this message checks whether it has been selected as next hop or not. If it has not been selected, then it just deletes the DATA message from its buffer and goes back to its initial state. If it has been selected then that node becomes the forwarder and starts the process again.

To increase the reliability of the protocol, once the SELECT message has been broadcasted, the forwarding node schedules a timer waiting for an ACK message. Thus, the last step consist of confirming the reception of the SELECT message by the new forwarding node. In the normal case, the next DATA message forwarding by the next hop serves as an implicit acknowledgement. However, if the new selected next hop is not resending the DATA message (e.g. has no neighbours to forward the message to and temporarily buffers the message) then it must send back an explicit ACK message.

If the previous forwarder does not receive an acknowledgement message it resends the SELECT message up to two additional attempts. If after that, the SELECT message is not acknowledged then the forwarding vehicle restarts the forwarding process. This mechanism ensures that a packet has been delivered from a forwarding vehicle to the next one. As we shall show in our simulation results, this idea of using opportunistic forwarding among nodes that already received the DATA message together with these retransmission schemes avoids the "Range Limit Problem", Fig. 3.15, which turned out to be the main cause of packet drops [51] in many VANET routing solutions.

In figure 3.16 the whole state machine of the protocol is shown. This diagram, gives the details of the states of the protocol as well as their transitions obtaining an idea of how this protocol behaves when it receives a packet depending on the state. These states are related to every unique packet.

On the other hand, unlike traditional geographic routing schemes, BRAVE does not use any recovery scheme to escape from local minima. That is, when a message reaches a vehicle having no

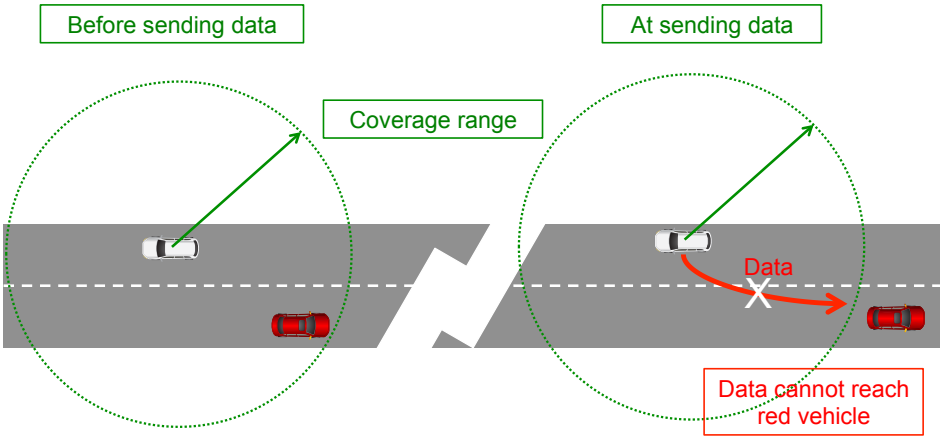


Figure 3.15: Range limit problem

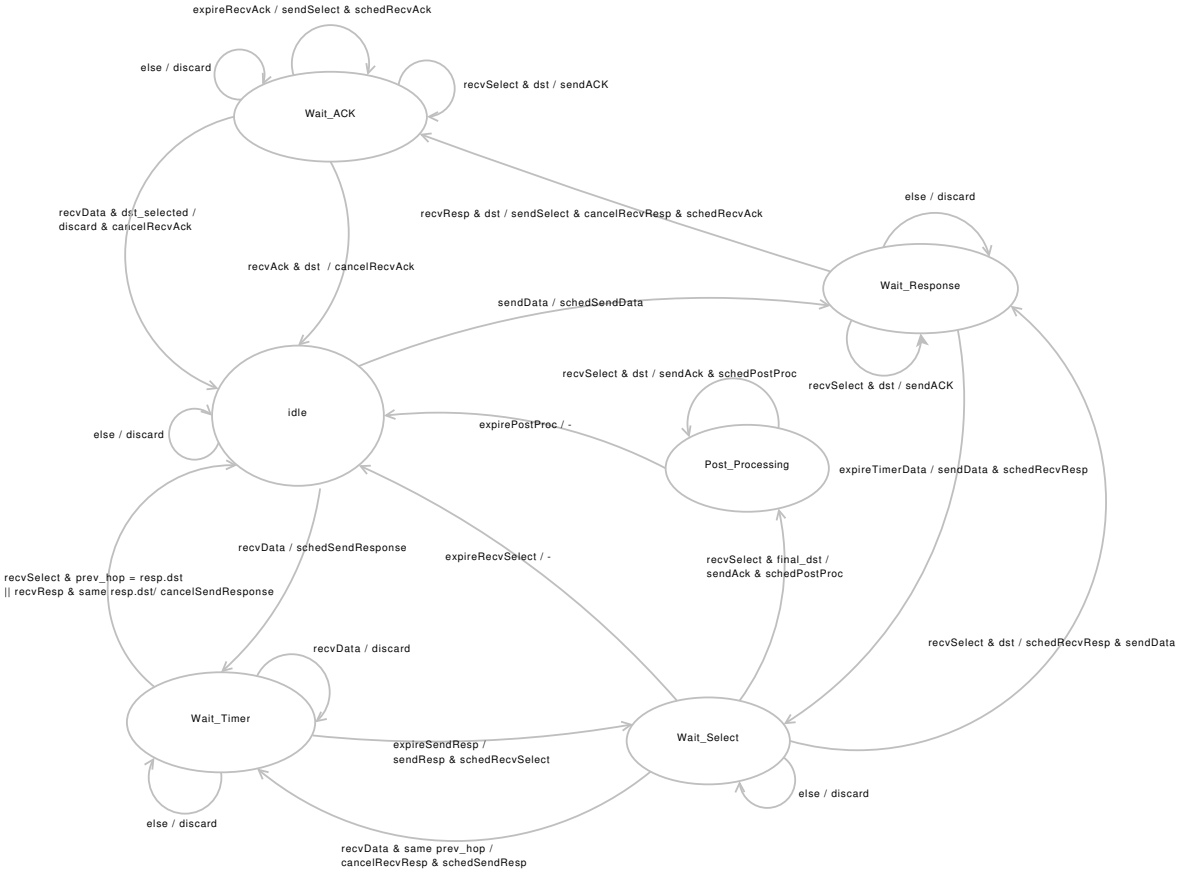


Figure 3.16: State Machine

neighbours closer to the next junction than itself. The reason is that perimeter mode has been proven to be of very marginal benefit in VANET topologies in which most escaping alternatives are along the same street. To deal with this situation (which are frequent in VANETs due to network partitions) in BRAVE we adopt a store-carry-forward approach. That is, we use a packet buffer to store the data packet until a new neighbour being better than the current node shows up. Note that these neighbours are discovered by receiving their periodic beacon messages, but information contained in beacons is not used to make forwarding decisions. The reason why beacons are not fully eliminated is because they are mandatory in the current IEEE 802.11p and DSRC standards under development. So, they would be present anyway.

The data forwarding algorithm is more formally described in the following pseudo-code both for the sender (current vehicle holding the message), Algorithm 1 and the receiver (candidate neighbour), Algorithm 2.

Algorithm 1 BRAVE from the sender's point of view

```
1: key ← generateKey(src_id, p) {Duplicate avoidance}
2: sendDataPacket(p, bcast);
3: finish ← 0;
4: while !finish do
5:    waiting_time = scheduleWaitingTimer();
6:    rcv_pkt ← receiveResponseMsg();
7:    if !hasExpired(waiting_time) then {waiting_time has not expired yet}
8:       src_id ← getSrcId(rcv_pkt);
9:       if rcv_pkt.key = key then
10:         selected_node ← src_id;
11:         sendSelect(key, bcast);
12:         ack_pkt ← receiveAck(); {Blocking instruction}
13:         src_id_ack ← getSrcId(ack_pkt);
14:         if ack_pkt.key = key then
15:            if src_id_ack = selected_node then
16:               finish ← 1;
17:            end if
18:         end if
19:      end if
20:   else {waiting_time timer expired}
21:      resendDataPacket(p, bcast);
22:   end if
23: end while
```

3.5.3. Improved data forwarding by adjusting waiting times

The basic data forwarding scheme explained above can be improved if we reduce the overhead of the protocol in terms of the number of RESPONSE messages which are needed. Another improvement would be to reduce the number of possible collisions due to many candidates sending their RESPONSE message at the same time. Both goals can be achieved at the same time by properly adjusting the timers used by candidates to schedule their RESPONSE messages. The idea is to adjust those timers so that the best candidates according to the particular routing metric answer first. In addition, all other candidates overhearing a response from a better candidate cancel their response.

Algorithm 2 BRAVE from the receiver's point of view

```
1: data_pkt ← receiveDataPacket();
2: src_id_data ← getSrcId(data_pkt);
3: dst_id_data ← getDstId(data_pkt);
4: key ← data_pkt.key;
5: response_time ← calculateTimerResponse();
6: if hasExpired(response_time) then
7:    sendResponse(key, bcast);
8:    select_pkt ← receiveSelect();
9:    dst_id_select ← getDstId(select_pkt);
10:   if ((select_pkt.key = key)and(id = dst_id_select)) then
11:      sendAck(key, bcast)
12:      if own_node_id = dst_id_data then
13:         finish();
14:      else
15:         Now the node takes the role of a sender node
16:         sendDataPacket(data_pkt, bcast);
17:      end if
18:   end if
19: else if responseReceived(key) then
20:   cancel();
21: else if selectReceived(key) then
22:   cancel();
23: end if
```

For the sake of simplicity, we assume without loss of generality the case in which we are using as routing metric the hop count our distance to destination. That is, from all neighbours the best one would be the one being closer to the next junction. The same can be done for other metrics (e.g. remaining time, delay, etc.) by just using normalizing the values between a minimum and a maximum waiting time.

To adjust timers based on the goodness of candidates, we define the progress that a neighbour n provides for a message addressed towards a junction d with respect to the current forwarder c as:

$$P(n, d, c) = dist(c, d) - dist(n, d)$$

where $dist(a, b)$ represents the Euclidean distance between the position of the nodes a and b.

The larger the progress provided toward the next junction by a neighbouring vehicle, the smaller the waiting time should be. We define the Forwarder Coverage Area (FCA) as the circle with center in the current forwarder and radius the theoretical radio range (r). Note that this radius is used as a reference but the protocol can work in situations in which the actual radio range is different from r. We divide the FCA into a number sub-areas of equal width. This is depicted in Fig. 3.17. We then assign waiting times so that all nodes in the same sub-area get the same waiting time, which is then modified by a random component to prevent collisions across nodes in that same area. The assignment function will be defined in such a way that the waiting times associated to each area are ordered according to their progress. Given a Number of Sub Areas (NSA) in which the FCA is divided, a node can easily compute the NSA in which it is located

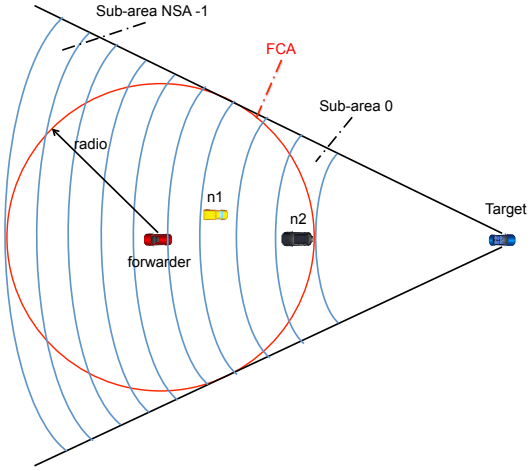


Figure 3.17: Division in areas

$$CSA = \lfloor NSA \times \frac{r - P(n,d,c)}{2r} \rfloor$$

The value of Common of Sub Areas (CSA) is an integer between 0 and $NSA - 1$ corresponding 0 to the area which provides the larger progress. Once the CSA is calculated, each vehicle can compute its waiting time as

$$T = (CSA \times \frac{Tmax}{NSA}) + random(\frac{Tmax}{NSA})$$

where $Tmax$ is the maximum time that the forwarder waits until receiving a RESPONSE message from any neighbour. The function $random(x)$ returns a randomized value between 0 and x.

By using this discrete function we ensure that vehicles from areas closer to the current forwarder only generate RESPONSE messages if there are not better vehicles in the sub-areas providing more progress. In addition, the random component prevents collisions among vehicles located in the same sub-area.

3.6. Performance Evaluation

To assess the performance of BRAVE, we have conducted a set of simulation experiments comparing the performance of existing VANET routing protocols in a realistic scenario. Below we give the details of the simulations and analyse the main results.

3.6.1. Simulation Setup

In order to evaluate the performance of our proposed solution, we have implemented GSR, SAR, A-STAR, gpcr, GeOpps and BRAVE protocols within *The Network Simulator ns-2*, version 2.33[1]. To generate the simulation scenario (street map) and the vehicular mobility patterns, we have used the

[1] http://www.isi.edu/nsnam/ns/

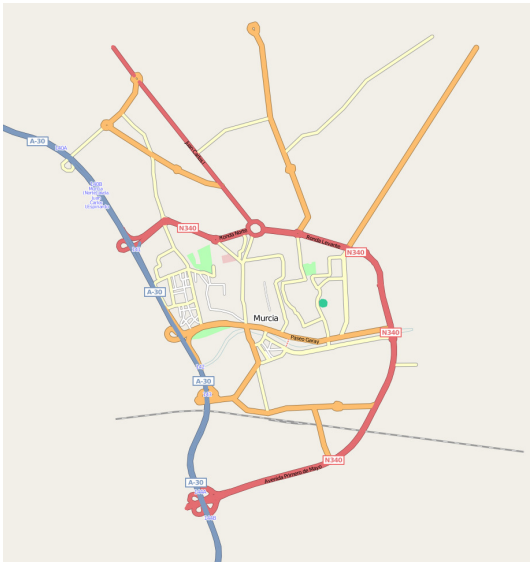


Figure 3.18: Map of Murcia city center and access roads used in our simulations.

well-known SUMO tool[2]. This tool allows to simulate realistic vehicular movements such as traffic jams and stops at intersections provoking disconnected networks and uneven distribution of vehicles. It also allows us to build a realistic street map.

In particular, we have run our simulations in an area of $5 \times 4km^2$ in the city center of Murcia, Spain (Fig. 3.18). We have selected the most relevant streets, so our scenario consists of 53 streets and 28 junctions. Vehicles move through 20 predefined routes at a maximum speed of $50km/h$ inside the city, and $80km/h$ on the highway that crosses the scenario. The routes followed by the vehicles have been selected according to realistic situations. We have also considered a wide range of traffic densities. Vehicles are injected into its route at a certain traffic rate. This rates are covered from 1/30 to 1/10 vehicles per second. In such a manner the $1/x$ rate means that each x seconds a new vehicle is injected into its route.

In our simulations, wireless signals propagate according to the *two-ray-ground* model. Vehicles carry out their communications via 802.11p interface card, implementing the enhanced *ns-2* 802.11 physical and MAC models[3]. Transmission power is adjusted allowing a maximum transmission range of $250m$.

We have simulated 10 independent runs for each configuration (figures in section 3.6.2 show the average of such runs). For each run, there are 100 different random data sources. Each one sends a 512 bytes data message towards a static station located in the center of the city. This is done to avoid introducing bias in our comparison due to considering different moving vehicles as destinations. We have employed a beacon interval of 2 seconds. Since the protocols employ geographic routing, each vehicle needs to know the position of the destination. For a fair comparison we have assumed that this information is known by all vehicles although they would use a location service in practice.

Regarding the protocols compared with BRAVE, our implementation of gpcr computes the corre-

[2]http://sumo.sourceforge.net/
[3]http://dsn.tm.uni-karlsruhe.de/english/Overhaul_NS-2.php

lation coefficient of the neighbours when a vehicle checks whether it is a coordinator or not. In the case of GSR, the list of junctions that a data message must traverse is included within the message header. The same criterion is applied to SAR, since it is strongly based on GSR. Moreover, SAR uses a message buffer to store messages that can not be forwarded at a particular moment towards the destination. Each message can be held in the packet buffer for 30 seconds. To be fair, we have not enriched any of the protocols with any additional information (e.g. street density) other than the city map. For A-STAR we implement its recovery strategy when a packet gets stuck in a local minimum. Finally, in the case of GeOpps, the useful lifetime before data messages get discarded has been set to 180 seconds.

3.6.2. Analysis of results

To assess the performance of the different protocols, we consider the PDR and the end-to-end delay. The PDR is the ratio of successfully received data messages at the destination over the total number of data messages generated. We measure the end-to-end delay as the time it takes a data message to make it to the destination from the time it is generated. In addition to that, we also perform a detailed study of the causes of packet drops to better understand the effectiveness of BRAVE to deal with those issues.

Figure 3.19 shows the PDR achieved by each protocol as the density of vehicles increase. We can see that BRAVE outperforms all other protocols, obtaining a delivery ratio around 0.8 and 0.9 for all evaluated densities. GeOpps is also sensibly better than the other schemes but this comes at the price of additional delay as we shall explain later on. We can see that the protocols based on geographic routing show a "bell-shaped" PDR graph. The reason is that for low densities the PDR is low because the network is highly disconnected. For high densities the PDR gets low again due to the high contention and transmission failures at the Media Access Control (MAC) layer. In particular, the "Range Limit Problem" that we mentioned in the previous section. This occurs because the higher the density, the higher the probability of finding a neighbour just near the transmission range of a vehicle. Therefore, is more likely to choose a neighbour which has a low probability of reception without errors. By contrast, BRAVE solves that problem thanks to its opportunistic next hop selection. In fact, the higher the traffic density the higher the delivery rate it obtains. When the density is low and there are no neighbours, BRAVE stores the packet until any new neighbour appear into the transmission range of the forwarder vehicle, and the packet is forwarded. This is the reason why BRAVE obtains good results across a wide range of vehicle densities.

This is clearly supported by Fig. 3.20 where we analyse the cause of packet drops. For lower densities drops for GSR, SAR, A-STAR and gpcr are mainly due to the absence of candidate neighbours (NBV). SAR is less affected than the others because the packets are not dropped unless they expires into the buffer of the vehicle. We show that drops due to expired packets as PEX in the figure.

Moreover, this figure also corroborates the explanation of the "bell-shaped" behaviour. The higher the traffic density, the lower the number of drops caused by the absence of neighbours, since exists a better connectivity. However, in those higher densities we can see an important increase of drops due to the "Range Limit Problem". That is, packets lost because they are not successfully decoded by the receiver. We mark those as (MAC) in the graph.

GeOpps is not very severely affected since it does not employ geographic routing. Messages are only handed over from one vehicle to another if its expected delivery time given its trajectory is

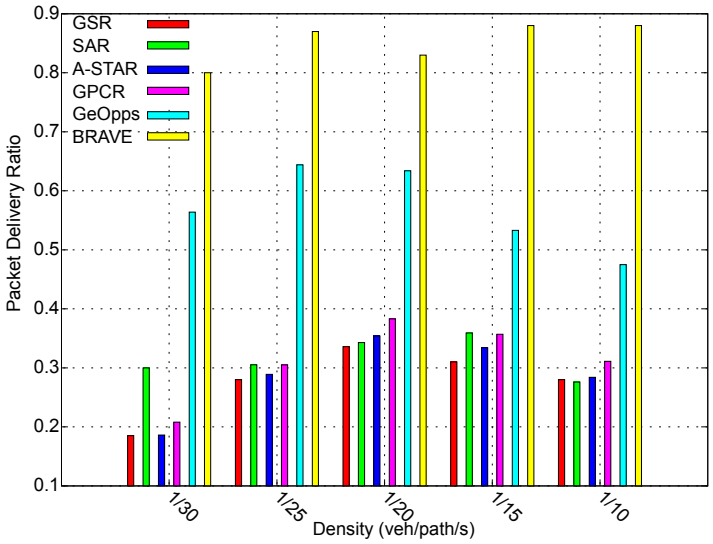


Figure 3.19: Packet Delivery Ratio

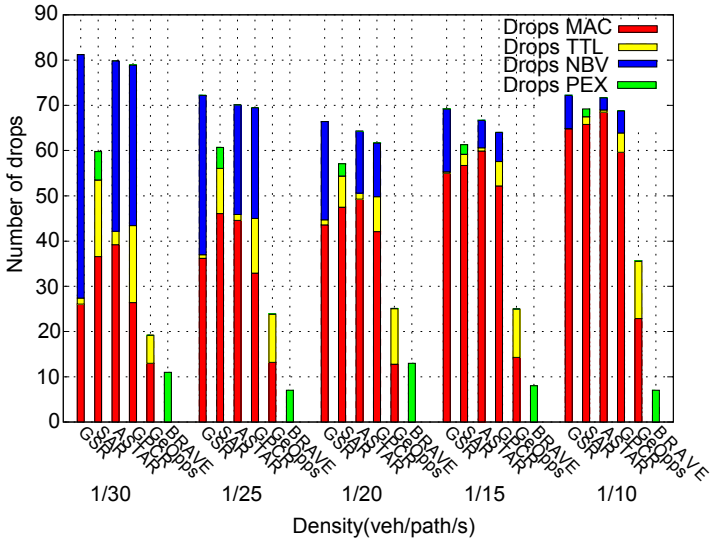


Figure 3.20: Analysis of the cause of drops

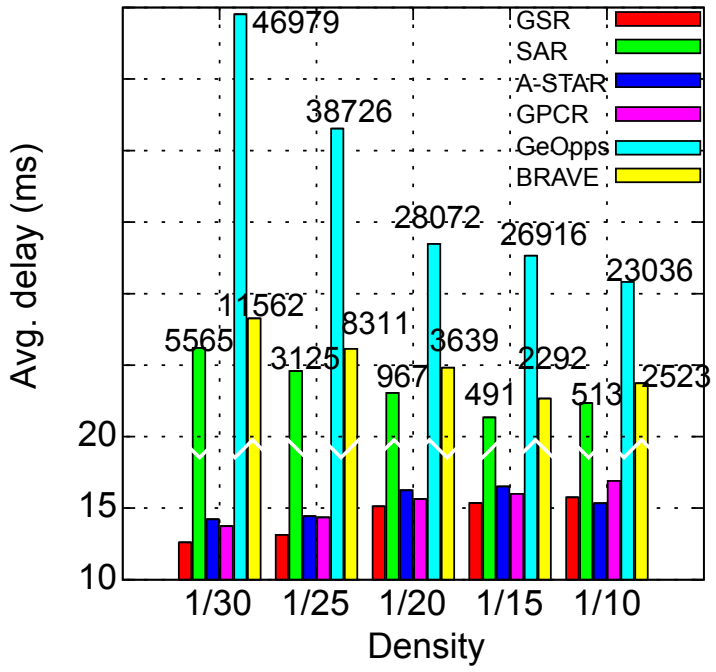


Figure 3.21: End-to-end Delay

better. So, there are fewer communications over lossy links which ends up in having a slightly better reliability at the expense of additional delay. The losses in GeOpps due to exceeded Time To Live (TTL) (packets that exceed the maximum hop count). This is a documented problem. (see Cabrera et al. [51] for additional details).

In the case of BRAVE, we can see that the protocol has only very few packet drops compared to the other solutions. All of them are produced due to expired packets. That is, a message that reaches a car that never finds a suitable forwarding candidate. We can see that our proposed protocol is not affected by the "Range Limit Problem" (MACMAC because the opportunistic neighbour selection does a great job at avoiding packet losses during the forwarding process. Key to this is the fact that only vehicles that successfully received the data packet are considered as candidate relays.

Finally, we study the end-to-end delay in Fig. 3.21. As we can see, those protocols that count on a buffer to deal with temporary network disconnections present additional delay, but at the same time they are able to deliver messages in scenarios with large disconnections in which the other protocols just fail. As the density of vehicles increases, the networks is more connected and allows protocols to reduce the end-to-end delay. By being delay tolerant, GeOpps is the one with larger delays. This is because the message is forwarded only to vehicles which can carry the message closer to the destination. In the case of BRAVE, the proposed protocol manages to get reasonable delays when the network is connected, while still being able to deliver messages in disconnected networks at the expense of higher delays. The other protocols have a much shorter delay, not because they better in finding paths, but due to the fact that they only manage to deliver the messages when the network is connected.

3.7. Conclusions

In this chapter we analysed the problem of efficient routing in vehicular networks. This is a challenging problem because of the intrinsic properties of VANETs such as frequent disconnections, variable topology, constrained mobility, etc.

We present a novel algorithm called Beacon-less Routing Algorithm for Vehicular Environments (BRAVE). The main ideas behind BRAVE are an opportunistic next hop selection within the routing process, and an improved overall path re-computation. The opportunistic neighbour selection allows the protocol to obtain a high reliability in relatively dense scenarios. The proposed schemes eliminates a lot of contention and guarantees that the next hop has successfully received the data packet. The second novelty refers to the fact that BRAVE allows intermediate nodes to recompute the trajectory (list of streets) towards the destination. Just by adding the positions of the next two corners the protocol is able to effectively follow the best paths while still working with local information.

Our simulation results show that the proposed protocol is able to outperform existing solutions in terms of PDR over a wide range of vehicle densities. In addition, it manages to get a good trade-off between delivery ratio and end-to-end delay.

All the protocols explained in this chapter assume the collaboration of the nodes of the network to hand over the messages to the destination. However, the existence of malicious users with a totally different motivation can impair the performance of these protocols and even obtain sensible information by overhearing the messages. This is precisely the motivation of the following chapter. Making VANET routing protocols more secure under hostile information where malicious users can be present.

Chapter 4

Evaluation of the use of guard nodes for securing the routing in VANETs

In the previous chapter, we discussed the advantages and disadvantages of the main routing protocols in the literature. Based on them, we developed a new protocol called BRAVE which outperformed theses ones.

All these protocols assume that every node in the network collaborates forwarding the packets the best they can according to the protocol. Nevertheless, there exist hostile scenarios where not all of the nodes are so collaborative, and where they act according to their own interests. These malicious nodes can impair the seriously the performance of the previous routing protocols because they were not designed taking into account this sort of actions.

In this chapter we tackle this topic, analysing the different threats that can affect the routing task and providing a mechanism to reinforce the delivery of packets.

4.1. Introduction and motivation

As we commented in Chapter 3, routing protocols are pretty important for VANETs because all the services and applications of the upper layers rely on them for delivering their messages. So if they do not work properly, all these services and applications will be affected by them.

Routing protocols are usually designed to deal with the problems of the signal propagation, the variability of the links among nodes and so on. But there exist other phenomena for which they are not designed.

Most of the proposed routing solutions for VANET have not considered security issues. That is, they are not able to deal with certain security threats such as *spoofing*, *Sybil attacks*, *selective forwarding* or *sink-hole attacks*, where malicious nodes try to impair the routing protocol by not forwarding the information to other nodes. For this reason, a variety of solutions to these attacks were proposed in the literature [78–80].

In addition, the IEEE 1609.2 standard [81] for securing Wireless Access in Vehicular Environments

(WAVE) addresses the issues of securing Vehicle-to-Vehicle (V2V) communication against spoofing and eavesdropping by using a Public Key Infrastructure (PKI). Thus, a Certification Authority (CA) will be responsible for generating and managing digital certificates. This standard proposes vehicles to sign the messages and piggyback the certificate of the sender, which contains its corresponding public key. Therefore, when the destination receives the message, it is able to validate the authenticity and integrity of the message. The drawback of this technique is that if every V2V message includes this public key, usually as a X.509 certificate, their size will be notably increased.

In this chapter, we extend our routing protocol presented in Chapter 3 called BRAVE by introducing guard nodes leading to our proposal called **S-BRAVE**. In S-BRAVE, messages are signed by taking advantage of the Public Key Infrastructure (PKI). Nevertheless, we have developed an efficient certificate exchange mechanism where the certificate will be inserted in V2V messages only if the other vehicle has not received it yet. Thus, authenticity and integrity is guaranteed for every message transmitted along the VANET. Since our target in this paper is that of securing the routing protocol we have not dealt with the issue of certificates revocation. However we have worked in this issue in a previous work already published where we exploited the capabilities of the Next Generation Networks(NGNs) to do it [82].

On the other hand, in the environment of WSNs, a proposal for strengthening the security capabilities of a routing protocol have obtained very good results in terms of packet delivery ratio. This proposal is also based on the concept of watchdog nodes or guard nodes. These are defined as neighbouring vehicles that overhear packet exchanges to ensure that the packet is forwarded by the intended next hop [83]. We have applied this technique in the VANET environment making BRAVE able to transmit messages in hostile scenarios where malicious nodes selectively forward messages in order to cause packet losses. For this purpose, using the aforementioned technique, neighbours watch other selected nodes to be sure that they forward packets to the next hop. If a node is selected to forward the packet and it does not transmit it, then neighbouring nodes will select themselves as a forwarder, being responsible for sending the packet to the next hop. The whole process is detailed in later sections.

The remainder of this chapter is organized as follows: In Section 4.2 we review the different studies and proposals related to the topic of this chapter. Section 4.3 describes the different routing-specific threats. Our proposed solution to make the routing protocol more secure is presented in Section 4.4. We evaluate the performance of our proposal in Section 4.5. Finally in section 4.6 we will comment the benefits obtained in this article as well as our next research steps related to this work.

4.2. Related Work

In 2005, Parno and Perrig [40], envisioned the future technology used for vehicle manufacturing employing wireless communications with a radio range of at least one kilometre. Among the different aspects of these vehicular networks, they identified the main challenges these vehicular network would have to cope with like the bootstrap with only a few vehicles equipped with this technology, its high-speed mobility, or security related aspects like authentication, message integrity or privacy.

In order to come early to the security issue, they outlined a first approach of the different adversaries these networks would have to face, as well as different kind of attacks that could affect their behaviour.

Regarding the adversaries, they identified five different types:

- Greedy Drivers. These drivers will attempt to maximize their gains, regardless of the cost to the system. For instance, broadcasting a warning message about a fake congestion ahead, making the ahead vehicles to clear the path to his destination.
- Snoops. This category of adversary encompasses either isolated nodes or private companies interested in obtaining drivers profile and sensible information raising serious privacy concerns.
- Pranksters. Pranksters include bored users probing for vulnerabilities and hackers seeking fame via their exploits.
- Industrial Insiders. Like employees and mechanics which can update the software on a vehicle. They can distribute keys or even create keys that would be accepted by all other vehicles.
- Malicious Attackers. Malicious attackers deliberately attempt to cause harm via the applications available on the vehicular network.

Although they did not own precise information about this technology, they enumerate fome of the more likely scenarios:

- Denial of Service (DoS). By jamming the communication channel use by the vehicular network a malicious user can prevent critical information from arriving.
- Message Suppression Attacks. Malicious nodes selectively drop packets altering the correct behaviour of the vehicles.
- Fabrication Attacks. An adversary can initiate a fabrication attack by broadcasting false information into the network.
- Alteration Attacks. A particularly insidious attack in a vehicular network is to alter existing data. This includes deliberately delaying the transmission of information, replaying earlier transmissions or altering the individual entries within a transmission.

This study was resumed later on by Raya et al. [3, 41] and Lin et al. [42].

They first characterize the kind of attacker defining four dimensions: **insider** vs. **outsider**; **malicious** (only aimed at harm the network) vs. **rational** whose target is to obtain benefit of the attack; **active** vs. **passive**; and **local** vs. **extended**.

They also provide a new classification of different attacks grouped in two groups depending on its difficulty. In the first one they include attacks like denial of service, inserting wrong information into the network, disclosure of vehicles ids.

In the latter group they include more sophisticated attacks like the wormhole attack where two nodes cooperate to disseminate erroneous information; or the hidden vehicle where the vehicle cheats its position.

However, they do not discuss these problems from the point of view of a routing protocol. That is, they describe only appropriate security mechanisms like the use of digital signatures, tamper-proof devices, key management, but not how to use them in an efficient way taking into account the overhead that these mechanisms would introduce into the routing protocol.

Another extensive study on different adversary models has been presented by Papadimitratos et al. in [84] where they enumerate the different possibilities that attackers have to harm the network taking also into account not only single node attacks but also colluding nodes.

Papadimitratos et al. [85] also dealt with the problem of securing beacon messages in their secure vehicular communication system. They propose to sign them attaching also the sender certificate into them. Geocast dissemination messages are also digitally signed and augmented with the certificate of the sender. Nevertheless, attaching always the certificate of the sender increases packet size. Given that both beacons and data messages are enlarged, collision probability increases reducing the reliability and increasing the number of retransmissions.

On the other hand, the problem of securing a VANET routing protocol is analysed by Harsch et al. [86]. In this case, they study Position-Based Routing with Distance-Vector recovery (PBR-DV) [55], developed within the context of the Network on Wheels (NoW) project [87], identifying its security threats and possible attacks. This analysis allowed them to provide a security extension to the protocol. This extension comprises the use of digital signatures and a set of plausibility checks to ensure the packet is sent by a non-malicious neighbouring node.

The main drawback of this contribution is the authors do not consider a malicious node which receives the packet and do not relay them. All the considerations taken into account involve the right reception of the packet by a non-malicious node. That is, they check that the packet has not been manipulated or altered by a malicious node. But if a malicious node does not forward the packet, it will impair the performance of the routing protocol.

PBR-DV, works in the following way. After knowing the position of the destination, it uses a greedy forwarding algorithm to reach it. Since, in urban scenarios streets constrain vehicles movements, the greedy process will often reach local optima where no neighbour provides advance to the destination. BRAVE solves this by introducing the next junction as a first destination to be reached into the first packet to be transmitted, which is the packet containing the data information to be sent.

Among the security techniques included in PBR-DV there exists an aspect which is worth highlighting. For packets to be delivered through more than one hop, they introduce two signatures, one for the source node, and another one for the sender (hop by hop). Thus, each packet will be signed twice, and after verifying these signatures, in each hop, the sender's signature will be removed introducing a new one corresponding to the next hop. By contrast, our proposal only introduces one signature in each packet. The first packet sent by BRAVE will be forwarded as transmitted by the source, without resigning it. On the other hand, the rest of the protocol's control messages, which are only one-hop messages, will be signed by the node that sends them.

Finally, Festag et al. [88] propose another security alternative with an interesting mechanism which is the Secure Neighbour Detection (SND) which is applied to a geocast routing protocol instead to a unicast routing protocol. One of the most strict requirements for the secure protocol to work properly is that all of the nodes must be synchronized. Such a requisite is pretty important because the mechanism to assure that a received beacon corresponds to a real neighbour calculates an estimated delay with the time-stamp of the beacon received and the own time-stamp of the receiving node. If the difference is within the pre-established limits the vehicle sending the beacon is considered a neighbour.

Other security threats like the Sybil attack which is explained in the next section or the manipulation of positions of other nodes are prevented by using tamper-proof units that perform cryptographic operations. This way, malicious nodes cannot illegitimately extract keys of other nodes or modify fields in geocast data or control headers.

Despite all the security threats dealt in this paper they do not mention the attack addressed in this chapter which is the selective forwarding attack where a node selectively or randomly prevents the

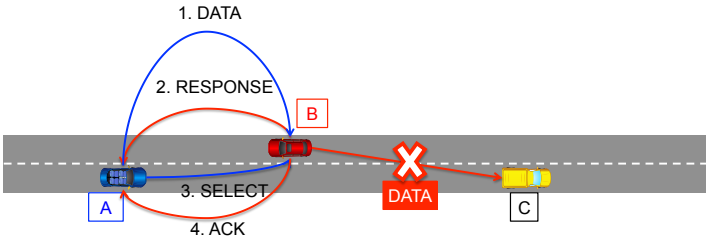


Figure 4.1: Selective forwarding attack performed by vehicle B.

forwarding of certain packets to the destination impairing the performance of the routing protocol.

4.3. Routing-specific Threats

There are different security threats depending on the layer that they are aimed at. Centring our attention to the network layer, attacks like *black hole*, *selective forwarding*, *wormhole* and the likes, are described in the literature [41,42,89,90]. Depending on the messages exchanged by routing protocols, some of them are more vulnerable to these attacks than others. Thus, it is important to analyse in deep the routing protocol behaviour to find the threats that affect it the most.

