NAV2V: Navigation Assisted V2V Routing Protocol for Urban Areas

Lincoln H. Teixeira and Árpad Huszák
Budapest University of Technology and Economics, Department of Networked Systems and Services, Budapest, Hungary
Email: {teixeira, huszak}@hit.bme.hu

Abstract —Vehicular AdHoc networks (VANETs) allow connectivity in dispersed vehicle environments and other difficult access areas, where conventional network systems are not satisfactory. One of the most significant technical challenges faced in implementing VANET is designing an efficient routing protocol that can provide a reliable path between the source and destination of the information. This paper discusses a routing scheme that incorporates the control strategy for transmitting messages and Global Navigation Satellite System (GGNS) information to optimize network routing. This scheme uses geolocation information to select the best path to forward the messages. To simulate communication in real-life scenarios, we used the *Simulation of Urban MObility (SUMO)* and *Network Simulator-version 3 (NS-3)* platform to compare our proposed algorithm to the traditional routing protocols scenarios when the number of source-destination pairs varies. Our results show that the proposed NAV2V algorithm can decrease the packet loss rate, end-to-end delay, and enhance network efficiency.

Index Terms—ITS, NS-3, routing protocol, SUMO, V2V, VANET

I. Introduction

The fundamental requirement for the future of vehicular communication is that the network arrangement is to be done in an autonomous, intelligent, and optimized way to forward the information among themselves without needing infrastructure. These communication methodologies combine to design functions on the roads and improve the transport system [1]. The vehicles are mobile nodes linked in a wireless set through AdHoc systems. Although these nodes are not displayed similarly, the information can pass through multiple intermediate nodes to assist routing protocols [2]. These moving cars are utilized as routers to provide a reliable mobile communication structure among the vehicles [3]. Interest in these networks is increasing, mainly regarding research aimed at working with government entities, the automotive industry, and transport sectors that ensure the transport system area's security and performance [4].

The following open questions have inspired the research:

- High Mobility: As the speed and route of vehicles within a given area are different, congestion arises and topological changes. Data transmission on vehicle-to-vehicle (V2V) networks is complex due to these irregular topological changes. The connection between nodes (vehicles) can take a few seconds, causing communication problems. Therefore, there is not enough time for the routing protocol to exchange information and build an efficient route table.
- Connection instability: Starting and maintaining the connection is necessary for routing in modern VANET architecture. Because of the dynamic configuration, the connection lifetime is limited. An effective routing scheme would aim to provide the life of a connection for as long as possible.
- Autonomy: Each node works autonomously, dealing with the information delivered across the network. This type of network has no infrastructure, so it transmits data to all nodes, allowing it to withstand communication failures. It will enable autonomous nodes to distribute information to their neighbors. Tracking vehicles' positions can help optimally redirect data, reduce delay, network overhead, and use the limited time to communicate with useful information.
- Fault tolerance: As VANET is exposed to architectural disconnections, it is difficult to establish a stable link between nodes. Therefore, an autonomous alternative routing protocol is required as a complement to the routing solution. An efficient routing algorithm should consider these factors and include a fault tolerance mechanism as part of the design.
- Scalability: The design of the vehicle network depends on the number of vehicles on the road. When the number of cars decreases, the need for assistance increases; however, as the number of cars decreases, the number of transmission demands is also expected to decrease. Vehicles require continuous data transfer; thus, in these situations, the connection must always be preserved. Therefore, the network must be able to increase or decrease based on requirements and other essential factors. The networks are defined along the entire axis of the highway generating many nodes. Therefore, the reception range needs to be expanded [4]. Mainly because we never know when it will be necessary to send information, especially if it is an emergency or a warning signal.

Manuscript received July 19, 2021; revised January 14, 2022.
Corresponding author email: teixeira@hit.bme.hu.
doi: 10.12720/jcm.17.2.74-89

- Low latency: As the transmission window is short and with ITS functions, the amount of requested transmission data can be very high; therefore, it requires a network with low latency.
- Limited capacity: Bandwidth is the amount of data or information packets sent over an existing communication network. As AdHoc VANET has no infrastructure, a weakening of the signal occurs due to electromagnetic interference in the radio signal and signal blocking, such as buildings and communication outside the line of sight [4]-[6]. The existence of dense buildings on the sides of the streets can degrade the performance of VANETs applications [2], resulting in a reduced capacity concerning the guided networks.
- QoS: The quality of service at VANET is essential due to the use of a wide variety of distributed mobile applications, including the delivery of traffic alerts, route planning, and file sharing [7]. However, maintaining the priority option for delivering packages at an appropriate level represents a challenge. Due to the dynamic change of topology, it is difficult to guarantee the necessary resource reserve [3].

Despite all the problems presented, we hardly believe that there will be a single solution for routing protocol for vehicular networks. The development of efficient routing schemes is a challenging task and significantly impacts communication, leading researchers to experiment with different strategies to solve it.

In this article, we examine and analyze the performance of four routing algorithms for VANETs: AdHoc On-demand Distance Vector (AODV), Destination Sequenced DistanceVector Routing (DSDV), the Optimized Link State Routing, Protocol (OLSR) and Greedy Perimeter Stateless Routing (GPSR), because they are the most used and researched in AdHoc communication in vehicular networks today.

Routing algorithms require optimization for AdHoc vehicle networks as traffic congestion increases, and a millisecond error in the traffic management network can lead to disastrous results [8]. This research sought to offer a solution that uses the position of vehicles at each moment as a parameter and dynamically updates the routing table to help each vehicle act as a network node and share information more quickly and efficiently. Therefore, a graph theory algorithm modified to help find the shortest path, based on the transmission time, is proposed to act as a catalyst when executed with existing algorithms. Then, an optimized routing protocol scheme is proposed to reduce packet transfer times which can also be used for faster communication in a real environment, analyzed in a network simulator.

A. Methods

Here, we present additional methods to improve the performance of our proposed algorithm, including (1) Using information from the position of neighboring nodes to make the selection of the best path and try to avoid nodes that do not connect to the final destination, to avoid loops of packages; (2) Selection of the best path, using a modified shorter distance algorithm, which takes time as the main weight and (3) We compare the performance of our proposed NAV2V with traditional protocols, analyzing the performance metrics, such as packet loss rate, end-to-end delay, and network performance. Significant contributions of this paper are summarized as follows:

- We have introduced the most used routing protocols in vehicular AdHoc communications nowadays.
- Then, we propose a new network routing scheme to improve data transmission using the information available on the vehicles' GNSS, which exploits the positions of neighbor nodes to forward the packets. We call this solution as NAV2V algorithm, which makes use of the concept presented in this work.
- Subsequently, we present the comparative results between the current protocols and the new concept introduced in this research using computer-based simulation software.

We organize the rest of this research paper in the following way: Section 2 reviews the related research work on network route protocol. Navigation-Assisted V2V Routing Protocol for Urban Areas approach is described in Section 3. In Section 4, we explain the simulation environment. We present the evaluation of the simulation results with discussions in section 5. Finally, Section 6 concludes the paper.

II. Related Work

VANETs are distributed and self-organized communication structures configured through automobiles as nodes to transmit messages. These nodes are mobile but have restricted ranges of mobility and communication due to the direction of the roads, width of the lanes, and limitations of pedestrian zones and buildings. The primary aim when researching VANETs is to create a fast and economical vehicle communication system to allow the distribution of data for the safety and comfort of passengers [9].

For the connection to be efficient, broadcast communication must be avoided. It is seriously affected by propagation problems, such as signal loss, rapid fading due to multipath transmission, and a hidden terminal problem [10]. New relay schemes are expected to be developed as a research field to improve the efficiency of V2V communication. For the communication to occur, the vehicle communicates with another vehicle directly if there is a direct wireless connection between them, forming a V2V communication. When there is no direct connection between the cars, it utilizes an assigned routing protocol to route data from one vehicle to another until it reaches the destination point, forming the communication between the multi-hop vehicle [11]. Routing protocols use local information from their network neighborhoods to determine which relay nodes will be used in data routing to perform packet transfers. These protocols must be flexible to the peculiarities and resources of VANET, where vehicles can face many obstacles, such as crossings, buildings, traffic lights, trees, resulting in insufficient channel quality and connectivity. Therefore, it is necessary to use an efficient and reliable routing protocol to communicate without losing data [12]. Various systems are involved through this modern notion of VANETs, such as traffic management, routing, handover, signal propagation, and autonomous vehicles [13].

We can still apply the existing AdHoc routing protocols in other types of networks to VANETs. However, the simulation result indicated that they suffer from performance problems for rapid vehicle changes and reduced chances of transferring information [14].

Wireless VANET communication provides vehicle-to-everything (V2X), and the topology can vary from dense to very sparse [7]. With dense network typologies, vehicles on the transmission route make it possible to provide end-to-end multipoint connectivity between the origin and destination of the node, rather than transferring messages through base stations. Communications on a sparse network can be achieved using the store-carry-and-forward paradigm, inspired by Delay Tolerant Networks (DTN). It was initially designed for communication between spaces but is also recognized for use in scenarios where the telecommunications infrastructure is unstable or unavailable due to disconnected areas, natural disasters, or emergency conditions.

Based on DTN networks, Vehicle Delay Tolerance Networks (VDTN) was created. The framework consists of vehicle nodes and other nodes, providing a low-cost connectivity solution in challenging scenarios where a telecommunications infrastructure is unreliable or unavailable [15], [16].

