Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

An implementation of IPv6 AODV protocol in Vehicular Ad-hoc Network using NS-3 simulator

Tayeb Diab[1], Abdelhak Mankour[2], Ali Sandid[3]
[1, 2, 3] *Faculty of Exact Sciences, University of Mustapha Stambouli, Mascara, Algeria*

ARTICLE INFO

Received: 30 Dec 2024

Revised: 05 Feb 2025

Accepted: 25 Feb 2025

ABSTRACT

Ad-hoc networks are designed to address specific communication needs in dynamic environments. Among them, Vehicular Ad-hoc Networks (VANETs) represent a special class of ad-hoc systems intended to enable reliable communication between vehicles and Road Side Units (RSUs) under conditions of high mobility and frequent topology changes. With the rapid growth of wireless-enabled devices, efficient routing has become a crucial challenge. One widely used routing protocol in VANETs is the Ad-hoc On-Demand Distance Vector (AODV). However, IPv4 addressing presents limitations in scalability and adaptability due to the large number of nodes and the highly dynamic nature of vehicular environments. To overcome these constraints, this work proposes an IPv6-based implementation of the AODV protocol for VANETs. Several simulation scenarios are conducted to evaluate and compare the performance of the IPv6 and IPv4 addressing schemes within the VANET context.

Keywords: VANET, AODV, IPv4 & IPv6 routing, NS-3 simulator, Ad-hoc wireless communications.

INTRODUCTION

Recently, with the rapid development of the automotive industry and wireless communication technologies, Vehicular Ad-hoc Networks (VANETs) have emerged as one of the most promising research areas. VANETs are a specific subclass of Mobile Ad-hoc Networks (MANETs), where mobile devices are replaced by vehicles capable of forming spontaneous wireless networks (Mohammadi et al., 2015). Introduced in 2001 under the concept of car-to-car communication and networking, VANETs enable vehicles to establish direct communication links and exchange information dynamically. They constitute a key component of the Intelligent Transportation Systems (ITS) framework, aiming to improve traffic safety, efficiency, and driving experience.

VANETs are typically defined as infrastructureless networks in which each node can act as a router to forward packets. Therefore, routing protocols play a fundamental role in ensuring efficient data dissemination among vehicles. Among existing MANET routing protocols, the Ad-hoc On-Demand Distance Vector (AODV) protocol (Perkins et al., 2003) is one of the most widely adopted due to its simplicity and effectiveness in dynamic environments. Developed originally by Nokia Research Center in 1991, AODV establishes routes only when needed and supports both unicast and multicast communication.

In this paper, we propose an IPv6-based implementation of the AODV protocol for VANET environments. The implementation is carried out using the NS-3 simulator. The remainder of this paper is organized as follows: Section 2 provides an overview of VANET networks and the existing AODV implementations in different simulators; Section 3 presents the proposed IPv6 AODV version and simulation setup; and Section 4 discusses the obtained results and performance analysis.

OVERVIEW ON VEHICULAR AD-HOC NETWORK (VANET)

VANET defines a specific type of mobile Ad-hoc network. Nodes within the network can be categorized in two types: mobile nodes “vehicles” and fixed nodes called RSU located in different environments (urban, rural or highway) and in critical locations such as service stations, intersections, etc (Diab et al. 2019a).

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The data exchange between nodes in the network is done in a wireless transmission space and in a multi-hop way. The vehicles use vehicle-to-vehicle (V2V) or inter-vehicular communication to communicate within each other and with the RSUs via vehicle-to-roadside communications (V2I). Each vehicle acts as receiver, transmitter and router to allow the exchange of data in the network.

VANET characteristics

VANETs are characterized by their unique characteristics that distinguish them from MANET. These special characteristics can be summarized as follows:

Mobility. Mobility of vehicles represents a critical characteristic that influences the behavior of communications within the network. VANETs have a specific mobility compared to several known MANETs:

High *mobility*. In VANET, the speed of vehicles is very high, and therefore the communication time between nodes will be shorter, which poses significant problems of radio propagation instability.

Mobility models. In VANET, vehicles follow specific directions according to the roads and the environment characteristics like buildings, traffic regulations, etc. This implies that vehicles follow regular and limited mobility patterns (Diab, 2020).

Predictable mobility. The availability of certain information as vehicles speed and the roads path can lead to predicting their next position, in which vehicle displacement models can play the main role in this prediction at least on the short run (Petit, 2011).

Energy. is a major constraint in traditional mobile networks and this affects the computing capacity and the quality of applications. However, in VANET, the communication entities have an efficient power system that provides sufficient energy capacity. Even when the engine is stopped, the embedded platform can use the battery device (Petit, 2011).

Density of vehicles. It ranges from high to low density depending on the geographic area (high traffic density in urban area) and the time factor (i.e. low traffic density during off-peak hours).

Network topology and connectivity. Given the mobility of nodes, the vehicle can join and leave the network in a short time, which changes the topology and causes network partitioning frequently (Petit, 2011).

Frequent exchange of information. As an Ad-hoc network and the presence of different types of nodes. VANETS featured with a high-frequency exchange. Nodes exchange information frequently to maintain the existence of the network (Rejab 2018).

VANET applications

VANETs support a wide range of applications (as represents **Figure 1**), most of the concerns of interest to MANET are of interest in VANETs, but the details differ (Journal of Computing 2010). E.g. vehicles tend to move in an organized fashion and they are restricted in their range of motion.

According to (Weil et al. 2011) VANET applications are categorized as safety, traffic efficiency and infotainment. illustrates a taxonomy of VANET applications.

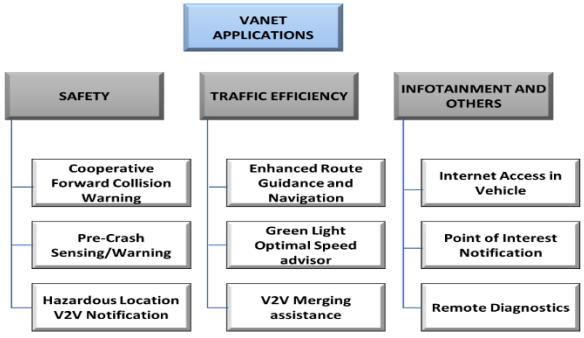

Figure 1. Taxonomy of VANET applications.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Safety Applications. Safety applications aim to warn drivers at the right time about dangerous situations in the road in order to enhance driving safety.