In BRAVE, the first packet to be transmitted contains the data information, and only nodes that receive this packet participate in the next hop selection mechanism. Hence, a black hole or sink hole attack consisting of a malicious node that silently discards or drops messages without informing the source that the data did not reach its intended recipient will not affect BRAVE at the time of selecting a new neighbour. However, an attacker might participate on the exchange of BRAVE packets and, by retaining the DATA packet and sending back an ACK packet, it can stop forwarding prematurely. Figure 4.1 illustrates this case.

In addition, BRAVE messages are not authenticated nor integrity-protected, enabling other kinds of attacks by a malicious node. Thus, he can manipulate the information stored in the message, for instance by changing the destination of the packet or altering its content. This issue can be alleviated by employing a PKI. In this way, vehicles sign data packets with their private keys and receivers can validate packets by using the public key contained within the digital certificate of the sending vehicle. In the following section, we describe the mechanism employed to exchange these certificates among nodes.

Another sort of attack for which BRAVE is vulnerable is the following: a malicious node can impair BRAVE by proposing itself as the best candidate to forward the packet to the destination when it is not. To do so, it will take advantage of the timer scheduling, by which the node that answers first to the DATA packet is selected as the next hop. So, a malicious node which answers first will be elected as next hop.

Malicious nodes can also harm BRAVE by not issuing the SELECT message once that they have sent the DATA packet. In this way, no neighbour will be selected as relay.

Other more elaborated attacks, like the Sybil attack or wormhole attack, can also be practised within the VANET environment.

In a **Sybil attack**, a malicious node presents multiple identities with different locations to other vehicles in the network. This attack is more sophisticated than the previous ones because, in this

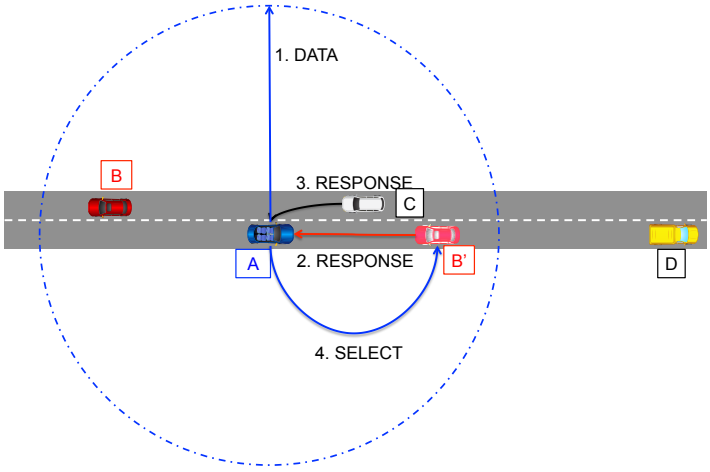


Figure 4.2: Sybil attack

case, the malicious node announces itself also in other locations, taking advantage of these positions to be selected as the best neighbour to forward a packet. For instance, in Figure 4.2 vehicle B creates a new identity B' in a more advantageous location. Hence, it is selected as the best forwarder to the destination.

The only way for a malicious node to create more than one entities is to have more than one pair of public/private keys. There are different alternatives for it, like the use of pseudonyms or installing several certificates within the vehicle. We simplify the problem by forcing a single certificate per vehicle, which is generated by a trusted CA. In such case, the Sybil attack gets reduced to its minimum exponent. That is, a vehicle could forge its position, but could not create multiple identities.

Finally, a **wormhole attack** requires the cooperation of at least two malicious nodes. It consists of two vehicles that create a tunnel between them, so that they can forge their distance to the destination. For instance, if the malicious nodes are far from each other more than one hop, by using the tunnel, for the rest of the neighbours it would be as if there were no distance between them. This attack is harder to perform because of the high variability of links among neighbouring nodes due to the high speed of the vehicles.

4.4. Securing the BRAVE protocol

In this section we develop S-BRAVE, an extension of the BRAVE routing protocol targeted at addressing the security threats that have been previously detailed.

First of all, we will provide authentication and integrity by using a PKI. Thus, the source vehicle signs data packets with its private key and the receiver uses the public key of the sender to check the validity of the packet. Since the receiver node requires the sender's certificate, a previous exchange is needed (introducing extra overhead). We propose a certificate exchange mechanism that tries to reduce the associated overhead. It is described in Section 4.4.1.

Although the use of a PKI is a building block of our protocol, it is not enough to avoid the threats discussed in Section 4.3. Therefore, in Section 4.4.2 we detail the modifications we have done to the behaviour of the original BRAVE protocol to deal with malicious nodes. Such additions include the

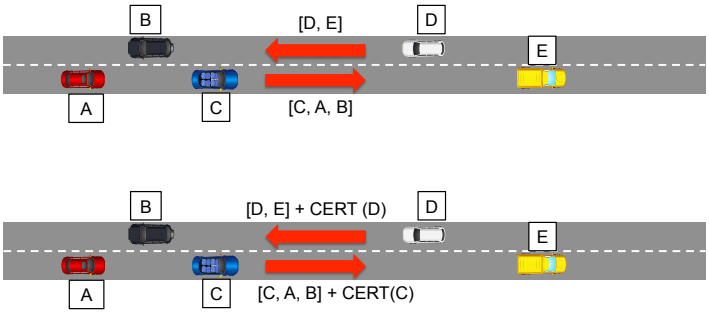


Figure 4.3: Certificate exchange via periodic beacons.

adoption of the mechanism of guard nodes.

4.4.1. Certificate exchange

Since a VANET is a distributed environment, vehicles must trust each other somehow. They cannot always access to the infrastructure to check the validity of the messages. Therefore, the most appropriate way to check the authenticity is to sign them with a private key, allowing other nodes to gather the corresponding public key to verify that signature. In S-BRAVE we assume a unique CA which is the same for all the vehicles in the VANET.

Thanks to the PKI, each vehicle owns a unique identifier and a pair of keys (public and private) as well as a certificate issued by the CA. So, if nodes have installed the public key of the CA they will be able to check the validity of other vehicles' certificates locally without asking to a third-party.

The first problem to address is that of exchanging certificates among vehicles. Every time a node receives a message it must have the sender's certificate in order to authenticate it and to check the integrity of the message. One possible choice is to include the certificate in every data message. However, this is not efficient because certificates have a big size and they would increase the overhead.

We propose a reactive certificate exchange method which minimizes the number of certificate exchanges. Every beacon sent includes a cache of known neighbours identifiers, being a known neighbour the one whose certificate is stored within the vehicle.

When a vehicle receives this beacon, it will be able to determine if its certificate is present in the cache of the neighbour just by looking for its own identifier in the neighbour list. If the certificate identifier is not present, then the vehicle will include its own certificate in the next beacon round. In case the cache is full the oldest element is removed allowing a new insertion.

Using this strategy only the first beacon will include the certificate, the following messages between those vehicles will not need to include certificates for validation. Besides, other nodes that receive a beacon with the certificate can take advantage of this exchange method to store the certificate for a possible use in the future. Figure 4.3 shows this exchange of messages.

Given that certificates are exchanged in advance, it is possible to authenticate routing messages (RESPONSE, SELECT and ACK) by only using digital signatures. However, in order to check the validity of a DATA message, a vehicle located farther than one hop of the sender needs a mechanism to get the certificate of the source. The reason is that DATA messages are signed by the source, but not by intermediate relays.

Our proposal to solve this problem is based on modifying RESPONSE and SELECT messages. A

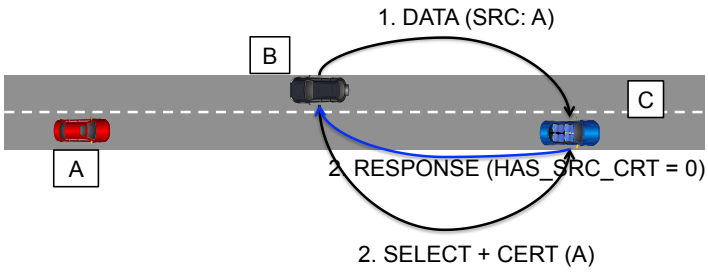


Figure 4.4: Certificate exchange of the source vehicle.

bit included in the RESPONSE message will indicate if the responding vehicle needs the certificate of the source. After receiving this RESPONSE, the SELECT message will be extended with the certificate of the source node depending on this bit (see Figure 4.4). This protocol modification entails an overhead decrease mainly when the path between source and destination is stable.

The CA is able to add malicious nodes to a Certificate Revocation List (CRL), so that their messages become invalid for the other vehicles. Thus, when a node detects a malicious node it can notify the CA to update its CRL by using an infrastructure network. This way, after receiving the updated CRL, nodes will discard the packets whose source node matches one of those identities listed in the CRL. As commented in the introduction of this paper, the issues related to distributing and updating CRLs are discussed in our previous work [82].

4.4.2. S-BRAVE operation

The certificate exchange scheme described before is a basic building block of our solution. However, BRAVE is still weak against a selective forwarding attack that can be accomplished in two ways. In the first one, a malicious node does not continue the forwarding of the DATA message. However, it answers its previous hop with an ACK message making it believe that it has forwarded it. In the second one, the malicious node does not send the SELECT message. For instance, if a node starts the exchange of messages but it does not send the SELECT message, there will not be a forwarding node and therefore the message will not be forwarded. In both cases, the previous hop may think that the forwarding was completed.

In order to avoid this type of situation, S-BRAVE employs the concept of watchdog nodes or guard nodes in the following way. Every neighbouring vehicle that provides advance to the destination will act as a guard node. Those vehicles not selected as the next forwarder will try to ensure that the whole DATA forwarding process is completed. They keep on listening to the next forwarder, checking whether it retransmits the DATA message. If a guard node does not receive this message, it will take the role of the next forwarder by taking the responsibility of sending the DATA message to the next hop. They also include the detected malicious vehicle in a black list, to avoid that it gets selected as the next forwarder in the future.

We have modified the original BRAVE protocol as indicated next.

First of all, we have modified the ACK message. A new bit has been added, which indicates the reason why this ACK has been sent. Thus, a vehicle can send this message by two reasons: DATA message has already been forwarded previously, or it has been buffered by the vehicle because it did not have any neighbours which provided advance towards the destination.

In addition, we have also defined a black list in which neighbours which do not forward messages are registered. This mechanism is used to avoid a malicious node to continuously impair the performance of the protocol by being selected by the same node one time after another. Thus, guard nodes will ignore the messages coming from a node of the black list. For instance, after a node sends an answer with a RESPONSE message, the neighbours that have this node into their black list, will also send their RESPONSE message instead of cancelling their timers. Besides, the sender of the DATA message will also ignore the RESPONSE of a node if its identifier is stored in the black list.

Finally, neighbouring vehicles that receive a RESPONSE or SELECT message do not go back to the initial state. Instead, they will keep the DATA message just received, awaiting for the right exchange of messages and the subsequent DATA message forwarding by the selected node. They also schedule a timer that waits for this exchange to happen within a period of time, otherwise guard nodes will come to the conclusion that a malicious node is attacking by preventing the packet from being delivered. In such case, they collaborate to forward the DATA packet.

In the following, we detail S-BRAVE and provide some pieces of pseudo-code of the most relevant operations that must be performed.

The sender vehicle, after issuing a DATA packet, schedules a timer waiting for responses from neighbouring nodes (*awaitingRESPONSE*). This packet is the one that triggers the next hop selection. Procedures 3, 4, 5 and 6 deal with the main message exchanges of S-BRAVE. In addition, Procedure 7 defines what vehicles do after their timers expire.

Procedure 3 processDATA (m:message, src:address, dst:address)

```
1: if (noActiveTimers) then {Node receives DATA in initial state}
2:    if (dst == ownAddress) then {Node is the destination of DATA}
3:       send(RESPONSE);
4:       scheduleTimer(awaitingToSELECT);
5:    else if (nodeProvidesAdvanceToDest(dst)) then
6:       scheduleTimer(awaitingToAnswer);
7:    end if
8: else if ((src == selectedNode) && awaitingACK) then
9:    exit; {Next hop, i.e. selectedNode, retransmit the packet}
10: else if (awaitingForwardedMsg) then {guard nodes}
11:    cancelTimer(awaitingForwardedMsg);
12:    if (nodeProvidesAdvanceToDest(dst)) then
13:       scheduleTimer(awaitingToAnswer);
14:    end if
15: end if
```

In Procedure 3, a vehicle that has received a DATA message can be in two states. The first one is the *idle* state, where the vehicle is at the beginning of processing the DATA message. If the node is the final destination of the packet it will immediately answer with a RESPONSE message, scheduling a timer to receive the SELECT message too. Otherwise, only the vehicle providing advance to the destination will schedule a timer to answer with a RESPONSE message. The receiver can also be in the *awaitingACK* state, meaning that it has nearly finished the exchange of messages but it is expecting the ACK message. After receiving the ACK, the node would go back to the *idle* state. Finally, if the node is a guard node and receives this DATA message it will cancel its timer of watching the packet, scheduling a new timer that depends on the progress provided with respect to the destination.

Procedure 4 processRESPONSE (m:message, src:address, dst:address)

```
1: if (awaitingRESPONSE && (dst == ownAddress)) then
2:    cancelTimer(awaitingRESPONSE);
3:    send(SELECT, src);
4:    selectedNode ⇐ src;
5:    scheduleTimer(awaitingACK);
6: else if (awaitingToAnswer) then
7:    cancelTimer(awaitingToAnswer);
8:    scheduleTimer(awaitingNextForwarderSelected);
9: end if
```

When a vehicle receives a RESPONSE message (Procedure 4), it will send back to the most promising forwarder a SELECT message, also scheduling a new timer. On the other hand, if the vehicle is not the best forwarder, it will schedule a new timer to overhear the messages exchange to act as a guard node.

Procedure 5 processSELECT (m:message, src:address, dst:address)

```
1: if (awaitingSELECT) then
2:    cancelTimer(awaitingSELECT);
3:    if (finalDest == ownAddress) then
4:       send(ACK,src);
5:       scheduleTimer(awaitingPostProc);
6:    else if (dst == ownAddress) then
7:       if (noneighbours) then
8:          send(ACK); {It buffers the DATA}
9:       else
10:         send(DATA);
11:         scheduleTimer(awaitingRESPONSE);
12:      end if
13:   else
14:      scheduleTimer(awaitingForwardedMsg);
15:   end if
16: else if (awaitingToAnswer) then
17:   cancelTimer(awaitingToAnswer);
18:   scheduleTimer(awaitingForwardedMsg);
19: else if (awaitingNextForwarderSelected) then
20:   cancelTimer(awaitingNextForwarderSelected);
21:   scheduleTimer(awaitingForwardedMsg);
22: end if
```

Procedure 5 describes what happens when a vehicle receives a SELECT message. If it has already sent a RESPONSE message, it will be selected as the next forwarder. Thus, it will cancel its waiting timer (*awaitingSelect*). In case the vehicle is the final destination, it will send an ACK message back to the previous hop. Otherwise, it will broadcast the DATA message unless it will not have any neighbours around it. In this latter case, it will store the message in a buffer, answering with an ACK which specifies this. Guard nodes will cancel their timers and will schedule new ones because the messages exchange is being performed correctly.

Procedure 6 describes the ACK reception process. If the vehicle that receives the ACK is the sender,

Procedure 6 processACK (m:message, src:address, dst:address)

```
1: if (awaitingACK) then
2:    cancelTimer(awaitingACK);
3:    exit;{Node goes back to initial state}
4: else if awaitingForwardedMsg then
5:    if (m.reason == Forwarded) then {reason is an attribute of the message m}
6:       cancelTimer(awaitingForwardedMsg);
7:       send(DATA);
8:       scheduleTimer(awaitingRESPONSE);
9:    else {m.reason == Buffered}
10:      if (noPromisingneighbours) then
11:         buffer(DATA);
12:      else
13:         cancelTimer(awaitingForwardedMsg);
14:         send(DATA);
15:         scheduleTimer(awaitingRESPONSE);
16:      end if
17:   end if
18: end if
```

it will cancel its timer assuming the whole message exchange is completed. On the other hand, guard nodes will analyse the reason for sending this ACK. In case the message indicates a forwarding not heard by them, they will take the role of forwarders by broadcasting the DATA packet.

Procedure 7 timerExpires(timer)

```
1: if (timer == awaitingToAnswer) then
2:    send(RESPONSE);
3:    scheduleTimer(awaitingSELECT);
4: else if (timer == awaitingPostProc) then
5:    exit;{Node goes back to initial state}
6: else if (timer == awaitingNextForwarderSelected) then
7:    send(DATA);
8:    scheduleTimer(awaitingRESPONSE);
9: else if (timer == awaitingForwardedMsg) then
10:   send(DATA);
11:   scheduleTimer(awaitingRESPONSE);
12: end if
```

In Procedure 7, if the vehicle state is *awaitingToAnswer*, it will send a RESPONSE message. This is the case where the vehicle has received the DATA packet and has scheduled a timer to answer to it. On the other hand, guard nodes (the last two cases) will take the role of new forwarders by broadcasting the DATA message.

In the next section, we analyse possible attacks and how our proposed solution behaves against them.

4.4.3. Threat analysis

Let us start with a way of selective forwarding attack in which a malicious node does not forward the DATA message as depicted in Fig. 4.4.3. The neighbours which provide advance to the desti-

nation, after receiving the DATA packet, will trigger a timer before sending their response (line 6, Procedure 3). The one which provides the highest progress towards the destination will answer first with a RESPONSE message (line 1, Procedure 7). However, the rest of these neighbours schedule a new timer waiting for a vehicle to be selected as the next forwarder (line 1, Procedure 4). After receiving this SELECT message, guard nodes cancel their timer, scheduling a new one to be sure that this new forwarder will deliver the message to the next hop (lines 19-22, Procedure 5). If any of the aforementioned timers expire, guard nodes will assume that the vehicle selected to forward the message is a malicious one. Hence, they will select themselves as new forwarders, taking the responsibility of sending the message to the next hop (line 6-12, Procedure 7). Not all timers expire at the same time. The vehicle whose timer expires first will start sending the DATA message. In order to reduce the overhead of the protocol, the other guard nodes will cancel the sending of this DATA when they overhear the DATA from another guard node. Depending on their relative positions to this new forwarder, the neighbours providing advance will act as guard nodes, scheduling a new timer, or just going back to the initial state.

On the other hand, if the malicious node replies with an ACK message as presented in Fig. 4.4.3, it must also insert an additional field called *reason* with two possible values (forwarded and buffered). If the guard nodes receive an ACK message with the **reason** of *message already forwarded* they will react by sending the DATA message (line 5, Procedure 6). However, if they receive an ACK with the *message buffered reason*, they will check their neighbour list to have an idea of how many neighbours there are around. If there are any other neighbours providing advance to the destination apart from the vehicle which sent the ACK, they will take the responsibility of sending the DATA message. In this process, the aforementioned mechanism to reduce protocol overhead by overhearing DATA transmissions takes place. Otherwise, guard nodes will buffer the packet until new neighbours come close to them (line 9, Procedure 6).

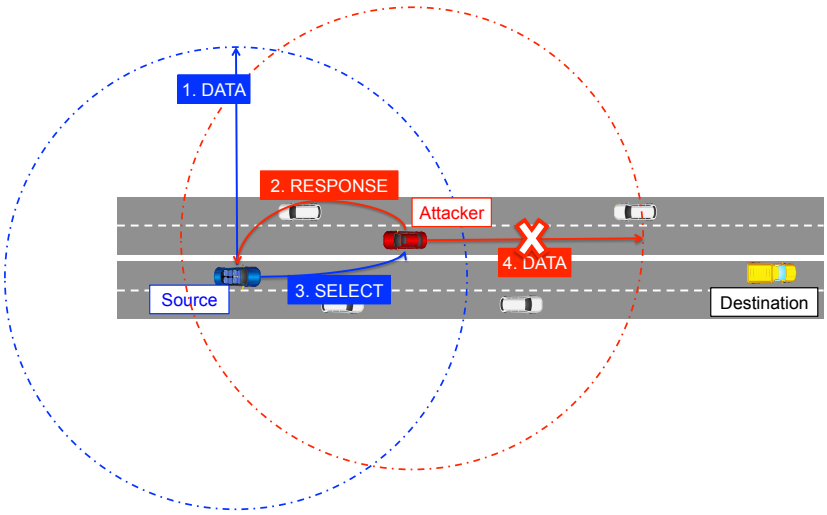
Thus, S-BRAVE is able to deal with the selective forwarding attack as well as providing integrity and authenticity to the messages. Any attacker can pretend to be the best forwarding node but the attack will not be successful if there are surrounding guard nodes. Packet identifiers are unique, so although more than one guard node would detect the attacker and therefore would forward the packet, duplicate packets will converge in the next hops. So, they can be detected and avoided.

Another variety of the selective forwarding attack occurs when a malicious node that has been selected as next forwarder goes on with the entire process to deliver the DATA to the next hop, but it does not allow any other neighbour to be selected by not sending a SELECT message. Thus, no other neighbour will receive the confirmation to forward the packet to the destination. S-BRAVE, in order to counter-attack this situation, uses the *awaitingNextForwarderSelected* and *awaitingForwardedMsg* timers. Thus, when these timers expire, a guard node will take the role of new forwarder starting the protocol to deliver the message to the next hop (line 6 and 10, Procedure 7).

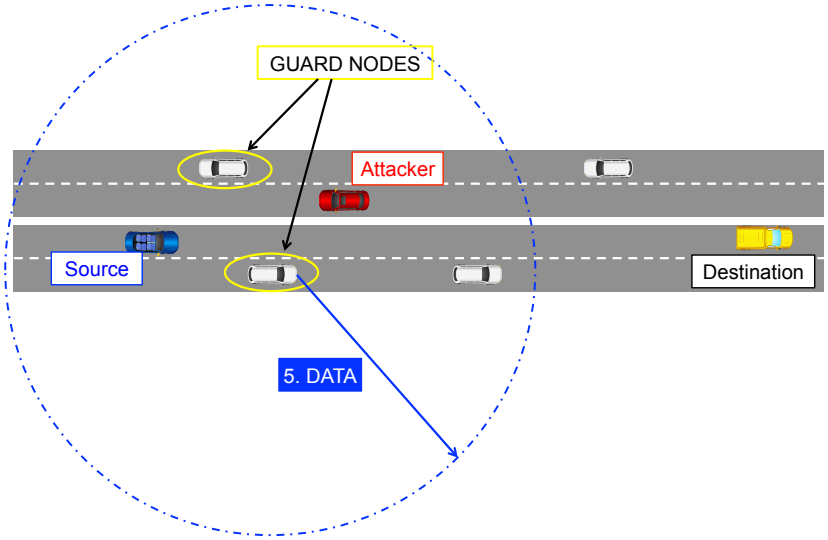
4.5. Performance Evaluation

We have compared both protocols BRAVE and S-BRAVE within the Network Simulator ns-2, version 2.33[1]. We consider a $5 \times 4km^2$ scenario which consists of the main access roads and streets of the city center of Murcia, Spain (Figure 4.7). It contains 53 streets and 28 junctions. This map, as

[1] http://www.isi.edu/nsnam/ns/

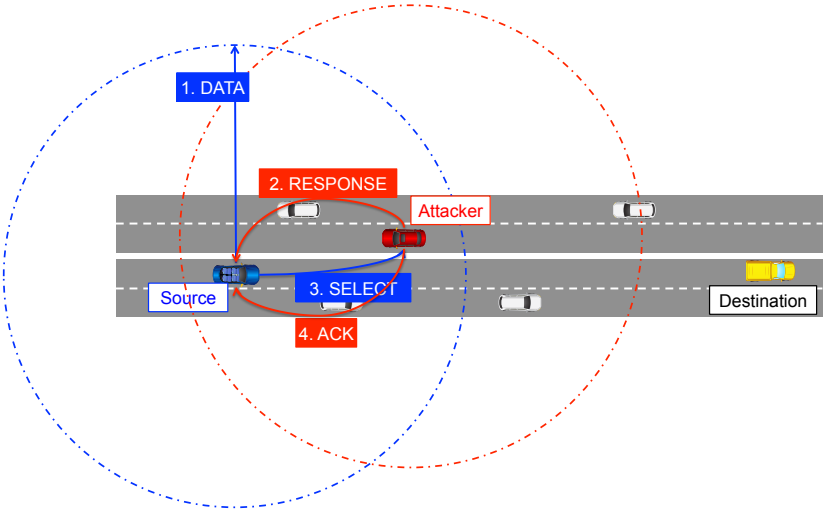


(a) Attacker does not forward DATA packet.

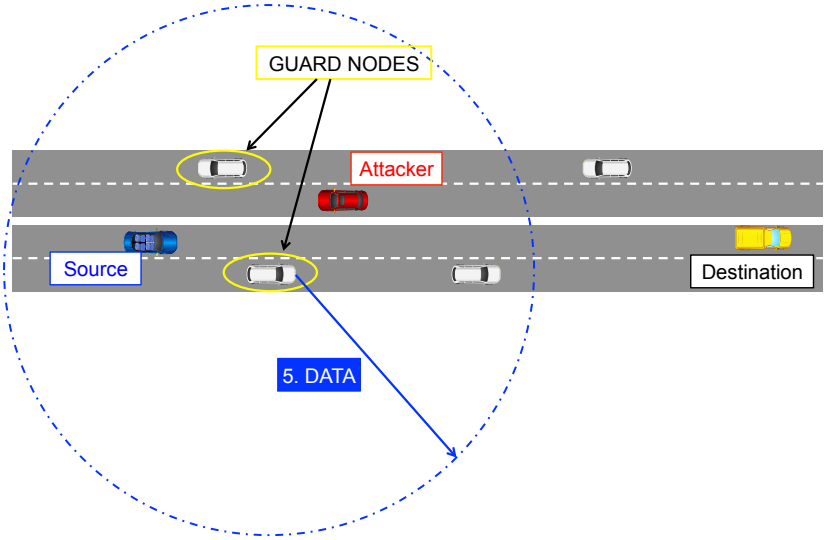


(b) Guard nodes do it.

Figure 4.5: First example of selective forwarding attack.



(a) Attacker responds with an ACK message.



(b) Guard react forwarding the DATA packet.

Figure 4.6: Second example of selective forwarding attack.

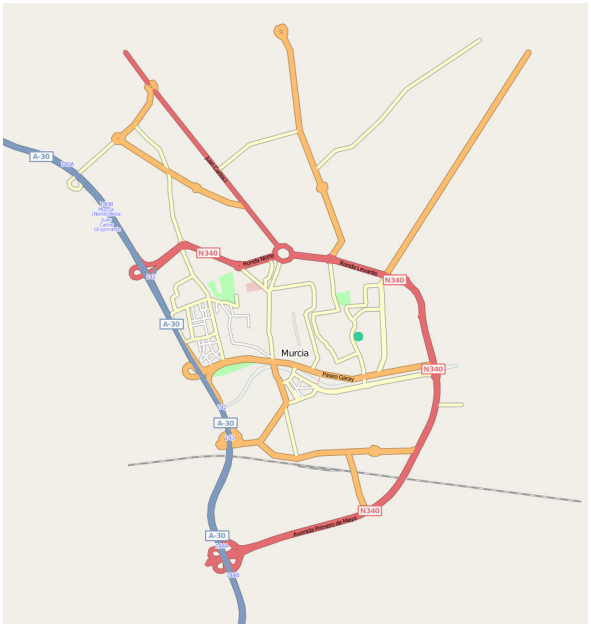


Figure 4.7: Map of Murcia city center and access roads used in our simulations.

well as the vehicular mobility patterns, have been generated by means of the well-known SUMO road traffic simulator[2].

Vehicles move through 20 predefined routes at a maximum speed of 13.8 m/s inside the city, and 22.2 m/s on the highway that crosses the scenario during 885 seconds. The routes followed by the vehicles have been selected according to realistic situations. We have also considered a wide range of traffic densities. Vehicles are injected into their routes at a certain traffic rate. This rate is varied from 1/45 to 1/15 vehicles per route per second. Thus, a $1/x$ rate means that each x seconds a new vehicle is injected into its route.

In our simulations, wireless signals propagate according to the two-ray-ground model. Vehicles carry out their communications via an 802.11p interface card, implementing the enhanced ns-2 802.11 physical and MAC models [91]. The transmission power is adjusted to allow a maximum transmission range of 250 meters. Within this scenario we have simulated 10 runs for each configuration, each of them with different traffic sources randomly selected for this purpose. Therefore, figures in this section show the average of such runs along with their corresponding 95% confidence intervals.

4.5.1. BRAVE vs S-BRAVE

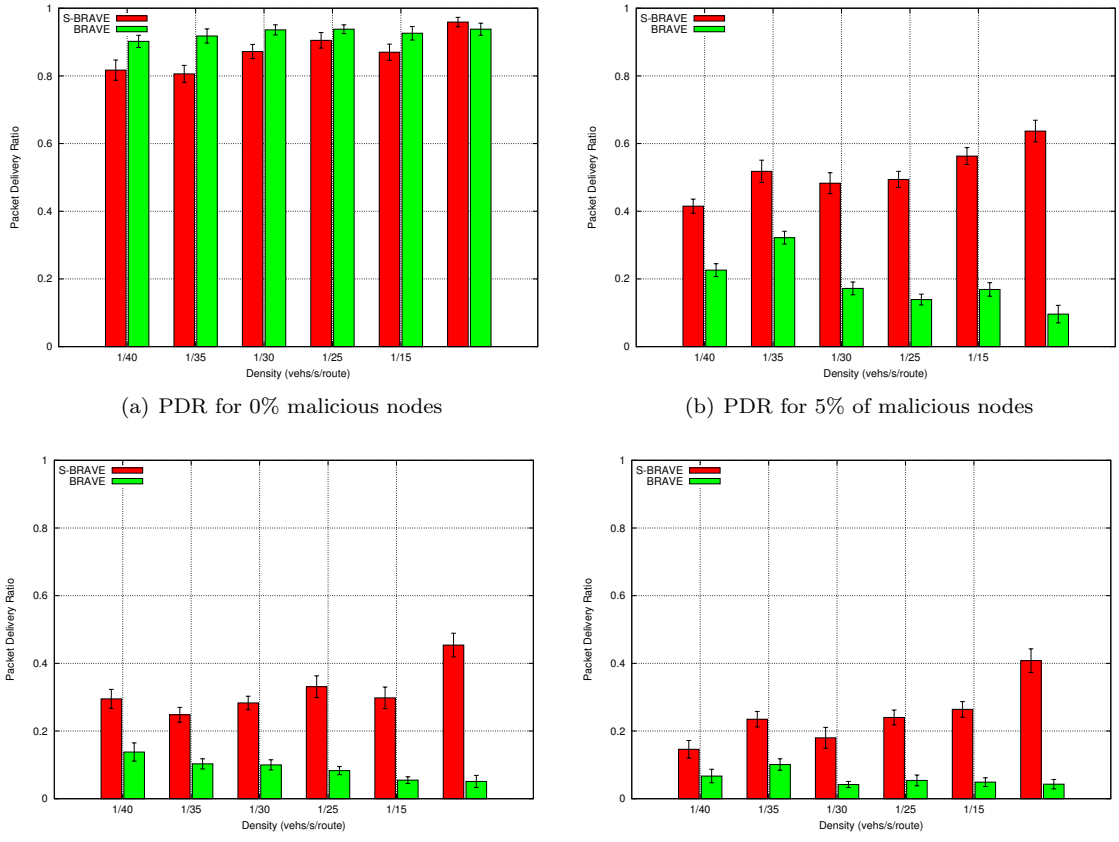
We have compared both protocols for a varying percentage (0%, 5%, 10% and 15% of the total number of vehicles) of malicious nodes that randomly apply one of the two modes of the selective forwarding attack already explained. Regarding the cache of known neighbours, for our simulation we have used an unlimited cache because for the time being OBUs do not have any storage limitations. In this cache, the vehicle will store all the neighbours met along the road. This way, if a vehicle meets

[2]http://sumo.sourceforge.net/

an old neighbour it will already have stored its certificate, avoiding a new message exchange to obtain it reducing the overload.

The firsts results presented in Fig. 4.5.1, correspond to the packet delivery ratio (PDR) obtained for both secure and non-secure routing protocols for different percentages (0%, 5%, 10% and 15%) of malicious nodes in the network.

The PDR of BRAVE is not new because it was provided in Chapter 3 where we analysed its performance in detail. Its delivery rate when no malicious nodes are present is about 90% as Fig. 4.8(a) shows. The performance of S-BRAVE approaches the obtained by BRAVE but it loses up to a 5% in sparse scenarios. This is caused by the false positives occurred during the simulation and their corresponding overhead. During the simulation, guard nodes watching the packets to be forwarded do not receive the forwarded message, making the decision of being themselves the new forwarders. However, the denser the scenario the better performance is obtained from S-BRAVE, reaching the same results as BRAVE.



(a) PDR for 0% malicious nodes

(b) PDR for 5% of malicious nodes

(c) PDR for 10% of malicious nodes

(d) PDR for 15% of malicious nodes

Figure 4.8: Percentage of PDR for 0% , 5%, 10% and 15% of malicious nodes.

The selective forwarding attack is a harmful one in light of the graphs corresponding to the different percentage of malicious nodes. Only with a 5% of malicious nodes, BRAVE is severely affected reducing its performance down to a maximum of 30% of the PDR in a sparse scenario, $1/35\ vehs/route/s$. From this point on, the performance decreases delivering only nearly the 10% of the packets to the

destination for the highest density. On the other hand, the use of guard nodes in S-BRAVE allows maintaining a PDR between the 40%, for the lowest density, and a 65% with the highest density outperforming BRAVE in at least a 20%.

As the percentage of malicious nodes is increased, the performance of both protocols is deteriorated Figures 4.8(c) and 4.8(d) reveal. Despite this, S-BRAVE outperforms BRAVE up to 40% in the former and 35% in the latter. In these scenarios BRAVE is only able to deliver up to 13% and 10% of packets with 10% and 15% of malicious nodes in the best case in sparse scenarios reaching in dense scenarios a 5% or 10% of the packets.

The reason for the routing protocols to be so sensible to selective forwarding attacks in sparse scenarios is the following. In sparse scenarios, the probability of finding more than a neighbour to be selected as the next forwarder is really low. So, the sending node usually finds one node at much which could be selected as a next forwarder. This way, if the node is an attacker and there are not other neighbours acting as guard nodes, the attack will succeed impairing both routing protocols. As the density increases, the technique of using guard nodes becomes more and more effective.

Regarding the end-to-end delay, Fig. 4.5.1 presents the delay for the different simulated densities with a 5%, 10% and 15% of attackers, respectively. As the percentage of malicious nodes increases, it is more difficult to find nodes which can act as guards, hindering the routing process due to the lack of nodes that take the responsibility of delivering the packets. Therefore, vehicles reach a local optimum more often having no trustable neighbour that provides advance. In that situation, vehicles make the decision of storing the packet until they find a suitable neighbour. This is the reason why as the percentage of attackers is increased the delay also increases.

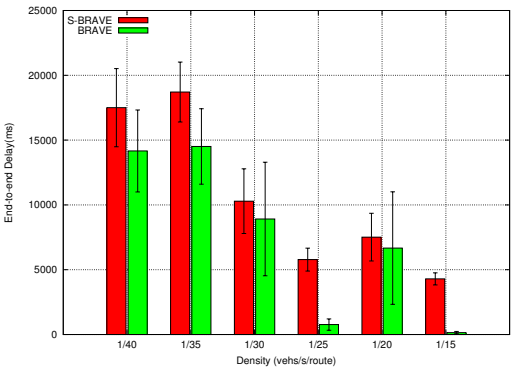
Despite these facts, BRAVE usually obtains a lower delay compared with S-BRAVE in high density scenarios. Looking back at PDR graphs, BRAVE for those densities was hardly able to deliver up to 20% of the packets. Matching these results, we can deduce that BRAVE is only able to deliver packets if the senders are near the destinations. If they are distant, it is more likely that it reaches a malicious node that drops the message.

