JTSE
JOURNAL OF TRANSACTIONS IN SYSTEMS ENGINEERING

Journal of Transactions in Systems Engineering
https://journals.tultech.eu/index.php/jtse.
ISSN: 2806-2973
Volume 4, Issue 2
DOI: https://doi.org/10.15157/JTSE.2026.4.2.651-666
Received: 18.05.2026; Revised: 16.06.2026; Accepted: 16.07.2026

TULTECH

Research Article

Integrating MPLS and SDN for Improved Routing in VANETs: A Comparative Study of AODV and OLSR Protocols

Ronild Hako*, Evjola Spaho

Faculty of Information Technology, Polytechnic University of Tirana, Tirana, Albania
***ronild.hako@fti.edu.al**

Abstract

Vehicular Ad Hoc Networks (VANETs) enable wireless communication among vehicles and roadside infrastructure, supporting safety, traffic optimization and infotainment in Intelligent Transportation Systems (ITS). However, the high mobility of vehicles causes frequent topology changes and link breaks, degrading routing performance. This paper investigates the integration of Multi-Protocol Label Switching (MPLS) and Software-Defined Networking (SDN) with two widely used VANET routing protocols, Ad hoc On-Demand Distance Vector (AODV) and Optimized Link State Routing (OLSR). We evaluate four progressive configurations for each protocol: the default protocol, the protocol with MPLS-enhanced forwarding, the protocol with SDN-based centralized route optimization, and the combined SDN-MPLS integration. The simulation environment uses a realistic urban topology with 50 vehicles and 5 Road-Side Units (RSUs). Results are analysed while using several traffic patterns and measuring Key Performance Indicators, such as: Packet Delivery Ratio (PDR), average end-to-end delay, jitter, throughput, and routing overhead. The simulations show that each integration layer contributes measurable improvements to the protocol performance, with the SDN-MPLS combination achieving the best overall results.

Keywords: VANET; AODV; OLSR; MPLS; SDN; Routing; Key Performance Indicators

INTRODUCTION

Vehicular Ad Hoc Networks (VANETs) are a specialized category of modern Intelligent Transportation Systems (ITS), which make possible real-time communication, exchange of safety alerts, traffic conditions, and infotainment data between vehicles and roadside infrastructure [1, 2]. As vehicular networks evolve toward full automation and cooperative driving, the volume and variety of network traffic continue to grow, placing increasing demands on routing protocols to maintain reliable communication under challenging conditions.

The main challenge in VANET routing is derived from the vehicular mobility. Vehicles move at varying speeds within an urban road infrastructure, causing rapid changes in network topology. A wireless link established between two vehicles may last for only a few seconds before one or both vehicles move out of range [3, 4]. Traditional Mobile Ad

Hoc Network (MANET) routing protocols, such as the Ad hoc On-Demand Distance Vector (AODV) and Optimized Link State Routing (OLSR), were designed to accommodate dynamic topologies but face performance degradation in vehicular environments with higher rate of topology changes [5].

By using complementary routing and forwarding technologies VANET's performance can be improved. Multi-Protocol Label Switching (MPLS) replaces hop-by-hop IP lookups with fast label-based forwarding using Label Switched Paths (LSPs) that are previously established, reducing processing delay and allowing for traffic engineering capabilities [6, 7]. Software-Defined Networking (SDN) is a model that separates the control plane from the data plane, by centralizing routing decisions in a controller. This controller maintains a global view of network topology and can compute optimized paths based on real-time conditions [8-10].

Prior studies on MPLS in VANETs [6][11] demonstrate the forwarding improvements, but evaluate MPLS in isolation, without centralized control. Moreover, SDN-VANET studies [4, 8, 9, 10] establish the architectural benefits of centralized routing, but do not measure its contribution relative to forwarding-layer optimizations. What remains unaddressed is a controlled evaluation that isolates the individual and combined contributions of MPLS and SDN under identical topology, mobility, and traffic conditions, for both reactive (AODV) and proactive (OLSR) routing protocols. This paper fills that gap by evaluating four progressive configurations: (1) the default protocol operating independently, (2) the protocol enhanced with MPLS label-switched forwarding, (3) the protocol with SDN-based centralized route optimization, and (4) the combined SDN-MPLS integration.

The assessment uses a realistic simulation environment modelled on the road network of Tirana, Albania, with 50 vehicles and 5 RSUs, across three different traffic patterns: Constant Bit Rate (CBR), Variable Bit Rate (VBR), and Burst. The main contributions of this paper are: performance evaluation of MPLS and SDN integration with routing protocols, under identical simulation conditions; analysis of the trade-off between delay reduction and routing overhead for each integration layer; and a comparison of how protocol architecture (reactive vs proactive) is affected by centralized optimization.

The remainder of this paper is organized as follows. Section 2 provides background and related work on VANET routing protocols, MPLS traffic engineering, and the SDN framework. Section 3 describes the proposed methodology, including the study area, and the integration approaches for MPLS and SDN. Section 4 presents the simulation setup and results. Section 5 concludes the paper with a summary of findings.

BACKGROUND AND RELATED WORK

VANET Routing Protocols

Routing in vehicular environments is complex by design, due to the high node mobility, frequent topology variations, and sporadic connectivity disruptions [3, 5]. Standard

MANET protocols face difficulties maintaining performance in vehicular domains, where control-message flooding, route-discovery delays, and outdated routing information lead to packet loss and increased end-to-end latency [1, 3]. VANET routing protocols are commonly classified as topology-based, position-based, cluster-based, and hybrid [5, 8].