Cooperative Forward Collision Warning. the goal of this use case is to avoid rear-end collisions with other vehicles by aiding drivers (Kosch et al. 2007).

Pre-Crash Sensing/Warning. This use case assumes that a crash is unavoidable and will take place. In such cases, the involved vehicles exchange in an efficient way the related information with neighbouring (Bako & Weber, 2011).

Hazardous Location V2V Notification. the goal of this use case is to share pertinent information about dangerous locations on the roadway (e.g., potholes, bottleneck, etc.) between vehicles in certain area, through V2V communications. These information's can also be transmitted from external service providers to RSUs.

Traffic efficiency applications. Traffic efficiency applications aim to enhance the efficiency of transportation systems by providing traffic related information to drivers or road operators (Kosch et al. 2007).

Traffic information systems. it enables the infrastructure owner to collect traffic data of a region to be used later for predicting traffic congestion on roadways, which use VANET communication to provide up-to-the minute obstacle reports to a vehicle's satellite navigation system which called Automotive navigation system (Yousaf et al. 2013).

Green Light Optimal Speed Advisory. it provides information related to the location of a signalized intersection and the signal timing to vehicles approaching the intersection.

Infotainment and others. Some of these use cases provide entertainment or information on a regular basis to drivers. Other ones are transparent to the driver and they play important role for improving the vehicle functions.

Internet Access in Vehicle. it allows drivers and eventually passengers to access the Internet via the VANET. In this case, RSUs act as internet gateways.

Point of Interest Notification. it allows traders and advertisement companies to advertise their business promotions to nearby vehicles. To this end, an RSU broadcasts the advertisement information (e.g., location, hours and pricing) to the contacted vehicles.

VANET communication architecture

Communication types in VANETs can be categorized into four types (Saeed et, al. 2012)**:**

a- In vehicle communication. It detects the inner system data or performance of the vehicle and determines factors such as driver exhaustion or drowsiness etc., Is crucial for public safety as well as driver safety (Liang et, al 2014). Essentially is between OBU of the vehicle and its AUs (Sofian et, al. 2020).

b- Vehicle to Vehicle (V2V). The data exchange between different vehicles via their OBUs (Sofian et, al. 2020) and does not rely on fixed infrastructure (Ravi et al., 2017).

c- Vehicle-to-road infrastructure (V2I). Is a bidirectional wireless communication between vehicles and infrastructure-connected RSUs to gather data & provides updates related to environmental sensing (Sofian et al. 2020).

d- Vehicle-to-broadband. cloud (V2B) communication. This allows communication of vehicles over broadband connections as 3G/4G which enhances the driver assistance and vehicle (Ravi et al., 2017) (Sofian et, al. 2020).

OVERVIEW ON EXISTING AD-HOC ON-DEMAND DISTANCE VECTOR (AODV) PROTOCOL

AODV is a loop-free routing protocol for Ad-hoc networks. It is designed to be self-starting in an environment of mobile nodes, with standing a variety of network behaviors such as node mobility, link failures and packet losses. Bypassing the Bellman-Ford "counting to infinity" problem offers efficient convergence when the topology shifts, typically when a node relocate in the network (Perkins et al. 2003).

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Structure of AODV protocol

In this section, we present the routing table structure and different messages of AODV protocol.

Routing table:

Nodes store all their information about routes in their routing tables, which consists of a set of entries, with the restriction that each has a different *DestIP*.

Every routing table entry contains the following fields as represented in **Table 1** (Perkins et al. 2003).

Table 1. AODV routing table attributes.

Attribute	Meaning
Destination IP Address	The IP address of the destination node
Destination Sequence Number	Indicate the freshness of routing information
Valid Destination Sequence Number flag	K = {kno, unk}, set to unknown (unk) when a routing-table entry is updated with information that is not equipped with sequence number *seq* itself
Next Hop	The closer node to the unicast destination (neighbor-node)
Other states and routing flags	(e.g. Valid, invalid, repairable, being repaired), an element of the set F = {val, inv}
Network Interface	The point of interconnection between a node and a network, but does not have to need a physical form
Hop Count	Number of hops needed to reach destination *hent*. The variable hops range over the type Integer (IN) and we make use of the standard function +1
List of Precursors	Containing the IP address for each of its neighbors that are likely to use it as a next hop towards each destination. The list is acquired during the *RREP* processing
Lifetime	Expiration or deletion time of the route, is either determined from the control packet, or it is initialized to ACTIVE_ROUTE_TIMEOUT

Figure 2 illustrates the routing table entries of every node subordinate to the network.

```
RoutingTableEntry (Ptr<NetDevice> dev = 0,Ipv4Address dst = Ipv4Address (), bool vSeqNo = false, uint32_t m_seqNo = 0,
                   Ipv4InterfaceAddress iface = Ipv4InterfaceAddress (), uint16_t  hops = 0,
                   Ipv4Address nextHop = Ipv4Address (), Time lifetime = Simulator::Now ());
```

Figure 2. The routing table structure in NS-3 simulator.

AODV messages:

Route request (RREQ), Route replies (RREP), and Route error (RERR) are the message types defined by AODV. These message types are received via UDP, and normal IP header processing applies. So, for instance, the requesting node is expected to use its IP address as the Originator IP *OperIP* address for the messages. For broadcast messages, the IP limited broadcast address is used (255.255.255.255). This means that such messages are not blindly forwarded. However, AODV operation does require certain messages to be disseminated widely, perhaps throughout the Ad-hoc network. The range of dissemination of such RREQs is indicated by the TTL in the IP header. Fragmentation is typically not required. As long as the endpoints of a communication connection have valid routes to each other, AODV does not play any role (Perkins et al. 2003).

The structure of these messages is shown later in this paper with a slight modification at the IP address level.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

AODV protocol operation

This section describes the scenarios to generate Route Request (RREQ), Route Reply (RREP) and Route Error (RERR) messages.

Routes generation:

In this section, we detail the route generation process in the AODV protocol, which relies on two main procedures:

Route discovery:

- Nodes discover routes in request-response cycles. A node requests a route to a destination by creating and broadcasting a *RREQ* message to all its neighbors including *OperIP, OperSeq, DestIP* and *DestSeq*.