In order to provide more insight onto the performance of the protocols, we also compute the overhead per delivered message, the overhead per delivered message per hop, and the number of delivered packets against the number of hops they have gone through.

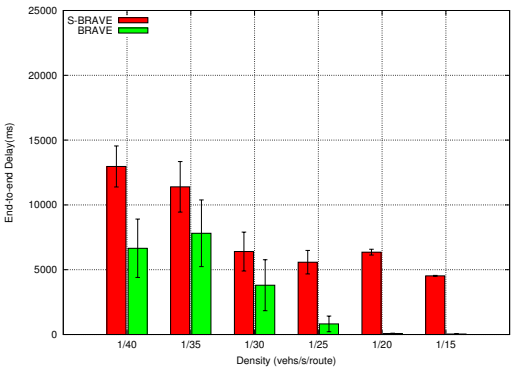
As expected, S-BRAVE has more overhead per successful delivery than BRAVE (Figure 4.10(a)). In fact, S-BRAVE sends certificates when needed while BRAVE does not use them. However, if we consider that overhead per successful delivery per hop we can see (Figure 4.10(b)) that S-BRAVE only adds little overhead compared to BRAVE, despite the need of certificates. The reason for the higher overhead in Figure 4.10(a) is that S-BRAVE manages to deliver packets to destination which are located far from the source (# of hops), while BRAVE just can not do it.

This is corroborated by Figure 4.11, where we have related the number of delivered packets by both protocols with the distance (in hops) they are able to reach. For this purpose we have contemplated, not the average value of the 10 runs of the scenario but the total amount of delivered packets of these runs. In light of these results, we can see that S-BRAVE delivers more messages than BRAVE, and it manages to deliver them even to vehicles located many hops away from the source.

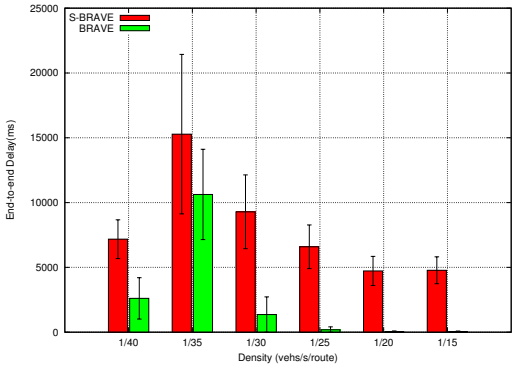
Despite the gap between BRAVE and S-BRAVE regarding the PDR, S-BRAVE still experiments up to a 40% of yield loss in lowly dense scenarios where a vehicle has few neighbours able to forward the messages towards the destination, compared with a scenario without malicious nodes. This means



(a) 5% of malicious nodes



(b) 10% of malicious nodes



(c) 15% of malicious nodes

Figure 4.9: Delay for 5%, 10% and 15% of malicious nodes.

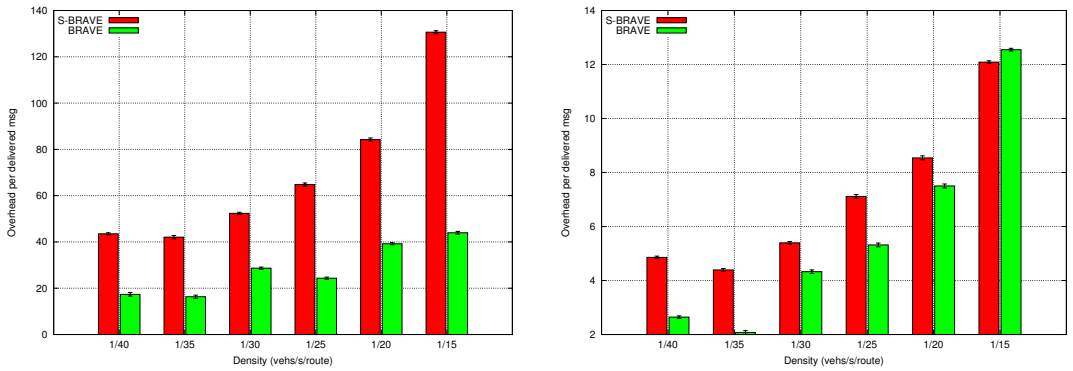
that with just these mechanisms, we are still far from a secure routing protocol able to countermeasure every attack of malicious nodes.

4.6. Conclusion

In this chapter we analyse the problem of secure routing in VANET. In particular, we focus on the BRAVE routing protocol, which is one of the best performing proposals so far. This is a challenging problem because of the high mobility of the vehicles as well as because BRAVE also uses the store-carry-and-forward paradigm to make nodes act as ferries for the packets if there are no promising neighbours around the vehicle.

For this purpose we have introduced a certificate exchange mechanism guaranteeing the authenticity and integrity of the messages as they traverse intermediate nodes until they reach their destination. Besides, we have also developed a way of securing BRAVE against selective forwarding attacks using neighbouring nodes as guard nodes. They watch for the message to be sent by the next forwarder and, in case this vehicle does not forward the message, they take the responsibility of sending the message to the next hop.

In order to compare both protocols we have implemented them in NS-2. In light of the results of the previous section. S-BRAVE outperforms BRAVE in terms of PDR. In spite of it, S-BRAVE is



(a) Overhead of both protocols for 5% malicious nodes per delivered msg.

(b) Overhead of both protocols for 5% malicious nodes per delivered msg per hop.

Figure 4.10: Overhead (number of BRAVE messages) per hop.

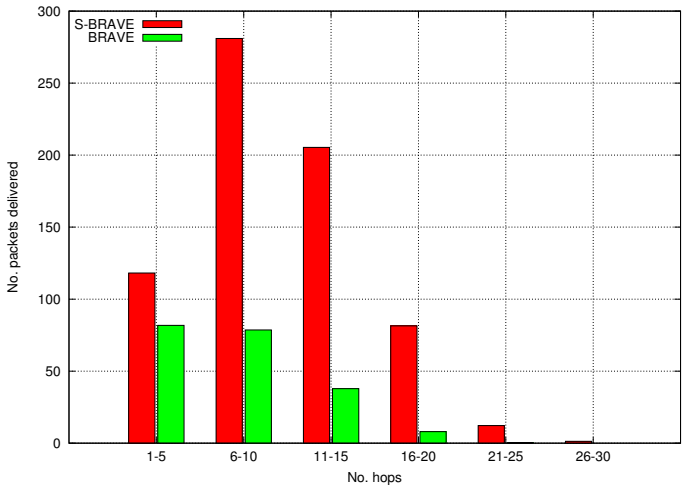


Figure 4.11: No. of delivered packets vs the distance both protocols can reach.

still far from a secure protocol where malicious nodes are not able to affect it. In low dense scenarios, S-BRAVE routing task is very arduous due to lack of neighbours. On the other hand, in very dense scenarios, its performance gap compared with BRAVE is up to a 50% of the PDR. In addition, this enhancement of performance is achieved without significantly increasing the delay nor the overhead. However, S-BRAVE still suffers from the influence of the attackers obtaining a maximum PDR of about 50%.

Chapter 5

Multi-hop routing in hybrid VANETs

In Chapter 3 we assumed vehicles in VANETs were only equipped with 802.11p wireless interfaces allowing them to communicate with neighbouring vehicles as they were in their transmission range. This way, BRAVE, by using vehicles as forwarding nodes is able to provide communication among nodes located far away ones from the others. Despite having obtained such good performance in terms of packet delivery ratio with our previous proposal, the set out model does not benefit from all the existent elements deployed along highways and in most of urban environments.

The concern of public traffic authorities regarding safety in our roads made them deploy traffic panels and surveillance cameras, besides numerous sensors to monitor their traffic state as well as providing real-time information to drivers as they passed by the aforementioned traffic panels. These elements which we are also referred as Roadside Units (RSUs), are connected to the infrastructure network in order to transmit information from and to the central servers. In addition, they can be equipped with wireless capabilities allowing them to communicate with the vehicles of the road.

On the other hand, thanks to the lack of energy consumption requirements that vehicles have, they can be equipped not only with powerful OBUs, but they can also incorporate different wireless technologies like cellular interfaces like UMTS or LTE, WiMAX, apart from the standard 802.11p interface. All these elements, together with other free WiFi APs provided by different private companies which can be deployed close to streets and in rest areas of highways, made the research community to set out a more enriched operating model for VANET-specific multi-hop routing protocols.

This model consists of taking advantage of RSUs and APs which have the ability to act as gateways to the infrastructure and the other recently mentioned wireless technologies to improve the performance of the routing protocols by deriving part of the traffic of the vehicular network to the infrastructure.

This is precisely our objective in this chapter. Here, we propose a routing solution being able to: (i) deal with different sort of wireless interfaces, prioritizing them to obtain the best performance in terms of delay and delivery ratio; and (ii) using the RSUs and APs, even when nodes do not have direct connection to them, to take advantage of the infrastructure network to provide short-cuts to the destination when that particular path is better than a complete multi-hop path through vehicles.

Our simulation results under a mixed urban and inter-urban scenario with a variety of vehicle

densities and RSUs densities show that the proposed routing protocol is able to benefit by further reducing the end-to-end delay, and increasing the packet delivery ratio.

5.1. Introduction and motivation

In Chapter 3, we motivated the use of multi-hop routing in VANETs as well as their main objective of making our roads safer. These networks on their own are able to hand over packets even nodes located far away from the source nodes, but in exchange a great effort is needed from the side of repair and recovery strategies. Actually, the more distant are source and destination, the less performance is obtained. Our previous routing solution (BRAVE) is able to obtain high packet delivery ratio even for distant communications, but this result is accompanied by an end-to-end delay which depending on the application could be acceptable or not.

There also exist many applications which rely on multi-hop routing to transmit information among the vehicles of a VANET. The first ones were safety related applications because of the high importance of reducing the lives lost in road accidents. Others were aimed at improving the driving quality by warning the drivers about traffic jams advising also possible detours to avoid them, or informing the driver about price lists of petrol stations. Finally, a latter group of applications were aimed at entertaining both passengers and drivers by allowing them to download or play music or films, watch the news and the likes connected to the Internet. Most of these applications requires Internet connectivity to provide such information to drivers and passengers.

The advantage provided by vehicles with respect to nodes of other sort of ad hoc networks such as sensors or any other mobile device is the duration of their energy source. They use batteries which can last for several years so they do not have energy constraints. So, they can be equipped with different wireless technologies like UMTS, LTE, or WiMAX among others which let them obtain such connection to the infrastructure.

On the other hand, most of the streets of our cities as well as the majority of our highways and motorways have been enriched with elements aimed at controlling and monitoring the traffic state. Among these elements we can enumerate the following: surveillance cameras, detectors, radars, and traffic panels. These elements also called RSUs could also serve as gateways to the infrastructure network if they incorporate wireless capabilities, creating this way a hybrid VANET, Fig. 5.1.

These hybrid networks are really attractive from different points of view due to the different value-added services they can provide. From the safety point of view, vehicles can receive real-time traffic information allowing them to take a different path in case of traffic jam, to react faster in case of a traffic accident avoiding a more dangerous one. On the other hand, From the point of view of entertainment, vehicles could share media content or request it to the infrastructure. This is the reason why a hybrid routing protocol is needed. Delivering packets to their destination is its responsibility.

The remainder of this chapter is organized as follows: In section 5.2 we describe the different hybrid routing proposals found in the literature. We present our hybrid VANET architecture in Section 5.3. Our hybrid routing protocol is developed in Section 5.4 detailing important elements like the use of different network interfaces or the RSUs to increase the packet delivery ratio. In Section 5.5, we evaluate the performance of our proposal by means of simulations. Finally Section 5.6 concludes the chapter.

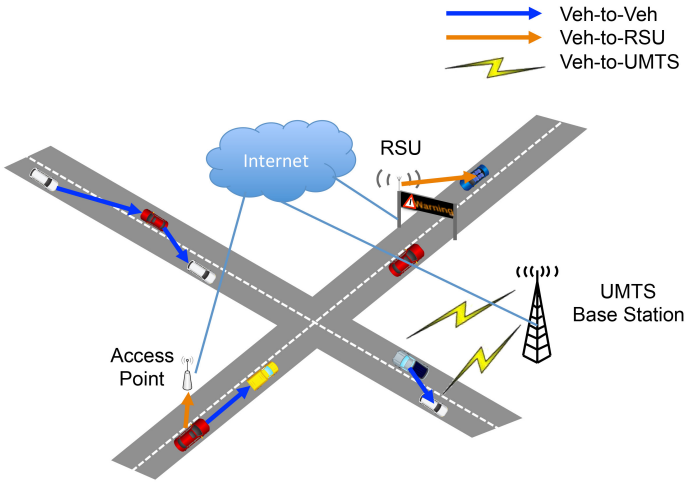


Figure 5.1: Hybrid routing example.

5.2. Related work

The design of our proposal coincided with the first and most active period of this research field. So, in the literature there were very few proposed solutions for hybrid VANETs because most of the researchers were focused on pure VANETs without infrastructure support.

A previous scheme was proposed by Gerla et al. [92] in 2006. They envisioned a VANET called Vehicular Grid (V-Grid), see Fig. 5.2, with the ubiquitous presence of the infrastructure and the opportunity to take advantage of it. This way, vehicles could communicate with Internet Servers through the Internet Infrastructure.

This architecture has different features and elements compared with the traditional Internet stack: They foresaw the necessity of unique addresses for nodes moving across the network that connect to different APs and they stated that geographical addresses were the most suitable ones for this environment. The challenge was that vehicles move continuously, so these geographical addresses vary as the time passes. To solve this issue, they propose two different geographical location services. One located within the infrastructure called Overlay Location Service (OLS) and a distributed one maintained entirely by the vehicle network in case the first one failed called Vehicle Grid Location Service(VLS).

OLS gathers the vehicle identifiers as they connect to the different APs of the architecture, storing also their geographical location.

They also tackled the problem of routing in V-Grid but at that time only AODV and OLSR were the most known protocols. However, they outlined a mechanism where vehicles after collecting information about the load of the APs of the architecture use this information to select the best path to reach the destination, whether using V2V or V2I.

Another proposal before our work was presented by J. Miller in 2008 [93]. He proposes an ITS architecture to connect vehicles to the infrastructure. To do so, the vehicular network is split into pre-configured zones grouping this way vehicles per zones. Their size is small enough such that two vehicles located at the furthest points of the same zone must be able to communicate with each other.

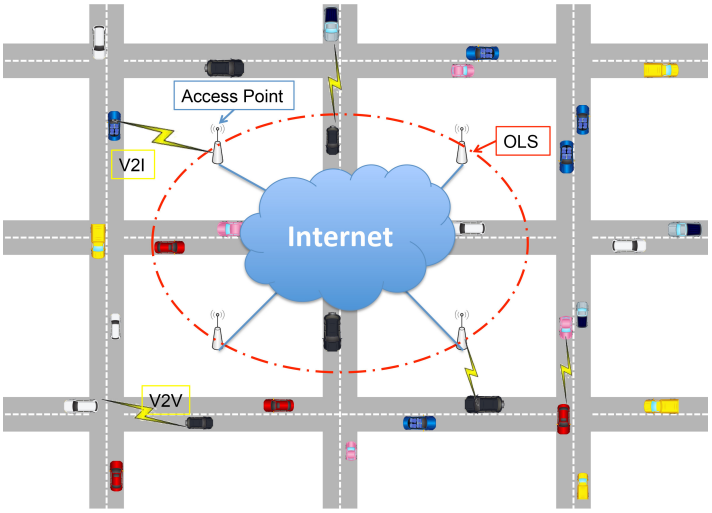


Figure 5.2: V-Grid architecture.

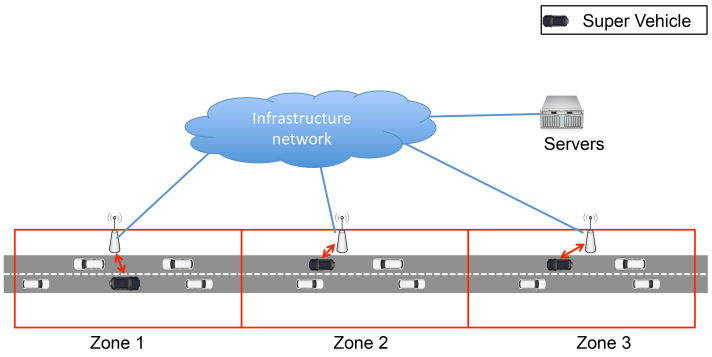


Figure 5.3: V2V2I Architecture.

Among all the vehicles present in each zone, one is elected as a Super Vehicle. This one takes the responsibility of communicating with the infrastructure, as well as with other Super Vehicles. A drawback of this this architecture is that it requires a central server which gathers positions and speeds of vehicles transmitted by the Super Vehicles. This architecture is outlined in Fig. 5.3.

Another drawback of this architecture is that it is only aimed at gathering information from the vehicular network. It does not contemplate the full interconnection between the vehicular network and the infrastructure. That is, neither a vehicle can take advantage of the infrastructure elements to send a message to another vehicle located far from its position, nor the infrastructure can connect to a vehicle of the network.

Hung et al. [94] in 2008 did not only propose an architecture called Heterogeneous Vehicular Network (HVN), but they also propose a routing protocol called Mobility Pattern Aware Routing Protocol (MPARP).

The proposed architecture comprises two elements (Fig. 5.4): vehicles and Base Stations(BSs). Vehicles are equipped with two different wireless interfaces 802.11 and 802.16 (WiMAX) while BSs are only equipped with the latter one which provides a wider transmission range. This way vehicles can

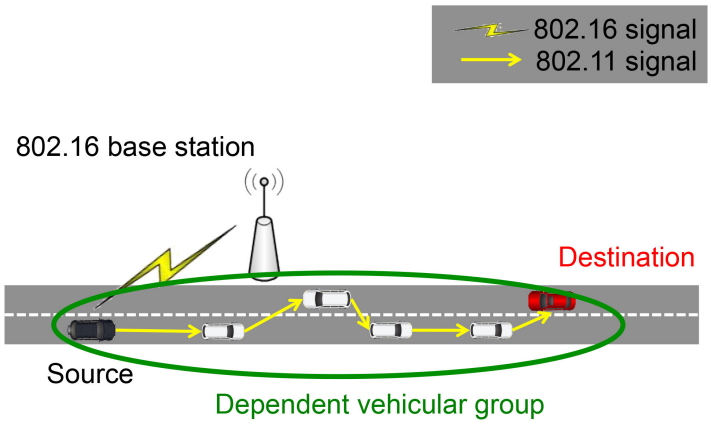


Figure 5.4: Proposed architecture.

communicate with other vehicles using the first interface, and vehicles associated with base stations can also take advantage of them to communicate with farther vehicles using the BSs to forward the packets.

In HVN, BSs work as a centralized location server gathering vehicles positions by receiving their beacons, data or request messages. Thanks to this information, BSs own more precise information than vehicles to make a routing decision. MPARP takes advantage of them as we will see next.

In MPARP, first of all, a source node sends a request message to the BS to know the best way to transmit a data packet to the destination. Since, BSs have the knowledge of the vehicles locations, they are in charge of deciding the best way to forward data packets. Their decision will depend on whether the vehicles in the path towards the destination can build a Dependent Vehicular Group (DVG) or not. If the answer is positive, the BS will answer indicating the source node to send the data via V2V communication. Otherwise, it will request the data to the source node to send it itself through the BS network towards the closest BS to the destination node.

A DVG consists of a group of vehicles with spatial dependency, temporal dependency and a lower relative speed.

This proposal has several drawbacks: First, nodes can only send information if they are associated with a BS, not fulfilling the hybrid VANET requirements. Second, the estimation of DVG can vary if vehicles change the route they are following, so if once the node is transmitting the packet through the vehicles composing the DVG they change their respective routes, the delivery ratio will be affected. And third, the routing protocol introduces a high overhead and also a corresponding delay to request the BS the best way to forward a packet.

Later studies like the one carried out in 2010 by Luo et al. [95] investigated how to improve the performance of routing protocols by using a support network. They claim that by using a bus network as a mobile infrastructure network operating in a different channel makes the routing task more efficient in terms of delivery ratio and throughput. Hence, buses have two different wireless interfaces, one to connect with the other vehicles with a transmission range R_1, and the other one to interconnect the bus network with a wider transmission range R_2 ($R_2 > R_1$) as Fig. 5.5.

Besides, instead of flooding a discovery message to know the position of the destination, the authors assume the use of a location service which has up-to-date information about nodes locations.

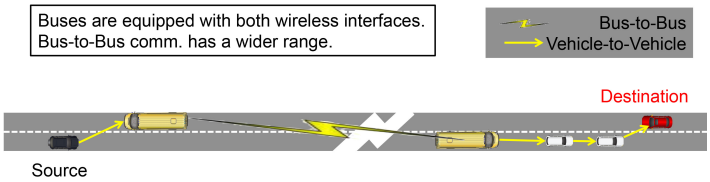


Figure 5.5: Example of urban scenario with a network of two channels.

They propose Mobile Infrastructure Based VANET Routing Protocol (MIBR), a reactive location based reactive routing protocol which also takes advantage of a digital street map and a table with the estimation of buses per road segment for its routing decisions.

The neighbour selection is done according to the information contained in the neighbour table (nbtable) that each vehicle collects thanks to the beacons of vehicles in their transmission range. So when a node using MIBR is going to forward a packet, it follows the next strategy which is called *bus first*:

1. If nbtable contains buses on the segment, then choose the closest to the junction after the next junction. Otherwise choose ordinary car closest to the junction after next.

2. If nbtable contains no vehicles and the packet is currently on a bus, then choose a bus closest to the next junction; else choose a vehicle closest to the next junction.

3. If nbtable contains no vehicles on the next road segment and the packet is now on a car, then choose a bus closest to the next junction. If not available, then choose a vehicle closest to the next junction.

4. If there are no better suitable forwarding nodes, then drop the packet.

With such a network with two different channels to operate with, the performance obtained in terms of delivery ratio is up to a 20% higher than with a network with only one channel. This is something expected because the congestion of the network is distributed in two channels lowering also access to the media.

Wu et al. [96] in 2012 also made a contribution to the hybrid VANETs routing. In this paper, authors analyse their proposed algorithm with two different routing algorithms: one called random walk where a node randomly selects a packet to send it to the next hop, and another called epidemic where a packet has n copies and each one takes a random walk. So none of them tries to deliver a packet towards a destination located far from the source node.

Their proposed algorithm comprises three main components: knowledge fusion, packet forwarding and buffer allocation.

The first one requires vehicles to maintain two different information tables. One with the locations of vehicles and timestamps, and the other with a list of packets carried by the vehicle.

The second component is invoked every time a vehicle encounters another one. When this happens, both vehicles exchange the aforementioned tables and update the delivery probability for all the packets they keep in their buffers as if they were forwarded by the neighbouring vehicle. Afterwards, they sort all the packets with this updated information.

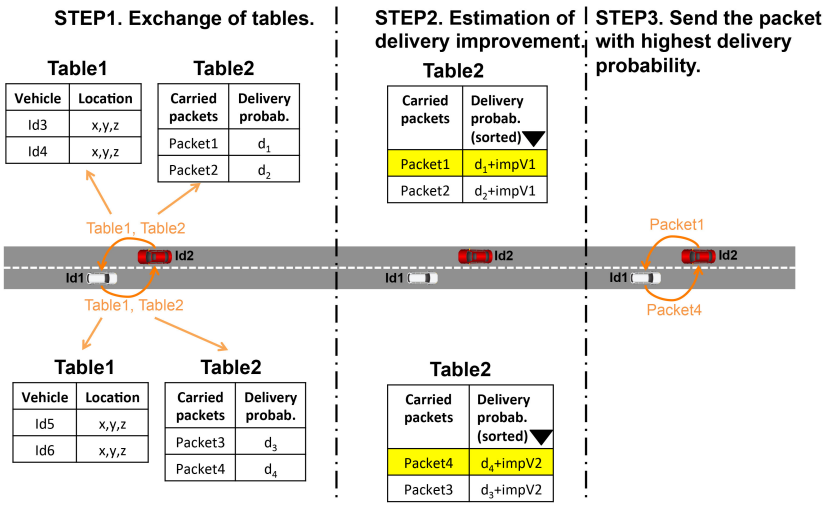


Figure 5.6: Exchange of messages.

Its authors assume that only one transmission per time slot is possible, and only if there is a positive increment in the delivery probability of the packet being forwarded. Otherwise the data packet remains in the current vehicle.

Finally, regarding the last component, APs use the buffer allocation scheme to place multiple copies of the packet in different APs. Since they assume a finite storage space, when new packets are stored in buffers old ones are dropped.

The drawbacks of this proposal are that forwarding only happens when a vehicle encounters another one and also that authors assume that only one packet transmission per slot is allowed. These two features limit the performance in terms of delivery ratio. Actually, their results when no infrastructure support is used is about 50% which increases up to 80% when APs are deployed.

Finally, Vegni and Little [97] in 2011 proposed a hybrid algorithm called **V2X** which is able to switch from V2V to V2I and vice versa. This protocol is based on a total cost function to decide whether the packet will use the infrastructure or not.

This function is based on two physical parameters: The radio resource utilisation time and the time interval needed to transmit the message over a path towards a destination. They also consider a data rate reduction factor associated to the distance from the sending node to the next hop.

With this function, authors estimate the total cost associated to the path from the source node to the destination which is used in determining the best path to follow to reach the destination.

Despite being one of the most complete algorithms, its assessment has only been tested under theoretical conditions. Another drawback of the proposal is the estimation path used by the cost function. Under so highly dynamic networks like VANETs, these paths are maintained during a little period of time, this is also one of the reasons why generic MANET routing protocols do not perform well on VANETs and geographic routing protocols do. Maintaining a path causes a great overload in the network because they usually break and new path re-calculation are required with the consequent increment of messages.

In the next section, we describe our proposed solution to take advantage of the infrastructure enriching the information at making routing decisions and enhancing the performance of the our

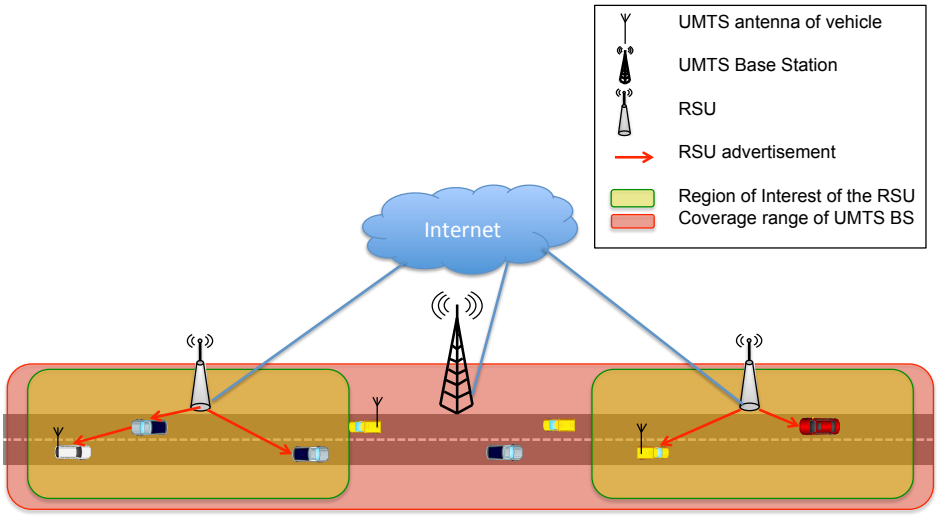


Figure 5.7: Hybrid VANET architecture.

routing protocol presented in the previous chapter.

5.3. Hybrid VANET architecture overview

The architecture set out in this chapter comprises different mobile and fixed elements which participate, in one way or the other, in the communication of the VANET. This architecture can interact with different wireless technologies: either V2V by an 802.11p interface, or vehicle-to-rsu by using the same interface or a WiFi standard 802.11a/b/g/n interface, or V2I by using RSUs as gateways or a cellular interface like GPRS, UMTS or more recently LTE. Fig. 5.7 presents these elements in a reduced vehicular scenario so as to clarify this architecture by an example.

Following a top-down view, firstly we find the cloud figure indicating the infrastructure network with Internet connectivity. Next there are two sort of base stations. The larger one corresponds to a cellular base station, concretely an UMTS, while the smaller ones represents the RSUs which act as gateways to the infrastructure network too. They have direct communication with the vehicles which pass by as they come into their transmission range.

Unlike the UMTS base station whose coverage range is up to $40km$, the transmission range of the RSUs is only up to $1km$. So, vehicles with a cellular interface have direct access. On the other side, RSUs must announce their presence somehow to vehicles farther than their radio range, gateway discovery protocols are responsible for this task.

Although having a nearby RSU alleviates the overload of the VANET increasing its performance in both packet delivery ratio and end-to-end delay., broadcasting the presence of RSUs to the whole VANET represents a huge overhead caused by both the announcement to the entire network, and the data packets of far away nodes aimed at the announced RSU to communicate with the infrastructure.

For this reason the concept of **Region of Interest (RoI)** is defined. The Region of Interest (RoI) is a controlled area where the advertisement of a RSU is broadcast. Its goal is to keep these messages alive in the region allowing vehicles which enter in it to know that a connection to the infrastructure network is possible by using the RSU. In this chapter we analyse the impact of the RoI size in the

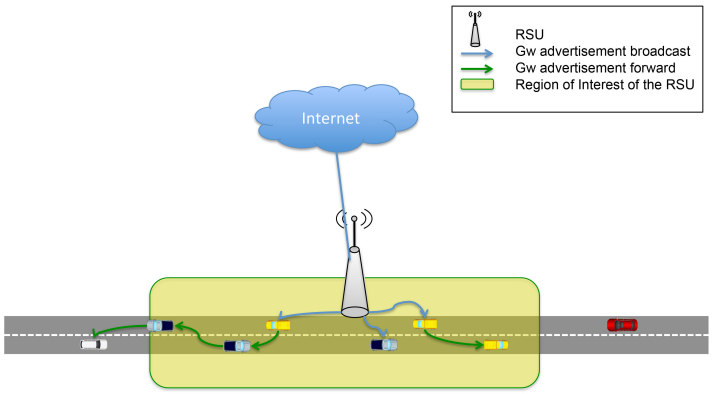


Figure 5.8: Example of Region of Interest.

performance of our routing protocol. Fig. 5.8 offers a deeper view of how it works.

The advertisement of the RSU includes not only its location but also the size of RoI which is usually determined by the two opposite corners of a rectangular area. This information is broadcast within the indicated area. When the message gets out of it, like in the left side of the region, this message is not forwarded. However, the white vehicle could enjoy the benefits of the RSU as it enters the region.

Finally, vehicles have an 802.11p wifi card, but only some of them are equipped with cellular interfaces. Thanks to them, they are able to connect to the infrastructure not only by using the RSU but also directly by this interface with a lower delay and a higher bandwidth.

Besides, vehicles equipped with an UMTS interface could also act as mobile gateways broadcasting their presence and allowing neighbouring nodes to connect to the infrastructure through this particular interface.

5.4. Beacon-less routing for hybrid VANETs

In Chapter 3 we proposed a VANET routing protocol called BRAVE suitable also for DTNs which obtained a notable performance in terms of PDR outperforming the other proposed solutions. In this chapter we extend this routing protocol to take advantage the goodness of connecting the VANET to the infrastructure through different elements as we have seen in the architecture recently presented.

Thanks to it, our hybrid routing protocol is able to take advantage of the infrastructure network to reduce the end-to-end delay when distant nodes communicate with each other. This way, exchanged messages, instead of following a path through intermediate nodes, are transmitted through the infrastructure network in case that is the most appropriate way according to their QoS requirements.

In Fig. 5.9 we present an example of how our proposed solution operates. The *source vehicle* is equipped with an 802.11p and an UMTS interface. Since it has also entered in the RoI of the RSU, it knows that it can also send messages to the infrastructure aiming them at the RSU. So basically, the source node has three ways to transmit information to a destination: (i) it can use the UMTS interface sending the messages directly through the infrastructure network; (ii) it can use V2V communication sending the information through intermediate vehicles until the messages arrive at the destination; or (iii) it can use the nearby RSU to shorten the previous path which is represented by *VirtInt1* which

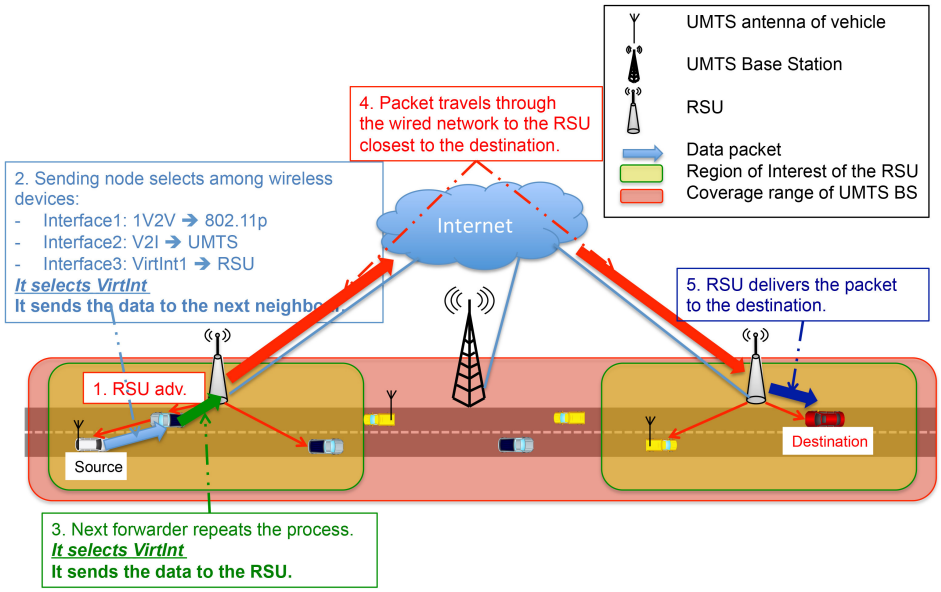


Figure 5.9: BRAVE routing in hybrid VANETs.

is thoroughly explained in the following section.

Therefore, the *source node*, as every intermediate hop, evaluates the best routing decision per data packet being aware of the QoS requirements as well as the cost of using the available interfaces. Among their available interfaces (UMTS, 802.11p and VirtInt1), it must select the best interface to deliver the data packet to the destination. In this example, its decision is that of using the closest RSU to transmit it including this information in the packet.

As we have already commented, the most promising neighbour, when it takes the role of next forwarder does the same evaluation process. In this case, it also comes to the same conclusion which is that of using again the *VirtInt1* sending the data packet to the RSU.

Now it is the turn of the RSU which evaluates the data packet making a new routing decision. Sending the data packet through the wired network to the other RSU which is the closest to the destination.

Finally, the other RSU at receiving the data packet evaluates it again sending it to the destination.

5.4.1. The concept of virtual interface

In the previous example, we exemplified the use of a hybrid VANET by a node that is intended to send some data to a destination located far away from it.

Among the interfaces available for the source node, there was a virtual one called *VirtInt1*. This interface would be selected when the routing decision is sending the traffic through the nearby RSU. Such a routing decision increases the reliability and packet delivery ratio reducing also the end-to-end delay because the packet will traverse the wired network freeing resources from the VANET.

Let us explain what these virtual interfaces are, and how they are managed by our routing proposal.

BRAVE in an isolated VANET uses the street map to establish a temporary destination closer to the position of the forwarding node as a smaller step to be achieved so as to reach the final destination.