AODV and OLSR, which are topology-based protocols, represent the most basic and deployed VANET routing models. AODV is a reactive distance-vector protocol that establishes routes on demand using Route Request (RREQ) and Route Reply (RREP) message exchanges. When a source node requires a route to a destination, it broadcasts an RREQ that flows through the network until it reaches the destination or an intermediate node with a valid route to it. The destination responds with an RREP that follows the reverse path, establishing forwarding entries at each intermediate node. AODV maintains routes only while they are actively used, reducing overhead in networks with low traffic but introducing route discovery delay when new paths are needed [5].

OLSR is a proactive link-state protocol that maintains complete routing tables through periodic exchange of Hello messages and Topology Control (TC) messages. OLSR introduces the concept of Multi-Point Relays (MPRs), a subset of one-hop neighbours selected in such a way to efficiently forward control messages, reducing the overhead of network-wide broadcasts. While OLSR provides low-latency route availability with pre-computed paths, it generates higher control-message overhead in large or highly dynamic networks [3, 5].

Both protocols must satisfy Quality of Service (QoS) requirements in vehicular environments, including application-specific latency, Packet Delivery Ratio or jitter requirements [2, 13, 14]. Traditional VANET routing lacks global optimization, redundant broadcasts waste bandwidth, and the absence of network-wide policies complicate node operations, motivating the integration of advanced traffic engineering and centralized control mechanisms [1, 4, 8].

MPLS Traffic Engineering

Multi-Protocol Label Switching is a traffic engineering technology widely deployed in traditional backbone networks. It establishes pre-computed Label Switched Paths (LSPs) used to forward packets based on fixed labels rather than IP address lookups [6, 11]. At the network ingress, a Label Edge Router (LER) classifies incoming packets and assigns a label corresponding to an established LSP. Intermediate Label Switch Routers (LSRs) then forward packets by comparing the incoming labels and swap the labels to the newly computed values, requiring very little delay per hop as the labels are processed. MPLS network architecture and design makes it suitable for delay-sensitive vehicular applications, including safety awareness messages and video streaming [7, 11].

Integrating MPLS with VANETs requires adaptation with mobility and frequent topology changes, as LSPs must be updated dynamically to respond to the current network state. When vehicles move out of communication range of a neighbouring node, the label tables need to be updated accordingly for the new network topology. Other simulation studies using NS-3 and SUMO revealed that MPLS-optimized VANETs have higher packet delivery ratios and lower average delays than traditional ad hoc routing protocols like AODV and OLSR. The label-switching mechanism removes redundant route discovery, while centralized label management ensures load-balanced across all links to be maximized.

Several research works focus on hybrid solutions of MPLS for backbone forwarding with traditional ad hoc routing for edge communication though frequent label updates can introduce additional control-channel overhead [6, 11].

SDN Framework

Software-Defined Networking decouples the control and data planes and introduces a logically centralized controller that has complete topological and traffic information about the network [8-10]. The SDN controller can be deployed on a dedicated server, an RSU, or a cloud node. It provides a programmable environment where routing algorithms and network policies can be deployed as software modules with minimal to no hardware intervention [7]. Through southbound interfaces such as OpenFlow, the controller installs or modifies forwarding rules in data-plane devices, while northbound interfaces communicate with applications for policy configuration and network monitoring [10, 16].

In vehicular environments, SDN can enable proactive congestion control, dynamic spectrum resource allocation, and flow prioritization for delay-sensitive packets [8, 10]. The centralized control model allows the SDN controller that has a global view of the network, the ability to compute optimal paths using Dijkstra or other shortest-path algorithms on the entire network topology, considering quality metrics that are unavailable to nodes operating with only local information. SDN-based VANETs can dynamically reconfigure routes, enforce QoS policies, and respond to real-time congestion events, addressing the fundamental limitations of distributed VANET routing [1, 4, 9, 12].

The SD-VANET architecture typically follows either a two-tiered or three-tiered hierarchical design: the vehicle and edge layer containing OBUs with wireless interfaces, the fog layer handling regional traffic management and low-latency decisions, and the cloud layer responsible for global network analytics and optimization [10, 15-17]. In the two-tiered designs, the fog and cloud layers are joint in the centralized network controller. Both approaches balance the need for global optimization with the latency constraints of vehicular applications, distributing the control across processing tiers. While these studies demonstrate the individual benefits of MPLS and SDN in vehicular networks, no prior work has systematically evaluated their progressive and combined integration with both reactive and proactive routing protocols under identical simulation conditions.

PROPOSED METHODOLOGY

Problem Description and Study Area

This study addresses the challenge of improving routing performance in urban VANETs through the integration of MPLS and SDN technologies with traditional routing protocols. The research examines if the integration of MPLS forwarding optimization, SDN centralized control, and their combination, provides clear and consistent performance improvements, while using different routing protocols and traffic patterns.