- When a node receives an *RREQ* but does not have a route to the destination, it rebroadcasts the *RREQ*. At the same time, it stores a routing table entry about the reverse-*route* to the requesting node including *OperIP*, *OperSeq*, *hent* to source node, IP address of node from which *RREQ* was received which can be used to forward subsequent responses. The rebroadcasting of *RREQ* won't continue indefinitely; it stops either when the source receives a *RREP* or when the *RREQ* lifespan is up.

- In the last case, the source node has to rebroadcast the original *RREQ* but this time with a longer lifespan and a new ID number if it is still needed. This process repeats until the *RREQ* reaches a node that has a valid route to the destination or the destination itself.

- This node responds with a unicast *RREP* message including *DestIP*, *hent* to destination, TTL "Time-To-Live along the *reverse-routes* of the intermediate nodes until it reaches the originator node.

- By default, the source node accepts the first fresh enough *RREP* message received. The freshness of the *RREP* is determined as follows:

- For every *RREP* received, the source node would first check its routing table to see whether it has a previous route to the destination or not. If it finds a route, it compares the *DestSeq*. If the *RREP* has a higher *DestSeq*, then the source node will consider *RREP* route as a fresh route and update its routing table; otherwise, if the destination has no previous entry in the route table, and the *RREP seq* is higher than the *OperSeq* sends in the last *RREQ* message, then the source node will register *RREP* in its route table, else the *RREP* is discarded (Mustafa et al. 2018).

- Thus, at the end of this request-response cycle a *bidirectional* route is established between the requesting node and the destination, data transmission can begin via this route. When a node loses connectivity to its next hop, the node invalidates its route by sending an *RERR* to all nodes that potentially received its *RREP*.

- On receipt of any of the AODV messages the nodes update the next hop *seq* and the *hent* of their routes in the routing table entry (Perkins et al. 2003).

Route Maintenance:

- The route maintenance of AODV is performed by detecting a link break before the complete failure of route. While data is being transmitted across, an active route may contain a broken link, it cannot be forwarded via this link to the next hop.

- In AODV, if a link fades while a route is active (or if it receives a forward request of data packet destined to a node that it does not have an active route), the node that detected the loss of the link immediately drops the data packets and increment its *DestSeq* in the routing table to infinity to invalidate the route and updates the lifetime field to the DELETE_PERIOD.

- The node then broadcasts a unicast *RERR* when it has only one link to this destination which must contain the newly incremented *DestSeq*. Any node that receives this packet, it updates this route entry accordingly and forwards the packet to the next upstream nodes to the source to inform these nodes of the unreachable destination.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Repeatedly, the *RERR* packet reaches the source node. After receiving it, it can reinitiate another route discovery if it still desires the route.

- A node that rebroadcasts *RREQ* control packets for the same destination does so with a new *seq* higher than that of the previously lost route and tries to find a new route for its destination. This dropping of data packets and regeneration of control packets is not only going to cause loss of packets and overhead, but also delays the process of data transmission.

- A more efficient way of repair is required. The suggested improvements by the original developers are *earlier detection of lost link* using local connectivity and/or *local route repair*. For the second suggestion, the repair mechanism to mitigate this problem of delay and packet drop when a link is lost.

- In this route maintenance scheme, when a link breaks in an active route, the upstream node of the break point that has data packet may choose to repair the link locally. It becomes the *RREQ* originating. It increases *DestSeq* and then broadcasts *RREQ* to the lost destination again. The repair attempt is therefore invisible to the other downstream nodes and the data source node, and so they will continue to transmit data packets. Therefore, as the repair node awaits the discovery process to respond with *RREP*, it buffers the incoming data during this period of repair (in case of data fragmentation).

- The local repair always has TTL which is always lesser than the global search TTL. However, AODV local repair attempt is only done when the break point is closer to destination than to the source node else this repair mechanism is not available and that *RERR* message is sent to the source node to rediscover the route.

- A local link repair is likely to increase the number of delivered data packets at the destination when the link is lost since data packets will not be dropped but rather redirected through alternative route to the destination. Delay is also minimised as destination is expected to be closer to the repairing node than the source node. *RREP* is expected to be received faster than if the *RREQ* was from the source node. However, the side effects may include the risk of repairing a route that is no longer needed and sometimes it also results in increased path length to the destination. Besides, cost of fail could be so dear; time wasted (delay), the entire route timing out and large amount of buffered data are dropped (ISAAC 2018).

Exceptional operating cases:

Nodes can keep track of connectivity to neighbors using available data link or network layer mechanisms. An error message processing can be broadcasted due to three, firstly when a node receives a data packet that's supposed to be forwarded but a path to the destination node isn't found.

Secondly when a node receives a *RERR* that causes one of its stored routes to be invalidated, if this happens the node will broadcast a *RERR* with all the new nodes which are now unavailable.

Finally, a node can detect its inability to communicate with one of its neighbors (*HELLO* message from a neighbor isn't received within expected interval) and thus mark all of its stored routes using dead node as invalid and broadcast *RERR* message to other neighbors that potentially received its *RREP* to perform the same operation (Mustafa et al. 2018).

Existing IPv4 AODV Implementations

Here, we cite some AODV implementations in IPv4 version. In NS-3, the 'ns3::aodv::RoutingProtocol' class inherits from 'Ipv4RoutingProtocol', making it IPv4-only (ns-3 Project 2025).

In OMNeT++ (INET), the 'inet::AodvRouter' compound module includes the AODV routing component and connects it to the 'ipv4.ip' (the 'inet::Ipv4' module) and 'ipv4.routingTable' (the 'inet::Ipv4RoutingTable' module), making it IPv4-based (INET framework 2025).

In NS-2, the AODV class (implemented in 'aodv.h' and 'aodv.cc') inherits from the Agent class and interacts with the simulator's native IP layer IPv4-only by design (Rehmani et al. 2009).

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

IMPLEMENTING IPV6 AODV IN VANET

The main idea of implementing the AODV IPv6 is based on expanding the header of all AODV packets *RREQ, RREP, RERR* and *Hello* messages, IP header in the internet layer, the ARP header in between data link layer and network layer, and the routing table as well.

From a simulation point of view, we put some changes and modifications to NS-3 module to be suitable to the AODV IPv6 routing protocol in VANET.

IPv4-Mapped IPv6 Address

Hybrid dual-stack IPv6/IPv4 implementations recognize a special class of addresses. The IPv4-mapped IPv6 addresses (Cotton M et al. 2013). These addresses are typically written with a 96-bit prefix in the standard IPv6 format, and the remaining 32 bits are written in the customary dot-decimal notation of IPv4 (Hinden et al. 2006).