This technology has a highly variable topology, with frequent partitions and, perhaps, low node density. Thus, delay-tolerant routing uses message storage, loading, and transferring paradigms, in which messages have a proper hop limit or time-to-live (TTL) and are stored until a satisfactory contact opportunity appears [17]. In this method, the node first stores the messages if there is no vehicle where it can be transported. Then, when other vehicles enter the transmitted radio line, the data will be relayed to the others [18]. VDTN routing algorithms make routing decisions by building and updating routing tables whenever mobility occurs [19].

Multiple replicas of messages can be generated to increase the likelihood of delivery at the cost of increasing network congestion. However, some nodes in a cooperative network may develop a selfish attitude to protect their resources, such as memory and energy, and not collaborate; that is, they do not forward messages. One solution to this dilemma is to adopt the Honesty Based Democratic Scheme (HBDS) introduced by [20] to find selfish nodes in cooperation for relaying messages. They test performance metrics, such as package delivery probability, package delivery delay, overhead rate, and the number of packages lost. The results indicate that the system can make nodes work together in a community to develop the network's performance. The proposed scheme can distribute information between nodes connected in the network to avoid problematic nodes [7].

Maintaining Quality of Service (QoS) on VANETs is crucial, while packet routing is a big challenge, especially when bandwidth is limited [3], [21]. Many protocols aim to provide high-quality services (QoS) while saving more resources [22].

Another problem in this type of approach is the hidden terminal: When there is no centralized communication coordination, the hidden terminal problem occurs in the VANETs, as shown in Fig. 1. This occurs when two nodes are not in the same communication range and transmit data for the same receiver [23], [24]. However, this problem is bypassed by using a synchronized solution in our work, where now the hidden terminal is accessible via a bridge from the terminal that accesses both nodes. This issue is a severe problem in the design of MAC protocols because it deteriorates the performance of the MAC layer. Only a few researchers have discussed this problem for MAC protocols on VANETs. Therefore, this issue needs to be studied and integrated into the design of MAC protocols in VANET [25].

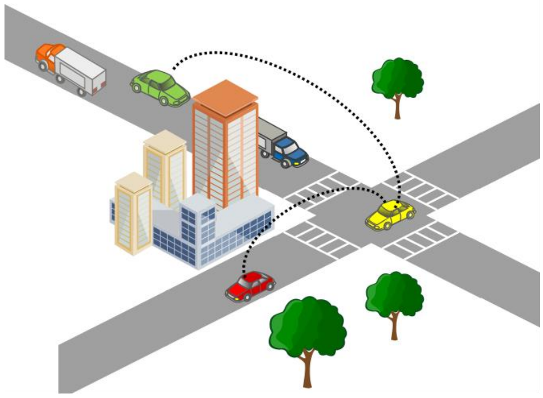

Fig. 1. Hidden terminal using a bridge to communicate.

Next, the main routing protocols available and currently studied on VANET networks, which are part of this study, will be described.

A. Routing Protocols

As routing is the central issue in networks where data transfer between network nodes is the fundamental requirement, this is an ongoing area of research and development. New routing protocol schemes continue to emerge as an improvement over established protocols [1].

Due to the high mobility of the nodes and the rapid changes in topology, designing an efficient routing protocol that can deliver a packet in a minimum period with few discarded packets is considered a critical challenge at VANET. In addition, many researchers have focused on designing a routing protocol suitable for dense environments that have a high density of vehicles with close distances between them. Developing an efficient routing protocol affects many improving factors; the first is to increase the system's reliability, take advantage of the percentage of package delivery, and reduce the extent of interference caused by tall buildings in the city environment. The third factor is that taking scalability into account is essential to avoid conflict if a simultaneous unicast routing request operation has been initiated. Another issue is delivering the package in the shortest time, especially in an emergency; this factor is considered critical [11].

There are three different routing protocols for AdHoc data networks, proactive, reactive, and hybrid protocols. The first is a proactive routing protocol based on the periodic transmission of the data network topology. Here, the protocol ensures that nodes always have up-to-date knowledge of the paths to other nodes. The second is a reactive routing protocol that only looks for a route when one is needed. Finally, the hybrid routing protocol represents a mix of sensitive and practical protocols [13].

These protocols can also be classified into the following categories of approaches. The first routing protocol category uses multiple on-demand paths to switch broken paths to another routing protocol automatically. The second category of routing protocols is position-based routing (or geographic routing). Most vehicles on the roads are equipped with GGNS and digital maps for navigation. Therefore, these vehicles can be easily aware of their geographic locations and then use this information to improve routing performance [26].

Now we will discuss the techniques and differences of the main routing protocols most used today.

1) AODV: The AdHoc Demand Distance Vector (AODV) protocol was developed in 1999 and was based on Dynamic Source Routing (DSR) and DSDV protocols [4]. Currently, AODV is the most common routing protocol that provides route discovery and maintenance techniques, with reduced bandwidth and CPU usage being one of its most significant characteristics, as it sends packets only on demand [3]. It is, however, a protocol known as reactive and the most widely used of all currently active protocols [27]. When it needs to send data packets to a destination point and has no path to the destination node before sending the data packets, it initiates a route scan. This network requires three procedures: (1) the path discovery method, (2) the generation of path messages, and (3) route management.

The route is created only when necessary; it needs less overhead about constructive routing requirements. Consequently, the low overhead required for data packets is one of the main advantages of an AODV. In this protocol, the routing information, which is also efficient in bandwidth, is not modified after a specific time. This strategy addresses the consequences of the initiated roads and the need for new paths to preserve routes. In [28], the authors suggested that the use of path information for each node as a parameter to choose the next hop during a route exploration process increases the efficiency of an AODV on a VANET. Its analysis, however, does not provide end-to-end latency, transit rate, and packet arrival rate and, it has not yet discussed the propagation effects to alleviate the flood problems [3].

AODV employs a route exploration method in the transmission mechanism. It also supports unicast, multicast and manages messages such as Route Request (RREQ), Route Responses (RREP), and Route Errors (RERR) and uses the notifications mentioned above to share information between the source and destination. “Hello” messages can be used to identify connections to and monitor neighbors where each active node periodically emits these messages to all neighbors. Since nodes periodically transmit these messages, a connection break would be detected if a node does not receive several “hello” notifications from a neighbor. It transmits a Route Request (RREQ) to that destination when a source has data sent to an uncertain destination. A path to the origin is formed at each intermediate node when an RREQ is sent. If the transmitting node has not already received this RREQ, it is not the endpoint and has no existing path to the destination; the RREQ is retransmitted. If the receiving node is the destination or has a current route to the destination, a Path Response (RREP) is generated. The RREP unicast is hop-to-hop to the source. Every intermediate node establishes a path to the destination as the RREP propagates. It records the route to the target when the source receives the RREP and can start transmitting the data. If the source gets several RREPs, the path that has the lowest hop count is chosen. Each node along the path updates the timers associated with routes to the source and destination, keeping the routes in the routing table. Data flows from source to destination. When a path has not been used for some time, a node cannot be sure whether the path is still valid; thus, the node excludes the route from its routing table. A path error (RERR) is sent to the data source hop-by-hop way if data is flowing and a link break is observed. Each intermediate node invalidates the routes to any inaccessible destination as the RERR propagates towards the origin. It invalidates the path and restarts the path discovery when the data source collects the RERR [29]. In AODV, networks do nothing except wait for links to be created; that is, nodes in the network need to send a connection request. Nodes that are not part of a particular route must not hold precise route information. Consequently, these nodes cannot allow the topology upgrade bundle to flow, or, in other words, they include only the path information they are on [1].

Studies by [3], [28] modified the AODV routing protocol, adding details on the vehicle's speed, orientation, and location. For nearby cars, the routing tables now provide additional driving detail. To pick the next hop, they used the path parameter and then implemented the transport and routing system to bring the information to the vehicle nearest to the destination, as long as the source can send the parcels to other vehicles that are heading in the same direction, making the route more stable.

When there are more nodes, AODV is more efficient, as it is continuously finding new routes. Furthermore, AODV has fewer end-to-end delays as it has a higher number of collisions [4].

2) DSDV: Destination-Sequenced Distance Vector routing is a constructive routing protocol based on a table which deals with the route estimation Bellman-Ford algorithm. The critical contribution of the routing algorithm DSDV was to solve the routing loop problem. For the network nodes, DSDV establishes a routing table. Since each node manages the routing table, the routing table will need to be changed more regularly if a router receives new information. With this information available in the routing table, through intermediate nodes [30], the source transmits the packet to the destination. Each node must identify the next hop to the collector and the number of hops away from the collector.

If a path already exists before the arrival of traffic, it would be conveyed without delay. Otherwise, traffic packets must wait in a queue before the routing information referring to their destination is obtained by the node. The node automatically changes the sequence number and transmits the information to its neighbors if a path fails to the next node. It checks its routing table when a node receives routing information. If the routing table does not locate this entry, the routing table will be modified with the routing information found. If the node discovers that it already has an entry in its routing table, the sequence number of the obtained information is matched with the entry in the routing table, and the information is modified. The information can be discarded with the lowest sequence number if it is smaller than the obtained information. The node will hold the knowledge that it has the shortest path or the least number of hops to the destination if the two sequence numbers are the same. In DSDV, the routing table maintains a routing table for the destination node. The table will include all the other nodes known directly or by any neighbors known. The entry in the routing table for each node includes details about the IP address of the node's last known sequence number and the hop count to access that node. The table also holds a record of the next-hop neighbor to enter the destination node along with this information, the timestamp of the last update obtained for that node [31].