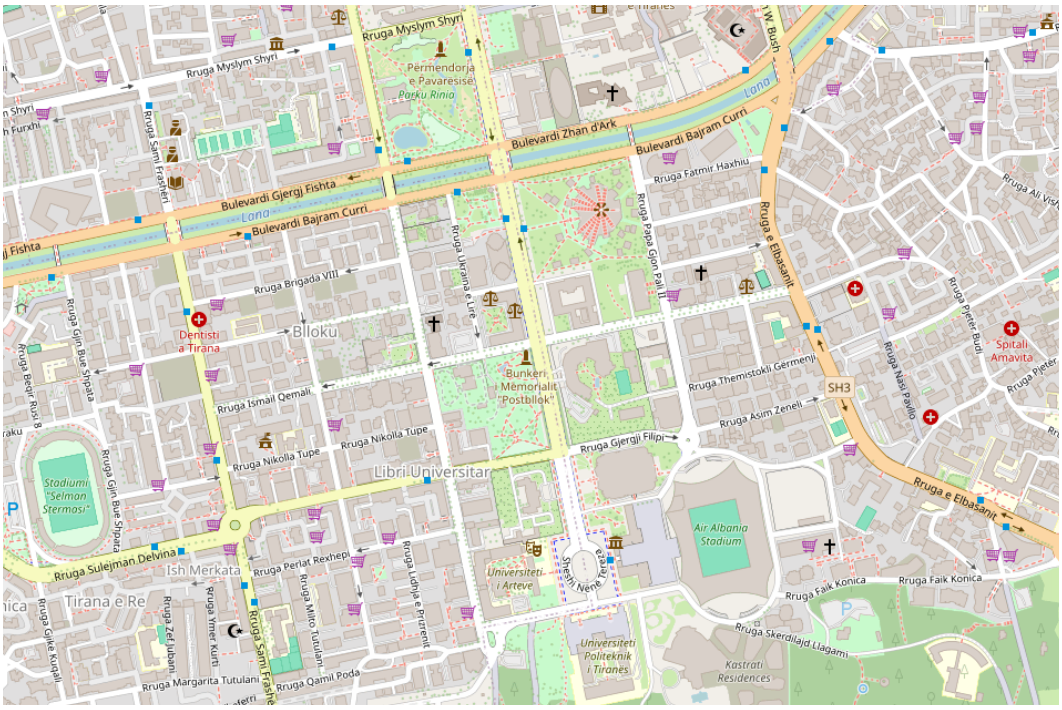

Figure 1. Considered Study Area, a Section of the Road Network In Tirana, Albania

The simulation is applied into a section of the road network in Tirana, Albania, covering an area of approximately 2.5 by 2 kilometres, as shown in Figure 1. The road topology is extracted from OpenStreetMap and processed through SUMO (Simulation of Urban Mobility) to generate realistic vehicular traffic patterns, including traffic signals, intersection and lane-changing behaviours. The simulation deploys 50 vehicles that enter and exit the network according to a fixed departure schedule, used across all simulations and scenarios in order to maintain the same vehicle flow.

Also positioned at strategic locations, in the city's bus stations, 5 Road-Side Units (RSUs) provide infrastructure coverage across the simulation area. RSUs are deployed in a ring topology. RSUs are connected to neighbouring RSUs using 1 Gbps fiber Ethernet links, creating a backbone that enables RSU-to-RSU communication independently of the wireless channel. RSU locations are chosen to maximize coverage, however wireless channel distance is limited by vehicles, since vehicles use IEEE 802.11p at 5.9 GHz, with 10 MHz bandwidth and 20 mW transmit power so vehicles have a range of about 300 m.

Three different traffic patterns are chosen for different network loads: Constant Bit Rate (CBR) where packets are sent every 0.5 s, Variable Bit Rate (VBR) where packets are sent at random intervals uniformly distributed between 0.1 and 2.0 s, and Burst that sends groups of 10 packets in rapid succession every 5 seconds. The simulation runs for 100 seconds with multiple seed values to ensure statistical reliability of the results. In Figure 2 is presented the tested scenario flowchart

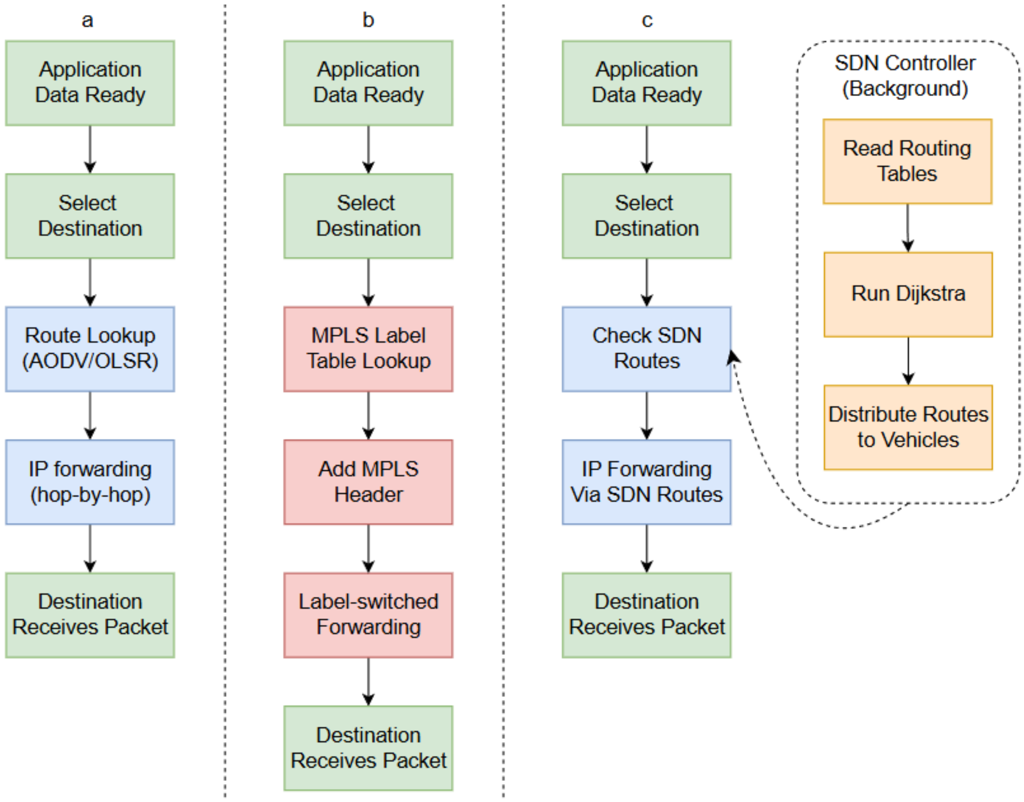

Figure 2. Tested Scenarios Flowchart: a) Default Protocol b) MPLS Enhanced c) SDN Integration

MPLS Integration

The MPLS integration adds label-switched forwarding to the existing routing protocols, without modifying their route discovery mechanisms. Each vehicle maintains a local MPLS label table, which is constructed from the routing information provided by the underlying protocol (AODV or OLSR), mapping destinations to outgoing labels. The label table is refreshed every 3 seconds to reflect topology changes discovered by the routing protocol.