Dual-stack sockets always require IPv6 addresses. The ability to interact with an IPv4 address requires the use of the IPv4-mapped IPv6 address format. Any IPv4 addresses must be represented in the IPv4-mapped IPv6 address format which enables an IPv6 only application to communicate with an IPv4 node. The IPv4-mapped IPv6 address format allows the IPv4 address of an IPv4 node to be represented as an IPv6 address. The IPv4 address is encoded into the low-order 32 bits of the IPv6 address, and the high-order 96 bits hold the fixed prefix 0:0:0:0:0:FFFF. The format of the "IPv4-mapped IPv6 address" is as follows in **Figure 3**.

Figure 3. IPv4-mapped IPv6 address format.

IPv6 AODV packets

In this paper, three AODV packets are modified; Route request (*RREQ*), Route replies (*RREP*), and Route error (*RERR*):

Modified RREQ message:

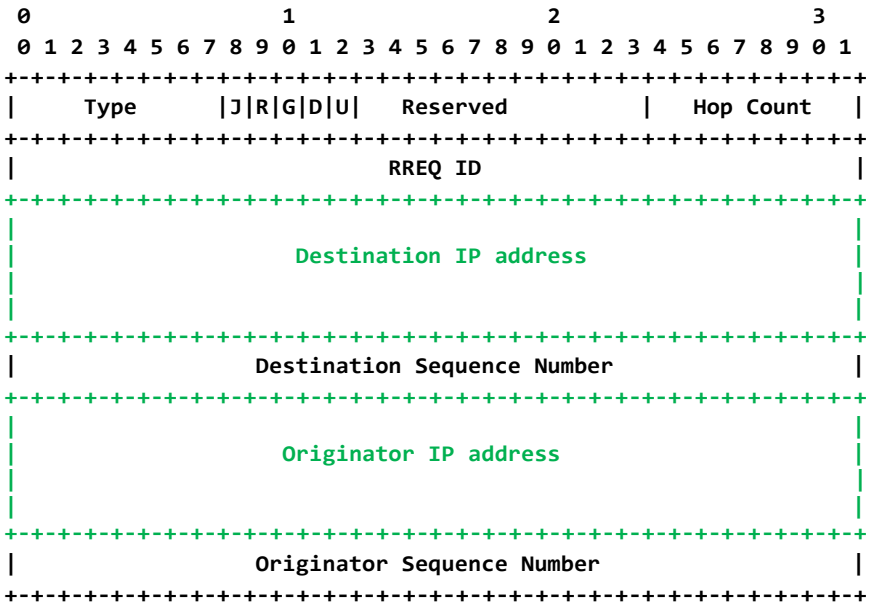

Figure 4. Modified Route Request (RREQ) format.

The format of the Route Request message is illustrated in **Figure 4**. The several fields included in RREQ are described in the following table (**Table 2**):

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Table 2. RREQ massage attributes.

Attribute	Meaning
Type	Type of the AODV message (here is 1)
J	Join flag; reserved for multicast
R	Repair flag; reserved for multicast
G	Gratuitous *RREP* flag; indicates whether a gratuitous *RREP* should be unicast to the node specified in the Destination IP Address field
D	Destination only flag; indicates only the destination may respond to this *RREQ*
U	Unknown sequence number; indicates the destination sequence number is unknown
Reserved	Sent as 0; ignored on reception
Hop Count	The hops number from the originator to the node handling the request (*hent*)
RREQ ID	A sequence number uniquely identifying the particular *RREQ* when taken in conjunction with the originating node's IP address
Destination IP Address	IP address (version 6) of the destination for which a route is desired (*DestIP*)
Destination Sequence Number	The latest sequence number received in the past by the originator for any route towards the destination (*DestSeq*)
Originator IP Address	IP address (version 6) of the node that originated the Route Request (*OperIp*)
Originator Sequence Number	The current sequence number to be used in the route entry pointing towards the originator of the route request (*OperSeq*)

Modified RREP message:

Figure 5 illustrates the format of the Route Reply message.

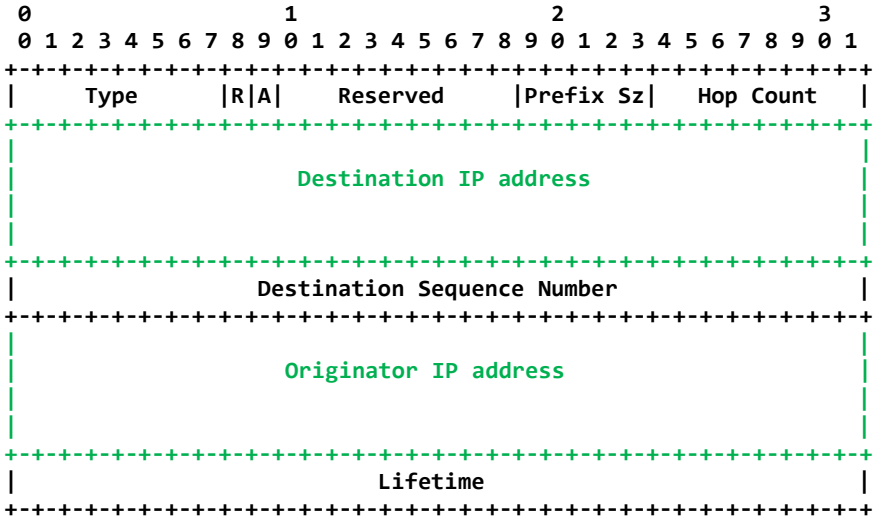

Figure 5. Modified Route Reply (RREP) format.

RREP message contains the following fields as represented in **Table 3**.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Table 3. RREP massage attributes.

Attribute	Meaning
Type	Type of AODV message (2)
R	Repair flag; used for multicast
A	Acknowledgment required
Reserved	Sent as 0; ignored on reception
Prefix Size	If nonzero, the 5-bit Prefix Size specifies that the indicated next hop may be used for any nodes with the same routing prefix as the requested destination
Hop Count	The number of hops from the originator to the destination. For multicast *RREQ* this indicates the number of hops to the multicast tree member sending the *RREP*
Destination IP Address	The IP address (version 6) of the destination for which a route is supplied
Destination Sequence Number	Destination sequence number associated to the route
Originator IP Address	The IP address (version 6) of the node which originated the *RREQ* for which the route is supplied
Lifetime	TTL (Time-To-Live): in milliseconds for which nodes receiving the RREP consider the route to be valid

Modified RERR message:

The format of the Route Error message is illustrated in **Figure 6**.