In DSDV, the distribution pace of the package falls dramatically, and one of the critical reasons for this is the use of already closed routes in the event of broken connections. The existence of paralyzed routes in DSDV does not mean that the path to the destination cannot be identified by [1]. The protocol provides a temporary connection to the desired destination through a neighbor that has a valid path [32]. However, as the number of vehicles is expanded relative to the other protocols, the outcome indicates a bad influence of the DSDV protocol. DSDV is also not a reasonable option if the number of vehicles rises considerably [33].

3) OLSR: The OLSR (Optimized Link State Routing Protocol) is a proactive routing protocol created for AdHoc mobile network types. This routing protocol depends on efficient periodic flooding of control information because it is a classic link-state routing protocol and when using specific nodes that act as Multipoint Relays (MPRs). OLSR uses a genetic algorithm to boost efficiency, adjusting the parameters and variations in the experiments tested.

Inside each node in the network topology, OLSR maintains a routing table to create a data transmission path. The basic principle used in this protocol is that messages are exchanged during the flooding process by the nodes chosen at the front. To discover its one-hop neighbors and even its two-hop neighbors through its replies, OLSR uses "hello" messages. This approach greatly decreases the overhead of the packet, contrasting a conventional flooding process in which all nodes retransmit each packet until the first copy of the message is received [9].

The features of this protocol are that the delay times created by sending data packets are short and ideal are well adaptable to topology changes and are easily integrated with various types of systems.

This protocol has two essential functions: discovering the neighborhood for each node and distributing the topology that transfers three different kinds of messages [13].

When using OLSR, in the event of faults and failures, there is a frequent exchange of messages about the topological details of the whole network. To minimize air traffic control, it uses the Multipoint Relay (MPR) technique. The OLSR operation regularly produces two messages on the network:

- Hello message: These messages allow up to two hops for each node to know its neighbors. Each node makes use of this information to pick its multipoint relay nodes.
- Topology control message: Each node transmits control messages known as network topology control messages to preserve the database necessary for routing packets. To build a set of selectors MPR [34], this message is transmitted periodically by separate nodes.

Consequently, for transmitting data packets, more bandwidth would be available. This protocol seems to improve packet transmission rate, latency, overhead routing, and throughput [34], based on the simulation analysis.

4) GPSR: Owing to the prevalence and performance of satellite navigation systems, researchers have suggested various routing protocols tailored to vehicular networks to develop routing protocols in VANET. This has become a hot spot for testing, and the Greedy Perimeter Stateless Routing Protocol (GPSR), is the most promising routing protocol category using this method. It uses a traditional routing technique based on a location that uses information from nearby vehicles to determine which neighboring node to receive the data would be chosen [12].

It uses greedy forwarding and perimeter forwarding. GPSR usually forwards data packets using a greedy algorithm; there is a local optimization problem, it uses perimeter forwarding to send data packets, ensuring that the packet forwarding does not pick the next hop with a greedy algorithm. In comparison, GPSR belongs to the protocols of greed routing and does not include a table of maintenance routing. The protocol functions practically stateless and has multipath routing capability [35].

Through the short-range position and location system, it is assumed that each node has its position coordinate information available. Nodes regularly share this information through beacon messages with their neighbors in a single hop. Therefore, within the contact spectrum, each node has the position information of all its neighbors at any time, as well as the position of the destination by beacon messages and location services. According to the greedy routing, the current node selects the best neighbor nearest to the destination based on the response of the beacon messages. However, after a timeout interval, if the current node does not obtain an answer from a neighbor, it finds the contact relation to be broken and deletes all entries from the neighbor table. Some conditions may occur in which there is no better neighbor than the node itself, which is regarded as a maximum local state.

The GPSR can no longer keep the greedy forwarding technique in this situation but instead transforms into a recovery mode to forward the packet to the next node. Both nodes obey the right-hand rule in the recovery mode technique to transmit the packet to the next node. Each node tests the packet header field after receiving the packets, either in Greedy mode or in Recovery mode.

In the perimeter routing mode of the GPSR algorithm, the right-hand rule can ensure that the GPSR protocol can get out of the “hole” but makes a choice arbitrary to a certain extent, which often leads to finding more leaps in the routing. An improved GPSR routing algorithm using the two-hand rule is proposed to solve the GPSR “hole” problem in wireless sensor networks. The double-hand rule includes the left and right-hand rules. The right-hand rule will guarantee that the GPSR protocol can get out of the “hole” in the perimeter routing mode of the GPSR algorithm but makes the decision random to some degree, always leading to more significant routing leaps.

A stable and improved variant of the Greedy Perimeter Stateless Routing (GPSR) protocol is introduced by the overlay. This protocol consists of two modules incorporating an upgrade that minimizes transmission delays and message control in the GPSR routing protocol [12]. In [35], the authors proposed an update to the GPSR routing protocol, which preserved its functionality, such as low complexity and simple realization. The simulation results in the NS-2 software show that the improved GPSR algorithm dramatically improved performance and demonstrated the efficiency of the proposed solution in the delay and packet arrival rates compared to the original GPSR.

Cross-layer data is often used as a complement under practical limitations of the protocols in general. A practical implementation using this information was used in the work of [36], which identified barriers and other unforeseen causes of fading, such as interference, contributing to the phenomenon of radio irregularity, to update and improve the GPSR protocol. This update was called GPSR over Symmetrical Links (GPSRSL). The experimental review shows the effectiveness of the GPSRSL compared to the original GPSR, where the results indicate that the proposed protocols allow a high packet delivery ratio without increasing energy consumption, maintaining a reasonable delay for the end-to-end application concerning the original GPSR.

Next, we will cover other protocols, not so well known, but that use geolocation information and inspire ideas to carry out this research work.

B. Vehicle Position and Route

There are many ways of relaying packets between various network nodes, as seen in previous protocols. In general, “hello” packages are often used between devices. Give these parcels to identifiable adjacent centers and become acquainted with their locations. The time to submit the course demand packages occurs as the initiating hub distinguishes its extensive hubs and locations using GGNS.

Contrary to several conventions, the Advanced and Enhanced Security Protocol for VANET (AESP-VANET) was established by [37], where the originating node does not send demands to each neighboring hub. Instead, it only sends them to adjacent hubs with increasingly desirable features, such as less travel speed than the source hub, less disparity between the source hub, shorter separation, and information obtained through the GGNS system.

Another technique that improves the “hello” message generation rate in rapid motion was developed by [38]. They have an OLSR extension called Proactive-OLSR (P-OLSR) that uses the details available from GGNS to weight the parameter of the expected transmission count (ETX). The relative speed and direction of motion of unmanned aerial vehicles (UAVs) [34] are also considered.

Solutions suggest the joint optimization of essential device parameters that systematically reside in various IOS layers to achieve the best detection capabilities. Nodes are responsible at the network level for gathering data from adjacent nodes to direct attitudes, while they are responsible for receiving warning notifications at the application layer level. Subsequently, with external knowledge found on other computers, such as the GGNS system, the local decision is taken.

Going on to [39], the vehicle cross-message authentication scheme in Wireless Access in Vehicular Environments (WAVE) of the IEEE is proposed to validate the protection application in the received message. The authentication process is the generation of a signature, the transmission of a periodic security message, and the message's authentication [40].

To create a path, [41] suggested an AODV routing protocol based on a global positioning system (GBAODV). To enhance routing performance, the influx of AODV routing packets using GGNS devices was restricted. It was found that their GBAODV technique decreases the load of the network more than the AODV technique. Therefore, it decreases the number of broken connections and the amount of packet failure. Also, the average end-to-end delay is shorter than AODV when using GBAODV. However, the efficiency of the GBAODV system for packet loss was not adequate when considering various path scenarios. Also, this analysis was focused on only eight nodes. To increase the efficiency of the routing protocols, [42] introduced vehicle movement details into the route discovery process based on an AODV for VANET applications. The proposed protocol to achieve more reliable routing considered the total route weight (TWR) (based on location metrics) and the expected expiration period. The suggested protocol decreases the routing burden even more and guarantees more reliable links, they found. However, there is no substantial change in the percentage of packaging disposal in the current protocol [3].

This article proposes a complementary scheme for the routing protocol in enhanced V2V networks that incorporates time savings and collision prevention at relay stations. We use the route information available in the geolocation equipment installed in vehicles today to make this possible. The simulated results assuming an intersection show that this significantly increases the average proportion of package delivery with the proposed scheme. Next, the proposed scheme and its implementation will be discussed.

III. A Proposed NAV2V Routing Scheme

In this chapter, a Navigation Assisted V2V Routing Protocol (NAV2V) for Urban Areas is proposed. NAV2V is a solution for vehicle connectivity problems in an urban area. The proposal utilizes global position navigation information to determine next-hop neighbors and the delivery time packet. We use graph theory to build the scenarios, and we propose an algorithm to solve the problem. Based on the number of neighbors, the algorithm selects the optimal host for forwarding a packet to the destination.

We divide this chapter into three parts: Section III-A describes how we can use the position information available in the vehicles to improve the routing delivery packet scheme. In contrast, Section III-B introduces our proposed solution to build all possible route communication using graphs, and III-C, we formalize our proposed algorithm to solve the problem and the rest details.