When a vehicle sends data, it checks its MPLS label table for the destination. If a valid entry is found, the vehicle adds on the packet a 4-byte MPLS header with the label value, before forwarding it. Intermediate nodes perform label lookup and forwarding without consulting the IP routing table, reducing per-hop processing time since the cross-check is done in the data-link layer. The MPLS header adds minimal overhead (4 bytes per packet) while providing the performance benefits of predetermined forwarding paths. Packet flow for this scenario is represented in Figure 2b.

Label distribution throughout the network is performed through periodic broadcast on the wireless channel. Each vehicle broadcasts its label bindings to neighbouring nodes every 3 seconds, consuming wireless bandwidth but ensuring that neighbours have current forwarding information. This approach adapts the traditional LDP (Label Distribution Protocol) model to the broadcast nature of wireless ad hoc networks, where all neighbours can receive label announcements simultaneously. Figure 3 depict network elements and interfaces in VANET topologies.

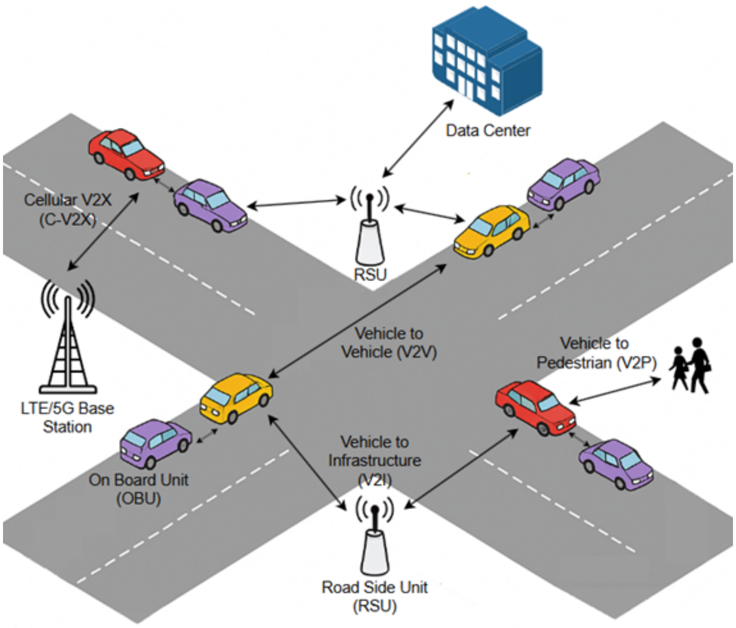

Figure 3. Network Elements and Interfaces in VANET Topologies

SDN Integration

The SDN integration introduces a centralized controller that maintains a global view of the network topology and computes optimized routes for all vehicles. The SDN controller is connected to all RSUs via the wired Ethernet backbone and receives topology information from the routing protocol running on each vehicle and RSU. By reading the AODV or OLSR routing tables from all nodes in the network, the controller builds a topology graph. The topology uses Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I) and Infrastructure-to-Infrastructure (I2I) interconnections as presented in Figure 3, each with the same metric value of 1.

Using the complete topology graph, the SDN controller runs Dijkstra shortest-path algorithm to compute the best routes for all source-destination flows. The computed routes are distributed to vehicles, where each one periodically reads the controller computed paths and installs corresponding /32 host routes in its IP routing table. Traffic flow and SDN tasks are displayed on Figure 2c.

These SDN-installed routes coexist with the routes maintained by the underlying routing protocol, with the SDN routes having priority due to their more specific prefix length. Packets following SDN-computed routes experience lower per-hop delay when compared to the baseline routing protocol, as the forwarding decision is previously defined by the controller rather than computed at each hop.

MPLS and SDN Integration

The combined SDN-MPLS integration, shown in Figure 4, represents the most advanced configuration, as it brings to the network topology both the centralized path computation of SDN and the fast label-switched forwarding of MPLS. The SDN controller computes optimal paths using the global topology graph and distributes both routing information and MPLS label bindings to vehicles.

In this configuration, the SDN controller operates in MPLS push mode, where it computes per-link MPLS labels for the optimized paths and distributes them to the relevant vehicles. Each vehicle receives label mappings for its computed paths and prepends MPLS headers when forwarding packets along SDN-optimized routes. This combination offers the advantage of two layers of optimization: the SDN controller ensures that packets follow globally optimal paths, while MPLS labels ensure that forwarding at each hop is performed through fast label switching rather than IP table lookups.

This combination can cause lower per-hop delay due to the reduced processing overhead of label-based forwarding, accounting for the combined benefits of centralized path optimization and label-switched forwarding. But it is also followed by an increased routing overhead caused by the MPLS headers and the SDN control traffic, from telemetry reports and label distribution messages.

