A Cross-layer Approach for MPTCP Path Management in Heterogeneous Vehicular Networks

Vadym Hapanchak, and António Costa

***Abstract*—Multipath communication has recently arisen as a promising tool to address reliable communication in vehicular networks. The architecture of Multipath TCP (MPTCP) is designed to facilitate concurrent utilization of multiple network interfaces, thereby enabling the system to optimize network throughput. In the context of vehicular environments, MPTCP offers a promising solution for seamless roaming, as it enables the system to maintain a stable connection by switching between available network interfaces. This paper investigates the suitability of MPTCP to support resilient and efficient Vehicle-to-Infrastructure (V2I) communication over heterogeneous networks. First, we identify and discuss several challenges that arise in heterogeneous vehicular networks, including issues such as Head-of-Line (HoL) blocking and service interruptions during handover events. Then, we propose a cross-layer path management scheme for MPTCP, that leverages real-time network information to improve the reliability and efficiency of multipath vehicular communication. Our emulation results demonstrate that the proposed scheme not only achieves seamless mobility across heterogeneous networks but also significantly reduces handover latency, packet loss, and out-of-order packet delivery. These improvements have a direct impact on the quality of experience for vehicular users, as they lead to lower application layer delay and higher throughput.**

***Index Terms*—V2X, MPTCP, Multipath, Vehicular communications, Path manager, cross-layer, Wireless, Connected Vehicles, Heterogeneous networks.**

I. INTRODUCTION

THE Intelligent Transportation Systems (ITS) have developed rapidly to enhance road safety and offer new services such as infotainment and in-car connectivity. Wireless communication plays an essential role in ITS, allowing vehicles, roadside infrastructure, and remote entities to cooperate to enhance the traffic efficiency and comfort of road users.

Current radio access technologies (RATs) have different characteristics regarding coverage area, data rate, central frequency, and modulation scheme. Combining multiple network technologies provides reliable broadband access for a vehicle user and helps to improve communication efficiency and throughput. Integration of multiple RATs is needed to support connectivity over various networks with seamless mobility and good service continuity. However, the heterogeneity of such technologies still causes severe concerns with data transfer and service continuity, as a vehicular user may frequently switch between different networks while traveling from one location to another. In a highly dynamic communication environment, where topology frequently changes and connectivity is often disrupted, meeting Quality of Service (QoS) requirements can be a challenging task. In addition, meeting the varying reliability and latency requirements of different applications in a moving vehicle can be a challenging task.

Several studies have proposed a tight integration of different wireless technologies to increase communication efficiency in heterogeneous vehicular networks [1]–[3], where each system offers unique benefits. Modern vehicles are often equipped with On-Board Unit (OBU) containing different radio interfaces such as Wireless Fidelity (Wi-Fi), dedicated short-range communication (DSRC), Long-Term Evolution (LTE), Bluetooth (BT), among others. OBU allows exchanging information with other road users and road infrastructure by using available network devices. For example, a vehicle user can connect to the Internet through the Roadside Unit (RSU) in ad hoc mode or through the infrastructure of the cellular network. However, several problems need to be addressed to make different technologies work together under highly dynamic vehicular environment [4]. Coordinating multiple RATs to select the network with the best connectivity while considering application requirements is a challenging task. The vertical handover (VHO) mechanism between heterogeneous networks, induced by vehicle mobility, must provide seamless and reliable connectivity anytime and anywhere. The network selection algorithm should consider multiple factors, such as signal strength, application requirements, available bandwidth, vehicle position, and speed.

Multipath connectivity is a promising approach to improve the network service for high-speed vehicles [5]–[8]. Equipped with multiple interfaces, the OBU enables vehicles to take full advantage of the available wireless technologies and maintain a continuous connection. Although multipath communication offers many advantages, its performance is currently poor in highly dynamic environments [9]. The constant handover events over heterogeneous networks due to vehicle mobility often lead to significant service interruptions. To address this issue, our paper focuses on improving Multipath TCP's [10], [11] handling of vehicle mobility at the transport layer. MPTCP is capable of maintaining several connections, known as *subflows*, and simultaneously transmitting a single data stream over them.Unfortunately, the current protocol implementation does not consider user mobility or the frequently

Manuscript received January 12, 2023; revised February 22, 2023. Date of publication March 28, 2023. Date of current version March 28, 2023.

The paper was presented in part at the International Conference on Software, Telecommunications and Computer Networks (SoftCOM) 2022.

Authors are with the University of Minho, Portugal (e-mails: b7768@algoritmi.uminho.pt, costa@di.uminho.pt).

Digital Object Identifier (DOI): 10.24138/jcomss-2022-0177

changing path characteristics of wireless links. Instead, the MPTCP uses the transport layer information to estimate the channel capacity and latency, which is insufficient in accurately reflecting the dynamic nature of vehicular networks. Consequently, link failure detection can be slow, especially when the path is abruptly broken, resulting in low communication performance under handover events.

Our main goal is to design a path management (PM) scheme that considers characteristics of available heterogeneous vehicular networks. In this work, we focus on a cross-layer approach, where the interaction among upper and lower protocol stack layers exists in order to improve path selection accuracy. Our research offers a comprehensive evaluation of MPTCP's performance in vehicular networks. Additionally, we propose a novel PM scheme that utilizes received signal strength (RSS) to dynamically manage interface usage. This approach accurately assesses path quality and predicts potential link failures in advance, resulting in better network performance. The effectiveness of the proposed technique in vehicular scenarios was demonstrated using a realistic emulation framework [12].

The initial version of our PM scheme was published in [13], where we analyzed the cross-layer approach for vehicular networks. This paper presents an extended version of the initial solution with new unpublished results, and it differs in several aspects, as outlined below:

- Section II contains an extended list of related works, a more detailed description of each work mentioned, and a major standardization efforts.
- Section III presents the theoretical background and the description of various issues related to the use of multipath communication in vehicular networks.
- Section IV first described the *mptcpd* tool, used to send and receive PM-related messages from the kernel. Next, an intelligent subflow control algorithm was designed, which is a novel contribution.
- Section V evaluates the performance of the proposed cross-layer PM for vehicular networks using experiments conducted in a realistic emulation environment to confirm previous results and obtain new outcomes.