A. Position Information

Geographic and Location-Based Services (LBS) for ITS exploit knowledge of where a user is located to deliver location-specific services. Therefore, LBS must identify its geographical location [43]. GNSS, short of Global Navigation Satellite System (GPS, Galileo, GLONASS), is the most accurate means of identifying the location (where the sight of enough of their satellites is known). ITS applications are location-based facilities. Time-critical and many other safety facilities, including collision avoidance, rely on the precise location of the vehicles involved. However, LBS's role in providing security services where knowledge of the exact location is vital. With the city map and the aid of GNSS on board the vehicles, it is possible to determine if the vehicles are on the same street, to determine the crossing points and if it is possible to have data communication between the hosts. This information is essential for defining the message routing hops and calculating the distances between the vehicles, as shown in Fig. 2. The communication area of a given vehicle is calculated using the area formula of an ellipse, according to equation 1. Where a is the Line-of-sight propagation (LOS) and b is the *StreetLength/2* [44], [30].

$$RadioCoverage = \pi * a * b \quad (1)$$

To determine whether two vehicles are within communication range, the following equation 2 was used. If the inequality results are less than or equal to 1, then the vehicle is within the communication range, and if the inequality is not satisfied, the car2 is outside the communication range [45], [30].

$$(car2_x - car1_x)^2 / a^2 + (car2_y - car1_y)^2 / b^2 <= 1 \quad (2)$$

As we can see in Fig. 2, the vehicle in the crossroad can be a bridge to the other two cars communicating. In this work, we list some assumptions and illustrations: Instead of infrastructure mode, vehicle-to-infrastructure (V2I), the vehicle sends the packets only in AdHoc mode, which means vehicle-to-vehicle (V2V) communication.

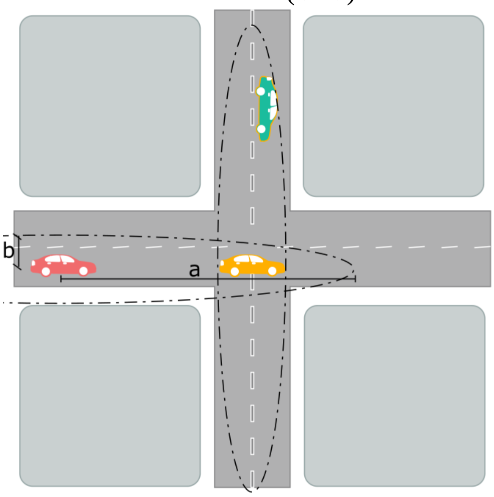

Fig. 2. Range communication system to calculate the distances and determine the crossing points to build the routing possibilities.

NAV2V is a geographic position-based routing protocol that assumes that each vehicle is equipped with GNSS to get the location information of itself and others as well if 802.11p is used. The IEEE 802.11p Cooperative Awareness Messages (CAM) contain information about the transmitter's vehicle speed, position, and direction.

Every vehicle knows its coordinates, and a central controller of the perimeter knows all vehicle positions.

Then, the controller calculated the best route using our proposed solution and distributed the route-table to the hosts that are part of a particular communication perimeter, for example, the downtown.

With the information of the routes of the vehicles that are part or will enter the perimeter of the controller, it is possible to create a graph with all the possibilities of communication between all hosts and use the time that this communication can occur, as the graph's primary attribute.

Now we will show the formalization of the route problem and detail the proposed algorithm to solve the issue.

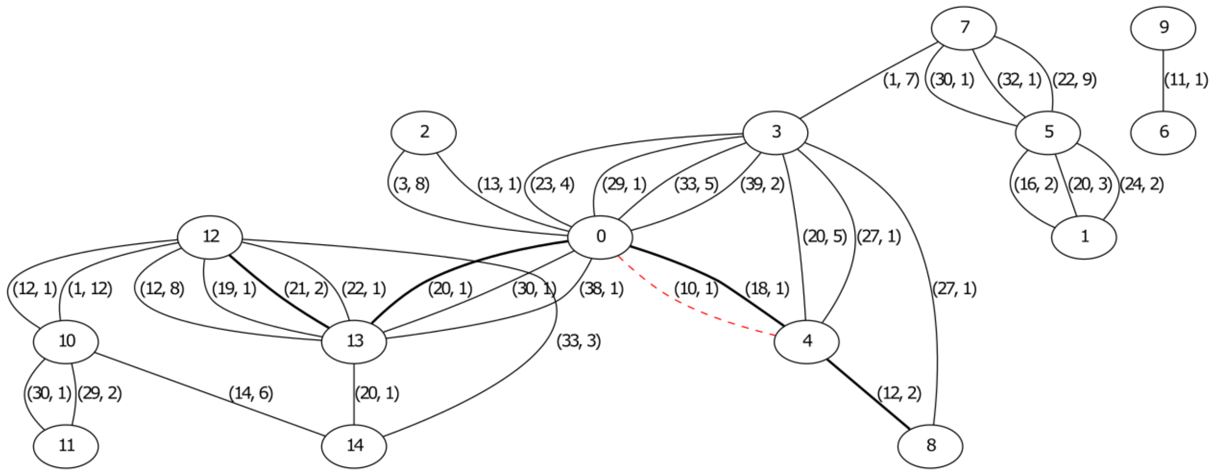

Fig. 3. Graph generated by calculating the distances between vehicles at each given time point. This graph demonstrates all the possibilities of communication between 15 cars in the same scenario.

B. *Algorithm*

The formalization of the problem:

Let $G<V,E,T,W>$ be the time-dependent direct graph G with multi-attributes, associated with the considered VANET topology, where the communication can occur in both directions. Where $V = \{n_0,...,n_W\}$ is the set of vertexes with $|V| = W$, and $E = \{(n_0,n_1)_1,(n1,n_m)_2, ...,(n_j,n_D)_M\}$ be the set of edges with $|E| = M$; and T and W are two sets of non-negative attributes functions. For every edge $e = (u,v) \in E$, there are two functions: timefunction $f_e(t) \in T$ and weight-function $w_e(t) \in W$ where t is a time variable and w is a window size variable. A time function $f_e(t)$ specifies what time the communication can occur from u to v. A weight-function $w_e(t)$ specifies how long the communication can occur from u to v, if departing from u at time t. We assume that time-function $f_e(t)$ is a piecewise constant function, calculated as follows:

$$f_e(t) = \begin{cases} w_1, 0 \le t \le t_1 \\ \vdots \\ w_n, t_{n-1} \le t \le t_n \end{cases} \tag{3}$$

Another important property is the *DeliveryTime* of the packet when the package will be delivered to the destination. This value is calculated using equation 4 and is used in the evaluation chapter to compare the gain of our proposed solution with other existing protocols. The maximum value of t is when the package will be delivered to the destination $P_k(n_{D-1},n_D,T,W)$.

$$DeliveryTime = \{T \mid P_k(n_{D-1}, n_D, T, W)\} \tag{4}$$

It may be possible to tie the shortest time in the transmission window. In such cases, use the following rules to choose the best path:

1) Which path has the least *hops*.
2) Which is the path with the smallest *Aggregate Path Loss,* as shown in equation 5 [46].

Path loss is an attenuation that decreases the power density of an electromagnetic wave as it transmits through the channel. There are distinct elements of path loss varying from the natural spread of the radio wave, diffraction path loss grows because of interference, to saturation path loss that exists for the existence of a signal that is not transparent to electromagnetic waves [47]. We use the sum of all path losses between the communication hops. As a variable in the equation, we have D as *Distance* between the transmitter and the receiver, and the λ is the free space wavelength defined as the ratio of the speed of light in meters per second to the carrier frequency in Hz as shown in equation 6. The equations are based on [48] work.

$$PathLoss_{Aggregate} = \sum_{0}^{e} \left(\frac{10 * log_{10}(16 * \pi^2 * D_e^2)}{\lambda^2} \right) \tag{5}$$

©2022 Journal of Communications

$$\lambda = \frac{c}{f_c} \quad (6)$$

For example: As shown in Fig. 3 bold edges, Let P_1 be the set of available transmission path for a source node n_8 and destination node n_{12}, while n_4, n_0, n_{13}, n_{12} is the forwarding nodes in the considered scenario, so $P_1 = \{n_8, n_4, n_0, n_{13}, n_{12}\}$. Among all the available paths, the algorithm tends to choose the one with the shortest time *t*, that is, the end-to-end delay will be the shortest possible, taking into account the time *t* attribute, where the time t of the neighbor must be greater than the t of the current node. Because of this, it is not possible to use the path between *n4* and *n0* marked in a red line because the attribute of the edge *(n4,n0,10,1)*, *T = 10* is less than the *T = 12* of the previous edge *(n8,n4,12,2*). So, it is necessary to use the route available at time *T = 18* in *(n8,n4,18,1).* Therefore, the bold path represents the best route selection to transmit the information.

As we can also see in Fig. 3, nodes 6 and 9 are disconnected from the rest of the nodes, this means that they can only communicate with each other.

Algorithm 1 represents the pseudo-code used to represent the V2V routing protocol assisted by controlled navigation for urban areas.