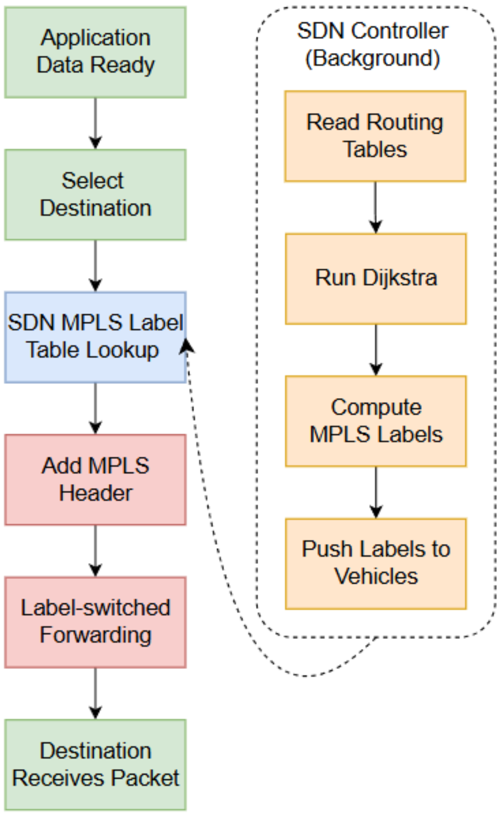

Figure 4. Tested Scenarios Flowchart: Integrated SDN with MPLS forwarding

SIMULATION RESULTS AND ANALYSIS

Simulation Setup

OMNeT++ 6.1 discrete event simulator with INET 4.5 framework for network protocol modelling and Veins 5.3.1 for vehicular network simulation coupled with SUMO for realistic traffic generation are used for the simulations. The simulation infrastructure, network elements and topology are presented in Figure 5. The simulation parameters and their values are summarized in Table 1.

Figure 5. Simulation infrastructure, network elements and topology

Table 1. Simulation Parameters

Parameter	**Value**
Simulation area	2500 m x 2000 m (Tirana, Albania)
Number of vehicles	50
Number of RSUs	5 (ring Ethernet topology)
Simulation time	100 seconds
Wireless standard	IEEE 802.11p, 5.9 GHz, 10 MHz
Vehicle Tx power	20 mW (~300 m range)
RSU Tx power	110 mW (~750 m range)
Data packet size	512 bytes
CBR send interval	0.5 seconds
VBR interval range	0.1 - 2.0 seconds (uniform)
Burst configuration	10 packets every 5 seconds
RSU backbone	1 Gbps fiber, 0.1 ms delay

Each scenario is executed with multiple random seed values to account for stochastic variations in vehicle departure times, wireless channel access, and routing protocol behaviour. The Key Performance Indicators (KPIs) reported are averaged across all vehicles and all seed iterations, and include Packet Delivery Ratio (PDR), average end-to-end delay, average jitter, throughput, and routing overhead.

Simulation Results

In this section are presented the simulation results for AODV and OLSR protocol families under CBR, VBR, and Burst traffic patterns. The results are organized by traffic pattern, with each table showing the progressive performance improvements from Default through MPLS, SDN, and SDN-MPLS configurations. The simulation results are presented in Tables 2 until 7.

Table 2. AODV Scenario Results Under CBR Traffic

Scenario	PDR (%)	Delay (ms)	Jitter (ms)	Throughput (kbps)	Overhead (%)
Default_AODV	55.37	862.56	1097.20	43.99	11.70
MPLS_AODV	55.22	786.57	1063.57	44.16	60.32
SDN_AODV	55.37	733.17	932.62	43.99	50.08
SDN_MPLS_AODV	55.22	629.26	850.85	44.16	90.14

Table 3. AODV Scenario Results Under VBR Traffic

Scenario	PDR (%)	Delay (ms)	Jitter (ms)	Throughput (kbps)	Overhead (%)
Default_AODV	62.17	731.18	969.44	23.89	20.86
MPLS_AODV	62.72	664.65	955.51	24.20	75.20
SDN_AODV	62.17	621.50	824.02	23.89	65.92
SDN_MPLS_AODV	62.72	531.72	764.40	24.20	94.73

Table 4. AODV Scenario Results Under Burst Traffic

Scenario	PDR (%)	Delay (ms)	Jitter (ms)	Throughput (kbps)	Overhead (%)
Default_AODV	51.52	776.16	906.43	44.53	11.15
MPLS_AODV	51.89	730.04	895.11	44.90	59.08
SDN_AODV	51.52	659.74	770.46	44.53	46.16
SDN_MPLS_AODV	51.89	584.04	716.09	44.90	89.12

Table 5. OLSR Scenario Results Under CBR Traffic

Scenario	PDR (%)	Delay (ms)	Jitter (ms)	Throughput (kbps)	Overhead (%)
Default_AODV	34.14	104.87	68.46	27.76	12.04
MPLS_AODV	33.48	82.22	70.51	27.42	82.00
SDN_AODV	34.07	92.53	54.61	27.97	53.37
SDN_MPLS_AODV	31.73	62.32	54.80	26.10	92.86

Table 6. OLSR Scenario Results Under VBR Traffic

Scenario	PDR (%)	Delay (ms)	Jitter (ms)	Throughput (kbps)	Overhead (%)
Default_AODV	33.96	110.65	79.25	13.73	22.23
MPLS_AODV	34.77	79.85	73.76	14.01	90.39
SDN_AODV	35.33	90.35	58.83	14.14	69.67
SDN_MPLS_AODV	33.92	60.12	55.30	13.63	96.16

Table 7. OLSR Scenario Results Under Burst Traffic

Scenario	PDR (%)	Delay (ms)	Jitter (ms)	Throughput (kbps)	Overhead (%)
Default_AODV	33.97	143.89	94.95	29.65	11.28
MPLS_AODV	33.33	123.34	109.85	29.45	81.98
SDN_AODV	33.90	118.20	75.28	30.03	49.06
SDN_MPLS_AODV	32.47	88.39	74.79	28.92	90.77

The results show a consistent pattern across both routing protocols and for the three types of traffic, showing that every integration layer provides clear and measurable enhancements to specific KPIs.