The rest of the paper is organized as follows: Section II reviews the literature on VHO methods in heterogeneous vehicular networks. Section III provides background and motivation regarding the usage of MPTCP in vehicular communications. Open issues of current multipath strategies are then discussed. The proposed cross-layer PM algorithm is described in Section IV, and its performance evaluation is presented in Section V. Section VI concludes the paper.

II. Related Works

The *Media Independent Handover (MIH)* service protocol [14] by the IEEE 802.21 Working Group is the major standardization effort for handoff mechanism between heterogeneous networks. The IEEE 802.21 standard specifies an interface that optimizes the handover mechanism among cellular and non-cellular systems. The set of basic services offered by the MIH allows interaction with the upper and lower layers through exchanging messages between entities. For example, MIH can detect the changes in physical channel condition and link state changes, and this information can be used at the upper layer to make VHO decisions and control the lower layers through a set of commands.

3GPP has recently proposed the use of multipath transport protocols for 5G networks to improve network interoperability [15]. As a result, the *Access Traffic Steering, Switching, and Splitting* (ATSSS) architecture was introduced to exploit transport layer multi-connectivity between 3GPP and non-3GPP networks. The recently introduced Path Aware Networking Research Group [16] aims to support research in bringing path awareness to transport and application layer protocols. It exploits interoperability mechanisms in order to combine different networks; discover information about the properties of a path, and design new algorithms for path selection and scheduling decisions based on this information.

Numerous studies are currently being conducted to address vehicular connectivity in heterogeneous multi-RAT environments. However, only a few of these works have taken into account the use of MPTCP for data transfer and seamless mobility in highly dynamic environments. A survey in [6] reviewed the existing studies on MPTCP for vehicular communications, with detailed discussions on the benefits and future research directions on MPTCP in vehicular networks. In [17], the authors discuss the use of MPTCP as a potential solution to address mobility-related issues. They emphasized the need for cross-layer assistance to be incorporated into the MPTCP solution. However, the design and implementation of such a mechanism were left for future work. The report in [7] presents a preliminary investigation into the use of MPTCP for V2I purposes. Measurements were performed on live networks and across the Internet using WiFi (802.11n, 802.11p) and 3G with several static and mobile scenarios. The presented results show that using multiple subflows was beneficial for both short and long flows, however not for heterogeneous links, when available links are highly diverse in terms of bandwidth, RTT or loss rate. The work in [18] proposes the cross-layer PM that exploit MAC-Layer information to estimate path quality. Experiments in mobile scenario, show that the proposed method is able to use the available subflows more quickly, and thus achieve better performance. Similarly, authors in [19] propose mechanisms based on cross-layer attributes measurements to improve a decision tree approach for intelligent path selection in MPTCP. Authors found that MAC-layer attributes have a strong relationship with the aggregated performance of MPTCP. The proposed decision tree approach decides which path to carry the incoming packets dynamically according to the prior learned schemes. The experiments were performed in the actual mobile scenarios and large-scale simulation. The work in [20] proposed a mobility-aware multimedia data transfer mechanism using MPTCP in a vehicular network. The proposed VHO method handles the link disruption caused by vehicle mobility. The distance between the vehicle and RSU is calculated to determine the unreliable path. The subflow congestion window size is set to 0 when the distance exceeds the communication range of RSU. The study in [8], propose a packet scheduling scheme for MPTCP, which tries to mitigate the impact of HoL blocking under handover events incurred

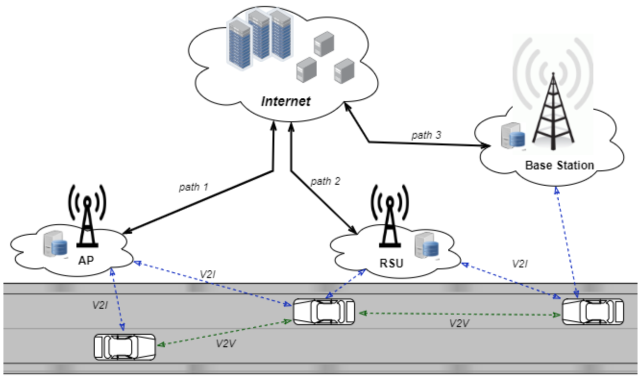


Fig. 1. Heterogeneous vehicular communication system.

by vehicle mobility. The dynamism of a wireless connection's characteristics were emulated in hardware set up, with the trace-based mobility model, to evaluate the impact of realistic vehicle mobility.

III. Background and Motivation

A vehicular network is considered highly dynamic due to the speed of the vehicles and radio propagation characteristics (Fig. 1). Vehicles can rapidly join or leave the network, leading to frequent disconnections and topology changes [4]. The Vehicle-to-Vehicle (V2V) communication allows vehicles to communicate with each other and to share safety-critical real-time information regarding their moving direction, position, velocity, acceleration, etc. Vehicles exchange data collected from sensors to increase perception capabilities. V2V permits direct communication among nearby vehicles without relying on fixed infrastructure support.

Vehicle-to-Infrastructure (V2I) communication allows fully mobile connectivity between road infrastructure and vehicles, connected through points of attachment within the different wireless technologies. V2I mainly provides information and data-gathering applications to drivers and passengers. In addition, V2I communication enables passengers to access various non-safety services when vehicles are moving, like internet connection, entertainment services, online multimedia services, and interactive communication. The communication range of the wireless technology and vehicle speed affect the connection time between the vehicle and the roadside infrastructure. The link availability is low, especially in highway scenarios, where the connection is often lost during data transmission. The rapid topology changes result in packet losses and reduced throughput, which negatively affect communication performance.

The advancement of wireless technology has enabled OBU devices to incorporate multiple low-cost network interface chips. Despite this, the management of handover processes for vehicular communication across diverse networks remains a complex issue. Traditional centralized handover mechanisms, commonly found in cellular networks, do not fit well within heterogeneous vehicular networks due to users frequently switching among different systems while on the move. To overcome this challenge, a decentralized mechanism is preferable. This approach empowers the vehicle user to select the most appropriate network based on local perceptions, providing a satisfactory level of quality of experience (QoE) while maintaining high mobility constraints.