Algorithm 1 Steps of the NAV2V algorithm:

```
Input: gpdDat, map.xml, source, destination
Output: path route between source and destination
Step 1: Identify positions and crossing points
  ∀ time/position available:
  Calculate distance between cars in the same street
if distance <= communication_range then
  create a list of possible neighbors
end if
  Calculate the time t neighbors will be available
  Calculate the PLoss neighbors
Step 2: Graph
  Generate all possible communications, with minimum time t
Step 3: Run SPT Algorithm
  Select the shortPath, based on time t
  if There is more than one option: then
    Choose the minimum Hop
    if There is more than one option: then
      Choose the minimum PLoss
    end if
  end if
Return: shortPath
END of the algorithm
```

We divide the algorithm into three steps. The first step identifies the vehicle positions at each timestamp, based on the input files of the city map and the gpsDat, which are the vehicles' routes in the same perimeter. Calculate the distance between vehicles and create a list with neighbors in different timestamps. Based on the distance between neighbors, it is possible to calculate the path loss based on equations 5 and 6. In the second step, we create a graph with all neighbors in unique periods. As this algorithm aims to deliver the package as soon as possible, the shortest delivery time available between the vehicles is chosen, and we discard the other information. In the third step, we choose the shortest path using the Short Path Temporal (SPT) algorithm, between the source and destination entry, based on delivery time information. If there are alternative paths, first, the algorithm chooses the one with the least number of hops and then the lowest aggregated path loss, as explained before.

Next, we will explain the Short Path Temporal algorithm.

C. Short Path Temporal Algorithm - SPT

To solve our graph problem, we need to find the shortest path of the graph, considering time as an essential attribute. Because of this, it is not possible to consider other algorithms for the short path, as they try to minimize the weight without considering the time dependence. In other words, the time constraint means that the next edge must have a time equal to or greater than the current node time, as the function *fe(t)* in equation 3.

As a solution to this time-dependent problem, we propose the following modification to the algorithm Bellman-Ford algorithm [49], and we call this solution as Short Path Temporal (SPT) algorithm, described in Algorithm 2.

Algorithm 2 SPT

```
Input: list vertices, list edges, vertex source
Output: distance[], predecessor[]
Step 1: initialize graph
for each vertex v in vertices do
  distance[v] := inf
  predecessor[v] := null
end for distance[source] := 0
Step 2: relax edges repeatedly
for i from 1 to size(vertices) - 1 do
  if w ⩾ distance[v] then
    if distance[u] > w then
      distance[v] := w
      predecessor[v] := u
    end if
  end if
end for
Return: distance[], predecessor[]
END of algorithm
```

Now we will explain how the simulation and evaluation process of the algorithm proposed in this research works compared to the existing and most used protocols in V2V communication nowadays.

IV. Simulated Environment

Researchers typically use computer simulation programs to test and analyze the results of their research, as they are more flexible and less costly compared to the real environment.

The proposed research work has used *Network Simulator 3 (NS-3) version 3.30* [50] to evaluate the performance of the proposed scheme. *NS-3* is a discrete event network simulator written in C++ and Python and is a computationally efficient and practical computer language. Under this, the module for *NetAnim 3.108* is included to animate our network scenario and observe the packet flow in the network while our simulated vehicles communicate with each other.

Using *NS-3*, the vehicle movements are simulated by uploading the map and vehicles set up first. The *mobilitytrace.cc* file is utilized from the *NS-3* directory to incorporate the NetAnim simulator code to be run directly from the *NS-3* simulator. Further, the *object.tcl* file is created in the same directory, which would help us create our own defined scenarios for each of the protocols used from the same source file with just a simple creation of a new object instance [1].

A generic grid city map was used as a scenario in our simulation, called Manhattan [51] grid. In this simulation work, the time agreed by us was 50 seconds, and the *Simulation of Urban Mobility (SUMO)* [52] was introduced to obtain a realistic scenario.

The *SUMO* software is open-source software that provides an accessible platform for simulating the land mode of transportation. It creates a simulation wherein the vehicles will be treated as nodes, and with pedestrians in view, a network is created. It comes with an improvised tools setup that provides a convenient platform for creating various scenarios that a researcher could use to experiment [1].

Fig. 4 represents the urban scenario set in the 5 x 5 grid topology. In this model, the movement pattern of mobile nodes in urban environments is simulated using a grid road topology. This mobility model contains horizontal and vertical roads, where nodes are randomly placed on the map at the beginning of the simulation and can change lanes. When a node reaches the intersection, it can continue by turning left, turning right, or going straightforward, randomly [23].

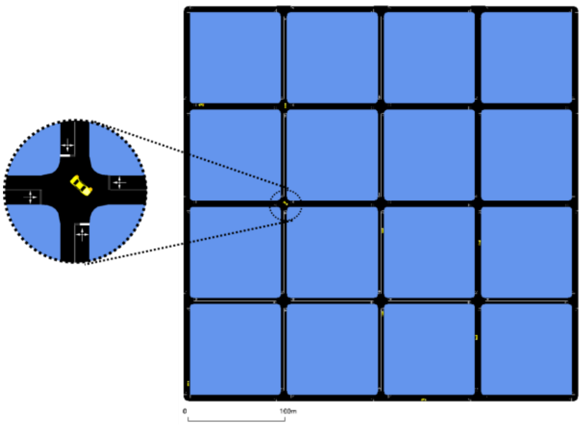

Fig. 4. Urban scenario set in the 5 x 5 grid topology simulation in SUMO. This model represents a typical square downtown.

The Diagram Process flow for the experimental setup is represented in Fig. 5, and several steps are necessary to set up this model. First, we used the *Netedit* tool to create the map (Manhattan grid style) and the buildings identified as the *map.xml* and *buildings.xml* files. These files are also used in NAV2V and NS-3, respectively. *OpenStreetMap* and *Polyconvert* could be used for the same purpose, as used in other works [2], [53]-[55]. After that, the *randomTrips* tool was used to generate the random routes of the vehicles and the number of vehicles in each simulation, identified as *route.xml*. To finish the first part of the simulation in the SUMO tool, the configuration file *sumo.cfg* was used. As a simulation output in SUMO through fcd-output, the *simtrace.xml* file is generated. The *traceExporter* tool makes it possible to create the *gpsDat* file used in the python simulation of the proposed NAV2V algorithm and the *mobility.tcl* file that will be used in NS-3.

TABLE I: SIMULATION PARAMETERS

Parameter	Value
BSM size	200 bytes
BSM rate	10Hz
Transmit power	20 dBm
Frequency	5.9 GHz
Channel bandwidth	10 MHz
Channel access	802.11p OCB
Tx range	0 - 300 m
Encoding	OFDM
Rate	6 Mbps
Propagation loss model	Two-ray ground
Simulation time	100 s
Fading Model	Obstacle Shadowing

Besides the *buildings.xml* file, it is necessary to generate the obstacles and the *mobility.tcl* file, which is the vehicle's movement file, it is necessary to use the vanet.cc file. This file is a modification of the *vanet-routing-compare.cc* file available in version NS-3.30.1 and contains all the configurations for the simulation. Table I represents the parameters used in the simulation.

Because of the NS-3 simulation, we have the pcap files for each host (vehicle) in the simulation as output. Therefore, it is possible to filter using the *tcpdump* tool, and the extracted information will be used to evaluate and compare the existing protocols and the proposed solution. For the simulation done in Python, the map.xml and gpsdat files were used and processed in the proposed NAV2V algorithm, as explained in the previous chapter. The relevant result information is filtered and used for comparison with the NS-3 results.

In this work, we consider that BSM (Basic Safety Message) [56] information will be distributed to all vehicles involved in a given urban area. Our simulation uses the mobility trace files produced by SUMO simulations in NS-3, during which every vehicle emits a BSM 10 times per second. For example, if we have 20 vehicles traveling in a common area and we intend to send BSM information in that specific area, that means a distribution of 380 messages, that is, the Permutation (20;2), as shown in equation 7. A given car is the source of the BSM message and all other vehicles involved in the same coverage area as the destination of the information. The NS-3 simulation protocols are AODV, DSDV, OLSR, and GPSR to compare with our proposed NAV2V protocol.

$$P_{(n,r)} = \frac{n!}{(n-r)!} \quad (7)$$

As a result, a *pcap* file is generated for each vehicle used in the simulation. With this file, it is possible to analyze all packets received by the host. So, we made a filter script to extract the time that a given source sent the first received packet.

Packet Delivery Ratio: Represents the fraction of the data packets delivered from the source nodes to the destination nodes over a communication channel [2].

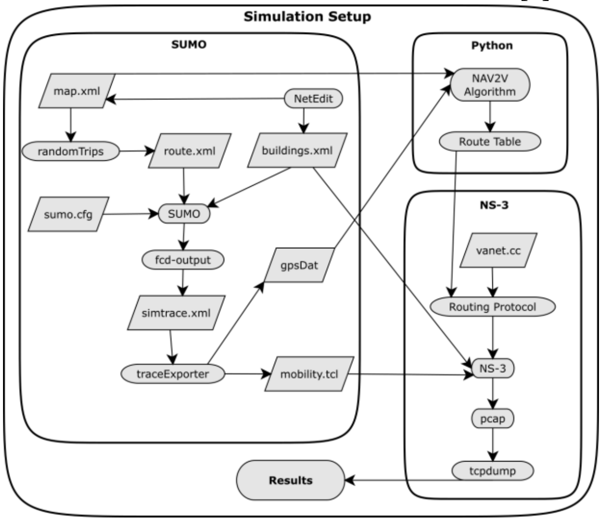

Fig. 5. Simulation configuration diagram to demonstrate the information flow used in this work.

Next, we will describe the obstacle model used in this paper.