Simulation Analysis

Delay and Jitter Reduction

From the simulation results, the most considerable and consistent improvement across all of the tested configurations is the progressive reduction in average end-to-end delay. For AODV, the delay reduction follows a clear hierarchy. Under CBR traffic, MPLS reduces delay by 8.8% (from 862.56 ms to 786.57 ms), SDN by 15% (to 733.17 ms), and the combined SDN-MPLS configuration achieves a 27% reduction (to 629.26 ms) as presented in Table 2. This pattern is consistent across VBR (reductions of 9.1%, 15%, and 27.3% respectively) and Burst traffic (5.9%, 15%, and 24.8%) (see Table 3 and Table 4).

For OLSR, the delay improvements are even more pronounced. Under CBR traffic, MPLS achieves a 21.6% reduction (from 104.87 ms to 82.22 ms), SDN reduces delay by 11.8% (to 92.53 ms), and SDN-MPLS by 40.6% (to 62.32 ms) (see Table 5). The SDN-MPLS configuration under VBR traffic achieves the best delay reduction of 45.7% (from 110.65 ms to 60.12 ms) (see Table 6). MPLS impact on OLSR can be linked with the proactive nature of OLSR, which maintains complete routing tables, providing MPLS with more stable and pre-computed label paths compared to AODV's reactive route discovery.

Jitter improvements show a different pattern between the two protocols. For AODV, MPLS provides minimal jitter reduction (1.2% to 3.1% across traffic patterns), while SDN delivers an approximate of 15% reduction, and SDN-MPLS achieves 21% to 22.5% improvement. This indicates that for AODV, jitter improvement is primarily driven by SDN's centralized path optimization rather than MPLS label switching.

For OLSR, the jitter behaviour reveals an interesting finding. MPLS alone slightly increases jitter under CBR (+3.0%) and Burst (+15.7%) traffic patterns, while reducing it under VBR (−6.9%) (see Table 5, Table 6 and Table 7). This increase occurs because OLSR's periodic Hello and TC messages interact with MPLS label distribution broadcasts, creating periodic channel congestion that introduces jitter variability. SDN integration, however, reduces OLSR jitter by 20% to 26% and the combined SDN-MPLS achieves 20% to 30% reduction.

PDR and Throughput Stability

PDR for both protocols is relatively stable for all four configurations. PDR values for AODV are close between all four configurations (55.22% to 55.37% for CBR, 62.17% to 62.72% for VBR, 51.52% to 51.89% for Burst) (see Table 2, Table 3 and Table 4). This stability indicates that neither MPLS nor SDN affects the packet delivery ratio. MPLS and SDN change how packets are forwarded along discovered paths, but they do not participate in path discovery.

While OLSR shows a slightly larger PDR variation (up to 2.41% under CBR), the SDN-MPLS configuration results in a slightly lower PDR than the default. This decrease can be attributed to the additional control traffic generated by the combined SDN telemetry and MPLS label distribution, which consumes wireless channel capacity that would otherwise be available for data packets.

Throughput remains consistent for all configurations and both protocols, as it is directly connected with PDR results. AODV maintains throughput between 43.99 and 44.90 kbps under CBR and Burst, and approximately 23.89 to 24.20 kbps under VBR (lower due to the variable and sometimes longer send intervals). OLSR throughput is lower overall (26 to 30 kbps for CBR/Burst, 13 to 14 kbps for VBR), consistent with its lower PDR.

Routing Overhead and Routing Protocol Impact

The routing overhead increases progressively with each integration layer, representing the trade-off between performance improvement and control traffic cost. For AODV under CBR, Default overhead is 11.70%, MPLS increases to 60.32% due to periodic label distribution broadcasts, SDN adds 50.08% from controller telemetry and route distribution, and SDN-MPLS combines both sources reaching 90.14%. OLSR follows a similar pattern with slightly higher overhead due to its proactive control messages. The overhead increase is most pronounced under VBR traffic, where the variable sends intervals result in a higher proportion of control traffic relative to data traffic.

VBR traffic consistently produces the best relative improvements from SDN and SDN-MPLS integration, with the highest delay and jitter reductions for both protocols. This occurs because VBR's variable send intervals create periods of lower channel utilization during which SDN-optimized paths can be more effectively utilized. Burst traffic shows the lowest PDR across both protocols (51.52% for AODV, 32.47% to 33.97% for OLSR) due to the concentrated packet transmissions that can cause channel saturation and collisions during burst periods.

Comparing AODV and OLSR reveals fundamental architectural differences in their response to MPLS and SDN integration. AODV achieves higher PDR (51% to 62%) compared to OLSR (31% to 35%) across all configurations, indicating that AODV's reactive route discovery is more effective in this urban topology with 50 vehicles. However, OLSR provides significantly lower absolute delay (60 to 144 ms vs. 531 to 863 ms for AODV), results' distribution over different simulation seeds are displayed in Figure 6 and Figure 7, reflecting the benefit of pre-computed routes that eliminate route discovery latency. The relative improvement from SDN-MPLS integration is larger for OLSR in delay reduction (38% to 46%) compared to AODV (25% to 27%), suggesting that centralized optimization is more efficient in proactive routing protocols. A summary for this comparison is shown in Table 8.