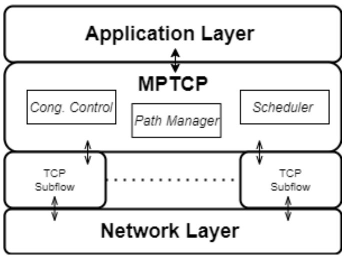


Fig. 2. MPTCP protocol stack.

A. *MPTCP: An Overview*

The Multipath TCP (MPTCP) is a well-known solution that provides an efficient end-to-end connection through various available paths. MPTCP is a set of TCP extensions, standardized in the Internet Engineering Task Force (IETF), that enables an application to send data over multiple IPs and interfaces. MPTCP functions transparently to applications and presents the standard socket API to the userspace, which means that applications are unaware of the use of multipath communication.

MPTCP is considered a decent solution for future LTE and 5G mobile networks, aiming to improve user experience with higher throughput and seamless connections [21]–[23]. Furthermore, MPTCP is compatible with the deployed networks since each subflow is handled in the middle boxes as a standard TCP session, so it does not suffer from any filtering on traditional network infrastructures. Therefore, only end devices, i.e., host and server, must support MPTCP for it to operate. The protocol configures the *Data Sequence Number (DSN)* to indicate the order of all packets in one MPTCP stream to ensure reliable, in-order data delivery over multiple dynamic paths. The ability of MPTCP to use multiple interfaces simultaneously allows applications to persist active in the case of VHO, whereas traditional TCP would break. In addition, MPTCP has the ability to establish, terminate, or modify the priority of each subflow at any point during an ongoing session. In this study, the proposed PM mechanism leverages this protocol capability to regulate subflow utilization as the vehicle transitions between different access networks.

MPTCP has been implemented on Linux Kernel and declares a modular structure with path manager, congestion control, and packet scheduler strategies (Fig. 2). When multiple communication paths are present, the packet scheduler method is needed to decide which data segment is transmitted over which subflow. Different scheduling strategies have been implemented so far [24]. The default MPTCP scheduler sends data on the subflow with the lowest round trip time (RTT) that its congestion window is not full. If one subflow fails during the connection, the scheduler retransmits lost data over available subflows. Furthermore, different congestion control algorithms have been proposed to guarantee fairness with other network users. MPTCP regulates subflow utilization through the PM module, allowing users to configure the specific operational mode for each MPTCP endpoint. The default *full-mesh* PM tries to create subflows between all pairs of IP addresses of the client and remote host, i.e., creates one subflow over each active interface. As a result, subflows are created immediately after the connection is established and removed when the MPTCP session is closed, which may prove insufficient for highly dynamic vehicular network environments.

In this work, the *"upstream"* implementation of MPTCPv1 [11] has been used. Unfortunately, this implementation isn't compatible with MPTCPv0 [10] and *"out-of-tree"* support of MPTCPv1 (available from the v0.96 version). The new protocol version has different design choices and fewer modifications to the original TCP stack than the previous version. It is included in the upstream kernel, and the recent Linux distributions have already enabled MPTCP support. Thus, modifying the system to install and use alternative kernels is no longer necessary. By default, each MPTCP connection uses a single subflow similar to plain TCP. Both the client and server should be appropriately configured, using, for example, the *iproute2* tool to enable use of multiple interfaces for applications. It is possible to specify a higher limit for the maximum number of subflows which each connection can have. In addition, it is possible to select the MPTCP endpoints (IP addresses) which will be announced or used for additional subflows. The work around the *"upstream"* MPTCP implementation is still in progress and the new features are still being developed [25]. As of the time of writing, the upstream implementation has only one packet scheduler model, implementing a simplified version of the *Blest* algorithm [26]. No coupled congestion controls are supported; regular ones per subflow are used.

B. Problem Statement

MPTCP can provide seamless mobility in case of handover in vehicular communications by aggregating multiple wireless connections transparently to the application. For instance, a vehicle's OBU with two network interfaces would be able to use them both simultaneously to transmit data while moving on the road. This allows to hold up peer-to-peer connection even when one link fails, such as when the vehicle moves out of the coverage range of a particular network. Although MPTCP is a promising solution that can improve V2I connections, several problems remain regarding service continuity during handoff scenarios. Due to the vehicular network's high dynamicity, the protocol must carefully evaluate link characteristics to avoid bad communication performance.

1) Seamless Handover: Transferring the connection from one access network to another is known as a VHO process. One of the main challenges in VHO management is maintaining continuous connectivity of mobile users when the connection needs to migrate between heterogeneous networks. Seamless VHO is a considerable challenge in vehicular networks caused by a highly dynamic communication environment.

MPTCP is engineered to enhance network resilience against link failures, leveraging multiple available paths to aggregate system capacities. Nevertheless, in situations characterized by unstable wireless links and high mobility, MPTCP's performance may not be optimal. Consequently, the protocol could not ensure service continuity during VHO in vehicular scenarios due to interruptions of data delivery to the application. The path failure detection method needs time to determine the path stale state, which can cause significant data transfer issues under the VHO. The aforementioned scenario has the potential to diminish the QoE of applications with real-time demands, as we shall demonstrate in the ensuing section.

2) HoL Blocking: Multipath protocols commonly encounter a head-of-line (HoL) blocking issue in heterogeneous networks, where there is a significant discrepancy in available bandwidths and latency of the utilized paths. Packet arrival disorder at the receiver may result from multipath communication, as data received through a faster subflow must wait for packets from a slower subflow to arrive and ensure orderly delivery. If the receiver buffer cannot accommodate waiting for packets on the slower path, previously received packets may be dropped, and the protocol will cause a complete blockage of data transfer. Therefore, avoiding packet reordering in vehicular multipath communications is crucial to increase overall performance. The HoL blocking problem can be solved by an adaptive PM approach or by an intelligent packet scheduler algorithm, which considers the heterogeneity of the utilized links and their correlation. Section V provides emulation-based studies demonstrating the HoL blocking under handoff events in the vehicular scenario.