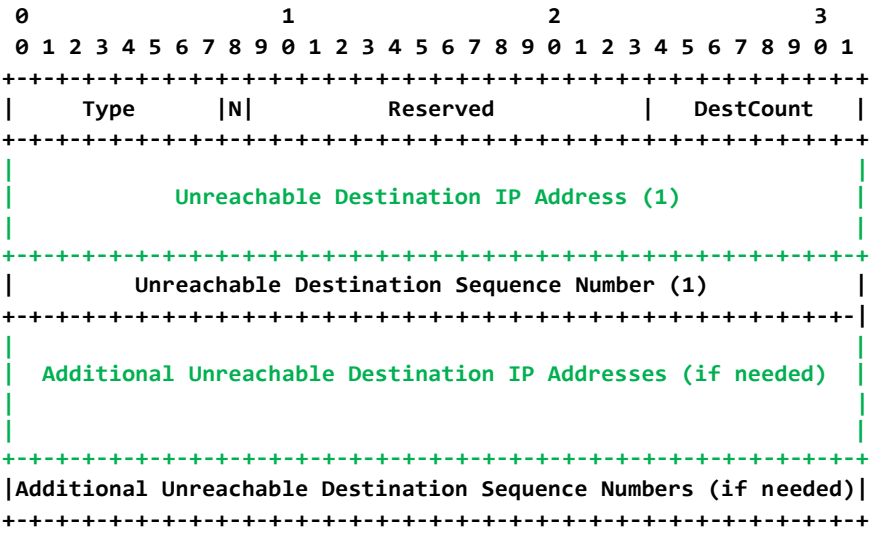

Figure 6. Modified Route Error (RERR) format.

Table 4 shows the RERR packet fields as below:

Table 4. RERR massage attributes.

Attribute	Meaning
Type	Type of AODV message (3)

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

N	No delete flag; set when a node has performed a local repair of a link, and upstream nodes should not delete the route
Reserved	Sent as 0; ignored on reception
DestCount	The number of unreachable destinations included in the message (≥1)
Unreachable Destination IP Address	The IP address of the destination that has become unreachable due to a link break *UDestIP*
Unreachable Destination Sequence Number	The sequence number in the route table entry for the destination listed previously *UDestIP*

IP header & modified IP header

Figure 7 describes the modified IP Header format for the IPv4 mapped IPv6.

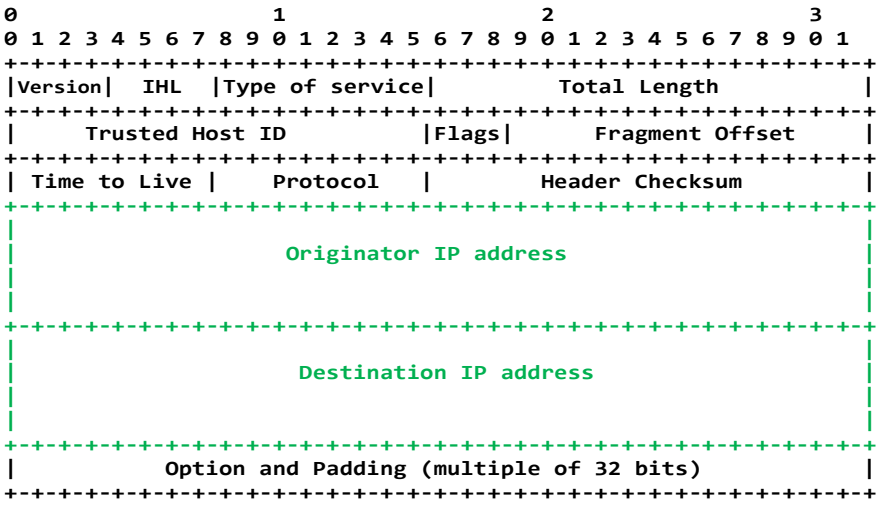

Figure 7. Modified IP Header format.

Address Resolution Protocol (ARP)

ARP is a protocol or procedure that connects an ever-changing Internet Protocol (IP) address to a fixed physical machine address, also known as a media access control (MAC) address, in a local-area network (LAN).

This mapping procedure is important because the lengths of the IP and MAC addresses differ, and a translation is needed so that the systems can recognize one another. The most used IP today is IP version 4 (IPv4). An IPv4 address is 32 bits long. However, MAC ad-dresses are 48 bits long. ARP translates the 32-bit address to 48 and vice versa.

The MAC address is also known as the data link layer, which establishes and terminates a connection between two physically connected devices so that data transfer can take place. The IP address is also referred to the network layer which is responsible for forwarding pack-ets of data through different routers. ARP works between these two layers (data link and network layer) (David, 1982). **Figure 8** presents the modified ARP header format.

Routing table entry

When a node receives an AODV packet from a neighbor, creates or updates a route for a particular destination or subnet, it checks its routing table for an entry to the destination. When no corresponding entry for that destination, an entry is created. The sequence number is either determined from the information contained in the control packet, else the valid sequence number field is set to false.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

In our AODV IPv6 implementation we subrogated the destination IPv4 address entry to the mapped IPv6 version in the routing table at the nodes level, i.e. the network won't be influenced by the IP replacement (network overhead won't be increased).

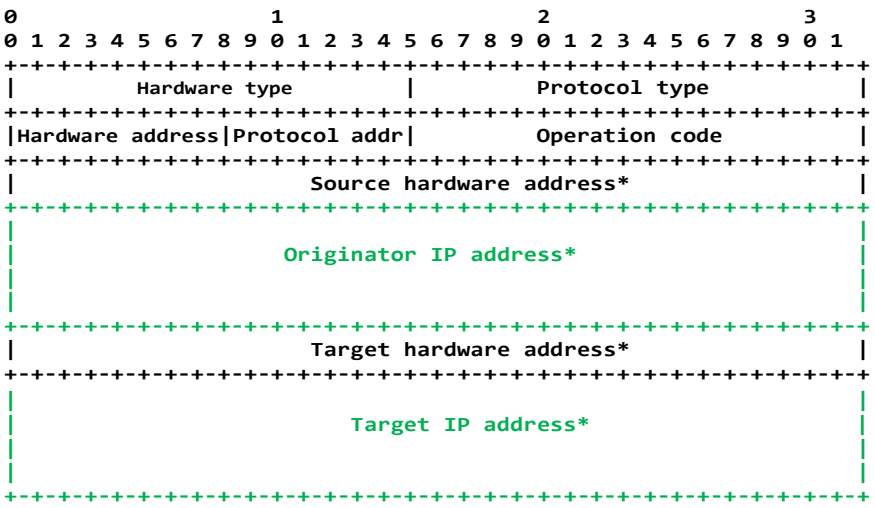

Figure 8. Modified ARP header format.