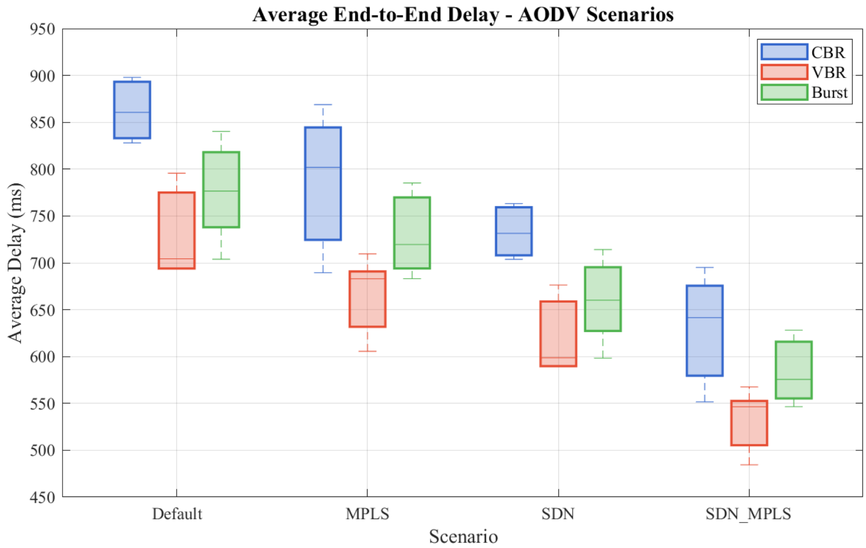

Figure 6. Results' Distribution of Average Delay for AODV Protocol

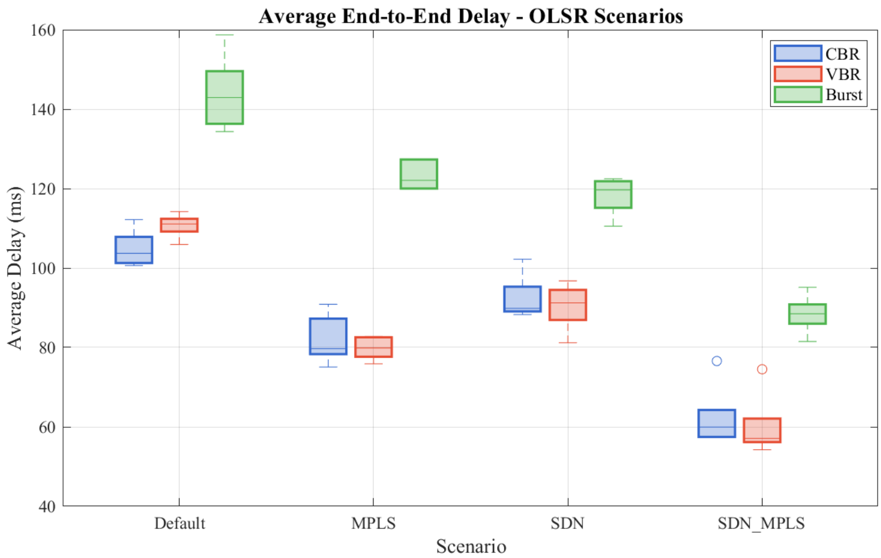

Figure 7. Results' distribution of Average Delay for OLSR Protocol

Table 8. Average Change Over Default Configuration for AODV & OLSR Scenarios.

KPI	AODV			OLSR		
	MPLS	SDN	SDN_MPLS	MPLS	SDN	SDN_MPLS
PDR	+0.44%	0%	+0.44%	−0.47%	+1.21%	−3.87%
Delay	−7.95%	−15%	−26.36%	−21.23%	−15.99%	−41.61%
Jitter	−1.92%	−15%	−21.53%	+3.92%	−22.24%	−23.79%
Throughput	+0.84%	0%	+0.84%	+0.05%	+1.68%	−3.06%
Overhead	+368%	+286%	+575%	+505%	+297%	+570%

Overhead and Deployment Trade-offs

The routing overhead reported in this study (up to 90.14% for AODV and 96.16% for OLSR in the SDN-MPLS configuration) represent the ratio of control traffic to total traffic, while using 512-byte data packets in the simulations. The large percentage of control overhead creates discussions regarding its practicality in a real-world deployment scenario, but there are also some factors to be considered.

Firstly, the absolute overhead is directly connected to the simulation's small packet size (512 bytes) and network scale. In a real IoT/VANET deployments using larger payloads (1500-byte Ethernet frames for infotainment or data aggregation), the same control message volume would represent a smaller percentage of the total traffic. Secondly, the SDN control traffic and MPLS label distribution is continuously shared via the 10 MHz wireless channel, which consumes the wireless capacity. This can be limited or integrated on existing beacon messages in a production deployment. Thirdly, the overhead reflects the results of the short broadcast intervals (every 3 seconds for MPLS label distribution and 1 second for SDN telemetry). In practice, these intervals are tuneable parameters. Increasing the label refresh interval to 5-10 seconds, or triggering updates only on detected topology changes rather than periodically, would reduce the network-wide overhead.

The trade-off between control traffic frequency and path optimization is a standard design parameter in SDN deployments, but it was not the focus of this study, which instead isolated the maximum achievable benefit of each technology layer. From a deployment perspective, the SDN-only integration offers the best overhead-to-benefit ratio (46-70% overhead for 12-18% delay reduction), making it suitable for bandwidth-constrained deployments. The SDN-MPLS configuration, despite its higher overhead, remains a good option for safety-critical applications (collision warnings, emergency braking alerts) where the 25-46% delay reduction directly affects reaction time. For infotainment or big-data applications with larger packets, the overhead percentage would decrease proportionally, making SDN-MPLS a feasible option under the current settings.

CONCLUSION

This paper presented the evaluation of progressive technology integration with two fundamental VANET routing protocols, AODV and OLSR. Simulations of four configurations for each protocol are conducted: the default protocol, MPLS-enhanced forwarding, SDN-based centralized route optimization, and the combined SDN-MPLS integration. A real-world environment modelled on the urban road network of Tirana, Albania, with 50 vehicles and 5 RSUs across three traffic patterns is used.