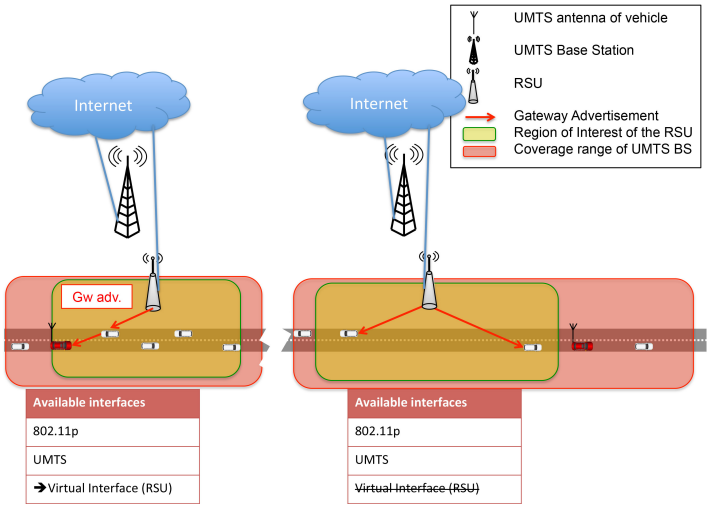


Figure 5.10: Example of BRAVE using virtual interfaces.

A similar approach is followed to aim packets to the RSU before being sent to the final destination. Since the packets of BRAVE are standard IPv6 packets, they can be extended using the hop-by-hop extension header [98]. This one is precisely the one that allows the packet to set a first destination to be aimed at before being transmitted to the final one. Nodes receiving a packet with this extension header, will have two different destination nodes: the final destination of the packet and a new one set in this extension header.

So, when a node receives the advertisement of a nearby RSU, it adds a new virtual interface to the available ones as Fig. 5.10 indicates. If new advertisements are received by other RSUs, new virtual interfaces will be added next.

When a node is intended to transmit information, BRAVE in this hybrid environment includes in its routing decision the available interfaces. If the selected interface is one of the available virtual interfaces, it adds to the data packet the hop-by-hop extension header. This makes the packet to go first towards the selected RSU instead of the final destination as Fig. 5.11 shows.

When *RSU1* receives the packet, it removes its extension header forwarding it through the wired network to the *RSU2* which in turn forwards it to the final destination.

5.4.2. Location service

Although the use of the infrastructure network to shorten the path length, i.e. the number of hops, when transmitting a data packet to a far away destination is a very attractive opportunity, it sets out the problem of finding an RSU near the destination location.

For this purpose a Location Service(LS) is needed. The LS has been also used by routing protocols to obtain the location of vehicles, updating them as they move along the VANET. Depending on the different alternatives proposed in the literature, vehicles periodically update their location like the most of the pure routing solutions, or as they passes by an RSU as we have recently reviewed in Section 5.2 [92–94]. So, among the information contained in the LS we can highlight the identifiers and locations of vehicles as well as a time-stamp when this information was updated.

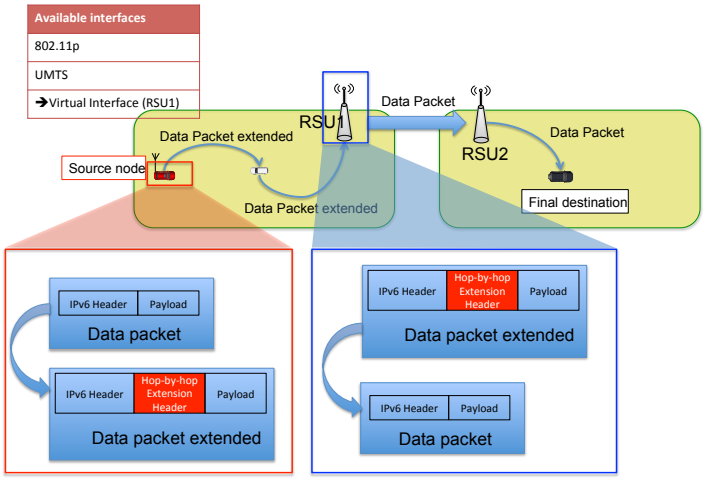


Figure 5.11: Use of IPv6 hop-by-hop extension header.

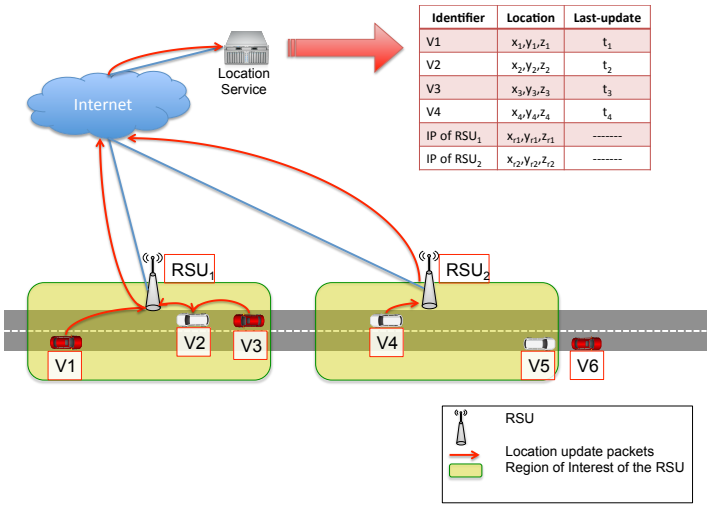


Figure 5.12: Location service. Location update flow of messages

This information is now enriched with the location and identifier (IP address) of RSUs deployed along the VANET as we present in Fig. 5.12.

In this example, vehicles located within the RoI and with an available path to the RSU regularly update their position. Since *V5* does not have a path to reach RSU_2 it cannot update its location. Likewise, due to not being within the RoI V6 cannot update its position using RSU_2.

Once that locations are available in the LS, it is possible to know the RSU closest to a specific destination by only querying the LS about it. The LS answers with the information of interest of the closest RSU consisting of its identifier and geo-location.

5.4.3. Using wired network to shorten the V2V path

The other challenge is to transmit geo-routing packets across the wired network allowing them to be forwarded later on in the VANET again. In Fig. 5.13 we detail the whole exchange of messages

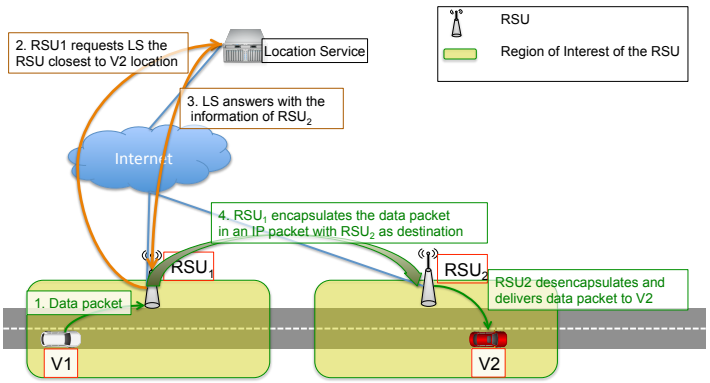


Figure 5.13: Geo-routing packets circulating across the wired network.

needed to deliver the VANET data packet through the wired network.

To solve this problem, when RSU_1 receives a data packet to be forwarded, it first queries the LS for the RSU closest to the location of the destination.This information is added in its list of possible routing decisions. The answer contains the identifier of the closest RSU, RSU_2, as well as its location. If the decision is to transmit the packet across the wired network, it encapsulates the geo-routing data packet in a new IP packet where the destination is the identifier of RSU_2.

At receiving this IP packet, RSU_2 desencapsulates it extracting the geo-routing data packet issued by $V1$ and forwarded via RSU_1. It makes another routing decision introducing the data packet again in the VANET. In addition, since $V2$ is in its transmission range, it delivers the data packet to the destination.

This mechanism was proved in real experiments thanks to the Spanish funded project called MARTA which also allows us to test not only this use case, but other more rewarding and complex.

In the next subsections we elaborate on the detailed operation of the routing decision stage of the protocol taking into account that now BRAVE deals with different wireless interfaces.

5.4.4. BRAVE Operation

Now that we have explained different new aspects that BRAVE must take into account so as to make a routing decision, we can proceed to take a thorough view of its operating mode.

We have followed the next notation to describe the mechanism of the routing protocol:

- c: Current node.
- m: The message. Through m.x the attribute x included in the header of the message is accessed: reliability, delay, cost, destination, auxiliary coordinate.
- I: Set of medium access interfaces.
- delay(m,a,b): Estimation of the time needed by the packet to go from a to b.
- cost(m,a,b): Estimation of the cost of sending a message from a to b.
- reliability(m,a,b): Estimation of the reliability that will be obtained by sending the message m from a to b.

- N(c): Set of c neighbours, i.e., the nodes that are one hop far from any network interfaces.
- f(n, m, d): Estimated utility of the node $n \in N(c)$ to send the message m to d.

Algorithm 8 Procedure forward(m:message)

```
1: if discard(m) then
2:    exit;
3: end if
4: if c == m.d then
5:    process(m); {Process if the own node is the destination}
6:    exit;
7: else
8:    m.d ← m.d − c; {Update the delay as it passes through node c}
9: end if
10: Candidates ← candidates(c); {Look for candidates to be the next forwarder around c neighbourhood}
11: Utility ← 0;
12: Selection ← null;
13: for all n ∈ Candidates do
14:    if f(n, m, d) > Utility then
15:       Selection ← n;
16:       Utility ← f(n, m, d);
17:    end if
18: end for
19: if Utility <= f(c, m, d) then {If the node does not have a promising next forwarder}
20:    buffer(m);
21: else
22:    forward(m, Selection);
23: end if
```

As we can see, Algorithm 8 is very straightforward. First of all, by using the function $discard(m)$, the node decides whether discarding the packet or not. This decision is based on the lifetime of the packet. If it has expired, the message is dropped and the algorithm ends. Otherwise, the evaluation of the message goes on checking if the current node is the destination of the message. Obviously, if this is the case, the message is delivered to the destination which process the packet by using $process(m)$.

Otherwise, the algorithm uses the $candidates(c)$ function to obtain the set of candidate nodes to be the next forwarders which is detailed in Algorithm 9. For each candidate the node evaluates its utility selecting the candidate with higher utility. The function f is used for this purpose. It computes the utility of a node based on its delay estimations, position, etc. An analysis of the impact of this utility function in the protocol is performed in the following subsection 5.4.5.

Finally, the utility generated by the best candidate to be next the hop is compared with the one generated by the current node. If the Utility of the best candidate is not greater than the current node it will buffer the message until new neighbours are detected. Otherwise, it forwards the message to the next hop.

It is worth mentioning, that although the forward process selects a candidate, if the candidate belongs to the neighbouring vehicles of the VANET, it will select it opportunistically as it was explained in Chapter 3. That is, if the routing decision is to forward the message to a vehicle of the VANET,

it will trigger all the sequence of the following messages: DATA, RESPONSE, SELECT and ACK to decide the best forwarder node.

Let us explain now how the function $Candidates(c)$ operates.

Algorithm 9 Function Candidates(c:node): SET

```
1: Candidates ← ∅
2: I ← AvailableInterfaces(c);
3: for all i ∈ I do
4:   for all n reachable via i do
5:     estimateddelay ← delay(m, n, d);
6:     estimatedcost ← cost(m, n, d);
7:     estimateddistance ← distance(n, d);
8:     if reliability(n, d) > m.reliability then
9:       if (estimated_delay < m.maxdelay) AND (estimated_cost < m.maxcoste) AND
         (estimated_distance < distance(c, d)) then
10:        Candidates ← Candidates ∪ n;
11:      end if
12:    end if
13:  end for
14: end for
15: return Candidates;
```

The *candidates*(c) function determines all the nodes that can act as a next hop, see Algorithm 9. Concretely, for each interface the current node evaluates all reachable nodes through itself. For each candidate three estimations are compared: delay, cost and distance, provided that the reliability constraint is satisfied. Here, it is worth clarifying what we understand as a directly reachable node through an interface.

For broadcast type interfaces like the VANET interface, reachable nodes are one-hop neighbouring nodes. That is, those located within the coverage range of the node. Thus, for a node, this information is updated periodically by a beacon based service of 802.11p called neighbour detection service. Nevertheless, for other interfaces such as UMTS or WiMAX, a reachable node is the other end node of the interface, for instance, for UMTS the base station.

On the other hand, the more suitable approach to route messages in pure VANET networks is opportunistic beacon-less protocols. These protocols operate in a reactive way, rather than based on neighbour tables built by beacons exchange. Thus, when a node broadcast a message, its neighbouring nodes are the ones who propose themselves as good candidates to forward the message to the destination. Various simulations and experiments have shown that this approach is the most appropriate for an unreliable network such as a VANET.

The estimates of cost, delay and distance are also dependent on the interface type. Thus, if a message is sent using the 802.11p interface, it will not have an associated economic cost and it will have an associated delay that will depend on the number of hops to the destination as well as a probability that a path exists. However, its reliability will not be very high. Nevertheless, if the node uses the UMTS interface, its associated cost will be higher but also its reliability. The delay of using this technology will be larger than the one of 802.11p for few hops, but this gap will be balanced as the number of hops is increased.

The utility of the VANET interface is calculated based on the neighbours information gathered by

the exchange of their beacons that can be outdated. If we decide to use the VANET interface and the sending fails, because no neighbours answer to the DATA message, all the interfaces are reconsidered but the VANET interface. This is extensive to the rest of cases. If one interface is chosen and the sending fails, only the rest of them are re-evaluated.

Once we have decided to buffer the packet, the node will try to send it again as soon as the node receives an event related to an update of any of its interfaces indicating the availability of new communication opportunities. In case of the VANET interface, such event can be a new beacon from a neighbour. When one of these events is triggered, the packet buffer is inspected trying to forward every message stored in it using the same forwarding algorithm as before.

5.4.5. The utility function in the routing protocol

The utility function used above is key for this routing protocol. This is, because this function is the one which analyses, for each packet to be transmitted by a node, the utility of sending it through the different opportunities that the node has, i. e., through the different directly reachable nodes.

Not all the packets transmitted along a network demand the same requirements. For instance, if the packet is that of a real-time video conference frame, it has severe restrictions regarding the time to be delivered to the destination. On the other hand, if the packet is part of an e-mail, its time-related restrictions will be more lax.

In the same way, each network interface offers different features that must be taken into account when a packet is going to be transmitted. For instance, if a node is forwarding a traffic warning message, the VANET interface is more suitable for broadcasting this message to the specific area. On the other hand, the UMTS interface will be more stable providing also the bandwidth necessary being more appropriated for surfing the Internet or for a film streaming session.

Therefore, the utility function, must be aware of these conditions to generate a result. Algorithm 10 offers an example of utility function.

Algorithm 10 Function f(c:current-node, n:neighbour, m:message): SET

1: $Utility \leftarrow 0$;
2: **if** $n.interface == UMTS$ **then**
3: $Utility \leftarrow u_{umts}$; {0,5}
4: **else if** $n.interface == RSU$ **then**
5: $Utility \leftarrow u_{rsu}$;{Initially set to 0,5}
6: **else if** $n.interface == VANET$ **then**
7: **if** $m.lifetime >= Threshold$ **then**
8: $Utility \leftarrow u_{vanet}$;
9: **else**
10: $Utility \leftarrow u_{vanetpenalized}$;{Initially Utility + 0.1}
11: **end if**
12: **end if**
13: $Utility \leftarrow Utility * (distance(c, m.destination) - distance(n, m.destination))$
14: $Utility \leftarrow Utility/(cost(n) * delay(n))$;
15: return $Utility$

Firstly, this utility function deals with the time-related constrains that a message can have, represented in this algorithm by the lifetime of the message. Thus, if the packet lifetime is less than a

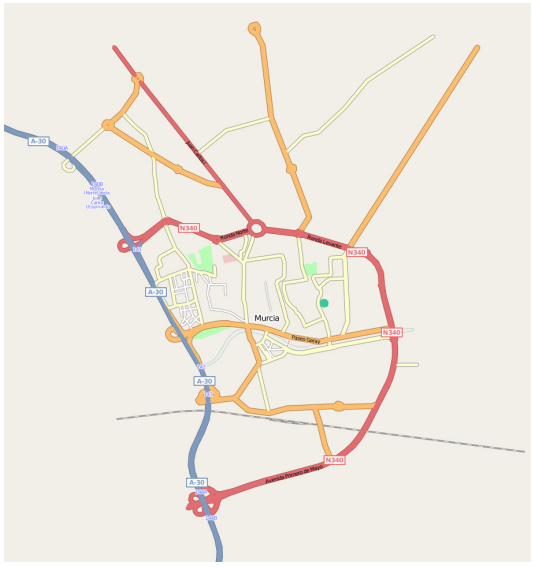


Figure 5.14: Map of Murcia city center and access roads used in our simulations.

pre-established threshold the VANET interface will be penalized against the others. In a similar way, the cost of using an interface affects this utility function, penalizing the UMTS interface because of its high cost. Finally, the other parameter taken into account in this utility function is the progress provided by a node. In the algorithm, nodes which provide more advance to the destination will be favoured.

So, to sum up, the utility function must accomplish the following characteristics: The different interfaces must be ranked according to their utility to hand over packets. This utility contemplates the benefit of using a determined interface, as well as its cost. Besides, QoS requirements must be contemplated to assure packets are not dropped because their lifetime has expired. For this reason, a threshold value is introduced. It is a trigger to select a different network interface (usually one with lower delay and higher bandwidth).

5.5. Evaluation of our proposed solution

To assess the performance of our proposed solution we simulate BRAVE over the urban scenario presented in previous chapters but including now RSUs. We have also studied the impact of the RoI size of the RSUs.

For this, we have used Network Simulator ns-2, version 2.33[1] to simulate an area of $5 * 4km^2$ in the city center of Murcia, Spain (Fig. 5.14). This map, as well as the vehicular mobility patterns have been generated using the well-known SUMO tool[2].

Particularly, for our simulations we have selected the most relevant streets, so our scenario consists of 53 streets and 28 junctions. Vehicles move through 20 predefined routes at a maximum speed of $13.89m/s$ ($50km/h$) inside the city, and $22.22m/s$ ($80km/h$) on the highway that crosses the scenario.

[1] http://www.isi.edu/nsnam/ns/
[2] http://sumo.sourceforge.net/

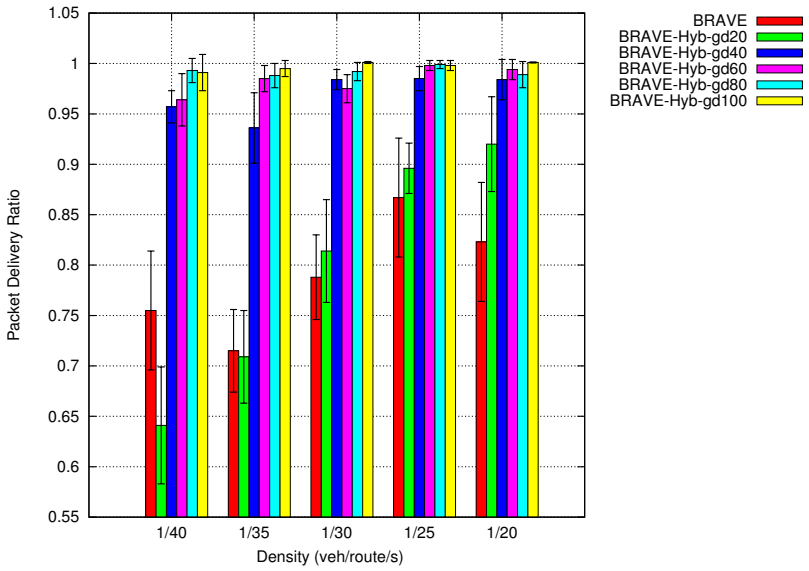


Figure 5.15: Packet Delivery Ratio

The routes followed by the vehicles have been selected according to realistic situations. We have also considered a wide range of traffic densities. Vehicles are injected into its route at a certain traffic rate. This rate is varied from 1/30 to 1/10 vehicles per route per second. In such a manner the $1/x$ rate means that each x seconds a new vehicle is injected into its route.

In our simulations, wireless signals propagate according to the two-ray-ground model. Vehicles carry out their communications via 802.11p interface card, implementing the enhanced ns-2 802.11 physical and MAC models. Transmission power is adjusted allowing a maximum transmission range of $250m$ and we have simulated 10 independent runs for each configuration (figures in following subsections show the average of such runs).

5.5.1. Performance in hybrid VANET

We have simulated different percentage of the scenario covered by the RoIs of the RSUs, 20%, 40%, 60%, 80% and 100%, comparing them against a pure VANET environment where there are not any infrastructure elements (RSUs or gateways) deployed within the scenario (0%).

Fig. 5.15 shows the packet delivery ratio (PDR) of these protocols as the density increases. Although with a 20% of the scenario covered by RSUs the difference between both proposals it is not very relevant because RSUs are not close enough to take advantage of the virtual interfaces. This gap is augmented as the number of RSUs is also increased. Thus, nodes can use RSUs as forwarders transmitting the packet in both wireless and fixed networks improving the performance as it is corroborated in the figure. From 40% on, this performance gap, in terms of PDR, between these proposals is extended by up to a 20% of the PDR in the lowest dense scenario, 1 vehicle every 40 seconds per route. Moreover, the PDR is maintained between 95% and 100% in these scenarios.

Another interesting result that we can state looking at the graph is that with a 100% of the scenario covered with gateways BRAVE is able to reach nearly a 100% of the PDR even in low density scenarios.

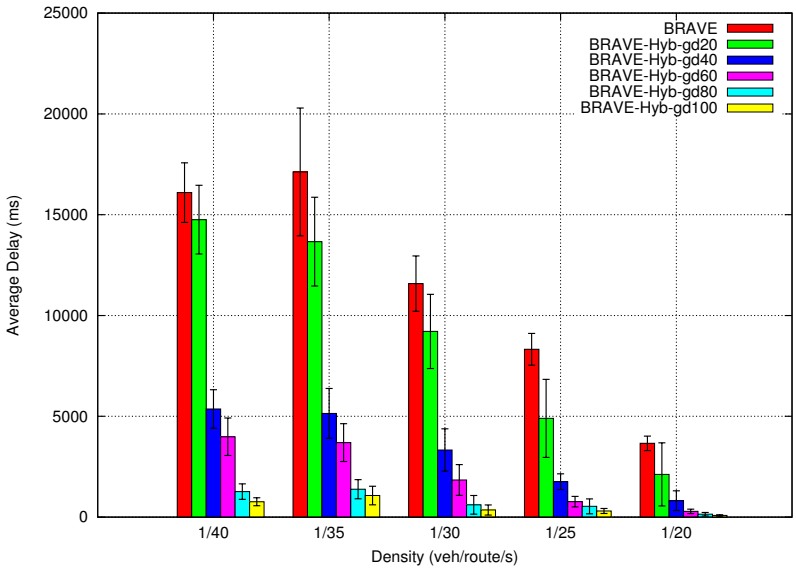


Figure 5.16: Average delay

The improvement of this protocol within a hybrid VANET environment is also verified in Fig. 5.16. The high PDR obtained by BRAVE for pure VANETs is reached due to the store-carry-and-forward paradigm, which incurs in a average high delay because vehicles themselves acts as ferries for the information until they encounter a new neighbour to deliver the information. However, BRAVE taking advantage of the RSUs too, can reduce this delay because it relies on the RSUs to deliver the information. This reduction is showed in low dense scenarios like in 1/40 $veh/s/route$ (1 vehicle each 40 seconds per router) where the improvement reaches up to 10 seconds. This gap is reduced as the density increases due to the increase of new neighbours able to forward the information. Nevertheless, in scenarios with high density there is a gap of a couple of seconds between both VANET environments.

5.5.2. Impact of the ROI size

Finally, the last analysis we have conducted within this research is related with the RoI size of the RSUs. We are interested in the impact this RoI size causes in the performance of the simulations because if the RoI size influences the performance, fewer RSUs could be deployed with a larger RoI size reducing the deployment effort of these elements. For this reason, we have varied the RoI size of the RSUs with the following values: 1000, 2000, 3000 and 4000m.

Fig. 5.17 shows the packet delivery ratio for a density of 1/35 $veh/s/route$. In the x-axis we display the percentage of the RSUs deployed in the scenario, whereas in the y-axis we display the PDR for these percentages. From the results of the graphs we can state that there is a huge difference between a 20% and a 40% of the scenario covered by RSUs. While with a percentage of 20% of the map covered by gateways a PDR result of a bit more than 70% is obtained, a density of 40% of gateways improves a 25% respect the previous density a with nearly 93%. Another interesting conclusion to highlight is that an effort to cover the 100% of the scenario with gateways is not worthy, because with a 60% of RSUs density we are able to obtain a performance very similar in terms of PDR.

The control overhead is shown in Fig. 5.18. This metric refers to the header size of different

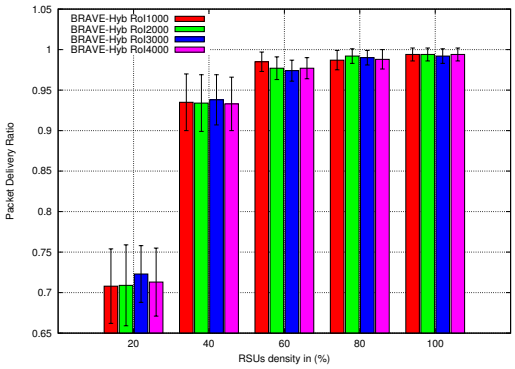


Figure 5.17: PDR for different ROI sizes and RSU densities.

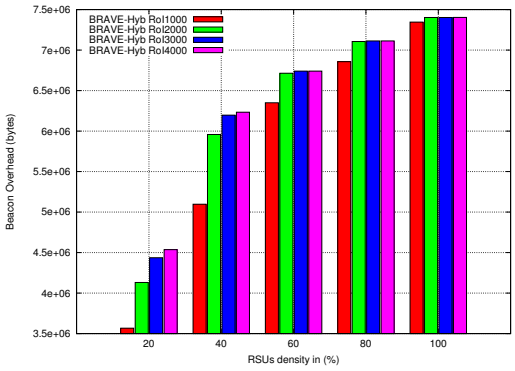


Figure 5.18: Control overhead for different ROI sizes and RSU densities.

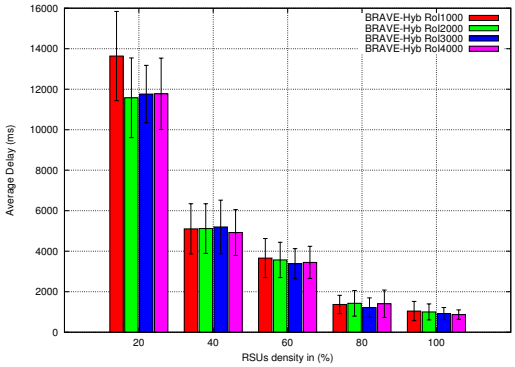


Figure 5.19: Average delay for different ROI sizes and RSU densities.

messages transmitted along the network without taking into account the data included in the data packets. So, it includes beacons and the DATA, RESPONSE, SELECT and ACK messages.

The advertisement of the RSUs is forwarded hop by hop by neighbouring vehicles by including the information of the RSU and its RoI in the beacon messages. Thus, if we enlarge the RoI size, the vehicles will maintain this information during more time increasing this way the control overhead. This behaviour is more noticeable in scenarios where only few gateways are deployed (20% and 40%). Thus, there is a trade-off between the RoI size and the overhead associated to it. A smaller RoI size will cause that less vehicles will propagate the information of the RSU allowing other vehicles to use it as a virtual interface, whereas if we have a huge RoI size this information will impair the performance of the routing protocol increasing its overhead.

Finally, Fig. 5.19 shows the average end-to-end delay. This metric reflects the importance of having the support of a wired infrastructure network. While with a gateway density of 20% we obtain nearly $14000ms$ to reach the destination, with a gateway density of 60% this delay is reduced to a value close to $4000ms$. These values reach their minimum values when the gateway density covers the 100% of the scenario, reaching nearly $2000ms$. Therefore, depending on the kind of traffic we are interested to deliver, a gateway density covering the whole scenario will be required. Otherwise, with a nearly 60% of the scenario we will have a good trade-off.

5.6. Conclusions and Future Work

In this chapter we have evaluated the performance of a hybrid VANET from the routing point of view. To achieve this goal we propose a routing protocol able to take advantage of the RSUs deployed by public traffic authorities as well as other wireless interfaces like GPRS, WiMAX or LTE among others. This protocol was developed taking the previous routing protocol BRAVE presented in the previous chapter.

We extended BRAVE with this capability providing a better performance by using the infrastructure in both PDR and average delay metrics.

We propose Beacon-less Routing Algorithm for Vehicular Environments (BRAVE) allowing it to deal with multiple-interfaces with an utility based interface selection mechanism based on the information of each packet to be transmitted. This protocol adapts itself to a hybrid scenario where

RSUs are deployed along the road and some of the vehicles are equipped with UMTS interface acting themselves as gateways for the others. A node, by this utility function will select the most promising forwarding node, among the different interfaces.

If the packet is transmitted using the VANET interface, BRAVE will follow an opportunistic next hop selection scheme to figure out the next hop. The opportunistic neighbour selection allows the protocol to obtain a high reliability in relatively dense scenarios. The proposed schemes eliminates a lot of contention and guarantees that the next hop has successfully received the data packet.

We have evaluated our protocol within a hybrid VANET environment with the purpose of analysing the impact of the amount of RSUs in this scenario allowing vehicles to use them as virtual interfaces and therefore getting that the packets can travels by both wireless and fixed network if needed in order to obtain a better performance. Our results show that an effort of deploying an amount of RSUs covering about 60% of the scenario obtains a good trade-off in terms of PDR, however, if there are non-delay-tolerant applications a 60% of the scenario is not enough to accomplish their requisites being necessary to cover up to 90% of the scenario to reduce the delay up to 1 second.

Another conclusion worth mentioning is that it is not necessary to cover the whole scenario with RSUs because a similar result is obtained with less effort (between the 60% and 80% of the scenario).

Chapter 6

Evaluation of the performance of pre-authentication in hybrid VANETs

In this thesis, we tackle different challenges in the field of VANETs. So far, we have addressed the topic of communicating nodes even when they are distant for both pure and hybrid VANETs.

To solve this issue, in previous chapters, we defined a complete routing protocol called BRAVE. Among their worth mentioning virtues we can highlight: its support for DTNs, a good trade-off between delivery ratio and end-to-end delay, its ability to communicate with the infrastructure and moreover to take advantage of it to deliver faster the data packet to a distant node.

Routing protocols are the baseline to provide communication among the nodes of a VANET as well as to connect to the infrastructure network. However, the access to the services of the infrastructure usually requires certain security aspects. Actually, most of the applications using the infrastructure to access a particular service require the authentication of the user so as to verify its identity and to know the services registered by the user.

In this chapter we address the current disadvantages of this authentication process in hybrid VANETs. We also propose a solution being able to take advantage of the mobility of vehicles to make this authentication process more agile.

6.1. Introduction and motivation

An isolated VANET without the support of some infrastructure is able to take on basic safety applications: for instance, an application which alerts of a traffic accident to nearby vehicles allowing them to reduce their speed avoiding a more dangerous one or even providing them alternative routes.

However, thanks to the lack of energy consumption requirements, vehicles can be equipped with more than one wireless device, employing other technologies like UMTS, LTE or WiMAX among others. In addition they can take advantage of the elements deployed along roads called RSUs which act as gateways to access the Internet.

The access to new information in the Internet provides additional benefits to by safety applications

like the mentioned above. They can improve their performance by accessing to real-time information provided by a server, or even allowing a faster distribution of the safety information among the nodes of the network through the different wireless interfaces.

Other applications have emerged thanks to this external connectivity. Public traffic authorities are not the only ones interested in these networks. VANETs have caught the attention of the private industry which can provide entertainment applications as well as other value-added services to improve the quality at driving.

Obviously, private industry will only offer its services to registered users who pay their fees, so in other words, these services will only be used by authenticated and authorized users. Therefore, an AAA infrastructure is needed to make these services possible in the network. In this field Coronado and Cherkaoui [99] studied AAA elements for service provisioning in VANETs evaluating both symmetric and asymmetric schemes opting for the latter ones despite their heavy computation requirements.

A typical authentication process, as we will see in Section 6.3, requires the exchange of multiple messages adding a significant latency to obtain network access. This problem is aggravated when a mobile node like a vehicle must repeat the process when it changes from one gateway to another as it moves. Another aspect to be aware of under a multi-hop network, is the cost of this process in terms of overhead and delay of the authentication.

In this chapter, our goal is not only to analyse the impact of the authentication scheme with the infrastructure over VANETs, but also the potential benefit of pre-authentication [100]. Using this pre-authentication scheme, a mobile node is able to carry out an authentication process with a gateway through the current gateway before it really starts using it. Thus, the node will not have to start the authentication process when attaching to this new gateway producing thus a faster hand-over.

Fig. 6.1 shows an example scenario where we can explain the use of pre-authentication. In the figure, vehicle A, follows the path indicated by the arrows. Thus, as soon as the vehicle discovers the presence of a gateway it starts an authentication process in order to use the services of the infrastructure network.

Gateways announce their presence by sending periodic advertisements with information about its position as well as other interesting information for the vehicles. Therefore, the vehicle will be able to authenticate with the closest one. However, the authentication process is a costly process that can take up to a couple of seconds to be completed. This means, that when the vehicle detects the gateway it will employ a part of the time in completing the authentication process, which could be used to send traffic to the infrastructure.

Using the pre-authentication scheme, when the vehicle detects a new gateway, it can carry out the authentication with the new gateway through the current one. Thus, when the node decides to attach to the new gateway it will save time and we also get as a result a better performance because the new gateway will be able to forward the traffic straight away.

The remainder of this chapter is organized as follows: In Section 6.2 we describe the main contributions published related to the authentication issue. The traditional authentication scheme is described in Section 6.3. Our proposal will be developed in Section 6.4 describing the pre-authentication process and their advantages. This work will be reflected in Section 6.5 where simulations and results are commented. Finally in Section 6.6 we will comment the benefits obtained in this article as well as our next research steps related to this work.

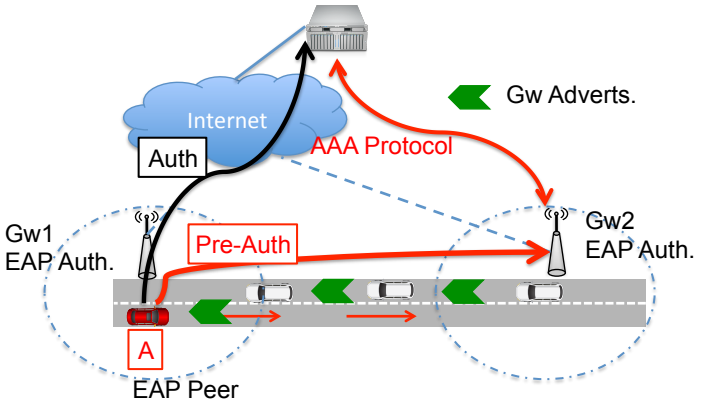


Figure 6.1: Example of scenario of pre-authentication.

6.2. Related work

A solution for authentication and authorization based on the deployment of PKIs is proposed by Casola et al. [101]. However, the deployment of purely-based PKIs solutions for network access control is less common than the use of AAA infrastructures. In this sense, some solutions like the ones proposed by Hafslund and Andersson [102] or by Moustafa et al. [103] provide access control and authentication in hybrid MANET by using deployed AAA infrastructures and EAP over 802.11i link-layer frames.

However, a protocol which transports Extensible Authentication Protocol (EAP) between the mobile node and the gateway through multiple hops independently of the underlying wireless technology is more appropriate in the context of MANETs or VANETs. The reason for this argument is that EAP is an authentication framework which supports multiple authentication protocols making it flexible depending on the situation.