A. Obstacle Model

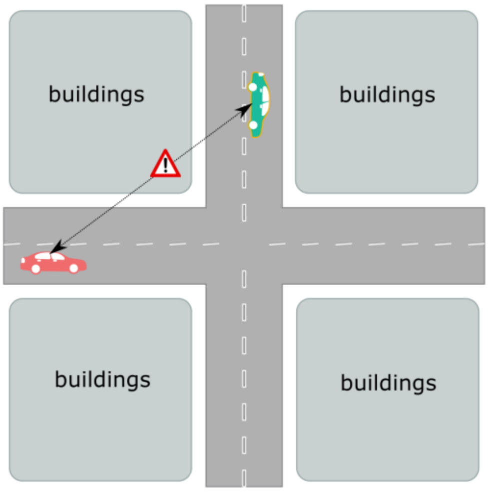

Fig. 6. Obstacle model used in our simulation, to represent the problem of interference of buildings in downtown.

Obstacles, such as buildings and trees interfere with radio wave signal propagation by contributing fading and shadowing effects. To produce results that accurately reflect real-world typologies, models must address the radio-interfering conditions that obstacles present. Failing to account for the effects of obstacles can, therefore, inaccurately overstate network performance. We implemented an obstacle shadowing model for the NS-3 network simulation toolset and tested using a script for wireless vehicular AdHoc network (VANET) scenarios and obstacle data from a map [57]. Fig. 6 represents the obstacle model used in our simulation.

V. Results and Discussion

To show the efficiency of the proposed algorithm, simulations were carried out with four existing protocols: AODV, DSDV, OLSR, GPSR, and our proposed scheme, NAV2V. Each protocol was simulated with different numbers of cars on the map; 5 to 30 cars were evenly distributed and used random travel to generate traffic, as explained in the simulation topic.

The following metrics were used to evaluate the performance of the proposed protocol: (1) Percentage proportion of packets delivered rate, that is, the number of packets that are received correctly at a destination with the number of packets that were sent by the origin. (2) Delay time in seconds, the end-to-end delay between source and destination. In addition, we use the (3) average aggregate delay time, where we average the delivery time of messages with different vehicle densities, to compare each protocol in a generalized way.

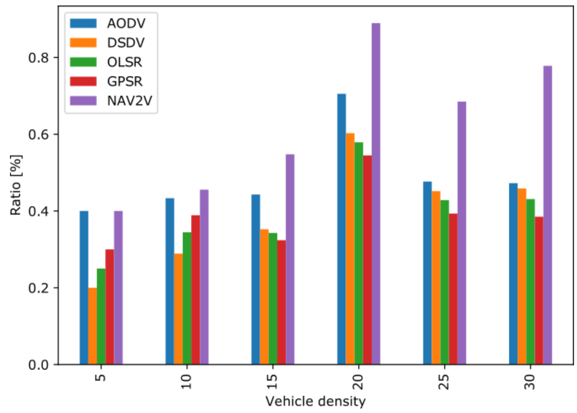

Fig. 7. Total possibilities of successful communication between all vehicles.

Fig. 7 shows us the total possibilities of successful communication between all vehicles, measured in percentage. As the result shows, it is the same as the AODV protocol in our proposed scheme when the network is sparse (with only five vehicles on the map). However, as the number of nodes increases in the simulation, the performance of the proposed NAV2V scheme is noticeably better, reaching a 30% gain.

The estimated point and confidence intervals of the aggregate number of possible communications of all simulation tests can be seen in Fig. 8. According to the obtained results, NAV2V can offer until 25% higher communication possibility compared to AODV. Because as the density of cars increases, so does the number of possible paths to transmit the data, and our algorithm can make the optimized selection.

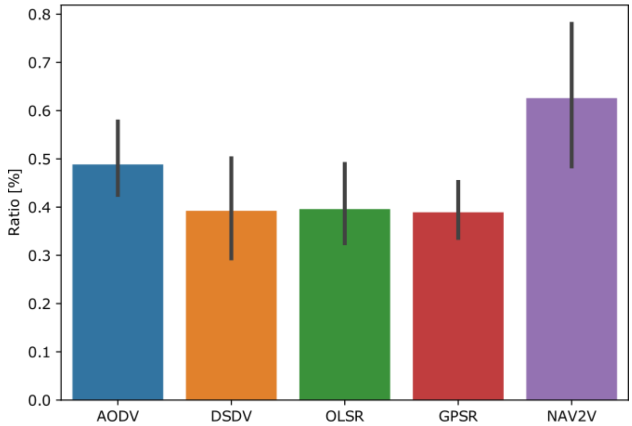

Fig. 8. Mean of possible communication and confidence intervals of the aggregate number of possible communications of all simulation tests.

After estimating if communication was possible, we measure the delivery time for these messages according to the previous graphs. The following figures represent the values in seconds in the absolute time of the simulation. We assume that all vehicles send the BSM message in a broadcast since the zero time of the simulation and continue sending the message at a rate of 0.1 messages per second. We used this method only to compare the protocols. In the natural environment, messages are triggered according to the occurrence of security events.

We illustrate the average delivery time of the packages in Fig. 9, separated by simulation. As many more packets are served, the average number of delays can also increase, but the value is close to the other protocols.

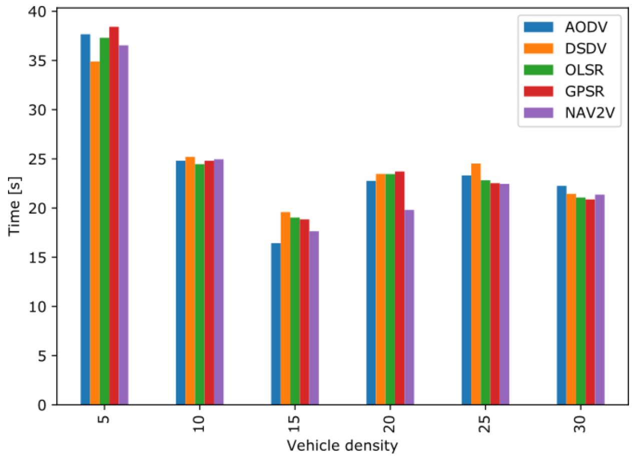

Fig. 9. Mean delivery time of packet.

Although the sum of the average package delivery time gain is small, as we can see in Fig. 10, we must remember that this scheme reached up to 30% more packages successfully delivered.

Then, to view the delivery time of the first package between origin and destination, we generate a boxplot, illustrated in Fig. 11, as the results show the differences in the time delivery packages between the routing protocols used in this work. Moreover, we can verify the time savings that the NAV2V scheme provided.

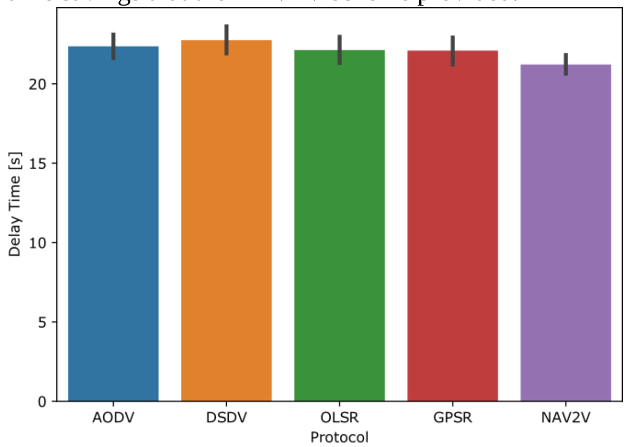

Fig. 10. Estimate delivery time and the respective confidence intervals.

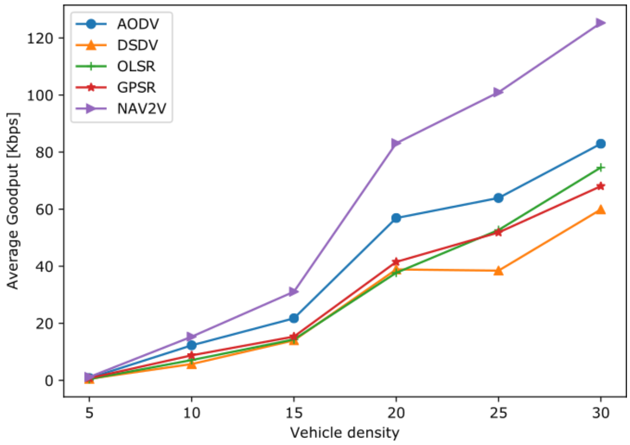

Fig. 11. Time delivery packets differences between the protocols.

To evaluate the behavior of NAV2V, we measured the time required to deliver the message from the source to the destination. Fig. 12 was generated to compare using proper bandwidth, that is, the Goodput in the simulations. In other words, the NAV2V solution has better use of the bandwidth when compared to the other protocols.

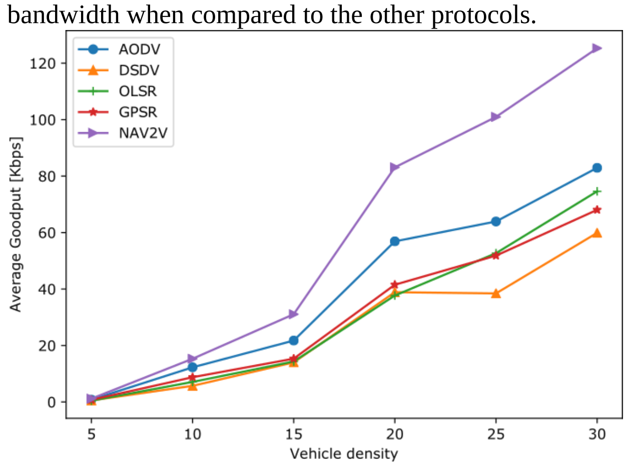

Fig. 12. Bandwidth (goodput) used by each protocol in the simulation.