IPV6 AODV IMPLEMENTATION EVALUATION ON VANET NETWORK

In order to evaluate performance, IPv6 implementation in the AODV protocol within a vehicular ad hoc network was tested under eight different scenarios, with node density varied as follows: 6, 8, 10, 12, 14, 16, 18, and 20 nodes.

Simulation settings

Table 5 summarizes the simulation details. Two parameters; the number of nodes and the mobility models were varied and minted manually. The following points were considered during the simulation:

- Each vehicle generally moves according to a road network pattern and not at random.
- The movement patterns of vehicles are occasional (stop, move, park, etc.).
- Vehicles must respect speed limitations and traffic signals.
- Vehicle behaviour depends on the behaviour of its neighbour vehicles as well as on the road type.
- Two nodes cannot exist in the same location at the same time.
- Nodes usually travel at an average speed lower than 120 Km/h (Algeria).

Table 5. Simulation settings.

Parameter	Value
CTS/RTS	2200 *bytes*
Data Packet size	128 *bytes*
Density (nodes number)	6, 8, 10, 12, 14, 16, 18, 20
Deployment environment	Urban (City)
Node type	Mobile (vehicles), Fixe (RSU)
Packet rate	1 *packet / sec*
Seed Manager	03, 11, 15
Simulation time	60 *seconds*
Topology	444 *m* x 444 *m*
Vehicles speed	10 *m/s*

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The simulation was carried out for three times for each scenario for each IP version. The three times refers to the way of generation of random values in NS-3. These values are based on the "seed[1]" value, which causes differences in the result values even for the same scenario. For that, we launch the simulation many times for each scenario to have a set with different possible values during the simulation with different seed values and average of observed values were taken to reduce estimation errors.

The simulation results were obtained for the existing work AODV IPv4 and then with the designed solution IPv6 mapped version. We have evaluated the performance and the impact of the node density on the two protocols versions in terms of Packet Delivery Ratio (PDR), Ent-to-End Delay (E2E), Overhead and Average packet size. The results are shown in line charts below.

Performance evaluation

Packet Delivery Ratio (PDR) results:

PDR is the ratio of a total number of delivered data packets to the total number of data packets transmitted by all sources as shown in Formula (1) (Diab et al. 2022). This evaluation metric will give us a concept of how well the designed solution is performing in terms of packet delivery at different network density.

$$Packet\ Delivery\ Ratio = \frac{Total\ No.of\ Packet\ Received}{Total\ No.of\ Packet\ Sent} \quad \ldots\ldots\ldots (1)$$

Figure 9 illustrates the effect of node density on packet delivery ratio. The PDR percentage for both addressing decreases by increasing the nodes number (traffic).

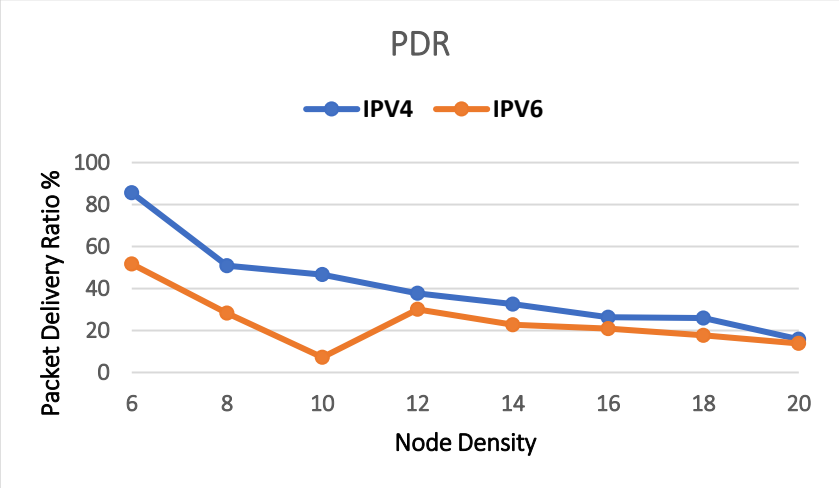

Figure 9. IPv6 Vs Ipv4 Packet Delivery Ratio (PDR) results in VANET network.

It is evident that IPv4 addressing outperform IPv6 addressing from PDR vision.

Overhead results:

This metric compares the total number of control packets sent during the simulation time compared to the total data packets delivered. It is calculated as the total number of routing control packets sent by all nodes divided by the total number of data packets successfully received by destinations -Formula (2)- (Diab et al. 2019b). Control packets consist of *RREQ*, *RREP*, *RERR* and *Hello* message in AODV protocol and *ARP* protocol in physic layer.

$$Overhead = \frac{\sum Control\ packets\ sent}{\sum\ Data\ packets\ delivered} \quad \ldots\ldots\ldots (2)$$

[1]Concept of Seed in Random Number Generator in C++: the seed acts as a starting point for the algorithm. Rather the algorithm picks numbers randomly from a distribution that the seed defines (e.g. Device time). If you provide the same seed to the algorithm, it will generate the same sequence of pseudo-random numbers.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

In **Figure 10**, overhead is plotted against the node density for both the original IPv4 and the proposed implementation IPv6.

Overhead is observed to increase with node density for both the original and IPv6 implementation in AODV mechanism. This is expected because more nodes mean unnecessarily repeat of broadcast of the control packets by each node, hence more control packets are committed during the route discovery phase. We also observed that the IPv6 it has a relatively higher overhead compared to IPv4. We debrief that the IPv6 need a bit more control packets to achieve data packet delivery to the destination.