Examples of these protocols are IKEv2, by Kaufman [46] or PANA, by Forsberg et al. [45] since they use UDP as transport protocol, although PANA provides a lighter way of operation. In this sense [104] proposes an extension to Mobile IPv4 to allow ad-hoc nodes to connect to the Internet. However this scheme is well-known to have performance limitations and the authentication mechanism is specifically designed for Mobile IPv4. Also, solution [105] designs a new public-key based protocol to provide an efficient and fast authentication process between the mobile node and the gateway, contacting the AAA server after the successful authentication. However, the solution assumes a complete change in the current AAA model and the standards defined for traditional network access control. In this sense, the pre-authentication scheme that we study in this paper follows the standard model for network access control based on EAP defined in [106].

Nevertheless, an authentication based on EAP may require the exchange of several messages as well as a certain time to process and complete the key exchange between the client and the server. The total time spent in completing an authentication can vary from several milliseconds to seconds as Georgiades et al. [107] indicated in their work.

In a VANET this problem is becoming important due to the innate properties of mobile networks like the network mobility or the link breaks that can cause a longer delay in the delivery of the data increasing the time necessary to complete an authentication process. Along the path covered by a

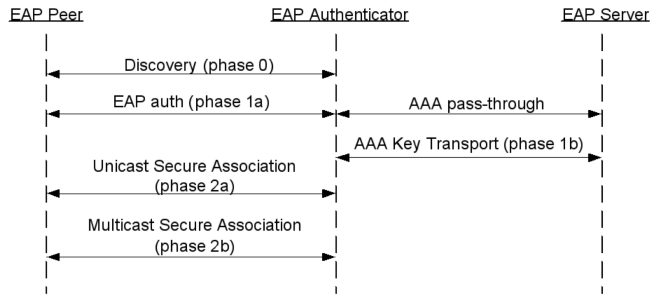


Figure 6.2: EAP's flow of messages to complete the authentication process.

vehicle this authentication process can be repeated several times with different gateways, so a lot of link losses will happen deteriorating the performance of the authentication process. This is the reason why it is interesting to reduce the time spent by a node to be connected and authenticated with a gateway.

In our previous contributions [108, 109], we proposed a solution to the authentication process in MANETs making it more efficient by the use of a media-independent pre-authentication scheme [110]. The proposal presented in these articles consists in an utility based control scheme employed to perform efficiently the pre-authentication process.

Every node willing to connect to the infrastructure network will execute this scheme periodically to select the most promising gateways to pre-authenticate with. Therefore, the pre-authentication will not be done with all the gateways that the node finds along its path but just with the better candidates.

The key aspect of this scheme is that of predicting the the next gateway that the node will be attached to after leaving the current one. This prediction is made taking into account the past and current distance from the node to the closer gateways. Thus, with this information, a new distance to the gateways is estimated for the next evaluation instant.

6.3. Access control in vehicular networks

As commented above, the infrastructure network enriches vehicular networks in such a way that a lot of services can be offered to drivers and passengers like a streaming film service, a news subscription application, a game platform to entertain the passengers and the likes. Hence, service providers need a secure access control scheme to know which users have acceded to what services and when this access took place.

For our purpose, we have used the Extensible Authentication Protocol (EAP) framework because it is one of the most common schemes and it is widely accepted within the Internet Engineering Task Force (IETF). Nodes using EAP over any transport protocol like PANA, or IKEv2, can be authenticated against an AAA server via a default gateway which is responsible for sending the node's credential to this AAA server.

Fig. 6.2 presents a sequence diagram which describes the flow of messages required to fulfil the authentication process by using EAP.

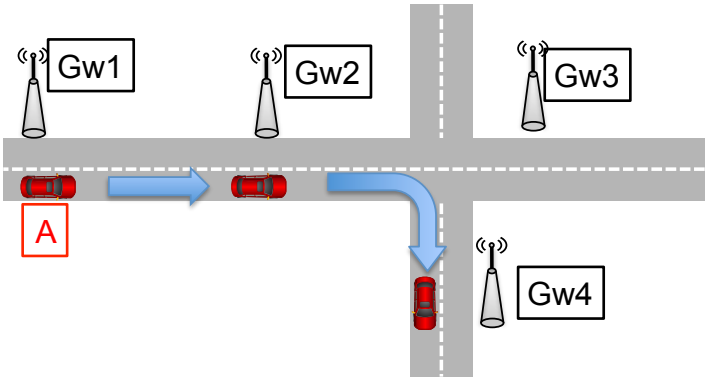


Figure 6.3: Example scenario of the authentication process carried out by Vehicle A as it moves along the path indicated by the arrows.

As we can see the authentication process comprises two phases. After a discovering stage where a gateway announces its presence to the nodes (phase 0), an EAP authentication starts between the EAP Peer (mobile node) and a EAP Server through an EAP Authenticator (gateway) in order to check the node's credential (phase 1a).

If the authentication succeeds, the EAP Authenticator receives a Master Session Key (MSK) from the AAA server (phase 1b), that the mobile node can also derive itself. This MSK is also used for further derivations obtaining new keys called Transient Session Keys(TSKs), that will be used to establish security associations between the mobile node and the gateway (phase 2a and 2b).

Let us consider an example scenario like the one illustrated in Fig. 6.3 in which **Vehicle A** uses the services offered by the wired network. The vehicle by connects and authenticates to the nearest gateways as it moves along its route.

Using the traditional authentication scheme, **Vehicle A** starts authenticating itself with gateway 1. As it moves, it will later receive a new message announcing the presence of a different gateway (gateway 2). When this new gateway becomes the closer one it starts the whole authentication process with this new gateway so as to continue enjoying the services the users were using. This event will happen as the vehicle moves when it encounters new gateways repeating the authentication process for each new gateway.

The same vehicle using a pre-authentication scheme could save time in future authentications taking advantage of the current gateway to complete a new authentication with another gateway to be used in the future. In addition, this pre-authentication process is quite similar to the authentication process we have just explained above. Indeed, the concrete messages exchanged by the gateway and the mobile node are the same. The main difference in the pre-authentication process is that the messages do not use the VANET to reach the new gateway, but the current gateway acts as a relay of the messages exchanged between the mobile node and the new gateway.

6.4. Pre-authentication in VANETs

Our proposed solution based on the pre-authentication operation consists in the following. First of all, gateways periodically announce their presence along a RoI allowing vehicles circulating within

this area to connect to the infrastructure by authenticating with them. A vehicle willing to access the infrastructure must authenticate with a gateway. So, it selects the closest gateway to start the authentication process. Once the authentication process is completed it will be able to enjoy the benefits of the infrastructure.

As it moves it will receive new periodic gateway advertisements. These messages contain important information like the gateway location, its radio range or its RoI. This information is very useful for the **gateway selection mechanism** to figure out the more promising gateway among the nearby ones to start the pre-authentication with.

Once we have obtained the promising gateways to pre-authenticate with, the node uses its current gateway as a relay to complete the authentication process with new gateways. So, the pre-authentication traffic follows the path *vehicle => authenticated_gateway => gateway_to_pre-authenticate_with*. It is worth noting that while the first part of the path involves the VANET the second part is only transmitted through the infrastructure network.

Thus, when the vehicle checks that one of this pre-authenticated gateways is the closest one it will not waste time completing the authentication, using this valuable time to send traffic information to the destination.

Thanks to the **gateway selection mechanism** pre-authentications will only be made with the gateways whose probability to be attached to in a near future is high enough. This way, the control overhead entailed to the pre-authentication scheme is reduced to pre-authentications with the promising gateways.

After an authentication or pre-authentication process, the AAA server is the entity making the decision of accepting or denying the access to the peer. This process ends storing in both ends, server and peer, security material pretty important to maintain the connection to the infrastructure.

This security material must be saved in a sort of memory. For this reason, we have used a cache to store it whose size is another important aspect in the pre-authentication process. In a cache with a small size, old gateways will be thrown off to allow new promising ones to be stored. This problem would not exist with a large cache, but a lot of pre-authentications will cause both a high use of the space of the storage device and a high overhead due to all the pre-authentications the cache allows.

Finally, the last aspect of the pre-authentication scheme to clarify is the own authentication process. We have chosen EAP-TLS [111] since it is one of the most common EAP methods nowadays. In addition, we use PANA [45] as a lower-layer protocol for EAP [106] due to its suitability for operating in multi-hop networks probed by Marin et al. [112] and Bernal et al. [113]. Finally, the last component needed to perform the authentication is the one responsible for delivery the EAP messages from the gateway to the AAA server. For this purpose we have selected Diameter [114] which is commonly used in 3G networks.

The complete authentication process requires exchanging several messages as well as the credentials of the mobile node. This makes it a heavy process that can last up to a couple of seconds in some cases.

During the authentication process using PANA under EAP-TLS several messages must be exchanged delivering some cryptographic material. This flow of messages is represented in Fig. 6.4. In this figure, we can see the different phases of the PANA protocol following the exchange of messages between a PANA client (PaC) and a PANA agent (PAA) corresponding to a mobile node and a gateway respectively. This flow has been slightly modified in the new release of the protocol but our

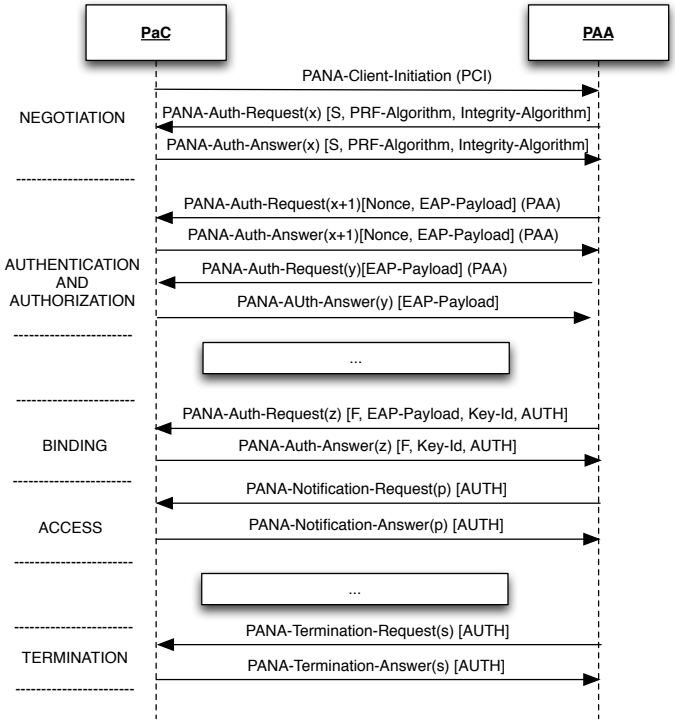


Figure 6.4: Flow of messages of PANA protocol.

simulations were made with the old one.

Table 6.1, on the other hand, shows other kind of information related to these messages, like the mean processing time, some statistical information as well as the packet size corresponding to each message exchanged. This information has been obtained empirically with a real implementation. These values are used in our simulations to make them more realistic.

6.4.1. Gateway selection mechanism

This mechanism provides the candidate gateways to perform the pre-authentication process with. It counts with a cache of authenticated/pre-authenticated gateways which stores the identifier of the gateway as well as the cryptographic material associated to its authenticated connection. So, if an element of this cache is removed, the cryptographic material will be also removed being removed from the pre-authenticated gateways.

The criterion used to insert one gateway in the cache is based on the distance between the gateway and the node. Thus, the most distant gateways are the first to be removed from the cache allowing new closer ones to be aware of.

Periodically, our scheme obtains the gateways of this cache to perform the pre-authentication with them, adding the cryptographic material into the cache.

This straightforward mechanism is very efficient in the following scenarios as we will see in the following section.

Messages	Mean processing time (ms)	Confidence int.(ms)	Packet size (bytes)
PCI	4.17	0.32	12
PSR	1.28	0.04	92
PSA	20.22	0.5	64
PAR	1.49	0.03	56
PAN	18.8	0.47	156
PAR	7.65	0.04	896
PAN	22.6	0.44	1056
PAR	1.13	0.05	116
PAN	21.92	0.82	56
PBR	1.33	0.07	188
PBA	n/a	n/a	96

Table 6.1: Mean processing time, confidence interval and packet size for PANA and EAP-TLS authentication.

6.5. Evaluation of the pre-authentication scheme

To evaluate the performance of the pre-authentication scheme we conducted a set of simulations using the Network Simulator 2 (NS-2) version 2.33 [115]. They consists of two different groups: the first group in which we evaluate the impact of the pre-authentication in VANETs and a second one where we evaluate how gateway density affects to the performance of the pre-authentication scheme.

Since the authentication process comprises the exchange of several messages with different sizes as described in Table 6.1 we have used these values to fairly simulate the authentication process adding to our simulations the different processing times too. This way, the authentication process will not only take into account the different message sizes but also the different delays introduced by their processing time according to the values of the table.

Regarding the communication parameters selected for the simulations, we have used the Two-Ray-Ground signal propagation model with a coverage range for all the elements of the network (vehicles and gateways/RoIs) of 250m. We have also used our previously presented routing alternative BRAVE to deliver the packets to the infrastructure and to other nodes of the network.

Finally, the advertisement of the gateways is made using GwDisc, a gateway discovery protocol proposed by Ros and Ruiz [116]. Since our objective in this chapter is that of analysing the pre-authentication scheme, we need nodes to receive advertisements of diverse gateways so as to have a wide enough list of available gateways to operate with. For this reason, we set the RoI size to the whole simulated area. Since the dissemination of the presence of the gateways is made by introducing this information inside the beacons the advertisements only incurs in a overhead of these messages.

6.5.1. Impact of the pre-authentication VANET environments

The assessment of the impact of the pre-authentication has been done simulating two kind of scenarios: an inter-urban scenario represented by a highway of $4km$, and an urban scenario represented by a grid of $1km$ x $1km$ with different vehicles densities and different number of traffic sources.

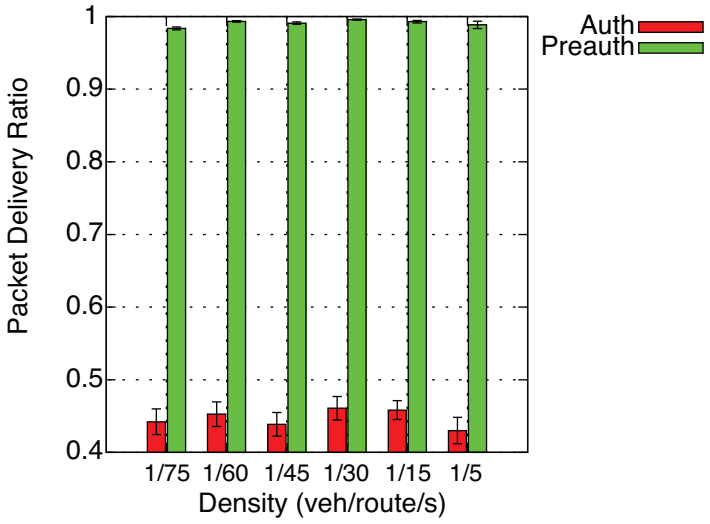


Figure 6.5: Packet Delivery Ratio for the inter-urban scenario.

Inter-urban scenario

The inter-urban scenario consists of a $4km$ highway segment with two lanes in each direction. Along the highway we have fixed 5 gateways uniformly distributed in the following positions: gateway 1 (414.6, 30), gateway 2 (1243.8, 30), gateway 3 (2073, 30), gateway 4 (2902.2, 30) y gateway 5 (3131.4, 30). We have also defined a common RoI for all of them which covers the whole scenario allowing vehicles to unlimitedly propagate their advertisements.

Vehicles, on the other hand, circulate at a maximum speed of $33m/s$ and are generated at a rate of 1/75, 1/60, 1/45, 1/30, 1/15 and 1/5 veh/path/s. That is, one vehicle is generated every 75, 60, 45, ... seconds in every defined path of the highway.

The last parameter to adjust in our simulation is the cache size of the authenticated/pre-authenticated gateways. For these simulations we choose a cache of *two entries* which is also defined in the traditional authentication scheme. Thus, both schemes have the possibility of having up to two authenticated gateways in their cache.

This scenario is very favourable for the pre-authentication scheme because every discovered gateway will be certainly used later on. Therefore, although the gateway's selection mechanism is very simple, (i.e. a promising gateway is just a gateway close to the node) this selection mechanism provides a good performance in this scenario as we shall see in the following figures.

In Fig. 6.5 we show the packet delivery ratio (PDR) with a 95% confidence interval of both schemes. The difference between them is notable with an improvement of 50%.

Our pre-authentication scheme takes advantage in the case where vehicles approach gateways and when they get away from them to send data through the infrastructure since they are pre-authenticated with them. So, the only wasted time in the authentication process is caused by the authentication with the first gateway. As a result the pre-authentication scheme obtains nearly a PDR of 100%. The traditional scheme must spend part of the approaching stage to make the authentication process wasting this important period of time without transmitting data to the infrastructure. Since the authentication process is a very cumbersome one, the PDR is reduced down to the 45%.

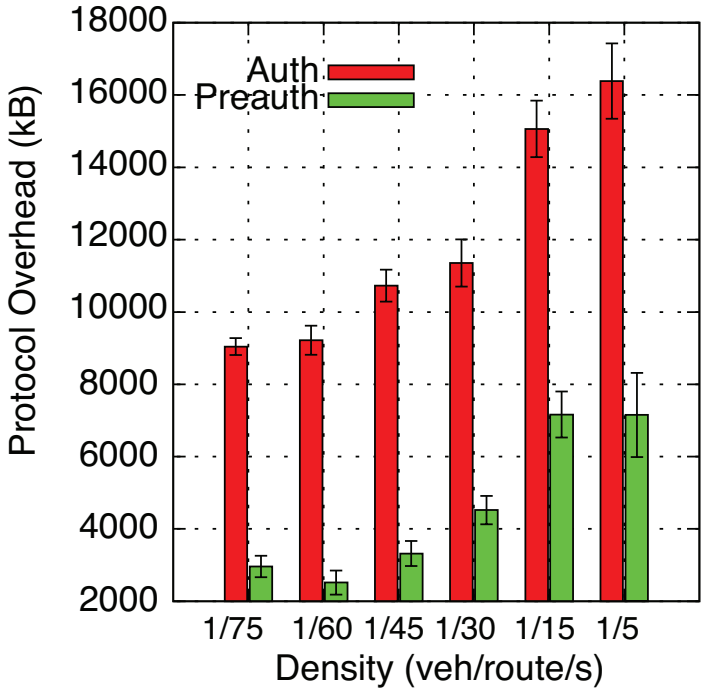


Figure 6.6: Control overhead introduced by both authentication schemes in the inter-urban scenario.

The control overhead of both schemes is presented in Fig. 6.6. This overhead takes into account only the messages related to the authentication process, i.e. we do not consider the gateway advertisements as control overhead, only the authentication/pre-authentication messages.

The pre-authentication scheme obtains an overhead twice lower than the traditional scheme. The reason is that gateway advertisements are forwarded by vehicles hop by hop, so a vehicle at the moment of receiving a gateway advertisement is very likely located far from it. With the traditional scheme, when the node selects it as its new gateway to the infrastructure, it is still far from it so the probability of losing authentication packets due to such a distance is very high.

With the pre-authentication scheme, the process is totally different. The vehicle takes advantage of its close distance to its current and authenticated gateway to start the pre-authentication to a promising gateway. Since the distance to its current gateway is really short, the overhead is reduced because the authentication packets sent will follow the path from the node to the authenticated gateway over the VANET and the infrastructure from the authenticated gateway to the gateway to pre-authenticate with. In addition, the graph also shows that the number of authentications and pre-authentications is less for the pre-authentication scheme than for the traditional one.

Finally, in Fig. 6.7 we can see the average delay of the data for both schemes. The better performance of the pre-authentication scheme reflects the utility of this scheme. With the pre-authentication scheme the average delay is about $1.5s$, whereas by using the traditional one it increases up to $5s$. So, the pre-authentication scheme provides a seamless handover changing from one gateway to another.

To sum up, since the gateway selection mechanism is more likely to hit in its prediction, a pre-authentication is always completed with a gateway to be attached in a close future. The pre-authentication process also takes advantage of the distance to its current gateway to achieve the

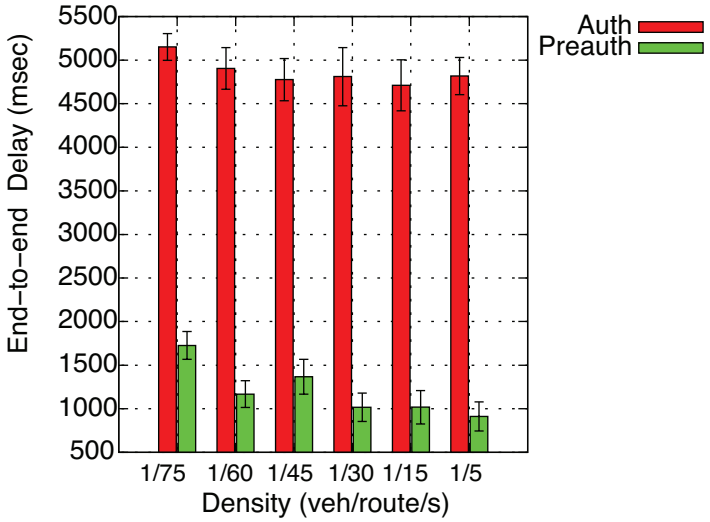


Figure 6.7: Average delay of the messages in the inter-urban scenario.

pre-authentication process improving the performance with respect to the traditional scheme. However, which is the performance of the pre-authentication scheme in an urban scenario? The next subsection answers this question.

Urban scenario

The simulated urban scenario consists of a grid of $1049m$ x $1049m$, with three roads horizontally and three vertically. We have placed 4 gateways in the following positions: gateway 1 (11, 505); gateway 2 (505, 11); gateway 3 (1045, 505); gateway 4 (505, 1045) with a coverage range of $250m$ as we can see in Fig. 6.8. As in the previous scenario, the RoI for all the gateways is the same covering the whole scenario. Regarding the density we have tested this scenario with 1/30, 1/25, 1/20, 1/15 and 1/10 $veh/path/s$. Vehicles in this scenario will travel at lower speeds than in a highway, about $16m/s$.

In this urban scenario, we have defined different flows of vehicles making them follow different paths and turns with the goal of covering the whole urban scenario with the same traffic density. That is, one flow of vehicles starts in a certain junction and it follows a determined path while another one, despite starting in the same point, follows a different one. This way, vehicles that at the beginning share the same route takes different paths at arriving in the next junction. Thus, we can evaluate the gateway selection mechanism since vehicles now have more than one gateway to pre-authenticate with.

Since vehicles at junctions now have more than one direction to follow, the predictions of the pre-authentication scheme are not 100% correct. This is the reason why the PDR presented in Fig. 6.9 are lower than with the inter-urban scenario. Despite this difficulty, the results maintain a high difference between both traditional and pre-authentication schemes. The traditional scheme has decreased its performance down to a 30% whereas the pre-authentication scheme is able to get a 90% as its best result, being all the values over the 80% of the PDR.

In this scenario, as Fig. 6.10 shows, the pre-authentication scheme also presents better performance

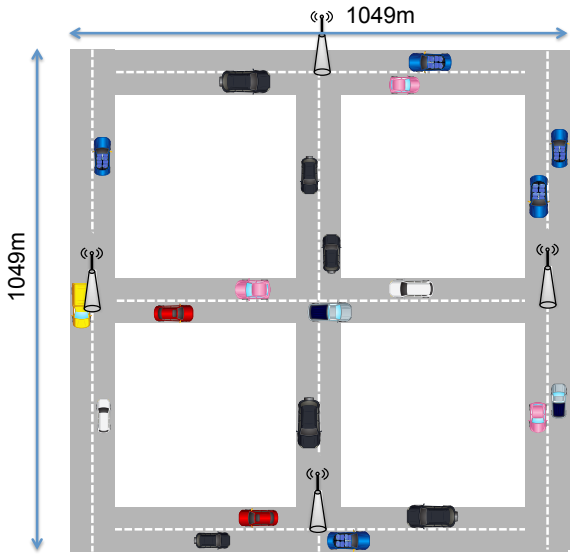


Figure 6.8: Grid of 1km x 1km with four gateways.

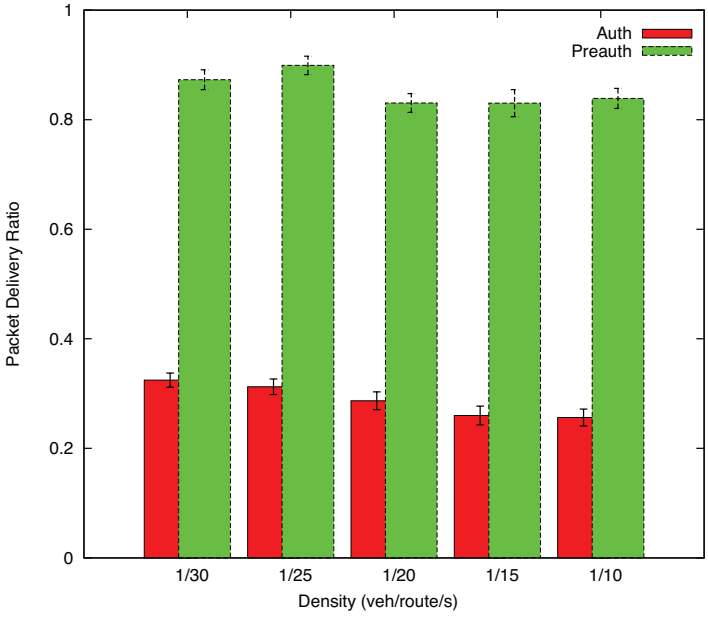


Figure 6.9: Packet Delivery Ratio for the urban scenario.

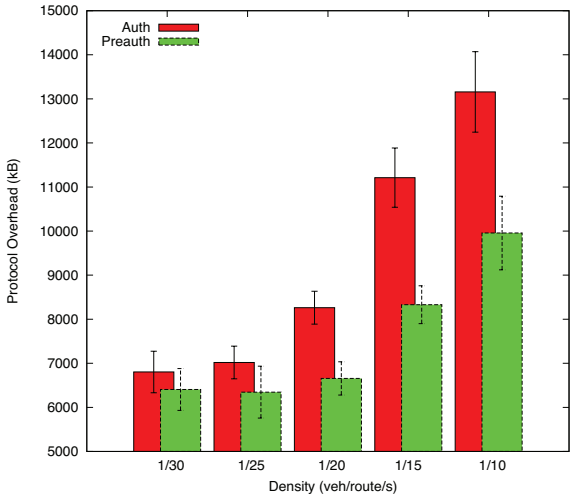


Figure 6.10: Control overhead introduced by both authentication schemes in the urban scenario.

in terms of control overhead with respect to the traditional scheme despite it also performs both authentications and pre-authentications. In lowly dense scenarios with 1/35 and 1/25 $veh/path/s$, both schemes have a similar performance. This means, that vehicle are less likely to propagate the gateways advertisements making both proposals to act similarly. In addition, pre-authentication predictions could be obtained without having the knowledge of the gateway located in the direction the vehicle will take.

In scenarios with high density, these gateway advertisements are more likely received. This is the reason why the difference between both schemes is notable. In light of these results, we can state that although the overhead for both schemes increases exponentially , the traditional one increases quicker than the pre-authentication scheme making the latter one more scalable.

The average delay also reflects this behaviour as you can see in Fig. 6.11 if we compare it with the results obtained in the highway scenario.

6.5.2. Impact of the gateways density in the pre-authentication scheme

In this second group of simulations we are interested in evaluating how the increase of the amount of gateways affects the performance of the pre-authentication scheme. This study is pretty interesting because an increase of the density of gateways could cause an increase of the authentication sessions affecting the behaviour and good results obtained from the previous group of simulations. Thus, in order to shed some light on this, we have run our simulations under a larger grid of $2km\ x\ 2km$, playing with different number of gateways.

In Fig. 6.12 you can see the location of the gateways for the different gateways densities in the grid: the first case consists of four gateways in the following positions (1010,0), (0, 1010), (2010, 1010)

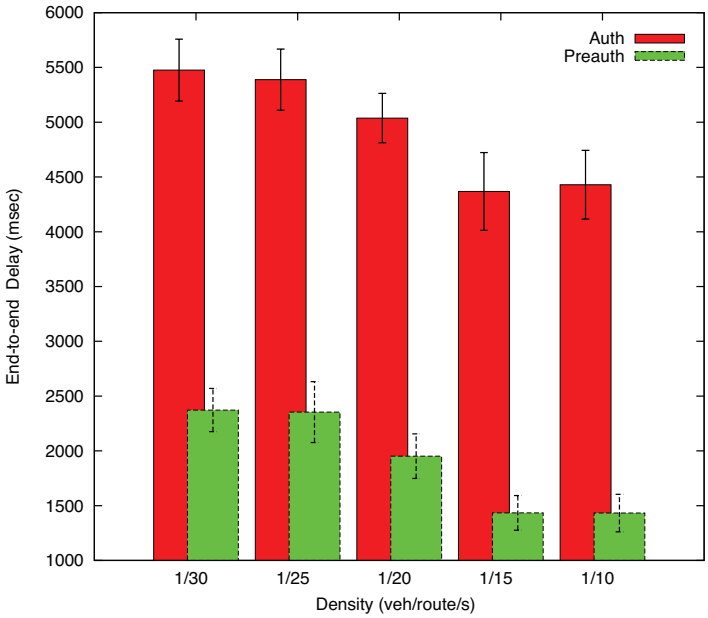


Figure 6.11: Average delay of the messages in the urban scenario.

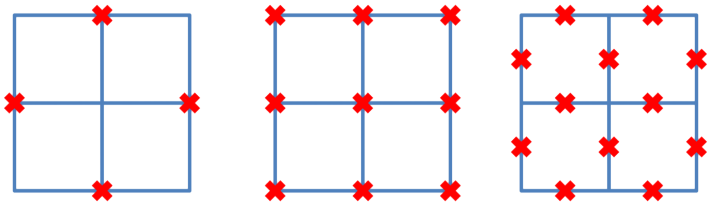


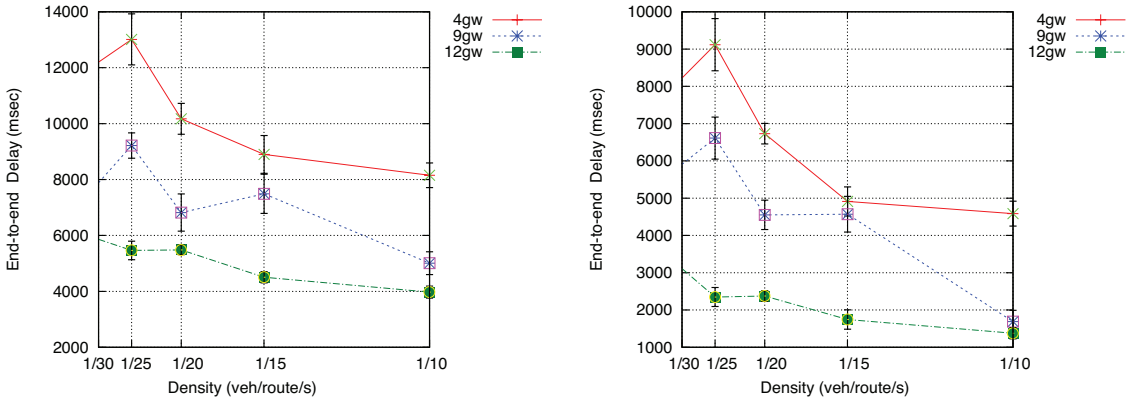
Figure 6.12: Grids of 2km x 2km with different number of gateways (4, 9 and 12).

and (1010, 2010) with a distance among them of 2 km following the path of the roads. The next one consists of nine gateways which have been located in (0,0), (1010, 0), (2010, 0), (0, 1010), (1010, 1010), (2010, 1010), (0, 2010), (1010, 2010) and (2010, 2010) thus every gateway is 1 km separated from the others. Finally, in the last one, we have placed twelve gateways under the same scenario in (510,0), (1510, 0), (0, 510), (1010, 510), (2010, 510), (510, 1010), (1510, 1010), (0, 1510), (1010, 1510), (2010, 1510), (510,2010) and (1510, 2010) increasing the amount of gateways advertised despite being separated 1 km (following the roads).

As in previous simulations, we have run our simulations varying the density of vehicles as well as the number of traffic sources as in the previous section. We have selected 1/30, 1/25, 1/20, 1/15, 1/10 $veh/path/s$ vehicle densities and 5, 10 and 15 traffic sources.

In this scenario vehicles can move with a speed up to $20m/s$. Since this grid was larger than the previous one we have also increased the size of the authenticated/pre-authenticated gateways cache to 3 entries.

The first metric that we are showing in Fig. 6.13 is the average delay of the data traffic sent by source nodes for both authentication schemes. This metric shows a clear evolution not only depending on the vehicle's density but also depending on the gateway's density. The more gateways deployed in



(a) Traditional scheme

(b) Pre-authentication scheme

Figure 6.13: Average delay obtained for both the traditional authentication scheme (left) and the pre-authentication one (right).

the scenario the less the delay obtained, this happens because we are reducing the distance between each node and a possible gateway, reducing the number of hops between them too and therefore reducing the delay.

Comparing both figures we can see that the average delay of the traditional scheme remains higher than the pre-authentication scheme in every case.

Taking a look at the PDR, as we can see in Fig. 6.14, this metric is affected by the gateway density. Since we are making our simulations in a large grid, the results gathered by the 4 gateways density implies that the distance from a node to a gateway so long that a packet will need to move through several hops in order to get to the gateway. The higher the gateway density the smaller the distance between a node and a gateway decreasing the number of hops to get to the destination and therefore increasing the PDR.

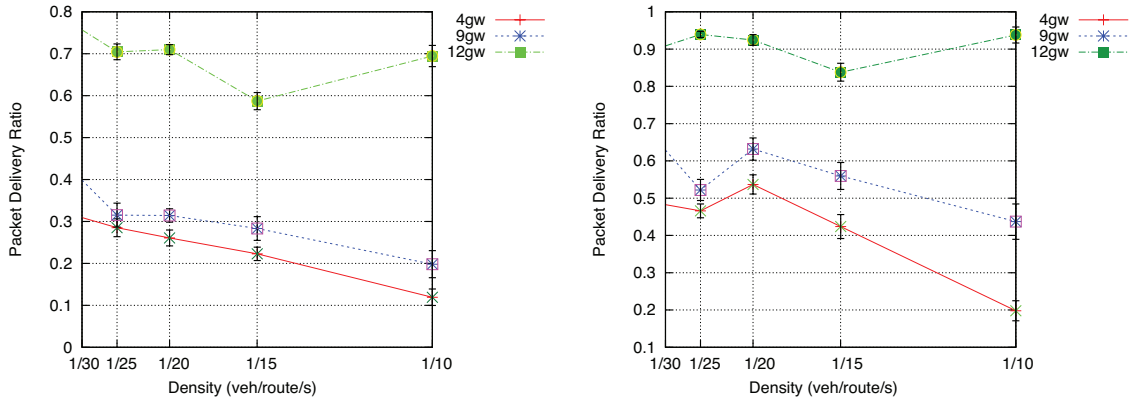
Even so, the performance of the pre-authentication scheme is better than the traditional scheme, because it takes advantage of the current authenticated gateway to carry out the pre-authentications with the promising gateways.

6.6. Conclusions

In this chapter we have studied the performance of the access control to the infrastructure by the nodes of a VANET. More specifically the impact of the pre-authentication scheme in VANETs. Firstly we have analysed this impact under two kind of VANET scenario, urban and inter-urban. Secondly, we have also studied the influence that the variation of gateways' density has in the authentication process.

The first study shows that the pre-authentication scheme proposed is a good alternative to the traditional scheme in environments with high mobility like VANETs. It reduces the overhead associated to the authentication process and also improves the delivery ratio achieving faster hand-overs between gateways.