3) Path under-utilization: Our findings reveal that heterogeneous networks lead to under-utilization of the recovered path for long-lived flows, which we discuss later in Section V. For example, when a vehicle moves beyond the range of a RSU, the associated subflow becomes unusable due to a weak signal, even though the WLAN interface maintains a valid IP address. In the event of packet loss, the protocol is unaware of the path's unavailability and attempts to restore the subflow by initiating a retransmission timeout (RTO). The RTO increases exponentially when the anticipated acknowledgment fails to arrive. This behavior is illustrated in Fig. 3. Here, the second subflow was interrupted and recovered during one MPTCP session. After the path recovers, the sender must wait for the successful retransmission occurs to determine the path availability. This unnecessary delay causes a vehicle to lose a significant portion of available connection time. The above analysis indicates that MPTCP, by default, cannot effectively leverage all available paths in highly dynamic networks with lossy wireless links. Additionally, standard RTT-based scheduling policies are insufficient to mitigate this issue in mobile scenarios.

4) Slow Handover: The transport layer protocols have well-known problems dealing with varying path conditions. Multipath TCP follows a similar approach to handling network status changes as the actual implementation of TCP. The standard TCP error recovery mechanism and congestion control are used to deal with abrupt changes in path availability. The subflow is considered potentially failed after one RTO, and the packet scheduler will stop using this subflow to transmit the data. In a vehicular scenario, the handover mechanism of MPTCP is either reactive, leading to temporary connection

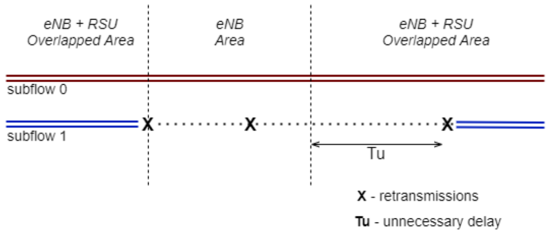


Fig. 3. Path under-utilization.

losses due to varying conditions of wireless networks. Consequently, the QoE is degraded as a vehicle can cross through multiple networks in a short time.

Therefore, advanced PM policies are required to control the subflow usage during the handover. This paper proposes a cross-layer PM mechanism to improve multipath communications in heterogeneous vehicular networks. In the next section, we describe in detail our PM solution that runs entirely in the *userspace* and interacts with the Linux kernel to control subflow usage.

IV. Cross-layer Path Manager

As previously mentioned, the ability of MPTCP to use multiple interfaces simultaneously can improve throughput and reliability. But, unfortunately, multipath vehicular communications still suffer from unstable wireless connections and high mobility. One possible solution to improve the connectivity in heterogeneous vehicular networks could be a proactive VHO mechanism, which can predict connection loss and seamlessly switch between available access technologies. Nevertheless, the PM module, which controls subflow utilization, resides entirely in the kernel, and the standard socket API exposed by the MPTCP does not allow selecting which path to use.

In [27], the authors propose to separate MPTCP data plane from the control plane by moving all PM functions into the userspace. The netlink inter-process communication mechanism was used to interact with the MPTCP kernel to obtain information about the MPTCP events and control established connections. A flexible API that exposes events and state information from the kernel was introduced, so the PM can react when a specific event is triggered.

A. MPTCP Daemon

The MPTCP daemon (*mptcpd*) was recently introduced to allow the development of customized PM strategies in userspace [28]. It leverages the *netlink* mechanism to interact with the Linux kernel and control subflow usage from the application layer through the set of instructions.

The *upstream* implementation of MPTCPv1 provides a netlink interface for PM-related operation, automatic endpoint configuration, and tracking per-connection information. As shown in Fig. 4, the MPTCP daemon handles connections through the generic API exposed by the Linux kernel. It provides a modular architecture that can easily integrate new PM strategies in the form of extensions - *plugins*. A plugin must control subflows through the set of available commands. At the same time, the kernel-userspace interaction is handled

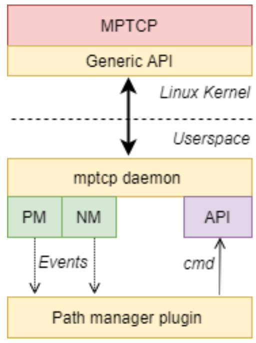


Fig. 4. The *userspace* path manager.

TAB. I. Path Manager Events

new_connection	A new MPTCP connection has been created
connection_established	New MPTCP connection has been established
connection_closed	MPTCP connection as a whole was closed
new_address	New address has been advertised by a peer
address_removed	Address is no longer advertised by a peer
new_subflow	A peer has joined the MPTCP connection
subflow_closed	A single MPTCP subflow was closed
subflow_priority	MPTCP subflow priority changed

by the *mptcpd* itself, while in-kernel events are forwarded to the plugin.

A plugin will obtain notifications for events it registered to, and should handle them accordingly. Corresponding callback functions are triggered reacting to various kernel events that can occur during MPTCP connection. The Path Manager (PM) module is responsible for delivering all PM related events. Typical PM events include connection established or closed, subflow created, addresses advertised, etc. Table I show a list of available PM events currently supported by the generic API. The *new_connection* event is triggered when a MPTCP connection is initiated, but not completed. The *connection_established* event is triggered on success of the three-way hand-shake procedure. It contains the information required to identify the connection, i.e., unique token local and remote addresses. The *new_subflow* and *subflow_closed* events are triggered once a new subflow has been established or terminated. Consequently, the *subflow_priority* event is triggered when a backup priority flag changes.

In addition, the Network Monitor (NM) module is responsible for delivering all network interface related events, as listed in Table II. The *new_address* event provides the IP addresses and ID of the endpoint announced by the remote host. Our PM algorithm uses this event to store the addresses of the remote host and establish new subflow only when the vehicle is in the coverage area of the serving network. This approach is more flexible than the existing in-kernel PM, which tries to create additional subflows immediately after the connection is established.

Similarly, the kernel-side can receive commands from the userspace to perform PM tasks. The generic API supports several types of commands that could change the state of MPTCP connection. As shown in Table III, the control level is sufficient for most vehicular scenarios. For example, it's possible to request MPTCP to create a new subflow with *add_subflow* command based on unique connection token,

TAB. II. NETWORK MONITOR EVENTS

new_interface	A new network interface is available
update_interface	Network interface flags were updated
delete_interface	A network interface was removed
new_local_address	A new local network address is available
delete_local_address	A local network address was removed

TAB. III. PATH MANAGEMENT COMMANDS

add_addr	Advertise new network address to peers
remove_addr	Stop advertising network address to peers
add_subflow	Create a new subflow
set_backup	Set priority of a subflow
remove_subflow	Remove a subflow
get_addr	Get network address
dump_addrs	Get list of MPTCP addresses
flush_addrs	Flush MPTCP addresses
set_limits	Set MPTCP resource limits
get_limits	Get MPTCP resource limits
set_flags	Set MPTCP flags for a local IP address

local and remote IP addresses and endpoint ID's, which should be known in advance. Similar commands allow removing subflow or change subflow priority.