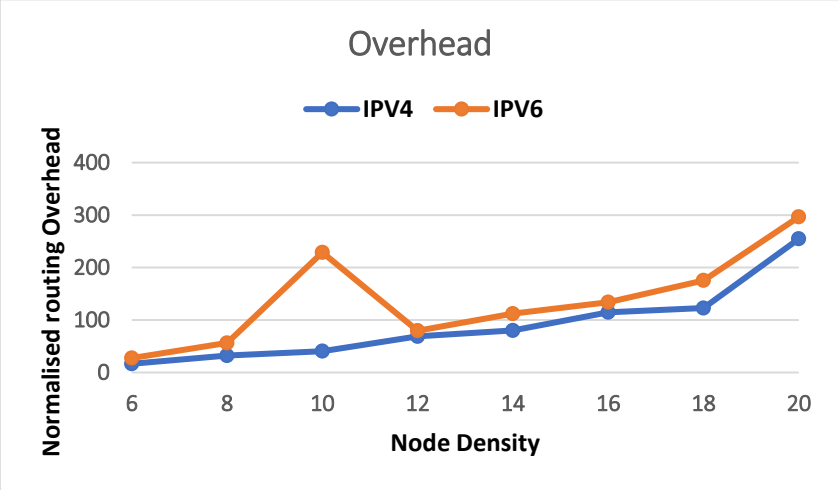

Figure 10. IPv6 Vs IPv4 Overhead results in VANET network.

End-to-End Delay (E2E) results:

This parameter is the average transmission time of data packets from source to destination as given in Formula (3). This traversal time is known as Average End to End Delay (E2E Delay). It is defined as:

$$E2E\ Delay = \frac{\sum^{n} [\ Recieved_{Time(t2)} - Sending_{Time(t1)}]}{Data\ Packets\ Received} \quad \ldots\ldots\ldots (3)$$

Figure 11 shows the average end-to-end delay of transmitted data packets from source to destination. The average E2E Delay increased when nodes velocity increases for both addresses but a comparison of the two showed that E2E is heterogeneous and unpredictable. So, we understand that the E2E Delay is not an efficient QoS measurement to esteem these scenarios rendering.

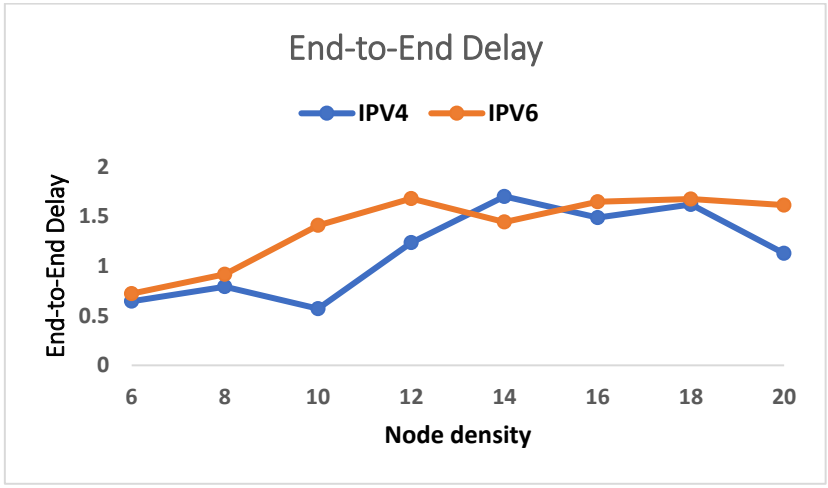

Figure 11. IPv6 Vs IPv4 End-to-End Delay (E2E) results in VANET network.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Average packet size results:

This metric calculates the total packet size transmitted during the simulation time compared to the total number of these packets. As expressed by Formula (4), the average packet size calculates the average size of sent packets (data and control packets). We use this metric to represent the quantity of packets (bytes) circulating in the network.

$$Average\ packet\ size = \frac{\sum Packet\_Size}{\sum Packet\ Sent} \quad \ldots\ldots\ldots (4)$$

Figure 12 is a plot of the quantity measurement against the network density. The graph shows a marginal increase in the average quantity with the node density for both addresses. Is observed that the IPv6 addressing have a much higher average quantity in density variations. Which mean that IPv6 protocol have more quantity packets circling in the network. These results totally make sense due to the packets expansion (e.g. IP header packet size became 44 bytes (from 20 bytes)).

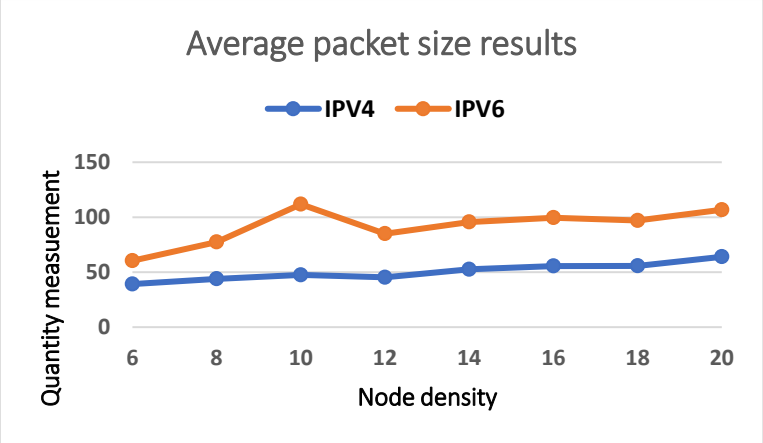

Figure 12. IPv6 Vs IPv4 Average packet size results in VANET network.

In the case of ten (10) nodes we remark that the overhead value for IPv6 mapped it increased strictly than the overhead average excess and an outgrowth in quantity packets circling in the vehicular Ad-hoc network. Also, we have noticed a dropout in the packet delivery ration to less than 7,3% while the average percentage is around 24.05%.

This irregularity, is probably consequent to a collision or link breakage in the network, which force the AODV to start the local maintenance process or to restate the route discovery phase from the beginning.

CONCLUSION

In this paper, we presented the implementation of the AODV routing protocol with IPv6 support using the NS-3 simulator and NetAnim as a visualization tool. The obtained results were analyzed according to several QoS metrics such as Packet Delivery Ratio (PDR), End-to-End Delay (E2E), routing overhead, and average packet size.