The second study analyses the behaviour of the authentication process, under different gateway's



(a) Traditional scheme

(b) Pre-authentication scheme

Figure 6.14: PDR obtained for the traditional authentication (left) and pre-authentication (right) scheme.

densities. In light of the obtained results, there exist a between both elements. Actually, both traditional and pre-authentication schemes obtain better results as the number of gateways is increased.

After all the simulations and results obtained, we can conclude the pre-authentication scheme is an appropriate mechanism to improve the authentication process in VANETs. It accelerates the handover among gateways carried out by the mobile nodes allowing them to make the most of the time connected to the gateway to access the desired services increasing this way the packet delivery ratio. In addition, the overhead introduced by the pre-authentication scheme does not impair the overhead of the traditional process confirming our proposal as a good mechanism to provide authentication and access control in this vehicular networks.

Although the obtained results yield by the simulations are very promising, the pre-authentication scheme can still be enhanced. The gateway selection mechanism used in our simulations is very straightforward and it does not take into account all the information provided by the advertisements of the gateways. A smarter gateway selection mechanism could also increase the obtained performance reducing the amount of authentication sessions and therefore the control overhead.

Chapter 7

Evaluation of the use of CS in data harvesting for VSNs

So far, we have addressed different aspects related to delivering packets across VANETs in this thesis. We started providing a routing protocol being able to deal with DTN networks with a high delivery ratio. We also provided a mechanism to strengthen the delivery of packets under hostile scenarios.

In a next iteration, we extended it allowing to use different wireless devices as well as the infrastructure to both connect to a node in the Internet, or by using it as a shorter path to a destination node of the VANET.

We also studied a different aspect of the security related to hybrid networks and their access to the infrastructure providing an agile mechanism called pre-authentication to allow nodes to make the most of gateways without wasting time in the authentication process.

This chapter faces a different challenge which is also interesting for vehicular networks, but from a different point of view. Vehicles are nowadays equipped with lots of sensors to measure the movement of the vehicle, its acceleration, velocity, direction, location, etc. They can also include other kind of sensors like a thermometer, humidity or pollution sensors, among others which can be of interest to several public authorities so as to check the pollution level of a determined area, its temperature and so on.

So, we can use vehicles circulating within some area to provide sensed information, instead of deploying lots of sensors within a determined urban environment. Such networks are usually called Vehicular Sensor Networks (VSNs).

In this chapter we study the harvesting process in VSN. This is usually done by the own vehicles acting as forwarders for the sensed data aimed at the Fusion Center(FC), the device responsible for analysing the sensed information.

7.1. Introduction and motivation

The issue of gathering information in a network is not new and has already been studied in the WSNs literature. Nevertheless, there are two main differences between WSNs and VSNs that affects

to the way the problem can be addressed. The first one is the mobility of the nodes. In WSNs sensors are deployed within a specific area being unable to move whereas in VSNs vehicles are moving with a high speed along streets and roads. The second one is related to energy consumption. Nodes in a WSN have a short battery life which is a great handicap in terms of performance, so all of the designed protocols and implementations are aware of it. On the other hand, in VSNs vehicles do not have such a hard requirement, so the variety of approaches to harvest information is extended to other designs that are not aware of the energy consumption.

In the area of information theory, a collection of sampling methods have emerged lately. They are very useful for making the harvesting process more efficient. This is known as **Compressed Sensing (CS)**, also called compressive sensing, compressive sampling or sparse recovery. Its main motivation is the idea of reconstructing the original information of a signal using only a few samples without losing accuracy in the reconstruction process.

Traditional harvesting protocols usually combine information of various nodes by appending their information inside a packet as they forward it to its destination with the consequent decrease of the amount of messages. By contrast, they increase the packet size by adding more data into the packet. This drawback is overcome by CS by being able to obtain an accurate approximation of the data by using a small number of generalised measurements, which are known as projections. We will explain the basic operation of CS in section 7.3.

The application of CS to WSNs could be viewed as a previous step to its application to VSNs. Nevertheless, since nodes in a VSN are moving along an urban area, they are not assigned to a determined location like sensor nodes are in a WSN. Thus, samples must include more information so as to know the specific location where the measurement was taken. In addition to this, the continuous movement of nodes cause effects like network partitions, platoons of vehicles and the corresponding high variability of the wireless links among them. These phenomena prevent a direct application of CS solutions for WSNs in a VSN scenario.

Our proposed solution consists in the application of CS to VSNs. It comprises two stages: (i) a query dissemination process where the FC broadcasts a message within a determined region of interest, and (ii) the harvesting stage where vehicles send their measurements back to the FC in an efficient way. In our case, efficiency refers to being able to send a low number of data packets having a smaller size compared to the case of gathering all the sampled data. To achieve that, the time at which nodes send their sampled data is controlled by a delay function. This function is designed so that nodes farther from the FC are the ones which start their transmission and intermediate nodes have enough time to append their sampled data as the data packet is forwarded. By using CS, the new data is combined with the one carried in a previous packet so that the overall size of the packet is hardly increased.

The remainder of this chapter is organized as follows. A review of the main related work is carried out in Section 7.2. Section 7.3 gives an overview of how CS works as well as the projections we have previously commented. In section 7.4 we give a thorough view of our proposal detailing every phase of the harvesting protocol. The evaluation of our proposal by means of simulations is achieved in Section 7.5. Finally, Section 7.6 concludes the paper.

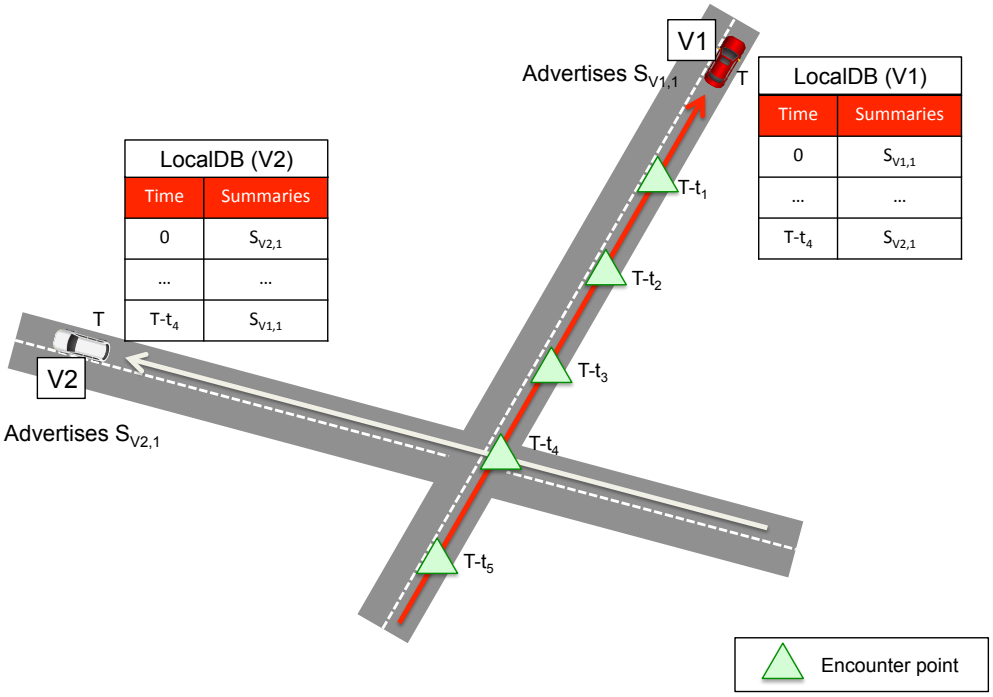


Figure 7.1: An image as example of a signal represented in traditional and wavelet basis.

7.2. Related Work

As commented above, our major interest in this paper is to define an efficient strategy to harvest the information of a determined urban Region of Interest (RoI) using the vehicles sensing equipment. Regarding this task, a survey of urban vehicular sensing platforms is provided by Lee et al. [117] where they analyse Mobeyes [118], FleaNet [119] and VITP [120] protocols.

Mobeyes, by Lee et al. [118], is a proactive urban monitoring strategy where every node that performs a measurement, sends a packet with a summary of its recent sampled information to its neighbours, Fig. 7.1. These summaries include, in addition to the sensed data, some other relevant information like timestamps and the position where the measurement was taken. In conventional sensor networks, this information is dispatched to the nodes responsible for analysing the sampled data. However, in VSN this mechanism is not practical due to the size of the generated data.

Besides, Mobeyes also allows an on-demand harvesting strategy (Mobeyes-ODH) that works as follows. First of all, the FC issues a query request, the receiving nodes immediately answer to this query with its data. Due to the fact that nodes answer as soon as they get the query, such scheme has the issue of generating a lot of packets that traverse and overload the network.

Later on, Lee et al. [119] propose FleaNet which is a *virtual flea market* service for urban vehicular networks. It is aimed at facilitating communications between buyers and sellers of goods and to efficiently find matches of interest because the will be very likely to end in transactions.

According to this protocol, nodes periodically broadcast their query for selling or buying items to its neighbouring nodes. These queries include a RoI where the advertisement must be spread in. After receiving the query, neighbours store the query without any further relying. This way, queries are only spread because of vehicle motion. FleaNet also provides a mechanism to deliver data using multi-hop

communications but as Mobeyes, nodes answer just after receiving the request message overloading the network with their answers.

The protocol comprises three kind of messages: query, match and transaction. First, a query message is broadcast to announce the goods a person possesses or seeks. When a node finds a query with a matching interest it issues a match message aimed at the query originator. Finally, a transaction message is transmitted to request for the transaction and its later reply.

Vehicular Information Transport Protocol(VITP) [120], allows nodes to aggregate information and report summarized results to the requester. This aggregation consists in piggybacking partial results so although they reduce the number of packets they still increase their size with each new partial result added to the packet.

These three sensing platforms are far beyond of our target in this chapter. They provide a mechanism to spread some information within a RoI avoiding flooding the network by only exchanging the information to the neighbouring nodes. VITP on its side it is also an application layer protocol, so it operates in a different layer and it needs a routing protocol to ensure the delivery of messages.

In the literature we can also find Delay-Bounded Vehicular Data Gathering(DB-VDG) proposed by Palazzi et al. [121]. This protocol follows the same strategy as we propose in this paper. Data harvesting is divided in two stages. In the first stage the FC node broadcasts a query message in a certain urban region. In the second one, nodes equipped with sensors that receive this query send back the sensed information taking into account the lifetime of the received query. Intermediate nodes will act as forwarders for this information by appending multiple samples into the same packet in order to reduce the overhead of the protocol.

The above proposals are not aware of one key aspect that we consider pretty important, the compression of the data packet information. These data packets usually contain information of more than one node, and their size depends on the number of intermediate nodes forwarding the packet or the areas of the region of interest that the packet must traverse to reach the destination. Thus, with an efficient compression technique we will be able to prevent the overload of the network.

This task has been widely studied in other research fields like WSNs, where the most efficient harvesting approaches have been proposed due to the particular constraints that they have to deal with to extend the sensors' battery lifetime. Among the different compressing techniques that have been applied, CS emerged with so much strength due to its high compression rate and accurate reconstruction of the original data.

Haupt et al. [122] give a review of CS as well as its application to WSNs. Two of these applications are worth mentioning: The application of CS directly to the transmitted information, even in the air by combining the signal using a technique called matched source-channel communication (Bajwa et al. [123]); and another scheme proposed by Feizi et al. [124] called *sparse distributed compression* which avoid the energy consumption in combining the sensing information within each node.

Although these techniques cannot be applied to our research domain there are other approaches that are more suitable for our research area. They compress the information using projection vectors.

Works like Chou et al. [125] apply CS in WSNs. In this paper, the authors propose an adaptive scheme managed by the FC. In a first step, nodes randomly send their measurement without any compression scheme to the FC. After receiving this information, the FC chooses a projection vector to obtain the information of the areas where it has a vague or imprecise information. This message is transmitted through the networks. Sensor nodes referred in the projection vector incorporate their

sensed information. Finally, once the projection vector is fulfilled it is sent back towards the FC.

Unlike WSNs, nodes in VSNs move inside an urban scenario. Therefore, even if compressing techniques in the network were applied, the FC will not have enough information about where these measurements have been taken. This is the reason why we do not apply compression in the network, but in the networked data using the projection vectors already mentioned.

On the other hand, more recently CS has been also applied theoretically to the VSNs in Yu et al. [126]. In this work, nodes gather information at intersections of the streets. After gathering several data, they apply CS to these samples that are transmitted to the FC.

Despite both works studied the application of CS, their approaches were only focused on the information that must be compressed without taking into account the phenomena and effects that mobile ad-hoc networks suffer. For instance, the variability of the wireless links between neighbouring nodes and other related issues phenomena like packets collisions, network congestion, or loss of signal which can also be produced and affect the quality of the communication.

This is our objective, analysing the impact of the CS in a VSN environment by means of network simulations. For this purpose we have defined an efficient harvesting scheme contemplating the whole process from the query broadcast by the FC to the reception of the compressed information of the sensed node by it.

7.3. Background

A straightforward introduction to CS is presented by Candès et al. [127]. In this paper, the authors motivate the use of this technology describing also its main purpose. The objective of CS is to capture efficiently the salient information of a certain signal of interest.

Traditional methods require to sample a signal at least at a frequency of twice the signal bandwidth (Nyquist-Shannon sampling theorem). However, all the elements of the signal are not always significant in terms of signal information and this is precisely the argument of CS.

For instance, if we ask somebody for a random number it will give it in base 10 by default. For instance 11, this base is called in signal context the *acquisition basis or sensing basis*, but we can change its representation using a different base like hexadecimal, obtaining a different value, b. Therefore, we have selected as *representation basis* the hexadecimal basis. Although we have two different values, 11 and b both of them corresponds to the same number, however b has only one digit while 11 has two of them.

The same happens to the signal representation. We can obtain a signal corresponding to an image whose components are the pixels and their colours, or we can use a different basis to represent it lowering the amount of information as Fig. 7.2 presents by using a wavelet transform (the black pixels of the wavelet transform does not contain information), being able to recover it without hardly losing accuracy.

CS asserts it is possible to recover certain signals from far fewer samples than traditional methods use. In order to do it, CS relies on two principles: **sparsity** and **incoherence**:

- A signal is **sparse** when the information rate is less than the signal bandwidth, i.e. only several of its components are relevant, being the majority of them dispensable and therefore not transmitted. For instance if the majority of the signal components have the same value, and only

Figure 7.2: An image as example of a signal represented in traditional and wavelet basis.

a few of them have a different one. Transmitting only these relevant ones in a compressed way, makes possible to reconstruct the original signal by the receiver while saving network resources.

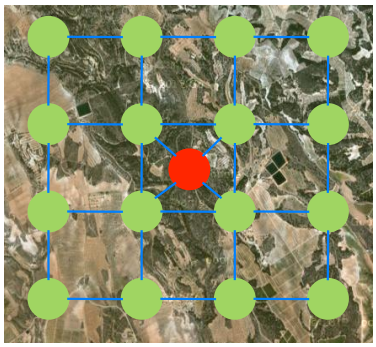
- **Incoherence**, on the other hand, measures the correlation of the data. The incoherence is applied to the sensing basis and the representation basis. Thus if a signal is sensed using a basis Φ while for its representation it is used another basis Ψ, the less correlation between the elements of both basis the more incoherence both basis will have. Samples must also be incoherent. Taking two samples, the more correlation between them the less information will be provided by the second data with respect to the first one.

Fulfilling the above principles, a signal is compressed in the following way:

$$\begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_k \end{pmatrix} = \begin{pmatrix} a_{11}a_{12}...a_{1n} \\ a_{21}a_{22}...a_{2n} \\ \dots \\ a_{k1}a_{k2}...a_{kn} \end{pmatrix} * \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_k \\ \dots \\ x_{k+1} \\ \dots \\ x_n \end{pmatrix} \Rightarrow$$
$$\Rightarrow y = \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_k \end{pmatrix} x \Rightarrow y = Ax \tag{7.1}$$

where $\boldsymbol{y}$ is the resulting compressed vector, A is the representation basis where the signal must be sparse, and $\boldsymbol{x}$ is the original signal also expressed as a vector. That is, the components of the signal x are ***projected*** onto the representation matrix that will be used to transmit the information in a compressed way expressed as the elements of the vector y, i.e. each component of y is the following way $y_k =< a_k, x >$.

CS guarantees that with a vector $\boldsymbol{y}$ with far fewer elements, k, of the dimension of the original signal, n, i.e. $k << n$, it is possible to recover with high accuracy the original signal $\boldsymbol{x}$.



Temperature in ºC

26	25.9	25.5	25.5
25.8	26	25.5	25.4
25.8	25.9	25.8	25.6
25.6	26	25.7	25.3

Figure 7.3: WSN measurements viewed as the components of a signal represented by a matrix.

Let us explain this projection operation with an example. Given a signal represented as a vector $x^T = (x_1, x_2, x_3)$, and a representation basis $A = (6, 2, 4)$ which for this example contains only one row. The projection is carried out like the next equation details.

$$y = Ax = \begin{pmatrix} 6 & 2 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 6x_1 + 2x_2 + 4x_3 = Z \tag{7.2}$$

So basically we have projected the vector $\boldsymbol{x}$ onto the projection basis $\boldsymbol{A}$ obtaining the projected value $\boldsymbol{y=Z}$. It is worth mentioning that initially we have a signal with three components x_1, x_2 and x_3 while the projected value give us only one component Z reducing the amount of information.

The application of this technique to WSNs is described by Haupt et al. [122] where the measurements obtained by each sensor of the network are seen as the components of a signal as Fig. 7.3 depicts, applying CS to compress this information and its following recovery.

Therefore, knowing the appropriate basis where this *signal* is sparse, nodes can take advantage of CS by means of projection vectors compressing the information as it is sent to the FC without a significant increase in the size of the packet.

Until now, we have explained the compression process with CS but we have not dealt with the other stage of CS which is the reconstruction process. It consists in the following minimisation of the l_1 norm:

$$\min_{\tilde{x} \in \mathbb{R}^n} ||\tilde{x}||_{l_1} \text{ subject to } \Phi\tilde{x} = y \tag{7.3}$$

where $||.||_{l_1}$ represents the l_1 norm of a vector that is $\sum |x_i|$.

Although the target of this paper is not the analysis of the different techniques to recover the original signal x, i.e. the minimization of the l_1 norm of the vector, there are several recovering procedures that are worth mentioning due to their good results and popularity. They are, a greedy algorithm called Matching Pursuit [128], as well as different improvements like Orthogonal Matching Pursuit [129] or Fast Bayesian Matching Pursuit [130], another statistical approach named Bayesian Compressed Sensing(BCS) [131], and a convex optimization approach [132].

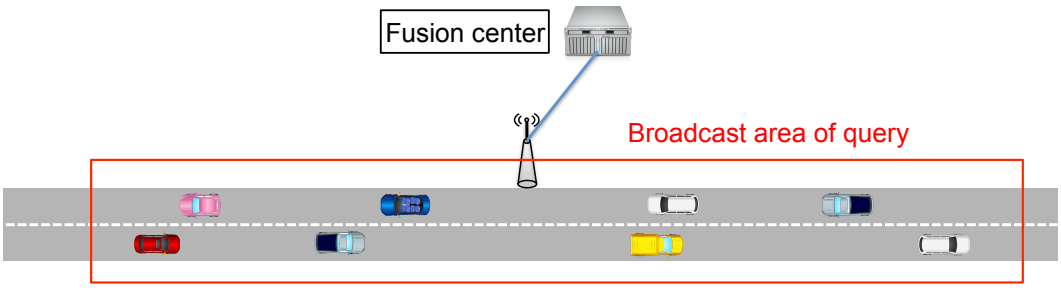


Figure 7.4: Example of RoI.

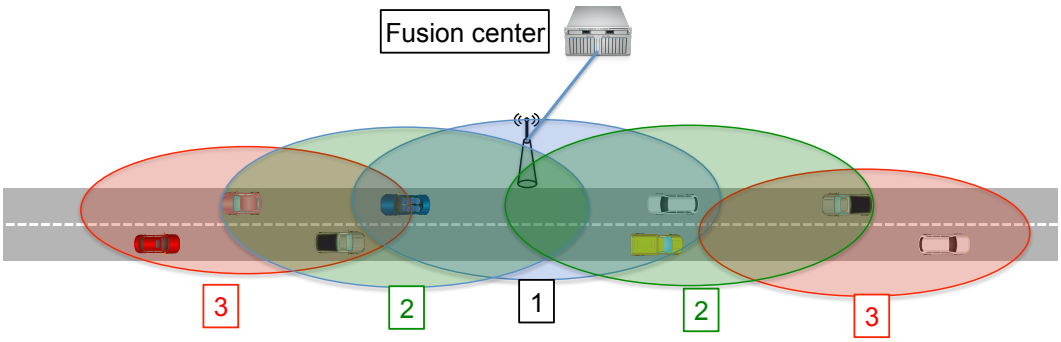


Figure 7.5: Query broadcast stage 2.

7.4. CS-based Vehicular Data Harvesting

Our main objective is applying CS to provide efficient data gathering protocols for the particular scenario of VSNs. For this purpose we have defined a solution called Compressed Sensing based Vehicular Data Harvesting (CS-VDH). It comprises two main stages:

In the first stage, as Figs. 7.4 and 7.5 show, the FC broadcasts a query message (CSQuery) within a specific Region of Interest (RoI). It is used to gather certain information like temperature, humidity, congestion, etc. This broadcast will be propagated by the nodes of the network which will forward the message until the RoI is covered.

The second stage, i.e. the gathering process, has been defined with the objective of reducing the number of transmitted packets as well as their length. So, as Fig. 7.6 depicts, the vehicles located farther from the FC are the first ones in sending their sensed information. Intermediate nodes take advantage of these data packets they have to forward to append their measured information. Rather than appending raw data, nodes use CS to include projected data so that extra information is added with very few additional bytes.

Fig. 7.7 provides a sequence diagram giving a complete view of the messages transmitted in our proposal as well as the processing activities that nodes must also fulfil so as to answer to the FC with the information they have sensed.

First of all, the FC broadcasts a query message which is flooded into the RoI. The underlying broadcast protocol is responsible for this task selecting the most appropriate neighbours to forward the CSQuery, in our example the CSQuery is forwarded by Node1 and Node2. Each node, after

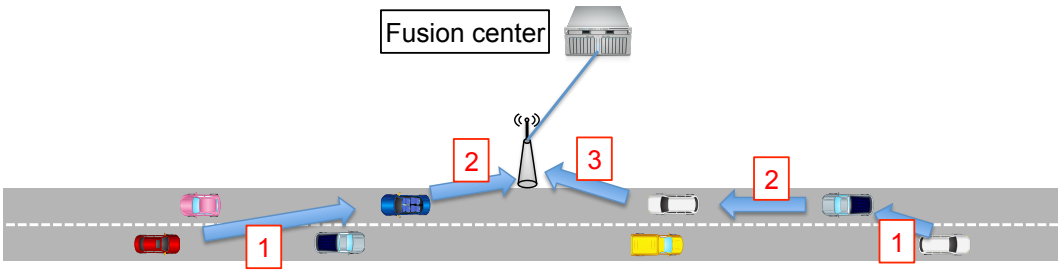


Figure 7.6: Harvesting stage.

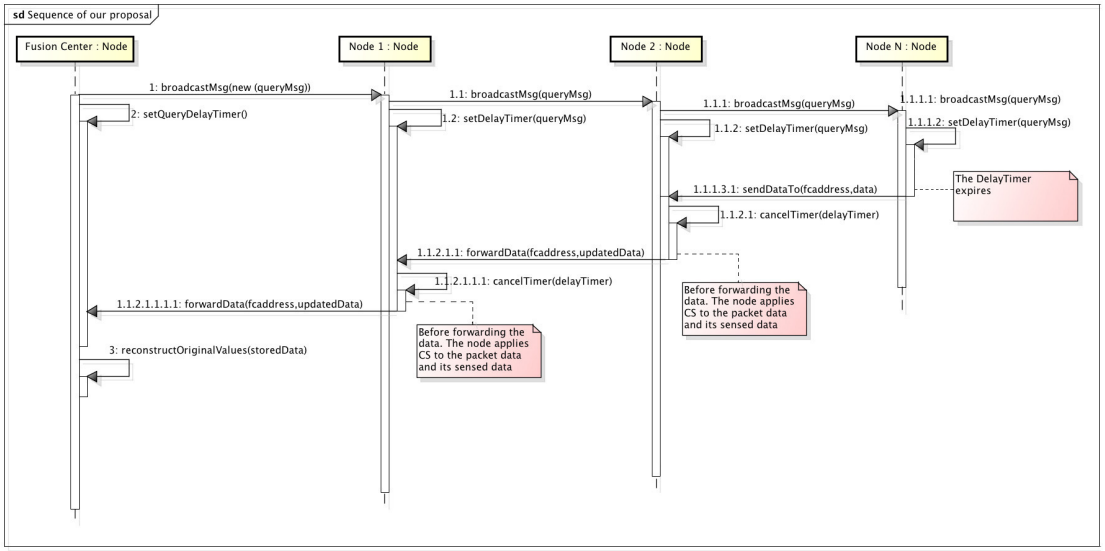


Figure 7.7: Sequence diagram of our proposal.

receiving the query message, sets up a timer, called delay timer. It represents the time they must wait before generating a new data packet with its sensed information. We will give a thorough explanation of this delay in following sections.

When the timer of Node N expires, i.e. the node farther from the FC, it starts sending back its sensed information to the FC. In order to do that, it generates a data packet and inserts this information as well as a boolean vector to indicate the areas where measurements were taken. As soon as an intermediate node receives the data packet, if its timer has not expired yet, it cancels the timer and combines its own measured data with that contained in the data packet. That combination of data is performed using CS as explained in section 7.3. This operation will be repeated until it reaches the FC.

On the other hand, when the timer of a node expires. It generates, as Node N did, a new data packet including only its own sensed information. Again, this packet will be sent to the FC passing through the intermediate nodes which will insert their information as previously commented.

7.4.1. Design issues

There exist several design issues worth mentioning in this paper.

Since the FC is not usually interested in measuring the whole urban scenario, it must **delimit the RoI** where nodes must take their measurements from. To do so, it specifies this region by the two opposites corners of a rectangular area for instance. This information is included in the CSQuery message and is used by the underlying geocast protocol to broadcast a message in that specific RoI.

One of the most challenging design issues is the **relation between positions and measurements**. Unlike in WSNs where nodes are located at fixed positions, in VSNs, nodes circulate through the streets along the urban scenario. So, basically in WSNs by just obtaining the identity of the source node which sent the data packet it is possible to identify the place where the measurement was taken. By contrast this relation does not exist in VSNs. Depending on the moment at which a measurement is taken by the same vehicle, the obtained data may correspond to different positions. Besides, a vehicle can gather information of different positions as it traverses along the road. So, the locations of the measurements must travel with the data somehow.

As commented in Section 7.3, the application of CS is based on the multiplication of two vectors of the same size. Since one of the vector, the projection vector must be sent within the CSQuery, we have to set its size a priori.

For this reason, we decided to divide the RoI in cells as Fig. 7.8 shows. Thus, instead of relating the measurement with the precise location of the nodes, we opted to make this relation with the cells of the RoI. This approach provides an extra advantage, if a node receives a data packet from another node and it already has the information of the cell where the node has taken its measurement, it will only forward the packet without wasting time to modify it.

Another interesting design issue is the **adaptation of the RoI matrix to a vector** since CS operates with vectors. This adaptation is straightforward. We concatenate the rows of the matrix obtaining a single row vector with all the cells of the matrix as shown in Fig. 7.9. Then, we insert in the CSQuery message the size of the rows of the previous matrix.

With this change, the operation is like in the previous CS example allowing us to compress the information of various locations.

$$y = Ax = \begin{pmatrix} 6 & 2 \\ 4 & 1 \end{pmatrix} \begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix} \Rightarrow \begin{pmatrix} 6 & 2 & 4 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \tag{7.4}$$

$$= 6x_1 + 2x_2 + 4x_3 + 1x_4 = U$$

What it is really interesting about CS is that the information can be combined at different moments. That is, if one node, for instance has the measurement of position (1,1). It can project its value obtaining $6 * x_1$, this information is transmitted in a data packet which is received and forwarded by an intermediate node towards its destination. But the intermediate one, can also combine its sensed information, for instance of position (2,2) just by adding its projected value $1 * x_4$ obtaining a new value $T = 6 * x_1 + 1 * x_4$ without hardly increasing the size of the data packet.

However, according to CS, a node can only combine its sensed data with that of the received data packet if the cells to which the data refers in both datasets do not overlap. So, a packet must include

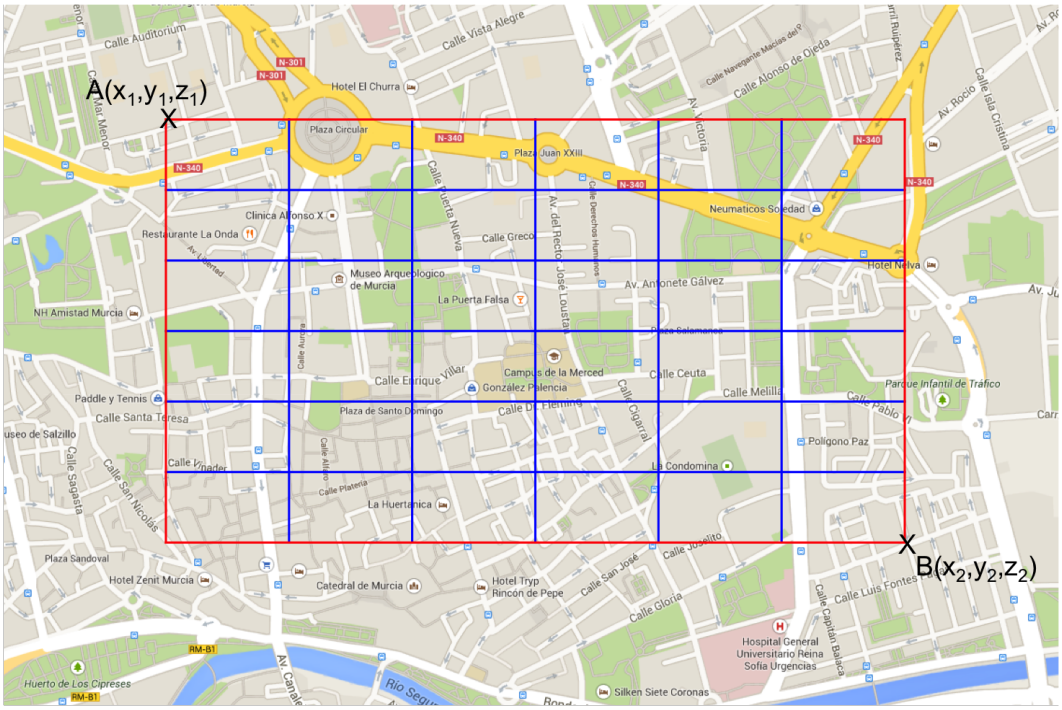


Figure 7.8: Definition of a RoI with 6x6 cells.

information about which cells contributed information to the carried data. In order to deal with this problem, we introduce a boolean vector indicating which cells provided information to the combined data. This vector has the same size as the number of cells in the RoI.

Based on the boolean vector, a node can easily check if its sensed information can be combined with that of a received data packet. This is done simply by checking that the cells with a *'1'* in its own boolean matrix are set to '0' in the received boolean matrix. Otherwise, the combination cannot be made. When a node combines data, it updates the cells of the resulting boolean vector accordingly.

Once we have commented the main obstacles that our proposed solution must overcome, in the next subsections we shed some light on the detailed operation of our proposed solution.

7.4.2. Query distribution

In the first phase of our proposal, the FC interested in harvesting a specific information of a certain RoI transmits a query message (CSQuery) that must be received by the nodes moving within such region. Thus, we must employ a broadcasting protocol to flood such query within the RoI. Broadcasting in VANETs is an issue that has been already investigated in VANETs and whose solution is the use of geocast routing protocols. Our proposed solution does not depend upon a particular broadcasting protocol. Thus, any of the latest solutions such as PIVCA [133] or AckPBSM [134] can be used.

As a matter of fact, we do not aim at defending a new broadcasting protocol. We just focus on deigning which information must be carried in the CSQuery message for CS to be applied.

As we motivated in Section 7.4.1, the following fields are necessary for the query's content:

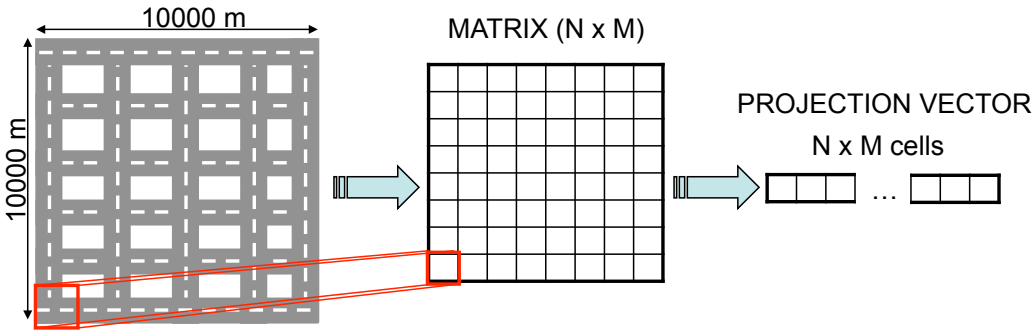


Figure 7.9: Matching process to obtain a projection vector.

- A sequence number to identify the packet.
- The FC address.
- The FC location.
- The projection vector.
- The number of cells per row.

From those fields, the address and location of the FC are needed for nodes to send back their sensed information. The number of cells is also important for the vector-to-matrix conversion. Finally, the main component needed to apply CS is the projection vector, which allow nodes to compress their information by projecting their sensed measurements.

The content of this projection vector is very important because it must guarantee the correct operation of the recovery stage of CS. The purpose of the projection vector is to combine the information in such a way that its recovery can be accurately performed. In Section 7.3 we explained that CS relies on two principles to its correct operation, **sparsity** and **incoherence**. The projection vector must transform the sensed measurements changing its representation in a sparse domain where only a few components are relevant. The main obstacle is that there is not a single representation basis that makes all kind of measurements to change their representation to a sparse domain. By contrast, this representation basis, and therefore the projection vector must be chosen according to the kind of measurements we want to take, CO_2 concentration, average speed of an urban sector, humidity, etcetera.

7.4.3. Harvesting process

The harvesting process is the most challenging task of our contribution from the point of view of the information of the data packets. It is where CS is performed to compress the information as it traverses the network. The way these packets are delivered from the source node to the destination is not the target of our work. Actually, there are many VANET routing protocols that are able to deal with this issue such as gpcr [57], GPSR [38], SAR [64], GeOpps [68] or BRAVE [135] to name a few.

On the other hand, the content of the data packet is also relevant for the application of CS and its later recovery process. Each data packet contains the projection value and a boolean vector with the

	Areas					**VALUES**
	1	2	3	4	5	
Node1	X	X				P1=A*X_1+B*X_2
Node2			X		X	P2=C*X_3+D*X_4
Combin.	**X**	**X**	**X**		**X**	**P1+P2**

Table 7.1: CS: Combination of data.

	Areas					**VALUES**
	1	2	3	4	5	
Node1	X	X	X			P1=A*X_2+B*X_2+C*X_3
Node2			X		X	P2=C*X_3+E*X_5
Combin.	**Not possible**					

Table 7.2: CS: Combination of data not possible.

same number of items as the projection vector transmitted in the query message. This boolean vector indicates the areas whose values are already appended to the data packet. That is, the first node in generating its data packet will send its sensed information marking those cells of the boolean vector corresponding to the areas where the measurements were taken. When intermediate nodes receive them, if they can apply CS to the packet inserting their own information, they will also mark in the boolean vector their areas corresponding to the appended information.

For our proposal we have defined two different operation modes for the harvesting stage. In the first one, when the delay timer expires or a data packet has been received by a node, it will insert its sensed data of its current location. In the other operation mode the node does not only insert its measurement in its current position but also those of previous locations.