B. Proposed PM Scheme

MPTCP can cause a reduction in communication performance during vertical handover when a moving vehicle switches between available wireless networks. Detection of link loss on the transport layer can be delayed, leading to a suboptimal user experience. The proposed cross-layer PM aims to predict connection issues by monitoring MAC-layer information on each available interface and estimating path quality based on RSS measurements. The main idea is to detect an imminent loss of connection before the link eventually becomes unavailable and data transmissions stops.

Our PM scheme, implemented as an *mptcpd* plugin, estimates MAC-layer information and adjusts subflow usage based on the perceived link quality. The handover decision is determined by a signal strength threshold, which selects the most suitable RAT and ensures uninterrupted user connections while the vehicle moves across diverse access networks. The system aims to enhance MPTCP reactivity by detecting link failures in advance, enabling seamless VHO, improving system resource utilization, and increasing resilience. By utilizing captured RSS information, our cross-layer PM can disable subflows with poor link quality and reactivate them when signal strength surpasses a pre-defined threshold. The cross-layer assistance is necessary, as the real-time network metrics, such as packet losses and RTT alone, are not always could indicate a forthcoming handover in a highly dynamic vehicular networks. The RSS parameter was chosen because it can reflect the overall channel performance and is the most commonly used indicator of the quality of the wireless environment. Naturally, the signal strength decreases as a vehicle moves away from the coverage area of the infrastructure node, and a handover occurs once the threshold is reached.

A vehicular user may incur multiple VHO during one MPTCP session, frequently connecting to and disconnecting from heterogeneous wireless networks across its path. The proposed PM algorithm enables dynamic creation, removal, or modification of subflow priority, providing control over handover execution in fast mobility scenarios. Unlike the default MPTCP scheme, our PM does not create any additional subflows immediately after the creation of the connection. Instead, the new subflow is created only if the path has a good signal quality. Only one subflow over each active interface towards the remote host is considered in this paper. The pseudocode in Algorithm 1 illustrates this path management procedure.

Algorithm 1: Pseudocode for PM algorithm

```
for each interface as I do
    if I is available then
        α ← smoothing factor (0 ≤ α ≤ 1)
        RSS_I(t) = (1 − α) ∗ RSS_I(t − 1) + α ∗ RSS_I
    end
end
for each subflow as S do
    I_s ← interface of S
    T_i ← threshold for I_s
    if I_s is not available then
        remove S and continue
    end
    if RSS_I(t) ≤ T_i then
        if S is active and S is not backup then
            set S as backup
        else
            if S is not active then
                remove S
            end
        end
    end
    if RSS_I(t) > T_i then
        if S is not active then
            establish subflow S
            set S as active
        else
            if S is backup then
                set S as non-backup
            end
        end
    end
end
```

For example, as the vehicle moves away from the RSU, a drop in RSS is observed, and once a particular threshold is reached, our PM algorithm will switch the subflow into *backup* mode. Once a subflow has turned into the *backup* mode, the MPTCP packet scheduler will only use it if the other subflow fails. Likewise, when a vehicle moves into the coverage area of the RSU, the PM will restore the subflow by removing the backup flag associated with it, so the packet scheduler can continue to use it for data transmission. In both cases, the protocol will send the MP_PRIO message to inform the remote host about the subflow status change.

Due to the high variance of vehicular networks, the RSS undergoes drastic fluctuations, which may lead to frequent changes in subflow operational mode. A simple low-pass filter mechanism is used to reduce the probability of the ping-pong effect when the subflow switches between two modes too frequently, resulting in unnecessary handovers. Here, the smooth factor a determines how quickly the filter responds to changes in the input signal. A higher smooth factor means that the PM will respond more slowly to changes in the RSS, resulting in a smoother output. However, too high of a smooth factor can cause the filter to become too sluggish, resulting in a delay in the output response. The choice of the smooth factor depends on the specific vehicular application and the desired balance between unnecessary handovers and maintaining a fast response time. The MAC-layer information and related attributes are expected to be retrieved from a vehicle's OBU.

V. Performance Evaluation

In our experiments, we intend that a vehicle can access a remote host via two heterogeneous wireless technologies (i.e., 802.11p and LTE), as shown in Fig. 5. The cellular network is considered to be always accessible by the vehicle, whereas a vehicle has casual ad hoc connectivity with RSUs, that are placed at fixed locations along the road.

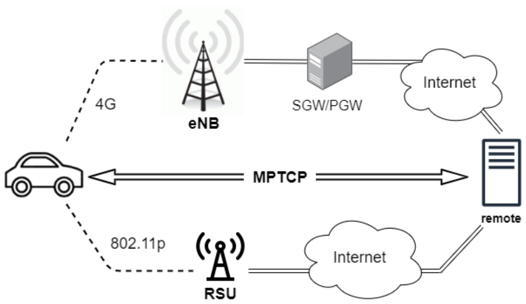


Fig. 5. Typical multipath scenario in vehicular networks.

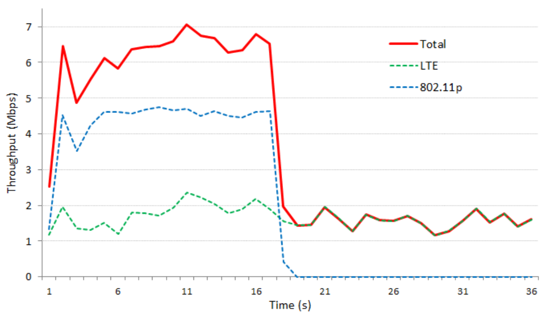


Fig. 6. MPTCP bandwidth aggregation.