The results show that the combined SDN-MPLS integration achieves the highest delay improvements across all configurations, reducing AODV delay by 25% to 27% and OLSR delay by 39% to 46% compared to the default protocols. Each integration layer brings forward improvements: MPLS accelerates forwarding through label switching (6% to 28% delay reduction), while SDN optimizes path selection through centralized computation (12% to 18% delay reduction). This combination provides benefits that exceed the performance of each technology, when applied independently.

PDR remains stable across all configurations for AODV indicating that the integration layers improve forwarding efficiency without affecting the routing protocol's ability to discover valid paths. OLSR shows modest PDR variation (up to 2.41%), suggesting that the additional control traffic from MPLS and SDN has a minor but measurable impact on the proactive protocol's higher baseline control overhead.

The routing overhead increase from MPLS (reaching 60% to 90% in combined configurations) represents the primary trade-off of this approach. In deployment scenarios where bandwidth is constrained, the overhead cost must be weighed against the delay and jitter improvements. The results suggest that SDN-only integration provides a favourable balance, achieving 12% to 18% delay reduction with moderate overhead (46% to 70%), while the full SDN-MPLS combination is justified for latency-critical applications where delay reduction is the primary objective.

Future work will consider Segment Routing (SR) as an additional traffic engineering mechanism, investigate the impact of different SDN topology construction approaches, and evaluate the performance under different vehicle densities and simulation areas.

CONFLICT OF INTERESTS

The authors would like to confirm that there is no conflict of interests associated with this publication and there is no financial fund for this work that can affect the research outcomes.

REFERENCES

1. Hako, R., Spaho, E. From VANETs to Software-Defined VANETs: A Comprehensive Review of Architectures, Routing Strategies, and Open Research Challenges. In *Advances in Internet, Data & Web Technologies*; Barolli, L., Petrakis, E. G. M., Hussain, F., Eds.; *Lecture Notes on Data Engineering and Communications Technologies*; Springer: Cham, Switzerland, **2026**, *290*.
2. Qur, K.N., Abdullah, H., Iqbal, S. Improving Quality of Service through Road Side Backbone Network in VANET. *J. Teknol.* **2016**, 78(2), 7-14.

3. Ghori, M.R., et al. VANET Routing Protocols: Review, Implementation and Analysis. *J. Phys.: Conf. Ser.* **2018**, *1049*, 012064.
4. Trivedi, H., Tanwar, S., Thakkar, P. Software Defined Network-Based Vehicular Ad Hoc Networks for Intelligent Transportation System: Recent Advances and Future Challenges. In *Futuristic Trends in Network and Communication Technologies*; Singh, P., Paprzycki, M., Bhargava, B., Chhabra, J., Kaushal, N., Kumar, Y., Eds.; *Communications in Computer and Information Science*; Springer: Singapore, **2019**, *958*.
5. Wahid, I., Tanvir, S., Ahmad, M., Ullah, F., AlGhamdi, A.S., Khan, M., Alshamrani, S.S. Vehicular Ad Hoc Networks Routing Strategies for Intelligent Transportation System. *Electronics* **2022**, *11*, 2298.
6. Kadir, G., et al. Combining MPLS with Ad-hoc routing protocols for a VANET congestion alert. *Passer Journal of Basic and Applied Sciences*, **2024**, *6*, 242-253.
7. Ku, I., et al. Towards Software-Defined VANET: Architecture and Services. *13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*; **2014**, 6849111.
8. Al-Heety, O.S., et al. A Comprehensive Survey: Benefits, Services, Recent Works, Challenges, Security and Use Cases for SDN-VANET. *IEEE Access* **2020**. *8*, 91028-91047.
9. Truong, N. B., Lee, G. M., Ghamri-Doudane, Y. Software Defined Networking-Based Vehicular Ad Hoc Network with Fog Computing. In *Proceedings of the IEEE Network Operations and Management Symposium*; **2015**, 7140467.
10. Mekki, T., Jabri, I., Rachedi, A., Chaari, L. Software-Defined Networking in Vehicular Networks: A Survey. *Emerg. Telecommun. Technol.* **2021**, *33*(10), e4265.
11. Fathy, M., et al., Improving QoS in VANET Using MPLS. *Procedia Comput. Sci.* **2012**, *10*, 1018–1025.
12. Ghafoor, H., Koo, I. CR-SDVN: A Cognitive Routing Protocol for Software-Defined Vehicular Networks. *IEEE Sens. J.* **2017**, *18*(4), 1761-1772
13. Yeferny, T., Hamad, S. Vehicular Ad Hoc Networks: Architecture, Applications and Challenges. *Int. J. Comput. Sci. Netw. Secur.* **2020**, *20*(2), 1-7.
14. Azzoug, Y., Boukra, A. Bio-Inspired VANET Routing Optimization: An Overview. *Artif. Intell. Rev.* **2020**, *54*, 1005-1062.
15. Chahal, M., Harit, S. Network Selection and Data Dissemination in Heterogeneous Software Defined Vehicular Network. *Comput. Netw.* **2019**, *161*, 32–44.
16. Kai, K., Cong, W., Tao, L. Fog Computing for Vehicular Ad Hoc Networks: Paradigms, Scenarios, and Issues. *J. China Univ. Posts Telecommun.* **2016**, *25*(2), 56-65.
17. Wijesekara, P.A.D.S.N., Sudheera, K.L.K., Sandamali, G.G.N., Chong, P.H.J. An Optimization Framework for Data Collection in Software Defined Vehicular Networks. *Sensors* **2023**, *23*, 1600.