In the first operation mode, when a node receives a data message from another node, it checks if the boolean vector has a free cell which matches its current position. If so, it modifies the boolean vector by marking the cell corresponding to its position and combining its projection to the current value of the other node. Otherwise, the node just forwards the message.

In the second one, the process is a bit more complex because the node that receives the packet may have also another one with its own measurements. In this case, the node must check if the two boolean vectors are compatible. This happens when both vectors are disjoint, i.e., when they do not share any values. Let us explain it better with a couple of examples:

Let us assume that the information of two nodes are the ones indicated in Table 7.1. Node 1 has sent a message with the information of the areas 1 and 2. That is, the projection P1 and the boolean vector (1, 1, 0, 0, 0) corresponding to the first row of the table. This message is received in its way to its destination by Node 2 who has already stored the information of the areas 3 and 5 so obtaining the projection P2 and the vector (0, 0, 1, 0, 1) (the second row of the table). Since their information is compatible the combination of both data is possible combining the information as the last row indicates.

However, in Table 7.2, the information of both nodes is incompatible. The projections cannot be combined because both sets of areas are not disjoint because both have inserted the value of the Area 3, and if we combine them we will introduce twice the term $C*X_3$ in the compression form which is not correct from the point of view of CS. In this case, Node 2 will forward Node 1's message generating a new data message with its own information too.

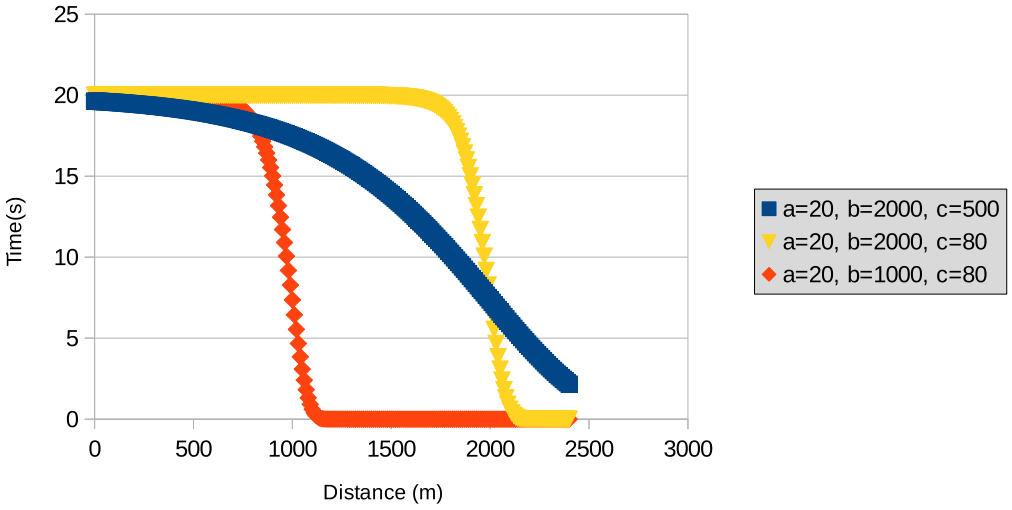


Figure 7.10: Different shapes varying the parameters of the formula.

7.4.4. Enhancement to the basic scheme

If every node inside the RoI transmits its sensed information at the same time, the network will be notably overloaded and the data communication will be jeopardized in such a way that only a few data will arrive to the FC due to the contention, lack of resources, etcetera.

This undesired behaviour can be mitigated by sorting the different responses in such a way that instead of having all the nodes sending their sensed information at the same time, they gradually generate these responses making intermediate nodes to take also advantage of the already generated responses to introduce their sensed information in them. So, nodes farther from the FC will be the first ones in answering with their sensed information and intermediate nodes take advantage of the packets they receive and must be forwarded to the FC to append their information.

For our proposal we decided to use a delay function expressed by the following equation

$$y = a * e^{-e^{\frac{x-b}{c}}} \tag{7.5}$$

and whose parameters are a, b, and c. We have used this function because of its flexibility. That is, by varying the values of its parameters, the function can change their shape as Fig. 7.10 presents.

We have adjusted the parameters to make the nodes farther from the FC more than a specific distance to answer instantaneously increasing this answer time exponentially to guarantee that intermediate nodes have enough time to receive and forward them appending their information. Each of the parameters model a determined part of the function. The parameter a limits the upper asymptote, that is, the maximum time (TMAX) a node must wait until it sends back its sensed information. By contrast, b models the distance from which the answer time must grow exponentially. And finally, c models the amount of points that the slope will have.

Therefore, after receiving the query message, each node calculates its distance to the FC. This

operation is straightforward because all the nodes are able to obtain their locations and the FC location is transmitted within the query message. Introducing this distance as the x variable it will obtain the time that it must wait until generating its own data packet setting it into de delay timer.

7.5. Evaluation

Our proposed solution comprises two different stages which need a different protocol to undertake each task. On the one hand, we need a geocast protocol to spread the query message inside the RoI and, on the other hand, we also need a routing protocol capable of dealing with the features of the VANETs that assures the delivery of the messages to its destination.

Among the existing geocast solutions we have selected AckPBSM [134] for the query's broadcast, because it is an adaptive protocol, suitable for vehicular scenarios and which have been probed to be efficient and reliable in vehicular networks. So, it guarantees that the distribution of the query packet among the different nodes of the network will be performed in an efficient way.

On the other hand, for the harvesting stage we have selected BRAVE [135], which has also obtained a good performance in terms of packet delivery ratio and which also provides support for Delay Tolerant Network (DTN) allowing nodes to carry themselves the packets until they find a suitable neighbour for their destination.

Finally, the last aspect that we must tune is the delay function used to set time the nodes must wait before generating a new data packet with its sensed information.

Since the simulated scenario has a RoI of $4047 * 5047\ m^2$ and we have set our FC in the center of the scenario we have adjusted the following parameters: $b = 2000$ which selects the distance where the slope of the function is located, and $c = 80$ indicating the number of values in the slope as well as the smooth of the slope. So, setting these parameters we ended up having the following equation

$$y = T_{MAX} * e^{-e^{\frac{x-2000}{80}}} \tag{7.6}$$

which corresponds to the second series of Fig. 7.10.

TMAX, which correspond to the a parameter, is the value of the lifetime of the response. That is, the maximum waiting time for the FC to receive the sensed data. For our simulations we have set a TMAX value of 20 seconds.

7.5.1. Simulations

We have evaluated our proposal's performance by means of simulations using *The Network Simulator NS-2*[1], version 2.33. For this purpose we have developed an urban scenario with the SUMO tool[2] taken from the main streets and highways of the city of Murcia, Spain, as shown in Fig. 7.11 where vehicles move during $445s$ and where the following vehicles' densities: 1/50, 1/45, 1/40, 1/35, 1/30, 1/25, 1/20, 1/15, 1/10 and 1/5 $veh/route/s$ have been defined.

Regarding the signal propagation model, we have used TwoRayGround for our simulations defining a coverage range of $250m$. In addition, we have run 10 different executions per scenario and vehicle's density showing in the graphs a 95% of confidence interval.

[1] http://www.isi.edu/nsnam/ns/
[2] http://sumo.sourceforge.net/

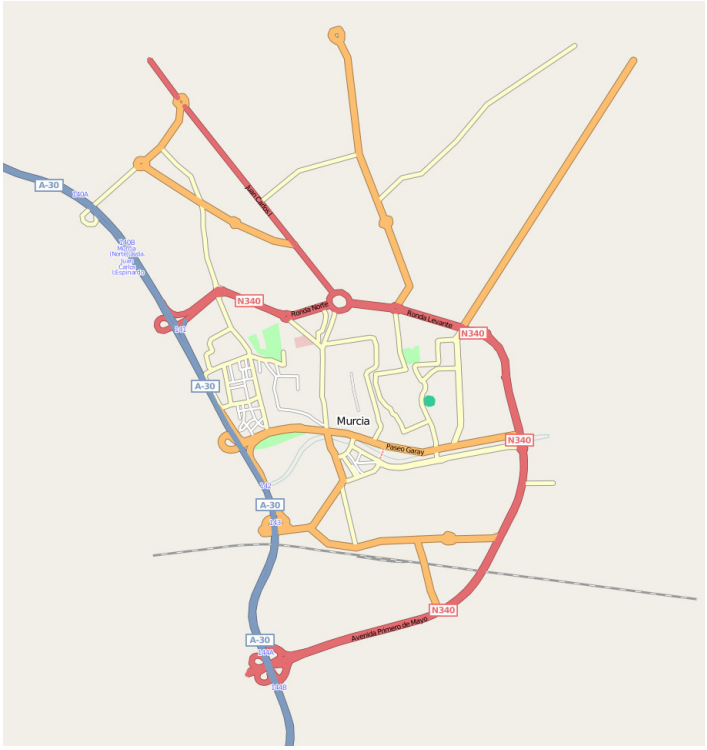


Figure 7.11: Urban scenario of the city of Murcia used in our simulations.

Since in the harvesting process there is only one FC, we have varied the seed of a random component of the timers making different vehicles to answer first in each execution of the simulation.

In Section 7.2, we presented different schemes found in the literature which deal with the harvesting process. Particularly, there are studies focused on WSNs and VSNs which deal with the application of CS to mobile ad-hoc networks. However, they only deal with the content of the packet without taking into account the mobility of the nodes and its effects in the vehicular network.

So, we will evaluate our proposed solution performance comparing it against DB-VDG, which is a VANET data gathering scheme. In addition, we will also compare variants of our proposed solution.

7.5.2. Comparison against DB-VDG

Comparing our proposal to DB-VDG as it was originally designed would be unfair because in DB-VDG, every node at receiving a packet inserts its sensed information without taking into account the area in the map where the measurement was taken. That is, if a packet is received by different nodes in the same area they will append their information dramatically increasing the packet size as it is forwarded by intermediate nodes to its destination. For this reason, we have adapted it to perform like our proposal does. That is, the urban region of interest is divided into areas and a node using DB-VDG will only insert into the received packets the sensed measurements of its current area, but only if this information is not already inserted in the packet.

After making these modifications to DB-VDG we have compared both proposals as Table 7.3 presents. As we can observe, the overhead of DB-VDG in its two variants using SBSS and DBSS is

Density	DBVDG-sbss	DBVDG-dbss	NOCS-TMAX20	CS-TMAX20
50	563728.4	514996.4	735.7	513.9
45	712439.3	592476.4	274.5	243.7
40	882573.4	685165.7	840.7	669.4
35	786394.4	597975.6	2390.6	2080.1
30	1007816.0	803683.1	2261.3	2022.6
25	1127205.2	878429.9	2216.6	1924.2
20	1114846.7	905167.5	3239.5	2725.2

Table 7.3: Control overhead of both proposals in number of messages.

huge. This enormous overhead avoids a good dissemination of the information along the network and complicates the delivery of both query dissemination and harvesting packets. The reason for such an overhead is caused by the heterogeneity of the scenario which makes DB-VDG to generate a great amount of messages for both the query flooding and gathering stage.

On the other hand, our approach, in all its variants, obtains a better performance in both the query dissemination and the harvesting process overhead which is about 200 times better. In addition, regarding the number of messages generated during the simulation, our compression proposal outperforms the one without compression reducing the amount of messages that are required. So, in light of these results we can state that DB-VDG does not scale in vehicular networks which is a main requirement in these networks. By contrast, our proposal seems to scale much better than DB-VDG as depicted in both figures.

7.5.3. Impact of CS in the overhead

We have shown that the improvement of our proposed solution compared with DB-VDG are notable. However with this comparison we do not have analysed the benefits of CS regarding the packets content. For this purpose we are going to focus on different variants of our proposal.

We have compared three variations of our strategy: two of them integrate CS whereas in the last one, intermediate nodes will simply append their sensed information to the packet without applying any compression technique. The difference between both CS variations consists of the two different operation modes that we have commented in Section 7.4 CS with and without accumulation. We refer to the first one by *CSPure*, to the second one by *CSAccum* and to the one without CS by *NoComp*.

In Fig. 7.12 we measure the amount of sensed values that nodes are able to carry in both approaches without taking into account the packet size.

We consider the *NoComp* approach the best possible solution in terms of amount of delivered packets due to its straightforward operation mode where new information is appended directly in the packet in case it does not exists.

Regarding the compression variants, both *CSPure* and *CSAccum* take more time to comprise the information being *CSAccum* the one with the most complex operation mode. Although by CSAccum, a node can insert more sensed values at once, the compatibility problem penalizes its performance. Despite that, its performance in terms of carried values is also good.

In light of this graph, the *NoComp* approach seems to be the best option to carry more sensed values but we do not have into account the overhead in the whole network. Figs. 7.13 and 7.14 show the control overhead of both proposals with respect to the simulated densities. This way, now we are

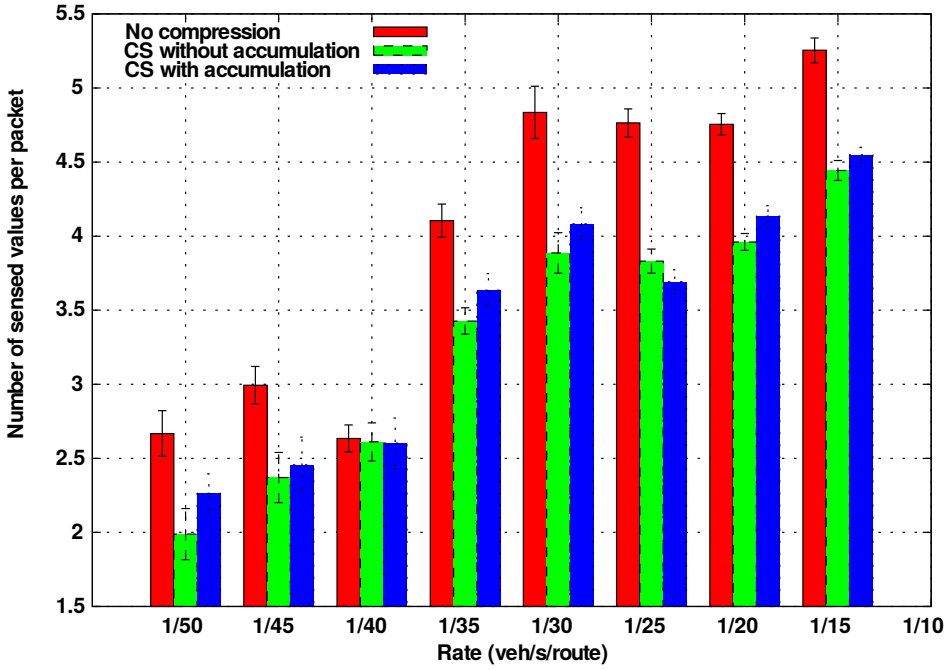


Figure 7.12: Number of sensed measurements carried in packets for CSAccum and CSPure for TMAX=20s.

measuring the amount of packets traversing the network.

As we can see the benefits of CS with respect to the overhead are more notable as the density of vehicles increases. The reason for this is that under lowly dense scenarios, for a distant node, the probability of finding a way to reach the FC is really low. So, since the timers of nodes close to the FC will expire with high probability. This reduces the possibility of combining data in CS. Thus, the performance is similar to the *NoComp* approach.

On the other hand, in dense scenarios like 1/15, 1/10 and 1/5 $veh/s/route$ the results are totally different. Communication among nodes is more fluent, i.e. finding with higher probability promising nodes which can forward data packets to the FC before their delay timers expire. This allows them to combine their information and thereby reducing the network overload.

In addition, overhead increases as the density does. This trend is smoother in both CS strategies with a maximum overhead of about 12000 messages for the highest density whereas in the strategy without compression the curve is more pronounced reaching nearly 18000 messages in the same scenario. This trend therefore confirms the better scalability of our approach in terms of control overhead.

7.5.4. Impact of the maximum waiting time

The delay function governs the time that nodes must wait for sending their sensed information issuing a data packet. This delay function is parametrized by three parameters: `a`, `b` and `c`. The parameter `a` is named also as TMAX and indicates the maximum waiting time for a node to send its information. Parameter `b` indicates the distance where the slope was placed in the graph. Finally, parameter c guarantees that there are enough points in the slope, that is, enough timestamps to assure

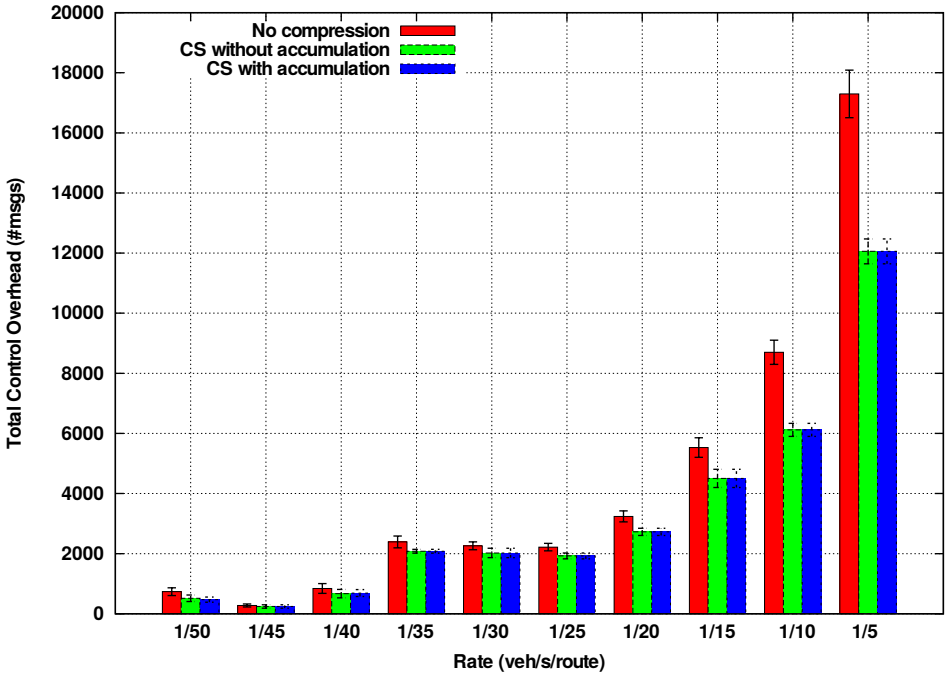


Figure 7.13: Overhead of compression strategy vs no-compression strategy in msgs for TMAX=20s.

that intermediate nodes obtained different delays

Variations of this parameter `b` will make nodes located located close to `b` meters far from the FC to answer immediately. The point is to set the parameter `b` with a value close to the distance between the FC and the limists of the RoI. This way, the slope of the graph will make nodes farther from the FC them to be the first ones to answer.

Parameter `c` guarantees that there are enough points in the slope, that is, enough timestamps to assure that intermediate nodes obtained different delays. A low value for this parameter makes no sense in our approach because it will impair the compression operation since intermediate nodes would not wait for the packets of the farthest nodes.

So, we conclude that an study of the TMAX parameter is needed since this parameter must be adjusted according to the lifetime of the gathered information. We have studied the impact of TMAX parameter with the following values 2, 5, 7, 10, 20, 40 and 50 seconds.

Fig. 7.15 presents the control overhead of both strategies with and without compression. We have obtained results for CS with different values of TMAX (2, 5, 10, 20 and 40 seconds) and we have compared it with the strategy without compression with a value of TMAX of 20 seconds.

In this heterogeneous urban scenario, CS outperforms the no-compression strategy when the TMAX value is higher than 20 seconds. In fact, in light of these results a TMAX of about 10 seconds produces a similar performance to the no-compression strategy which is therefore a threshold value in this not favourable scenario. So, if the requirements of the lifetime of the data is higher than 10 seconds the CS approach will be a suitable technique to harvest information of the vehicles in our scenario.

Another interesting conclusion we can obtain of these graphs is that the advantages of using CS are more notable as vehicles density increases. In low density scenarios, vehicles are less likely to find

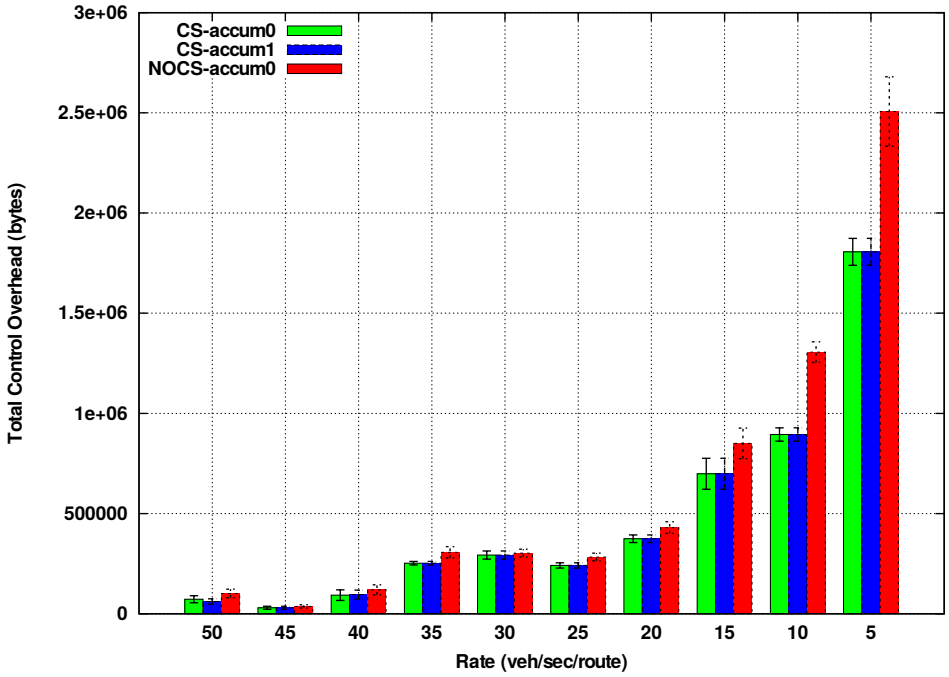


Figure 7.14: Overhead of compression strategy vs no-compression strategy in bytes for TMAX=20s

soon a neighbouring node to forward the data to the destination. For this reason, their delay timer expires very often despite having a TMAX value of near 1 minute. However as the density increases the probability of finding a neighbour increases in such a way that even a difference of a couple of seconds is notable in the graph too.

This also occurs when measuring the carried sensed values. Fig. 7.16 again presents a rising trend as the TMAX values increases for every density. This trend reaches its maximum in highly dense scenarios where the maximum number of data is reached with a value of TMAX of 20 seconds, e.g. with a density of 1/30 $veh/s/route$. In addition, we can figure out that there is a critical gap of the 10 first seconds in which the number of data is more variable.

Regarding the differences between using an accumulation strategy, given that the TMAX value increases, there is not a significant variation between using CS with or without accumulation. That is, it influences equally both strategies.

7.5.5. Reconstruction of the data

Although our interest in this work is to analyse the goodness of CS within a VSN as a good approach to harvest information, in this section we give an overview of the reconstruction process without detailing its operations. We have used the Bayesian Compressed Sensing (BCS) technique to achieve the reconstruction. So, we have introduced the collected data of one of the harvesting operations from one execution of the simulations with density 1/15 $veh/s/route$ in Matlab [136] to reconstruct the original data, that is the real values measured by the nodes, using the code of BCS using a Relevant Vector Machine(RVM)[3].

[3]http://people.ee.duke.edu/ lcarin/BCS.html

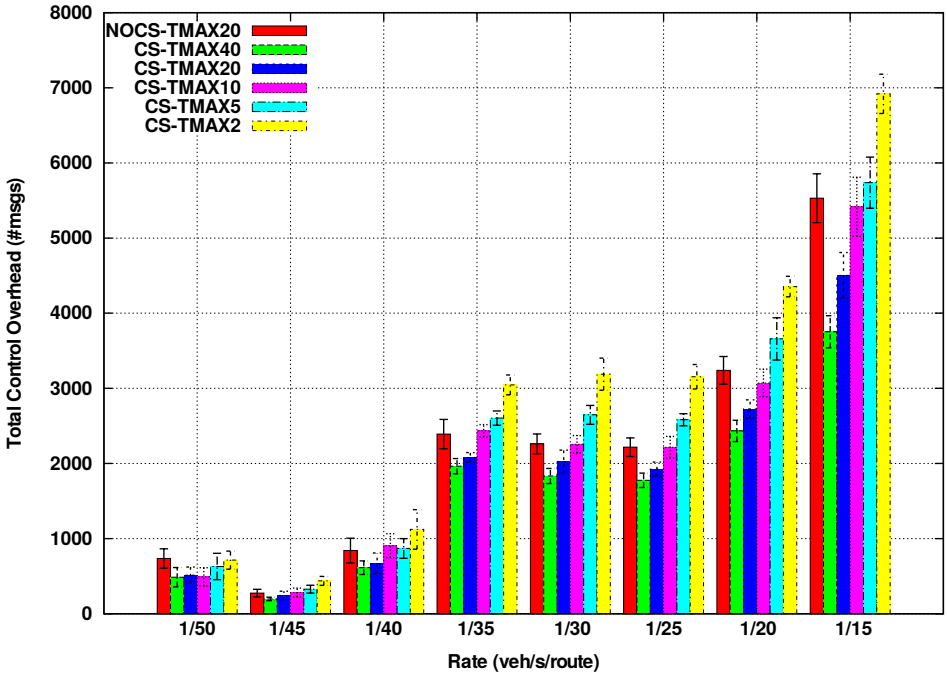


Figure 7.15: TMAX impact onto the overhead without accumulation.

Original data				
30.00	29.80	29.90	30.20	30.00
29.70	29.60	29.20	29.00	29.70
28.90	28.60	28.90	29.00	28.90
29.50	29.80	29.90	30.00	29.50
29.80	29.90	30.10	30.00	29.80

Table 7.4: Table of original data.

Initially we set the temperature according to a real temperature map of the city of Murcia to be the most realistic as possible. From this map we extracted a matrix of temperature, in ^oC, that corresponds to the table 7.4.

Nodes will perform their projections taking the difference between its current value and the average of the previous measured also inserted in the query message.

In this execution the FC has received 51 data packets with some replicated information. So, in Table 7.5 we present the data in 18 rows without duplicates. Each data packet received by the FC corresponds to each row.

Let us focus on the first packet received by the FC, i.e. the first row of the table. The sequence of 0s and 1s that we extracted to Table 7.6 corresponds to the cells of the RoI where the measurements were taken. The first five elements corresponds to the cells of the first row of the RoI. The second five elements to the second row and so on (see Table 7.7). On the other hand, the right most column provides the projection values associated to the measures received by each node.

Although, there are several cells whose data is not provided like 4 (4, 1), 5 (5, 1), 6 (1, 2), 11 (1, 3) and 20 (5, 4) with CS we have obtained the reconstruction of the data showed in Table 7.8.

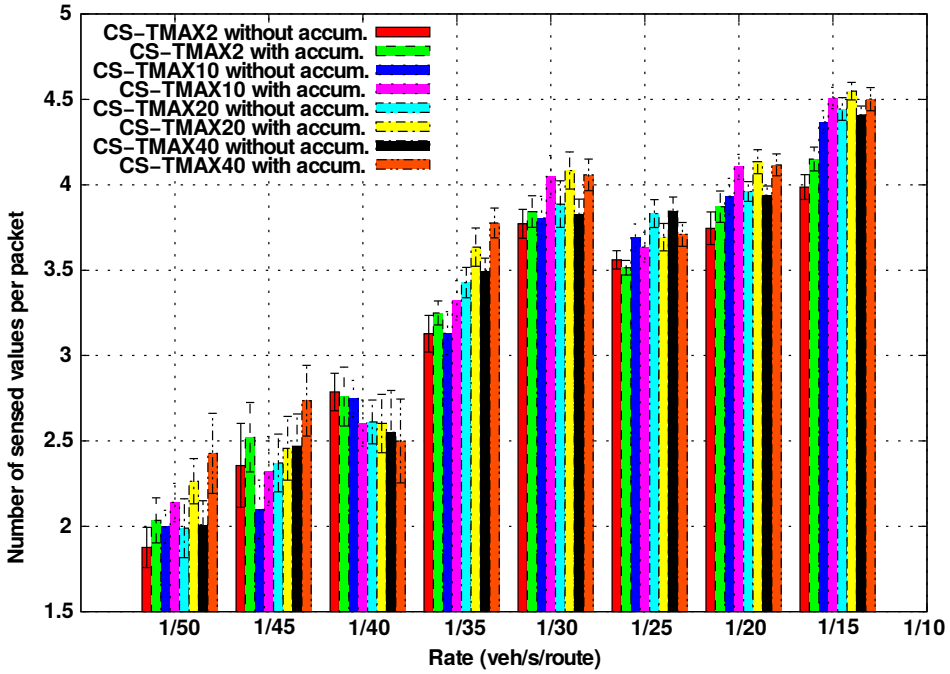


Figure 7.16: TMAX impact onto the carried data by nodes.

So, observing both tables we can conclude that the reconstructed data are really close to the original values obtained by only the projections of the nodes.

In Figs. 7.17 and 7.18 we present the root mean squared relative error (RelError) obtained for each simulated density of vehicles and varying also the TMAX parameter.

As we can observe, the *RelError* is lower than 0.45% for *CSPure* and 0.3% for *CSAccum*. In light of these results we can also manifest that the error in the reconstruction is independent on the TMAX parameter. On the other hand, there exists a relation between the *RelError* and the density of vehicles. This relation makes sense because the denser the scenario the more data the FC will receive. Therefore the more accurate reconstruction it can obtain.

7.6. Conclusions and future work

In this chapter we have presented a new harvesting protocol for vehicular sensor networks. This proposal makes use of compressed sensing, more concretely the concept of projection to compress the information without increasing the packet size at it traverses the network being forwarded by intermediate nodes towards its destination.

This technique outperforms previous proposals like DB-VDG in terms of overhead. We have also compared our proposal with a variation without compression. Although the *NoComp* variation is able to carry nearly up to one sensed value per packet more than our CS solution, the total overhead of *NoComp* is higher than the CS approach. In addition, the benefits of CS where it is not necessary to receive all the values to accurately reconstruct them make the difference in terms of the amount of carried values negligible.

We have also studied the variations of the parameter TMAX of the delay function which sets the

Obtained boolean vector and associated projections																							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	**1.2360**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	**1.1355**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	**1.3472**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	0	**1.6474**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	**0.5229**
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	1	1	**0.7597**
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	1	1	**0.6594**
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	1	0	0	0	**0.0590**
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	**-0.5531**
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	1	0	0	0	0	**-0.3414**
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	1	1	0	0	0	**-0.0290**
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**-0.3878**
0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	**-2.1380**
0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	**-1.2409**
0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1	1	0	0	0	1	0	0	0	0	**-1.0292**
0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0.1472**
1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0.1481**
1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**-0.4400**

Table 7.5: Table of received data: boolean vectors and projections.

Boolean vector received by one node																								
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0

Table 7.6: Information corresponding to the first row.

maximum time for nodes to answer to the FC with their sensed information. Due to the particularities of our scenario, we have seen that a TMAX value higher than 10 seconds ensures that CS-based schemes outperform the other alternatives in terms of delivered sensed values.

As future works we are considering the application of CS on different urban scenarios. Another interesting task is that of setting the FC in a different place, even outside the scenario as well as evaluating the use of more than one FC. The simulation of the deployment of gateways with access to the infrastructure network in the scenario can be another interesting future work. Finally, we also propose an study of the best representation of the information to obtain a higher accuracy in CS.

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	0	1	1	0

Table 7.7: Boolean vector matched to the cells of the RoI.

Reconstructed data				
30.10	29.95	29.87	30.31	30.09
29.65	29.55	29.18	29.02	29.44
29.31	28.83	28.98	29.13	29.02
29.36	29.79	29.86	29.95	29.97
29.77	29.83	30.10	30.05	29.78

Table 7.8: Table of reconstructed data.

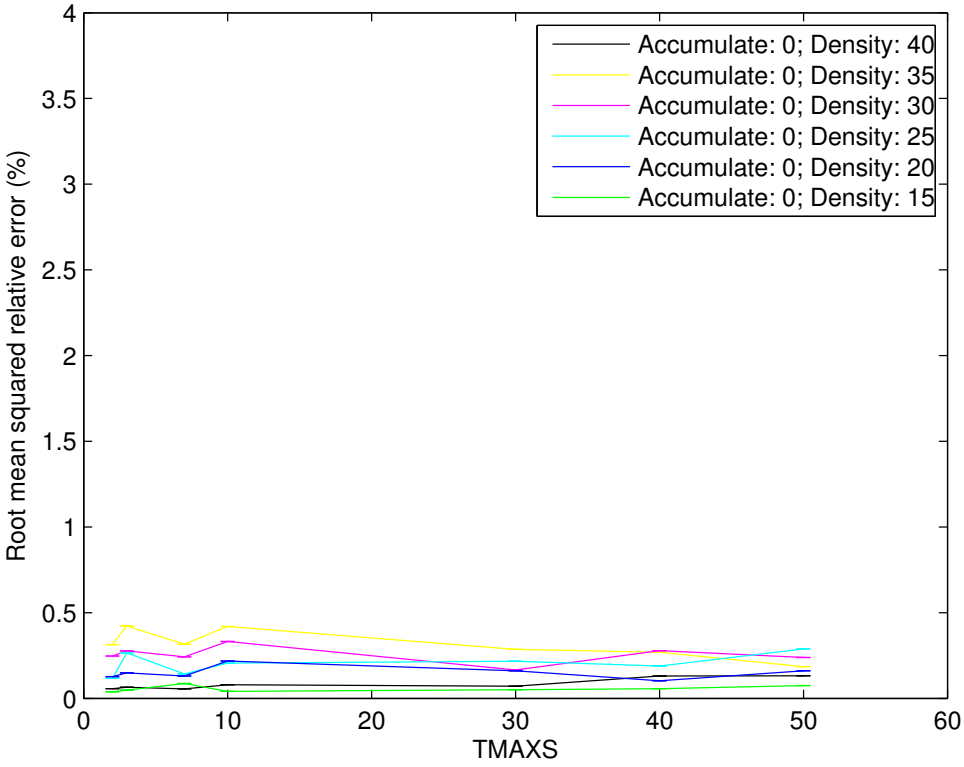


Figure 7.17: R.M.S Relative Error for CSPure.

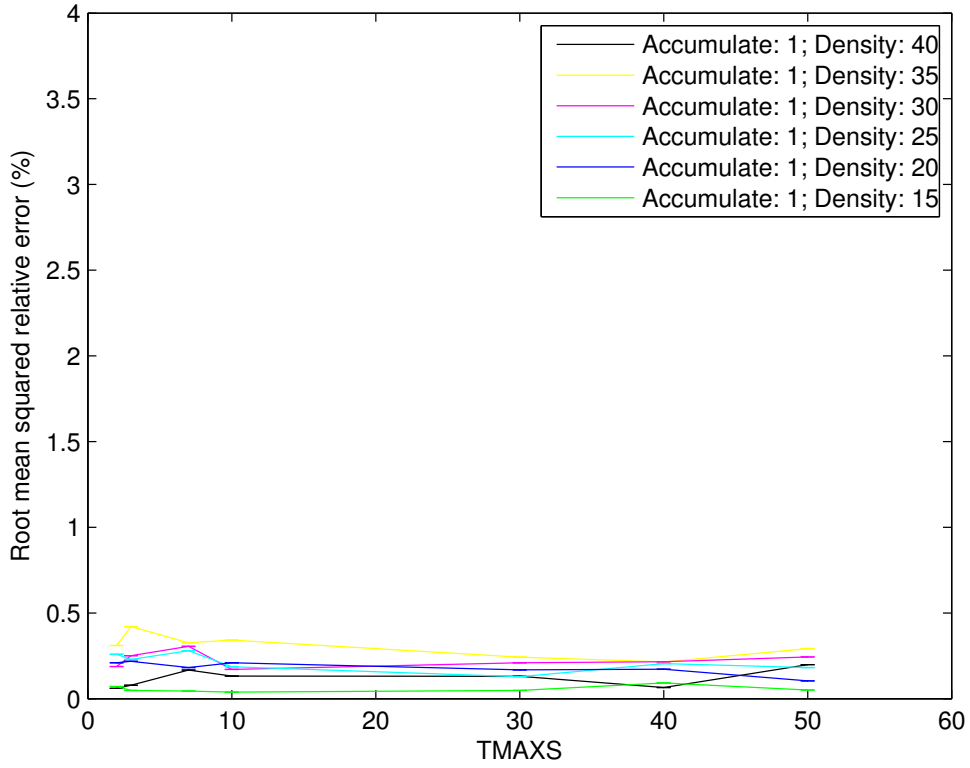


Figure 7.18: R.M.S Relative Error for CSAccum.