The platform for experiments was configured according to frameworks presented in [12], [29] to get reliable results. The cellular 4G and ad hoc IEEE 802.11p wireless technologies are simulated by the ns-3 network simulator, whereas the realistic vehicle mobility is provided by the SUMO traffic simulator. The proposed cross-layer PM runs in an isolated namespace connected to the simulated vehicle. All experiments were performed in real-time with upstream Linux kernel v.15.7 and mptcpd v0.9. The latest version of *iptools2* were used to configure a single subflow per interface for each MPTCP connection, while the *iperf3* were used to generate traffic between the vehicle and remote host.

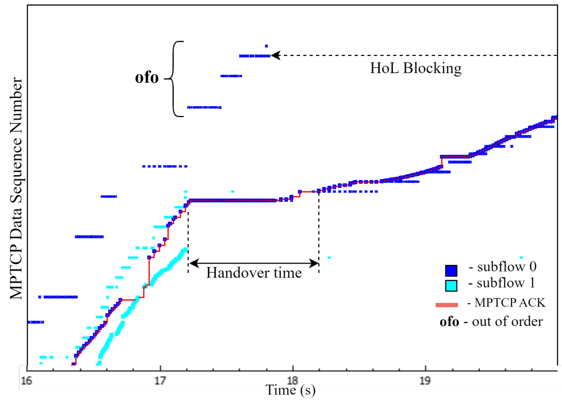
The proposed PM scheme continuously monitors the IEEE 802.11p interface status registering RSS value for every received packet from the RSU. Then, a subflow can be suspended or released based on the decision of the PM. Vehicles and RSUs run a basic road safety service defined by European Telecommunications Standards Institute (ETSI) and periodically broadcast *Cooperative Awareness Messages* (CAM) to other vehicles nearby. The CAM messages are exchanged among road users to notify their position and status in a single hop distance. The RSS samples are collected upon reception of CAM coming from the RSU. The threshold for the IEEE 802.11p interface was set to -82dBm, at which the transmission and reception of data are close to becoming unreliable.

A. Seamless Vertical Handover

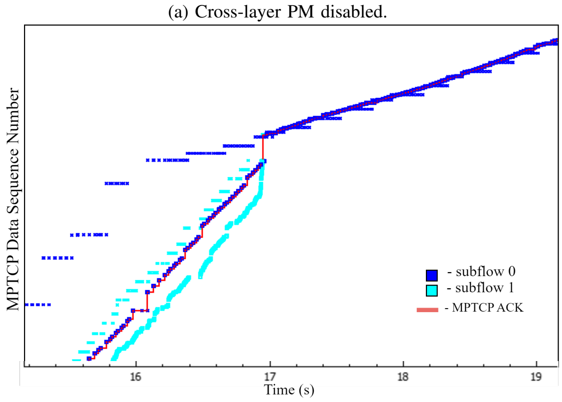
Fig. 6 displays both the average throughput per interface and the total achieved throughput during the MPTCP session. As shown, MPTCP can utilize two parallel connections to spread a single data stream and increase available bandwidth. Consequently, when both the 802.11p and LTE interfaces are used simultaneously, the achieved total throughput is higher. When the path through the ad hoc network is lost as a vehicle moves out of the RSU coverage area, the total throughput drops. However, a data stream continues to flow through the cellular network even when the 802.11p link becomes unavailable for packet transmission. Therefore, a moving vehicle can perform a VHO over cellular and roadside infrastructure without service interruption.

Upon careful inspection of the packet traces, it is evident why the default MPTCP path manager is not optimal for vehicular networks. Fig. 7 and Fig. 8 illustrate the evolution of *Data Sequence Numbers (DSN)* over time when the vehicle moves out of the coverage area of the RSU and the 802.11p link (*subflow 1*) is lost.

1) Aggregation Mode: Initially, the connection's data is distributed among all available interfaces, with most of the data sent over the ad hoc network with the lowest round-trip time (RTT). When the 802.11p link becomes unreliable and starts losing packets, the VHO process is triggered to redirect the data flow to a more reliable path (*subflow 0*). The handover process is not instantaneous, as MPTCP requires time to detect link failure and adjust the connection accordingly. This means that there can be a delay before the handover is complete, as shown in Fig. 7a. The packets through the LTE network (*subflow 0*) are arrived out of order (*ofo*), that could lead to HoL blocking with more in-flight packets accumulated at the receiving end. Since MPTCP guarantees in-order data delivery, packets received on *subflow 0* must wait for previous segments with lower sequence numbers to arrive, resulting in glitches in the application layer. This can be particularly problematic for real-time applications such as video conferencing or online gaming, where even small delays can be noticeable and disruptive to the user experience. In addition, MPTCP's scheduling mechanism continues to allocate packets to *subflow 1* since

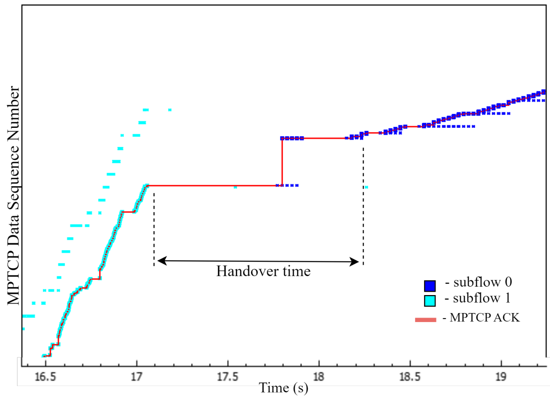


(a) Cross-layer PM disabled.

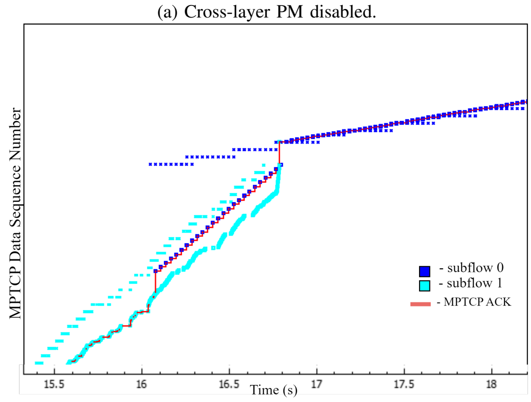


(b) Cross-layer PM enabled.

Fig. 7. Handover event in detail (aggregation mode).



(a) Cross-layer PM disabled.



(b) Cross-layer PM enabled.