The simulation results showed that AODV with IPv4 performs better than AODV with IPv6 in most QoS parameters. This difference mainly arises from the larger header size in IPv6 packets compared to IPv4. However, the transition to IPv6 addressing is inevitable due to the limitations of IPv4, including address exhaustion, security weaknesses, and geographical scalability issues especially with the increasing deployment of VANET and IoT technologies.

As a future direction, we plan to generalize the IPv6 implementation to other VANET routing protocols, such as DSR, OLSR, and GPSR, and to explore its adaptation in different vehicular communication scenarios. This will help develop more scalable and efficient routing mechanisms for next-generation vehicular networks.

REFRENCES

[1] *Ad Hoc On-Demand Distance Vector (AODV)"* — Model Library, ns-3. Available at: https://www.nsnam.org/docs/models/html/aodv.html. Accessed on October 21, 2025.

[2] *AodvRouter (*compound module).” INET Framework API Documentation, version 4.4.0. Available at: https://doc.omnetpp.org/inet/api-4.4.0/neddoc/inet.node.aodv.AodvRouter.html. Accessed on October 21, 2025.
[3] Bako, B., & Weber, M. (2011). Efficient Information Dissemination in VANETs. in Advances in Vehicular Networking Technologies, Miguel Almeida (ed). IntechOpen, London, p. 20.
[4] BITAM S and A. MELLOUK. Editors. ISTE Ltd and John Wiley Sons: Bio-Inspired Routing Protocols for Vehicular Ad Hoc Networks, Inc, 27-37 St George’s Road, London SW19 4EU, UK. 111 River Street, Hoboken, NJ 07030, USA, first edition (2014).
[5] Cotton Michelle; Leo Vegoda; Brian Haberman (April 2013), Ronald. Bonica (ed.). Special-Purpose IP Address Registries. Internet Engineering Task Force (IETF), Juniper Net-works. sec. 2.2.3. doi 10.17487/RFC6890. BCP 153. RFC 6890. Table 20.
[6] David C. Plummer. (Nov 1982). An Ethernet Address Resolution Protocol. *Network Working Group*.
[7] Diab, T., Gilg, M., Lorenz, P., & Drouhin, F. (2022). Using I2P (Invisible Internet Protocol) Encrypted Virtual Tunnels for a Secure and Anonymous Communication in VANets: I2P Vehicular Protocol (IVP). *Wireless Personal Communications*, *127*(3), 2625-2644.
[8] Diab Tayeb: Security management in vehicular Ad-hoc networks. University of Haute-Alsace, Laboratory (IRIMAS) Institut de Recherche en Informatique, Mathématiques, Automatique et Signal (January 2020).
[9] Diab, T., Gilg, M., & Lorenz, P. (2019a). A secure communication model using lightweight Diffie-Hellman method in vehicular ad hoc networks. International Journal of Security and Networks, 14(2).
[10] Diab, T., Gilg, M., Drouhin, F., & Lorenz, P. (2019b, December). Anonymizing communication in VANets by applying I2P mechanisms. In 2019 IEEE Global Communications Conference (GLOBECOM) (pp. 1-6). IEEE.
[11] Hinden Robert M; S. Stephen E Deering (February 2006). IP Version 6 Addressing Architecture. *Network Working Group*. doi:10.17487/RFC4291. RFC 4291.
[12] Isaac Otchere Nyameamah. (June 2018). Dynamic route maintenance scheme for aodv routing protocol using joint nodes. The university of Ghana, college of basic & applied science, legon in partial fulfilment of the requirement for the award of MPhil computer science.
[13] Journal of Computing: "A Comparative study of MANET and VANET Environment". Vol 2, ISSN 2151-9617(July 2010).
[14] Kosch,R. Baldessari, B. Bodekker, M. Deegener, A. Festag, W. Franz, C. C.¨ Kellum, A. Kovacs, M. Lenardi, C. Menig et al. (2007), “Car-2-car communication consortium manifesto,”.
[15] Mohammadi Morteza Zanjireh, Hadi Larijani: A Survey on Centralised and Distributed Clustering Routing Algorithms for WSNs. IEEE 81st Vehicular Technology Conference. Glasgow, Scotland (May 2015).
[16] Mustafa Hala, Noureldien A. Noureldien (2018). Detection of Route Discovery Misbehaving Nodes in AODV MANETs: A Survey, International Journal of Networks and Communi-cations, p-ISSN: 2168-4936, e-ISSN: 2168-4944; 8(4): 115-122.
[17] Perkins C, E. Belding-Royer, S Das: Ad hoc On-Demand Distance Vector (AODV) Routing (RFC 3561). In: Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications. pp. 99-100, Nokia Research Center (JULY 2003).
[18] Petit, J : Surcoût de l’authentification et du consensus dans la sécurité des réseaux sans fil véhiculaires. PhD Thesis Université Paul Sabatier (2011).
[19] Ravi Tomar, Manish Prateek, & G. H. Sastry. (2017). Vehicular Ad-hoc Network (VANET) - An Introduction. International Journal of Control Theory and Applications, International Science Press, 9(18), 8883–8888, ffhal-01496806.
[20] Rehmani, M. H., Doria, S. & Senouci, M. R. (2009). *A Tutorial on the Implementation of Ad-hoc On Demand Distance Vector (AODV) Protocol in Network Simulator (NS-2)*. Version 1, June 2009.
[21] Rejab HAJLAOUI : Résolution à base d’heuristiques du problème de routage dans les réseaux ad hoc de véhicules. PhD dissertation, University of Franche-Comté (2018).
[22] Saeed A, M. Faezipour, M. Nourani, and S. Addepalli, (2012) “Progress and challenges in intelli-gent vehicle area networks”, Communications of the ACM, vol. 55, no. 2, pp. 90–100.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

[23] Sofian Hamad and Taoufik Yeferny (February 2020). Vehicular Ad-hoc Networks: Architec-ture, Applications and Challenges, International Journal of Computer Science and Net-work Security (IJCSNS), VOL.20 No.2, College of Science, Northern Border University, Arar, Saudi Arabia

[24] Weil T, G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, and B. Jarupan, K. Lin (2011). “Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions”, IEEE Communications Surveys Tutorials, vol. 13, no. 4, pp. 584–616.

[25] Yousaf Saeed, Suleman Aziz Lodhi, Khalil Ahmed (June 2013). "Obstacle Management in VANET using Game Theory and Fuzzy Logic Control". ACEEE International Journal on Compu-ting. Volume 4.