Packet reception rate is an important performance metric to compare the efficiency of different VANET routing protocols. As well as the useful bandwidth, we can see that the packet reception rate is also noticeably better than the proposed algorithm, highlighted in Fig. 13 per simulation time and Fig. 14 as the average rate in all simulation time.

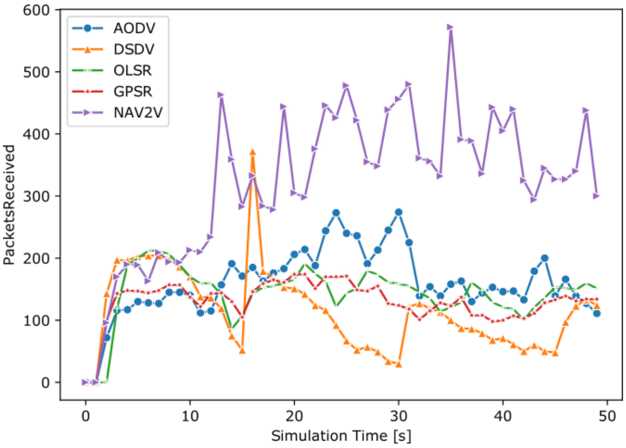

Fig. 13. Comparison of packets received from different protocols.

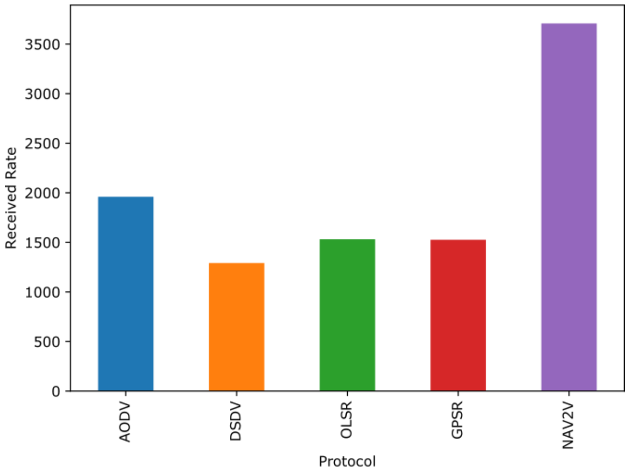

Fig. 14. Average rate of receiving packets by simulation.

A. Discussion

A new algorithm for AdHoc communication between vehicles has been proposed. The algorithm uses the GGNS information already available in the vehicles' onboard equipment. Simulations were made using the latest applications available for this environment. The results and improvements in the performance of the algorithm were proven in the simulations. This proposal can be widely used for the V2V communication system and later adapted for the V2X system for future research.

VI. Conclusion

In VANET, the nodes travel quickly, and the topology also changes, so the routing architecture is a challenge. The next generation of vehicular network communication systems must work collaboratively to provide excellent performance. In this article, we have investigated in depth the routing protocols AODV, DSDV, OLSR, and GPSR, which are the most used and adequate today in this type of system. A scheme to complement the routing, called NAV2V, has been proposed. This scheme uses the geolocation information in a Global Navigation Satellite System (GGNS), embedded in automobiles nowadays. These are essential items for the evolution of communications between vehicles until their evolution is fully autonomous cars.

The results of the simulations showed that there were problems with connectivity and selections for the next hop. NAV2V addressed connectivity and next-hop selection problems and also used route mechanisms to forward packets to the next available node for forwarding to the destination. We compared the performance of NAV2V with the other routing protocols implemented in the NS-3 simulation tool. The simulated results show that the proposed scheme significantly improves the performance of V2V communication. The terms of package delivery fees and end-to-end delays have been considered. The simulation results showed that the package delivery rate for NAV2V could reach up to 30% better in some cases.

As presented in this work, the current protocols and their constant change with optimization schemes are essential for the evolution of routing protocols since each has its characteristic of solving a specific problem. The proposed solution is a support to the other existing protocols in order to optimize the communication in the vehicular networks.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Lincoln Herbert Teixeira conducted the research, simulations and prepared the paper; Árpád Huszák proposed an idea, analyzed the data end review the text. All authors had approved the final version.

References

[1] V. Kumar and R. Kumar, "Smart re-route routing protocol for vehicular ad-hoc networks," in *Proc. Second International Conference on Inventive Research in Computing Applications (ICIRCA)*. IEEE, July 2020, pp. 758–763.

[2] S. Jobaer, Y. Zhang, M. A. I. Hussain, and F. Ahmed, "Uav-assisted hybrid scheme for urban road safety based on vanets," *Electronics (Switzerland)*, vol. 9, no. 9, pp. 1–19, 2020.

[3] A. Ahamed and H. Vakilzadian, "Impact of direction parameter in performance of modified AODV in VANET," *Journal of Sensor and Actuator Networks*, vol. 9, no. 3, p. 40, 2020.

[4] C. P. S. Canar, J. J. T. Yĕpez, and H. M. R. Lopez, "Performance' of reactive routing protocols DSR and AODV in vehicular ad-hoc networks based on Quality of Service (QoS) metrics," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 4, pp. 2033–2039, 2020.

[5] H. Fatemidokht and M. K. Rafsanjani, "QMM-VANET: An efficient clustering algorithm based on QoS and

monitoring of malicious vehicles in vehicular ad hoc networks," *Journal of Systems and Software*, vol. 165, pp. 1–16, 2020.

[6] A. Silva, N. Reza, and A. Oliveira, "Improvement and performance evaluation of GPSR-Based routing techniques for vehicular ad hoc networks," *IEEE Access*, vol. 7, pp. 21722–21733, 2019.

[7] Z. Du, C. Wu, X. Chen, X. Wang, T. Yoshinaga, and Y. Ji, "A VDTN scheme with enhanced buffer management," *Wireless Networks*, vol. 26, no. 3, pp. 1537–1548, 2020.

[8] S. Zeadally, R. Hunt, Y. S. Chen, A. Irwin, and A. Hassan, "Vehicular ad hoc networks (vanets): Status, results, and challenges," *Telecommunication Systems*, vol. 50, no. 4, pp. 217–241, 2012.

[9] T. S. Chouhan and R. S. Deshmukh, "Analysis of DSDV, OLSR and AODV routing protocols in VANETS scenario: Using NS3," in *Proc. International Conference on Computational Intelligence and Communication Networks, CICN 2015*, 2016, pp. 85–89.

[10] A. Katiyar, D. Singh, and R. S. Yadav, "State-of-the-art approach to clustering protocols in Vanet: A survey," *Wireless Networks*, vol. 26, no. 7, pp. 5307–5336, 2020.

[11] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network," *Journal of Network and Computer Applications*, vol. 37, no. 1, pp. 380–392, 2014.

[12] M. Houmer, M. Ouaissa, M. Ouaissa, and M. L. Hasnaoui, "SE-GPSR: Secured and enhanced greedy perimeter stateless routing protocol for vehicular ad hoc networks," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 14, no. 13, p. 48, 2020.

[13] W. K. Ahmed, M. N. B. M. Warip, W. K. Abduljabbar, and M. Elshaikh, "Comparing and assessing the enhancements of DYMO and OLSR in VANETs," in *Proc. IOP Conference Series: Materials Science and Engineering*, vol. 917, p. 012078, Sept. 2020.

[14] R. Hernandez-Jimenez, C. Cardenas, and D. Munoz Rodriguez, "Modeling and solution of the routing problem in vehicular delay-tolerant networks: A dual, deep learning perspective," *Applied Sciences (Switzerland)*, vol. 9, no. 23, 2019.

[15] V. N. Soares, J. J. Rodrigues, and F. Farahmand, "GeoSpray: A geographic routing protocol for vehicular delay-tolerant networks," *Information Fusion*, vol. 15, no. 1, pp. 102–113, 2014.

[16] J. A. Dias, J. J. Rodrigues, F. Xia, and C. X. Mavromoustakis, "A cooperative watchdog system to detect misbehavior nodes in vehicular delay-tolerant networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7929–7937, 2015.

[17] J. N. Isento, J. J. Rodrigues, J. A. Dias, M. C. Paula, and A. Vinel, "Vehicular delay-tolerant networks? a novel solution for vehicular communications," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 4, pp. 10–19, 2013.

[18] N. Uchida, G. Sato, and Y. Shibata, "Vehicular Delay-Tolerant networks with image recognition-based adaptive ARRAY ANTENNA FOR WINTER ROAd surveillance in local areas," *Future Internet*, vol. 11, no. 9, pp. 1–13, 2019.

[19] H. Nascimento, P. R. Pereira, and N. Magaia, "Congestion-aware geocast routing in vehicular delay-tolerant networks," *Electronics (Switzerland)*, vol. 9, no. 3, pp. 1–15, 2020.

[20] G. U. Rehman, A. Ghani, M. Zubair, S. A. K. Ghayyure, and S. Muhammad. (2020). Honesty based democratic scheme to improve community cooperation for IoT based vehicular delay tolerant networks. [Online]. pp. 1–19. Available: http://arxiv.org/abs/2004.08698

[21] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, "VANet security challenges and solutions: A survey," *Vehicular Communications*, vol. 7, pp. 7–20, 2017.