Fig. 8. Handover event in detail (backup mode).

is not immediately aware of the path's status, and it may take several RTO periods before the MPTCP recognizes that the path is unavailable. As a result, a significant amount of data can be lost when an interface fails and the underlying path becomes unusable, requiring retransmission through an available subflow.

The proposed path manager aims to anticipate packet loss and suspend the unreliable subflow before the link eventually breaks. As the vehicle moves away from the RSU, a gradual drop in signal strength is observed in the transition region. The faster a vehicle moves away from the RSU, the more abrupt a drop in RSS. Thus, according to our algorithm, the subflow must be suspended as the RSS drops below the pre-defined threshold. The proposed PM with the cross-layer assistance has the ability to detect and mitigate bad path conditions by automatically removing the unreliable *subflow 1* from the ongoing connection, before the interface becomes unavailable, as shown in Fig. 7b. Furthermore, our PM can dynamically reactivate the subflow when the path quality improves, continuously optimizing network performance by adapting to changing conditions.

2) Backup Mode: When MPTCP establishes a new connection, it can generate subflows that serve as *backup* paths, activated only in the event of a regular subflow failure. Therefore, in our next experiment, we configured the endpoints to utilize a WLAN network as the primary path and switch to the cellular network only when the WLAN link fails. Fig. 8a Figure 8a illustrates how MPTCP performs the activation of the backup path without the cross-layer assistance when the primary subflow 1 fails due to the vehicle moving out of the RSU coverage. At the start of the MPTCP connection, the 802.11p interface is used as the primary path, while the LTE interface is designated as a backup. In case of poor radio conditions causing the primary path to fail, MPTCP detects the failure and switches the traffic flow to the backup path. However, this approach may not be optimal for vehicular scenarios as it results in a delay during the handover process. Furthermore, there is a period where no subflow is active, which can negatively affect service continuity. The issue is that the protocol will continue to retransmit lost data over the broken interface because the subflow was not closed properly, thus worsening the situation.

The proposed cross-layer PM scheme employs a dynamic approach to subflow management. When the RSS falls below the configured threshold, it determines that the primary subflow is underperforming and takes corrective action. Initially, the PM removes the backup status of the cellular network, and for a short period, MPTCP uses both interfaces simultaneously. If the backup subflow was not established initially, an additional subflow is created over the cellular network to ensure network redundancy. Subsequently, the PM handles the lossy path by switching it to *backup*, thereby preventing it from being used for data transmission. Additionally, the underperforming subflow can be removed from the MPTCP session if necessary. By completing the VHO procedure while the ad hoc network is still available, the proposed scheme significantly reduces handover delay, as demonstrated in Fig. 8b.

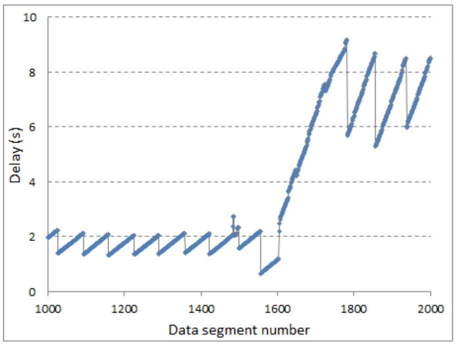
The results show that the proposed scheme can effectively ensure continuous V2I connectivity, even in the presence of fast mobility scenarios and heterogeneous wireless environments. The scheme's ability to prevent handover delay and HoL blocking further underscores its potential to enhance network performance and provide a seamless user experience. Overall, these findings support the use of the cross-layer scheme as a viable solution for improving connectivity and network performance in challenging wireless environments.

B. Application Delay

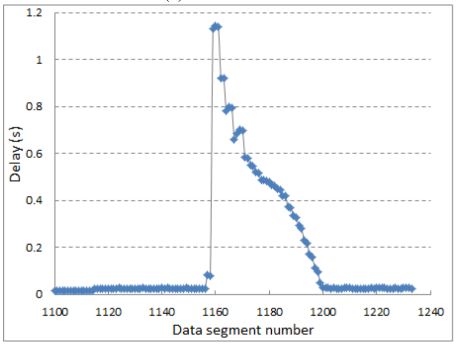
As we saw in previous sections, the multipath communication can cause significant data transport issues in vehicular networks. The applications with real-time requirements, such as media streaming, are most affected by frequent handovers and path variation, resulting in high data delivery latency and perceivable jitter. To monitor the variation in application delay, we generate traffic and include a timestamp in each data segment. The end-to-end delay is measured from the time the packet leaves the source application to when it arrives at the destination application, and upon reception of each data segment, we record its arrival time. By tracking the difference between these timestamps, we can determine the end-to-end application delay. Our emulation setup uses the same hardware clock on each communication endpoint, ensuring accurate measurements.

Fig. 9 presents a comparison of MPTCP performance in terms of application delay during the handover in a vehicular scenario for kernel v15.5, kernel v15.7, and our proposed cross-layer PM scheme. The results demonstrate that the application layer delay significantly increases when the handover occurs with a standard MPTCP scheme. However, when using our proposed PM scheme, the delay is significantly reduced, indicating improved network performance and reduced service disruption during handovers. Following the failure of the IEEE 802.11p link, the end-to-end delay can increase significantly, reaching up to 9 seconds on kernel v15.5 (as illustrated in Fig. 9a). This high application delay is primarily attributed to the very basic packet scheduler algorithm that fails to consider heterogeneous network characteristics. As a result, we can observe a very poor performance in terms of data delivery speed to the application, as out-of-order packets suffer from prolonged queuing times before being delivered to the application. The MPTCP packet scheduler was improved with better HoL-blocking estimation in Linux kernel v15.7. A scheduler algorithm was modified to handle heterogeneous paths better, as represented in Fig. 9b. Still, connection glitches occur when the vehicle moves out of coverage of RSU, and a handover is experienced.