Chapter 8

Conclusions

8.1. Summary and Main Contributions

This thesis focuses on studying different aspects of Vehicular Ad-hoc Networks (VANETs) like routing, security and data gathering. These problems have been addressed following a methodology which ensures the suitability of our solutions.

Our first contribution consists of a VANET-specific routing protocol called Beacon-less Routing Algorithm for Vehicular Environments (BRAVE). This protocol takes advantage of the different flaws and design problems of previous proposed solutions found in the literature. Unlike previous proposals, routing decisions are made opportunistically by neighbouring nodes. This way, instead of being the current forwarder the one that selects the next hop, neighbouring nodes propose themselves by sending a response after overhearing the DATA packet. A delay function is used to sort these responses in time. Thus, neighbouring nodes providing more advance towards the destination answers first to the current forwarder. BRAVE also takes advantage of the urban map to build a graph whose vertices are the the different junctions of the map and whose edges are the streets that connects them. This graph is used to figure out the shortest path to the destination, but it is also used to assist the routing protocol introducing in the closest junction as a first destination for the data packet to be sent. Finally, its Delay Tolerant Network (DTN) support makes it a very reliable protocol being able to obtain really high packet delivery ratio with a reasonable end-to-end delay.

Despite these good results, our design, as well as the other proposed solutions found in the literature ideally assume the collaboration of all the nodes of the network to achieve the task of handing over packets to their destinations. However, in practice there can be malicious users that can hinder the network by employing different kinds of attacks. This is the reason why we continued our research by making routing protocols more secure. Our second contribution strengthens the behaviour of our previous routing protocol under hostile scenarios. To do so, it adopts the public and private pair of keys of PKI allowing nodes to sign their messages avoiding their manipulation by malicious users. A certificate exchange strategy is also proposed to reduce the overhead of the network. Finally, we evaluate the use of guard nodes to reinforce the packet delivery when they reach malicious nodes which propose themselves as the best forwarders, but they drop these messages.

These two contributions consider a VANET comprised only by vehicles equipped only with an 802.11p interface. That is, only Vehicle-to-Vehicle (V2V) communication is considered. Although this

special case of VANET is possible, as vehicles enter in urban environments they can find different APs offering connectivity to them to the infrastructure network. In addition, they can also be equipped with other wireless technologies which allow them to directly communicate to the infrastructure, for instance 3G, 4G or LTE technologies. This new scenario sets out a new challenge in terms of both routing and security.

We extended BRAVE to take advantage of the new available elements enhancing both packet delivery ratio and average end-to-end delay. It incorporates the following new features. It is able to select the most suitable network interface depending on the QoS requirements of the packet to be sent. It is also able to use the connectivity to the infrastructure to shorten the path followed by packets when source and destination nodes are far away one from the other, if this is the most appropriate path. It can also take advantage of nearby RSUs to communicate with the infrastructure even when they are several hops away. Our new protocol is able to reduce the average end-to-end delay down to 15 times the one obtained in an isolated VANET.

Since users are now able to connect to the infrastructure they will demand accessing their subscribed services. However, if vehicles access the infrastructure through RSUs and APs deployed along streets and roads, the time such elements are available is limited. Since the authentication process can last up to a couple of seconds and nodes desire to make the most of this precious time in accessing their services, we use a different authentication strategy. We propose the use of a pre-authentication scheme which takes advantage of the current authenticated gateway to perform the pre-authentication with a new promising gateway. This saves time as the node comes closer to this new one. This strategy offers very promising results according to our simulations. It increases the packet delivery ratio, and reduces the control overhead and the end-to-end delay.

Finally, we also contribute to the application of VANETs to gather information in urban environments. In such scenario, vehicles act as mobile sensors that provide such information in urban environments. We propose the use of a compressing technique well known in the field of information theory called CS. This technique allows data to be compressed in such a way that the size of a compressed value corresponding to many sensed values can have the same size as the one with a single value. In addition, this technique allows an accurate recovery of the compressed data. This technique has probed to be very suitable for VSN because it reduces the overhead of the network reducing the payload of the packets obtaining a recovery error of up to 0.5%.

Next section summarizes the main work items that are derived from this thesis, and the we expect to address in the near future.

8.2. Future Work

In this thesis we have studied several aspects of VANETs to make them more efficient. We started our contributions designing and developing **BRAVE**, a routing protocol with DTN support whose routing decisions were made opportunistically by the immediate neighbours providing advance towards the destination. This proposal obtained more than 90% of PDR with a reasonable end-to-end delay.

Although further improvements on this solution would provide a marginal benefit in terms of performance because the improvement gap is really low, we also designed a more complete routing solution to deal with **hybrid-VANET** environments. In such scenarios, vehicles are capable of handling different network interfaces, and using the RSUs deployed along streets and roads to communicate

with the infrastructure network even when nodes do not have direct access to them. This complete solution employed an utility function to select the most appropriate network interface depending on the QoS requirements. Further improvements can be applied to it so as to make it more efficient. In addition, in our simulation, we only were aware of the RSUs of the scenarios but we did not take into account other wireless interfaces like UMTS or WiMAX which can also provide different results.

Regarding our proposal to strengthen the delivery of packets under hostile scenarios. We employed the technique of **guard nodes** to make neighbouring nodes to overhear the packet and its consequent forwarding. This proposal has still a great margin to be improved, we were able to increase the packet delivery ratio up to a 60% in the best cases. Sparse scenarios where malicious nodes are more likely to be the neighbours of a vehicle that provide advance towards the destination are the ones which need more improvement. So our future works will go in this direction.

The **pre-authentication** proposal also took as it first step a straightforward gateway selection mechanism which only considered the current distance to the gateways to include them in a cache of authenticated gateways. Simulations where only made over highway and grid scenarios. An urban one, with more candidate gateways would provide us more information about the validity of our gateway selection mechanism allowing us to improve it by predicting new positions considering this information in the selection of future gateways.

Finally, the last contribution presented in this thesis came in the scope of Vehicular Sensor Networks (VSNs). It consists in a **harvesting information** strategy which used a compressing technique called Compressed Sensing (CS) to reduce at maximum the size of the payload of the data packet when a lot of sensed values are carried in the same packet. Since in our simulations we place the Fusion Center (FC) in a specific position within the scenario. Different placements must be considered in future works. In this sense, it is also interesting the use of more than one FCs in the urban scenario. On the other hand, since CS is sensible to the representation of the data, a deeper study regarding this issue is also important to provide a good compression rate and the posterior recovery of the data. In the next section, we list the most relevant publications derived from our work.

8.3. List of Publications

In this section, we list the papers which are related to the contributions and development of this thesis. Only peer-reviewed international publications are considered.

8.3.1. Book chapters

- Francisco J. Ros, Juan A. Martinez and Pedro M. Ruiz, “Mobility models, topology, and simulations in VANET”, Mobile Ad Hoc Networking: Cutting Edge Directions(Stefano Basagni, Marco Conti, Silvia Giordano and Ivan Stojmenovic, eds.), Wiley-IEEE Press, 2nd edition, Chapter 15, pp. 545-576, March 2013.

- F.J Ros, V. Cabrera, J.A. Sanchez, J.A. Martinez and P.M. Ruiz, “Routing in Vehicular Networks”, Vehicular Networks: Techniques, Standards and Applications (Hassnaa Moustafa and Yan Zhang, eds.) Auerbach Publications, Chapter 5, April 2009.

8.3.2. Journals and magazines

- Francisco J. Ros, Juan A. Martinez and Pedro M. Ruiz, "A survey on modeling and simulation of vehicular networks: Communications, mobility, and tools", Computer Communications, Vol. 43, No. 1, p1-15, May 2014.
- Juan A. Martinez, Daniel Vigueras, Francisco J. Ros, and Pedro M. Ruiz, "Evaluation of the Use of Guard Nodes for Securing the Routing in VANETs", Journal of Communications and Networks, Vol. 15, No. 2, April 2013.
- J.A. Martinez and P.M. Ruiz, "Performance Evaluation of Pre-Authenticated Handover Across Gateways In Vehicular Networks", Adhoc & Sensor Wireless Networks, Vol. 15 Issue 1, p47-64, 2012.
- P.M. Ruiz, R. Marin, F.J. Ros and J.A. Martinez, "Enhanced Access Control in Hybrid MANETs Through Utility-based Pre-authentication Control", Wiley Wireless Communications and Mobile Computing Journal (WCMC), 2009.

8.3.3. Conferences

- P.M. Ruiz, V. Cabrera, J.A. Martinez, and F.J. Ros, "BRAVE: Beacon-less routing algorithm for vehicular environments", in Proc. IEEE Mobile Ad-hoc and Sensors Systems (MASS), pp.709-714, 2010.
- J.A. Martinez, P.M. Ruiz and R. Marin, "Impact of the Pre-authentication Performance in Vehicular Networks", in Proc. IEEE 72nd Vehicular Technology Conference Fall (VTC 2010-Fall), 2010.
- I. Lequerica, J.A. Martinez and P.M. Ruiz, "Efficient Certificate Revocation in Vehicular Networks using NGN Capabilities", in Proc. IEEE 72nd Vehicular Technology Conference Fall (VTC 2010-Fall), 2010.

Bibliography

[1] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, "A review of routing protocols for mobile ad hoc networks," *Ad hoc networks*, vol. 2, no. 1, pp. 1–22, 2004.

[2] V. Cabrera, F. J. Ros, and P. M. Ruiz, "Simulation-based study of common issues in vanet routing protocols," in *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pp. 1–5, IEEE, 2009.

[3] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security*, vol. 15, no. 1, pp. 39–68, 2007.

[4] N. H. T. S. Administration, "Fatality analysis reporting system (fars) encyclopedia."

[5] M. del interior. Dirección General de Tráfico, "Anuario estadístico de accidentes 2012," 2012.

[6] S. Tsugawa, "Issues and recent trends in vehicle safety communication systems," *IATSS research*, vol. 29, no. 1, pp. 7–15, 2005.

[7] M. Jerbi, S.-M. Senouci, Y. Ghamri-Doudane, M. Cherif, *et al.*, "Vehicular communications networks: Current trends and challenges," *IGI Global*, pp. 251–262, 2010.

[8] J. Hedrick, M. Tomizuka, and P. Varaiya, "Control issues in automated highway systems," *Control Systems, IEEE*, vol. 14, no. 6, pp. 21–32, 1994.

[9] O. Gehring and H. Fritz, "Practical results of a longitudinal control concept for truck platooning with vehicle to vehicle communication," in *Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on*, pp. 117–122, IEEE, 1997.

[10] C. V. infrastructure systems, "Links," 2009.

[11] IEEE, "Ieee 1609 - family of standards for wireless access in vehicular environments (wave)," 2013.

[12] A. International, "Astm e2213-03 - standard specification for telecommunications and information exchange between roadside and vehicle systems - 5 ghz band dedicated short range communications (dsrc) medium access control (mac) and physical layer (phy) specifications," 2003.

[13] I. O. for Standardization, "Iso 21217. intelligent transport systems – communications access for land mobiles (calm) – architecture," April 2010.

[14] E. T. S. Institute, "Etsi en 302 665 v1.1.1. intelligent tranport systems (its); communications architecture," September 2010.

[15] M. Nekovee, "Sensor networks on the road: the promises and challenges of vehicular adhoc networks and vehicular grids," in *Proceedings of the Workshop on Ubiquitous Computing and e-Research*, 2005.

[16] I. Panda, "A survey on routing protocols of manets by using qos metrics," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 10, pp. 120–129, 2012.

[17] L. M. Feeney, "A taxonomy for routing protocols in mobile ad hoc networks," *SICS Research Report*, 1999.

[18] E. M. Royer and C.-K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *Personal Communications, IEEE*, vol. 6, no. 2, pp. 46–55, 1999.

[19] G. V. Kumar, Y. V. Reddyr, and D. M. Nagendra, "Current research work on routing protocols for manet: a literature survey," *international Journal on computer Science and Engineering*, vol. 2, no. 03, pp. 706–713, 2010.

[20] J. Luo and J.-P. Hubaux, "A survey of research in inter-vehicle communications," in *Embedded Security in Cars*, pp. 111–122, Springer, 2006.

[21] J. Chennikara-Varghese, W. Chen, O. Altintas, and S. Cai, "Survey of routing protocols for inter-vehicle communications," in *Mobile and Ubiquitous Systems: Networking & Services, 2006 Third Annual International Conference on*, pp. 1–5, IEEE, 2006.

[22] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *Vehicular Technology Magazine, IEEE*, vol. 2, no. 2, pp. 12–22, 2007.

[23] Y.-W. Lin, Y.-S. Chen, and S.-L. Lee, "Routing protocols in vehicular ad hoc networks: A survey and future perspectives.," *J. Inf. Sci. Eng.*, vol. 26, no. 3, pp. 913–932, 2010.

[24] B. T. Sharef, R. A. Alsaqour, and M. Ismail, "Vehicular communication ad hoc routing protocols: A survey," *Journal of Network and Computer Applications*, vol. 40, pp. 363–396, 2014.

[25] F. J. Ros, V. Cabrera, J. A. Sanchez, J. A. Martinez, and P. M. Ruiz, "Routing in vehicular networks," in *Vehicular Networks: Techniques, Standards, and Applications* (H. Moustafa and Y. Zhang, eds.), ch. 5, Auerbach Publications, April 2009.

[26] S. Ruhrup, "Theory and practice of geographic routing," *Ad Hoc and Sensor Wireless Networks: Architectures, Algorithms and Protocols*, p. 69, 2009.

[27] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals," *Communications, IEEE Transactions on*, vol. 32, no. 3, pp. 246–257, 1984.

[28] T.-C. Hou and V. O. Li, "Transmission range control in multihop packet radio networks," *Communications, IEEE Transactions on*, vol. 34, no. 1, pp. 38–44, 1986.

[29] G. G. Finn, "Routing and addressing problems in large metropolitan-scale internetworks," tech. rep., DTIC Document, 1987.

[30] E. Kranakis, H. Singh, and J. Urrutia, "Compass routing on geometric networks," in *in Proc. 11 th Canadian Conference on Computational Geometry*, Citeseer, 1999.

[31] I. Stojmenovic and X. Lin, "Power-aware localized routing in wireless networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 12, no. 11, pp. 1122–1133, 2001.

[32] I. Stojmenovic and X. Lin, "Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 12, no. 10, pp. 1023–1032, 2001.

[33] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, "Speed: A stateless protocol for real-time communication in sensor networks," in *Distributed Computing Systems, 2003. Proceedings. 23rd International Conference on*, pp. 46–55, IEEE, 2003.

[34] K. R. Gabriel and R. R. Sokal, "A new statistical approach to geographic variation analysis," *Systematic Biology*, vol. 18, no. 3, pp. 259–278, 1969.

[35] G. T. Toussaint, "The relative neighbourhood graph of a finite planar set," *Pattern recognition*, vol. 12, no. 4, pp. 261–268, 1980.

[36] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wireless networks*, vol. 7, no. 6, pp. 609–616, 2001.

[37] J. Gao, L. J. Guibas, J. Hershberger, L. Zhang, and A. Zhu, "Geometric spanners for routing in mobile networks," *Selected Areas in Communications, IEEE Journal on*, vol. 23, no. 1, pp. 174–185, 2005.

[38] Brak Karp and H.T. Kung, "*GPSR: Greedy Perimeter Stateless Routing for wireless networks*," in *In Proceedings 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '00)*, pp. 243–254, 2000.

[39] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad-hoc wireless networks," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, pp. 1360–1369, IEEE, 2001.

[40] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *Workshop on hot topics in networks (HotNets-IV)*, pp. 1–6, 2005.

[41] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing vehicular communications," *IEEE Wireless Communications Magazine, Special Issue on Inter-Vehicular Communications*, vol. 13, no. LCA-ARTICLE-2006-015, pp. 8–15, 2006.

[42] X. Lin, R. Lu, C. Zhang, H. Zhu, P.-H. Ho, and X. Shen, "Security in vehicular ad hoc networks," *Communications Magazine, IEEE*, vol. 46, no. 4, pp. 88–95, 2008.

[43] M. Razzaque, A. Salehi, and S. M. Cheraghi, "Security and privacy in vehicular ad-hoc networks: survey and the road ahead," in *Wireless Networks and Security*, pp. 107–132, Springer, 2013.

[44] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "Extensible authentication protocol (eap)," June 2004.

[45] D. Forsberg, Y. Ojba, B. Patil, H. Tschofenig, and A. Yegin, "Protocol for carrying authentication for network access (pana)," May 2008.

[46] C. Kaufman, "Internet key exchange (ikev2) protocol," May 2005.

[47] S. Kent and K. Seo, "Security Architecture for the Internet Protocol." RFC 4301 (Proposed Standard), December 2005.

[48] W. Diffie and M. E. Hellman, "New directions in cryptography," *Information Theory, IEEE Transactions on*, vol. 22, no. 6, pp. 644–654, 1976.

[49] H. Moustafa and Y. Zhang, *Vehicular Networks: Techniques, Standards, and Applications*, ch. 5. Boston, MA, USA: Auerbach Publications, 2009.

[50] F. Li and Y. Wang, "*Routing in Vehicular Ad Hoc Networks: A Survey*," in *IEEE Vehicular Technology Magazine*, 2007.

[51] Victor Cabrera and Fran J. Ros and Pedro M. Ruiz, "Simulation-based Study of Common Issues in VANET Routing Protocols." In Proceedings IEEE 69th Vehicular Technology Conference (VTC2009-Spring 2009), April 2009.

[52] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *in Proc. IEEE Workshop on Mobile Computing Systems and Applications*, p. 90, 1999.

[53] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, pp. 153–181, Kluwer Academic Publishers, 1996.

[54] V. Namboodiri, M. Agarwal, and L. Gao, "A study on the feasibility of mobile gateways for vehicular ad-hoc networks," in *VANET '04: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, (New York, NY, USA), pp. 66–75, ACM, 2004.

[55] S. Schnaufer, H. Füßler, M. Transier, and W. Effelsberg, "Unicast ad-hoc routing in vehicular city scenarios," tech. rep., University of Mannheim, 2007.

[56] V. Naumov and T. Gross, "Connectivity-aware routing (car) in vehicular ad-hoc networks," in *Proc. 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, (Anchorage, Alaska, USA), pp. 1919–1927, May 2007.

[57] B.C. Seet, G. Liu, B.S. Lee, C.H. Foh, K.J. Wong, and K.K. Lee, "*Geographic Routing in City Scenarios*," in *In Proceedings of 3rd International Networking Conference IFIP-TC6*, pp. 989–999, December 2004.

[58] K. C. Lee, U. Lee, and M. Gerla, "To-go: Topology-assist geo-opportunistic routing in urban vehicular grids," in *Wireless On-Demand Network Systems and Services, 2009. WONS 2009. Sixth International Conference on*, pp. 11–18, IEEE, 2009.

[59] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 4, pp. 351–369, 2003.

[60] Z. Mo, H. Zhu, K. Makki, and N. Pissinou, "Muru: A multi-hop routing protocol for urban vehicular ad hoc networks," *Mobile and Ubiquitous Systems, Annual International Conference on*, pp. 1–8, 2006.

[61] W. Sun, H. Yamaguchi, K. Yukisama, and S. Kusumoto, "Gvgrid: A qos routing protocol for vehicular ad hoc networks," in *14th IEEE International Workshop on Quality of Service*, pp. 130–139, 2006.

[62] K. C. Lee, J. Härri, U. Lee, and M. Gerla, "Enhanced perimeter routing for geographic forwarding protocols in urban vehicular scenarios," in *Globecom Workshops, 2007 IEEE*, pp. 1–10, IEEE, 2007.

[63] Christian Lochert, Hannes Hartenstein, Jing Tian, Holger Fler, Dagmar Hermann, and Martin Mauve, "*A Routing Strategy for Vehicular Ad Hoc Networks in City Environments*," in *In Proceedings of the IEEE Intelligent Vehicles Symposium 2003*, pp. 156–161, June 2003.

[64] J. Tian, L. Han, K. Rothermel, and C. Cseh, "*Spatially Aware Packet Routing for Mobile Ad Hoc Inter-Vehicle Radio Networks*," in *In Proceedings of the IEEE Intelligent Transportation System Conference*, pp. 1543–1551, October 2003.

[65] K. C. Lee, P.-C. Cheng, and M. Gerla, "Geocross: A geographic routing protocol in the presence of loops in urban scenarios," *Ad Hoc Networks*, vol. 8, no. 5, pp. 474–488, 2010.

[66] K. C. Lee, M. Le, J. Harri, and M. Gerla, "Louvre: Landmark overlays for urban vehicular routing environments," in *Vehicular Technology Conference, 2008. VTC 2008-Fall. IEEE 68th*, pp. 1–5, IEEE, 2008.

[67] J. Lebrun, C.-N. Chuah, and D. Ghosal, "Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks," in *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, vol. 4, pp. 2289–2293, 2005.

[68] I. Leontiadis and C. Mascolo, "Geopps: Geographical opportunistic routing for vehicular networks," in *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pp. 1–6, IEEE, 2007.

[69] P.-C. Cheng, K. C. Lee, M. Gerla, and J. Härri, "Geodtn+ nav: geographic dtn routing with navigator prediction for urban vehicular environments," *Mobile Networks and Applications*, vol. 15, no. 1, pp. 61–82, 2010.

[70] M. Jerbi, R. Meraihi, S.-M. Senouci, and Y. Ghamri-Doudane, "Gytar: improved greedy traffic aware routing protocol for vehicular ad hoc networks in city environments," in *VANET '06: Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, (New York, NY, USA), pp. 88–89, ACM, 2006.

[71] B.C. Seet, G. Liu, B.S. Lee, C.H. Foh, K.J. Wong, and K.K. Lee, "*A Mobile Ad Hoc Routing Strategy for Metropolis Vehicular Communications*," in *In Proceedings of 3rd International Networking Conference IFIP-TC6*, pp. 989–999, Diciembre 2004.

[72] J. Zhao and G. Cao, "Vadd: Vehicle-assisted data delivery in vehicular ad hoc networks," in *IEEE INFOCOM'06*, 2006.

[73] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "Mddv: a mobility-centric data dissemination algorithm for vehicular networks," in *VANET '04: Proc. of the 1st ACM international workshop on Vehicular ad hoc networks*, pp. 47–56, 2004.

[74] Y. Ding, C. Wang, and L. Xiao, "A static-node assisted adaptive routing protocol in vehicular networks," in *VANET '07: Proc. of the fourth ACM international workshop on Vehicular ad hoc networks*, pp. 59–68, 2007.

[75] B.-C. Seet, G. Liu, B.-S. Lee, C.-H. Foh, K.-J. Wong, and K.-K. Lee, "A-star: A mobile ad hoc routing strategy for metropolis vehicular communications," in *NETWORKING 2004. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications*, pp. 989–999, Springer, 2004.

[76] Juan A. Sánchez and Rafael Marin-Pérez and Pedro M. Ruiz, "Beacon-Less Geographic Routing in Real Wireless Sensor Networks," *Journal of Computer Science and Technology*, vol. 23, pp. 438–450, May 2008.

[77] P. M. Ruiz, V. Cabrera, J. A. Martinez, and F. J. Ros, "Brave: Beacon-less routing algorithm for vehicular environments," in *Mobile Adhoc and Sensor Systems (MASS), 2010 IEEE 7th International Conference on*, pp. 709–714, IEEE, 2010.

[78] J. R. Douceur, "The sybil attack," in *Peer-to-peer Systems*, pp. 251–260, Springer, 2002.

[79] J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis & defenses," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pp. 259–268, ACM, 2004.

[80] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad hoc networks*, vol. 1, no. 2, pp. 293–315, 2003.

[81] I. T. S. Committee *et al.*, "Ieee trial-use standard for wireless access in vehicular environments-security services for applications and management messages," *IEEE Vehicular Technology Society Standard*, vol. 1609, p. 2006, 2006.

[82] I. Lequerica, J. A. Martinez, and P. M. Ruiz, "Efficient certificate revocation in vehicular networks using ngn capabilities," in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pp. 1–5, IEEE, 2010.

[83] R. Marin-Perez and P. M. Ruiz, "Sbgr: a simple self-protected beaconless geographic routing for wireless sensor networks," in *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*, pp. 610–619, IEEE, 2011.

[84] P. Papadimitratos, V. Gligor, and J.-P. Hubaux, "Securing vehicular communications-assumptions, requirements, and principles," in *Workshop on Embedded Security in Cars (ESCAR)*, vol. 2006, 2006.

[85] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure vehicular communication systems: design and architecture," *Communications Magazine, IEEE*, vol. 46, no. 11, pp. 100–109, 2008.

[86] C. Harsch, A. Festag, and P. Papadimitratos, "Secure position-based routing for vanets," in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pp. 26–30, IEEE, 2007.

[87] A. Festag, G. Noecker, M. Strassberger, A. Lübke, B. Bochow, M. Torrent-Moreno, S. Schnaufer, R. Eigner, C. Catrinescu, and J. Kunisch, "'now–network on wheels': Project objectives, technology and achievements," 2008.

[88] A. Festag, P. Papadimitratos, and T. Tielert, "Design and performance of secure geocast for vehicular communication," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 5, pp. 2456–2471, 2010.

[89] J. T. Isaac, S. Zeadally, and J. S. Cámara, "Security attacks and solutions for vehicular ad hoc networks," *Communications, IET*, vol. 4, no. 7, pp. 894–903, 2010.

[90] C. Laurendeau and M. Barbeau, "Threats to security in dsrc/wave," in *Ad-Hoc, Mobile, and Wireless Networks*, pp. 266–279, Springer, 2006.

[91] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, "Overhaul of ieee 802.11 modeling and simulation in ns-2," in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, pp. 159–168, ACM, 2007.

[92] M. Gerla, B. Zhou, Y.-Z. Lee, F. Soldo, U. Lee, and G. Marfia, "Vehicular grid communications: the role of the internet infrastructure," in *Proceedings of the 2nd annual international workshop on Wireless internet*, p. 19, ACM, 2006.

[93] J. Miller, "Vehicle-to-vehicle-to-infrastructure (v2v2i) intelligent transportation system architecture," in *Intelligent Vehicles Symposium, 2008 IEEE*, pp. 715–720, IEEE, 2008.

[94] C.-C. Hung, H. Chan, and E.-K. Wu, "Mobility pattern aware routing for heterogeneous vehicular networks," in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pp. 2200–2205, IEEE, 2008.

[95] J. Luo, X. Gu, T. Zhao, and W. Yan, "A mobile infrastructure based vanet routing protocol in the urban environment," in *Communications and Mobile Computing (CMC), 2010 International Conference on*, vol. 3, pp. 432–437, IEEE, 2010.

[96] Y. Wu, Y. Zhu, and B. Li, "Infrastructure-assisted routing in vehicular networks," in *INFOCOM, 2012 Proceedings IEEE*, pp. 1485–1493, IEEE, 2012.

[97] A. M. Vegni and T. D. Little, "Hybrid vehicular communications based on v2v-v2i protocol switching," *International Journal of Vehicle Information and Communication Systems*, vol. 2, no. 3, pp. 213–231, 2011.

[98] S. Deering and R. Hinden, "Rfc 2460: Internet protocol, version 6 (ipv6) specification," 1998.

[99] E. Coronado and S. Cherkaoui, "An aaa study for service provisioning in vehicular networks," in *In Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN 2007)*, pp. 669–676, 2007.

[100] Y. O. (ed), "Extensible Authentication Protocol (EAP) Early Authentication Problem Statement. draftietf-hokey-preauth-ps-12." IETF Internet Draft, Jan. 2010. Work in Progress, 2010.

[101] V. Casola, J. Luna, A. Mazzeo, M. Medina, M. Rak, and J. Serna, "An interoperability system for authentication and authorization in vanets," *International Journal of Autonomous and Adaptive Communications Systems*, vol. 3, pp. 115–135, 2010.

[102] A. Hafslund and J. Andersson, "2-Level Authentication Mechanism in a Internet connected MANET," in *6th Scandinavian Workshop on Wireless Ad-hoc Networks, May 3-4, Johannesberg Estate, Stockholm.*, 2005.

[103] H. Moustafa, G. Bourdon, and Y. Gourhant, "Aaa in vehicular communication on highways with ad hoc networking support: a proposed architecture," in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, pp. 79–80, ACM, 2005.

[104] U. Jönsson, F. Alriksson, T. Larsson, P. Johansson, and G. Q. Maguire, Jr., "Mipmanet: mobile ip for mobile ad hoc networks," in *MobiHoc '00: Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, (Piscataway, NJ, USA), pp. 75–85, IEEE Press, 2000.

[105] J. Choi, S. Jung, Y. Kim, and M. Yoo, "A fast and efficient handover authentication achieving conditional privacy in v2i networks," in *NEW2AN '09 and ruSMART '09: Proceedings of the 9th International Conference on Smart Spaces and Next Generation Wired/Wireless Networking and Second Conference on Smart Spaces*, (Berlin, Heidelberg), pp. 291–300, Springer-Verlag, 2009.

[106] B. Aboba, D.Simon, and P. Eronen, "Extensible authentication protocol (eap) key management framework.." RFC 5247, Aug., 2008.

[107] M. Georgiades, N. Akhtar, C. Ploitis, and R. Tafaziolli, "Aaa context transfer for seamless and secure multimedia services over all-ip infrastructures," in *5th European Wireless Conference (EW'04), Barcelona, February 24-27*, 2004.

[108] R. Marin, P. M. Ruiz, F. J. Ros, J. A. Martinez, and A. F. Gomez, "Pre-authentication based enhancement for access control in hybrid manets," in *In Proc. of IEEE Symposium on Computers and Communications, ISCC'07, pages 595-600, Aveiro, Portugal, July*, 2007.

[109] P. M. R. Martinez, R. M. Lopez, F. J. Ros, and J. A. Martinez, "Enhanced access control in hybrid manets through utility-based pre-authentication control," *Wirel. Commun. Mob. Comput.*, vol. 10, no. 5, pp. 688–703, 2010.

[110] A. Dutta, D.Famolari, S.Das, Y.Ohba, V. Fajardo, K. Taniuchi, R. Lopez, and H. Schulzrinne, "Media-independent pre-authentication supporting secure inter-domain handover optimization," *IEEE Wireless Communications*, vol. 15, no. 2, pp. 55–64, 2008.

[111] D. Simon, B. Aboba, and R. Hurst, "The EAP-TLS Authentication Protocol." RFC 5216 (Proposed Standard), March 2008.

[112] R. Marin-Lopez, F. Pereniguez-Garcia, A. F. Gomez-Skarmeta, and Y. Ohba, "Network access security for the internet: protocol for carrying authentication for network access," *Communications Magazine, IEEE*, vol. 50, no. 3, pp. 84–92, 2012.

[113] F. Bernal, F. Pereniguez, R. Marin, and A. F. Skarmeta, "A network access control solution based on pana for intelligent transportation systems," in *The 3rd International Conference on Connected Vehicles and Expo (ICCVE 2014)*, 2014.

[114] Y. Ohba and D. Frascone, "Open diameter, http://sourceforge.net/projects/diameter."

[115] S. McCanne and S. Floyd, "Ns network simulator 2, available at: http://www.isi.edu/nsnam/ns."

[116] F. J. Ros and P. M. Ruiz, "Efficient gateway discovery algorithms for delay-tolerant and delay-constrained data traffic in vehicular ad-hoc networks," in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pp. 1–5, IEEE, 2010.

[117] U. Lee and M. Gerla, "A survey of urban vehicular sensing platforms," *Comput. Netw.*, vol. 54, pp. 527–544, Mar. 2010.

[118] U. Lee, B. Zhou, M. Gerla, E. Magistretti, P. Bellavista, and A. Corradi, "Mobeyes: smart mobs for urban monitoring with a vehicular sensor network," *Wireless Commun.*, vol. 13, pp. 52–57, Oct. 2006.

[119] U. Lee, J.-S. Park, E. Amir, and M. Gerla, "Fleanet: A virtual market place on vehicular networks," *Mobile and Ubiquitous Systems, Annual International Conference on*, vol. 0, pp. 1–8, 2006.

[120] M. D. Dikaiakos, S. Iqbal, T. Nadeem, and L. Iftode, "Vitp: an information transfer protocol for vehicular computing," in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, pp. 30–39, ACM, 2005.

[121] C. E. Palazzi, F. Pezzoni, and P. M. Ruiz, "Delay-bounded data gathering in urban vehicular sensor networks," *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 180–193, 2012.

[122] J. Haupt, W. U. Bajwa, M. Rabbat, and R. Nowak, "Compressed Sensing for Networked Data," *Signal Processing Magazine, IEEE*, vol. 25, pp. 92–101, Mar. 2008.

[123] W. Bajwa, "Matched source-channel communication for field estimation in wireless sensor networks," in *Proc. the Fouth Int. Symposium on Information Processing in Sensor Networks*, pp. 332–339, 2005.

[124] S. Feizi-Khankandi, M. Médard, and M. Effros, "Compressive sensing over networks," *CoRR*, vol. abs/1012.0955, 2010.

[125] C. T. Chou, R. K. Rana, and W. Hu, "Energy efficient information collection in wireless sensor networks using adaptive compressive sensing," in *LCN*, pp. 443–450, IEEE, 2009.

[126] X. Yu, Y. Liu, Y. Zhu, W. Feng, L. Zhang, H. F. Rashvand, and V. O. K. Li, "Efficient sampling and compressive sensing for urban monitoring vehicular sensor networks," *IET Wireless Sensor Systems*, vol. 2, no. 3, pp. 214–221, 2012.

[127] E. J. Candes and M. B. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, vol. 25, pp. 21–30, Mar. 2008.

[128] S. Mallat and Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3397–3415, 1993.

[129] T. T. Cai and L. Wang, "Orthogonal Matching Pursuit for Sparse Signal Recovery With Noise," *Information Theory, IEEE Transactions on*, vol. 57, pp. 4680–4688, July 2011.

[130] P. Schniter, L. C. Potter, and J. Ziniel, "Fast Bayesian Matching Pursuit: Model Uncertainty and Parameter Estimation for Sparse Linear Models," *IEEE Transactions on Signal Processing*, Mar. 2009.

[131] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.

[132] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.0 beta." `http://cvxr.com/cvx`, Sept. 2013.

[133] C. E. Palazzi, M. Roccetti, and S. Ferretti, "An intervehicular communication architecture for safety and entertainment," *Trans. Intell. Transport. Sys.*, vol. 11, pp. 90–99, March 2010.

[134] F. J. Ros, P. M. Ruiz, and I. Stojmenovic, "Acknowledgment-based broadcast protocol for reliable and efficient data dissemination in vehicular ad hoc networks," *IEEE Trans. Mob. Comput.*, vol. 11, no. 1, pp. 33–46, 2012.

[135] P. M. Ruiz, V. Cabrera, J. A. Martinez, and F. J. Ros, "Brave: Beacon-less routing algorithm for vehicular environments," in *Mobile Adhoc and Sensor Systems (MASS), 2010 IEEE 7th International Conference on*, pp. 709–714, IEEE, 2010.

[136] MATLAB, *version 7.10.0 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010.