[22] W. Yanbin, W. Zhuofei, Z. Jing, L. Zhijuan, and M. Xiaomin, "Analysis and adaptive optimization of vehicular safety message communications at intersections," *Ad Hoc Networks*, p. 102241, 2020.

[23] B. E. Bilgin and V. C. Gungor, "Performance comparison of IEEE 802. 11p and IEEE 802. 11b for vehicle-to-vehicle communications in highway, rural, and urban areas," *International Journal of Vehicular Technology*, vol. 2013, no. 2013, pp. 1–10, 2013.

[24] S. Kumar and H. Kim, "Packet rate adaptation protocol based on bloom filter for hidden node avoidance in vehicular ad-hoc networks," *IEEE Access*, vol. 7, pp. 137446–137460, 2019.

[25] Y. B. F. Essaadi, M. Dahchour, and Abstract, *Lecture Notes in Electrical Engineering 397 Advances in Ubiquitous Networking 2*, 2016.

[26] C. F. Wang, Y. P. Chiou, and G. H. Liaw, "Nexthop selection mechanism for nodes with heterogeneous transmission range in VANETs," *Computer Communications*, vol. 55, pp. 22–31, Jan. 2015.

[27] A. Di Maio, M. R. Palattella, and T. Engel, "Performance analysis of MANET routing protocols in urban VANETs," *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11803 LNCS, no. 4, pp. 432–451, 2019.

[28] O. Abedi, M. Fathy, and J. Taghiloo, "Enhancing aodv routing protocol using mobility parameters in vanet," in *Proc. IEEE/ACS International Conference on Computer Systems and Applications*, 2008, pp. 229–235.

[29] I. D. Chakeres and E. M. Belding-Royer, "AODV routing protocol implementation design," in *Proc. International Conference on Distributed Computing Systems*, 2004, vol. 24, pp. 698–703.

[30] M. Manjunath and D. H. Manjaiah, "Spatial DSDV (S-DSDV) routing algorithm for mobile ad hoc network," in *Proc. International Conference on Contemporary Computing and Informatics, IC3I 2014*, pp. 625–629, 2014.

[31] T. Mantoro and M. Reza, "Performance analysis of AODV and DSDV using SUMO, MOVE and NS2," in *Proc. International Conference on Informatics and Computing, ICIC 2016*, 2017, pp. 372–376.

[32] K. U. R. Khan, A. V. Reddy, R. U. Zaman, K. A. Reddy, and T. S. Harsha, "An efficient DSDV routing protocol for

wireless mobile ad hoc networks and its performance comparison," in *Proc. EMS 2008, European Modelling Symposium, 2nd UKSim European Symposium on Computer Modelling and Simulation*, 2008, pp. 506–511.

[33] M. Abdelgadir, R. A. Saeed, and A. Babiker, "Mobility routing model for vehicular ad-hoc networks (VANETs), Smart City Scenarios," *Vehicular Communications*, vol. 9, pp. 154–161, 2017.

[34] P. Kumar and S. Verma, "Implementation of modified OLSR protocol in AANETs for UDP and TCP environment," *Journal of King Saud University - Computer and Information Sciences*, 2019.

[35] L. Lai, Q. Wang, and Q. Wang, "Research on one kind of improved GPSR algorithm," in *Proc. 2012 International Conference on Computer Science and Electronics Engineering, ICCSEE 2012*, 2012, pp. 715–718.

[36] A. Benzerbadj, B. Kechar, A. Bounceur, and M. Hammoudeh, "Surveillance of sensitive fenced areas using duty-cycled wireless sensor networks with asymmetrical links," *Journal of Network and Computer Applications*, vol. 112, no. 8, pp. 41–52, 2018.

[37] A. Ganesh, S. Ayyasamy, and N. M. S. Kumar, "Enhancing the security of routing in vehicular ad hoc networks," *International Journal of Communication Systems*, no. 5, pp. 1–14, 2020.

[38] S. Rosati, K. Kruzelecki, G. Heitz, D. Floreano, and B. Rimoldi,˙ "Dynamic routing for flying ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, pp. 1690–1700, 2015.

[39] S. Biswas and J. Misiˇc, "A cross-layer approach to privacy-preserving authentication in wave-enabled vanets," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 2182–2192, 2013.

[40] R. Bhatt and I. Sharma, "Prediction-based next-hop forwarding strategy in vehicular ad-hoc networks," *International Journal of Vehicle Information and Communication Systems*, vol. 4, no. 1, pp. 1–19, 2019.

[41] Y. Sun, Y. Chen, and Y. Xu, "A gps enhanced routing protocol for vehicular ad-hoc network," in *Proc. IEEE International Conference on Robotics and Biomimetics*. IEEE, 2011, pp. 2096–2101.

[42] X. Yu, H. Guo, and W. C. Wong, "A reliable routing protocol for vanet communications," in *Proc. 7th International Wireless Communications and Mobile Computing Conference*. IEEE, 2011, pp. 1748–1753.

[43] B. Williams, *Intelligent Transport Systems Standards*, 2008.

[44] E. Giordano, R. Frank, G. Pau, and M. Gerla, "Corner: A realistic urban propagation model for vanet," in *Proc. Seventh International Conference on Wireless On-demand Network Systems and Services (WONS)*, Feb. 2010, pp. 57–60.

[45] U. Daepp, P. Gorkin, and K. Voss, *Finding Ellipses: What Blaschke Products, Poncelet's Theorem, and the Numerical Range Know about Each Other*, American Mathematical Soc., 2018, no. 34.

[46] M. Giordani, T. Shimizu, A. Zanella, T. Higuchi, O. Altintas, and M. Zorzi, "Path loss models for v2v mmwave communication: Performance evaluation and open challenges," in *Proc. IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*, Sep. 2019, pp. 1–5.

[47] A. Sari and A. Alzubi, "Path loss algorithms for data resilience in wireless body area networks for healthcare framework," in *Proc. Security and Resilience in Intelligent Data-Centric Systems and Communication Networks*, Elsevier, 2018, pp. 285–313.

[48] T. J. Rouphael, "Chapter 4 - high-level requirements and link budget analysis," in *RF and Digital Signal Processing for Software-Defined Radio*, T. J. Rouphael, Ed. Burlington: Newnes, 2009, pp. 87–122.

[49] L. Ford Jr and D. Fulkerson, "Flows in networks. Princeton landmarks in mathematics and physics," Princeton University press, 2010.

[50] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and Tools for Network Simulation*. Springer, 2010, pp. 15–34.

[51] G. Jayakumar and G. Gopinath, "Performance comparison of manet protocols based on manhattan grid mobility model," *Journal of Mobile Communication*, vol. 2, no. 1, pp. 18–26, 2008.

[52] P. A. Lopez, M. Behrisch, L. Bieker-Walz, *et al*., "Microscopic traffic simulation using sumo," in *Proc. 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018.

[53] A. Validi, N. Polasek, L. Alabi, M. Leitner, and C. Olaverri-Monreal, "Environmental impact of bundling transport deliveries using sumo: Analysis of a cooperative approach in austria," in *Proc. 15th Iberian Conference on Information Systems and Technologies (CISTI)*. IEEE, 2020, pp. 1–5.

[54] P. Singh, A. Agarwal, G. Nakum, D. Rawat, and S. Nandi, "Mpfslp: Masqueraded probabilistic flooding for source-location privacy in vanets," *IEEE Transactions on Vehicular Technology*, 2020.

[55] W. Arellano and I. Mahgoub, "Trafficmodeler extensions: A case for rapid vanet simulation using, omnet++, sumo, and veins," in *Proc. High Capacity Optical Networks and Emerging/Enabling Technologies*. IEEE, 2013, pp. 109–115.

[56] M. A. Javed, J. Y. Khan, and D. T. Ngo, "Joint space-division multiple access and adaptive rate control for basic safety messages in vanets," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2014, pp. 2688–2693.

[57] S. E. Carpenter and M. L. Sichitiu, "An obstacle model implementation for evaluating radio shadowing with NS-3," in *Proc. ACM International Conference Proceeding Series*, 2015, pp. 17–24.

Copyright © 2022 by the authors. This is an open access article distributed under the Creative Commons Attribution License (CC BY-NC-ND 4.0), which permits use, distribution and reproduction in any medium, provided that the article is properly cited, the use is non-commercial and no modifications or adaptations are made.

Lincoln Herbert Teixeira received his M.Sc. degree in Electrical Engineering and Industrial Informatics from Federal University of Technology – Paraná UTFPR, Brazil, in 2009. He is currently pursuing the Ph.D. degree with the Faculty of Electrical Engineering and Informatics at the Budapest University of technology and economics - BME in Hungary. His research interests include routing protocols; mobile networks; data communication; Cooperative Intelligent Transport Systems (C-ITS).

Ass. Prof. Dr. Árpád Huszák received his M.Sc. degree in 2003 as an Electrical Engineer from the Budapest University of Technology and Economics (BME) at the Department of Telecommunications. He received Ph.D. degree in electrical and computer engineering in 2010. Currently, he is with the Department of Networked Systems and Services as assistant professor, but he also joined the Mobile Innovation Center Hungary (MIK). He is a member of the IEEE and HTE. Recently, he has been involved in many European projects (FP6-IST, FP7-ICT, and Celtic). His research interests focus on network protocols, mobile computing, and adaptive multimedia communication over wireless networks.