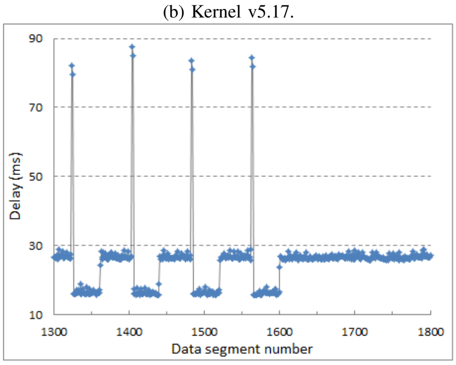
Fig. 9c illustrates the end-to-end delay when our PM is enabled. The figure illustrates that both subflows are utilized concurrently to transmit data, leading to a negligible variation in application delay until the transmission of data segment number 1600. The slight increase in delay can be attributed to the heterogeneity of the network paths. During VHO, the proposed cross-layer PM scheme can effectively minimize the end-to-end delay by suspending the underperforming subflow before link failure occurs. This ensures that the end user is not impacted by any disruptions, and network performance is optimized. Specifically, in the evaluated scenario, the average application delay at the moment of handoff decreases by one order of magnitude, from a range of 1 to 10 seconds to 50 to 80 milliseconds. These results confirm the efficacy of our proposed method in enhancing MPTCP performance and improving network QoS in vehicular scenarios.



(a) Kernel v5.15.



(b) Kernel v5.17.



(c) Cross-layer PM enabled.

Fig. 9. End-to-end application delay.

C. Recovered Path

For effective V2I communication, vehicles on the move must perform seamless VHO between heterogeneous networks, when switches its connectivity from one technology to another. Each byte of data sent in an MPTCP connection has an associated sequence number so that an available subflow can recover lost packets. However, the transport service on each subflow is provided by the regular TCP, that will start a retransmission timer after sending a packet of data. If a

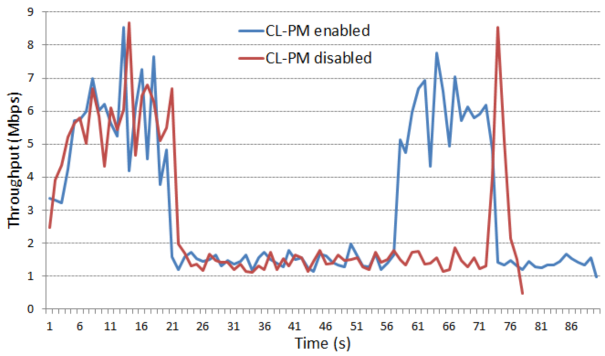


Fig. 10. MPTCP throughput with recovered path.

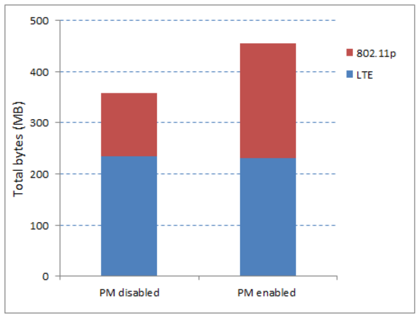


Fig. 11. Total transferred bytes.

sender does not receive an acknowledgment before the timer expires, it will assume the segment has been lost and retransmit it. When the sender misses too many acknowledgments, the MPTCP assumes that packet loss has appeared on the path somewhere between the vehicle and the remote host and decides to stop sending data over that path. After some time, the protocol starts testing the path availability by sending one packet over the same path again. After the first attempt, the exponential backoff is applied to the RTO, doubling the timeout value between successive retransmissions. Once the 15th retry expires (by default), the subflow is considered broken.

Section III gives insight into the problem of the under-utilization of a recovered path in vehicular networks. With regular MPTCP, there is nothing that the application can do to start using the recovered path other than waiting for successful retransmission. This unnecessary delay after the path recovers causes the vehicle to lose a significant fraction of available connection time. As the vehicle comes in or out of range of a particular network, additional subflows can be created or removed dynamically. The proposed cross-layer PM can detect path loss or availability much faster due to signal strength monitoring of accessible networks. For instance, when the vehicle moves into the network's coverage area, it can utilize the wireless interface as soon as RSS rises above the predefined threshold. In long-lived data flows, this can result in throughput gains since there is no idle period after the path recovers, as shown in Fig. 10. In this example, the additional subflow is created as soon as the IEEE 802.11 interface detects an RSU. Similarly, the subflow is removed from the MPTCP connection as the RSS falls below the receiver sensitivity threshold.

D. Long-lived Simulation

The cross-layer PM has been evaluated in a Manhattan-like urban scenario with bidirectional streets and one lane per driving direction. This generic urban scenario has been chosen to analyze the effectiveness of the proposed algorithm for V2I communication with different deployments of infrastructure nodes. All vehicles were equipped with LTE and IEEE 802.11p interfaces, running CAM service to exchange real-time information. The scenario includes a single LTE base station covering the simulated area, with a certain number of RSUs uniformly distributed at random intersections. A vehicle can lose and regain connection when it crosses through different RSUs while moving on the road. There are regions around infrastructure nodes where the wireless networks exhibit poor link quality due to bad radio conditions. Therefore, the connection time between the vehicle and RSU is changes according to the vehicle's speed and traffic conditions.

We conducted two identical simulations to compare the effectiveness of our cross-layer PM with the default MPTCP scheme. Each simulation lasted for 30 minutes and involved uploading files of varying sizes from the vehicle to the server. In one test, we disabled our PM, while in the other, we enabled it. After completing the simulations, we compared the total amount of bytes transferred by each interface, as shown in Fig. 11. It is evident from the results that our proposed scheme, aided by its efficient handoff mechanism, can utilize the available bandwidth more effectively than regular MPTCP. Furthermore, other experimental outcomes demonstrate that our PM substantially enhances the quality of experience for vehicular users in terms of achievable throughput, packet loss, retransmissions, and duplicate packets.

VI. Conclusions and Future Work

Multipath communication provides many advantages, such as improved connection reliability and increased throughput. This article discusses the MPTCP as a potential solution to address service continuity in highly mobile vehicular networks. A thorough investigation of MPTCP and its limitations in heterogeneous vehicular networks was conducted, highlighting the issues associated with traditional path management strategies. To address these issues, this paper proposes a PM algorithm operating at the application layer, which can dynamically adjust subflow usage under the frequent network topology changes. In our studies, we focused on the practical V2I scenario, wherein a vehicle is equipped with both ad-hoc and cellular interfaces, and is capable of communicating with a remote host while moving on the road. The proposed PM scheme utilizes cross-layer assistance to enable faster detection of path failure and recovery compared to the standard MPTCP. As a result, our PM method has been shown to yield a significant improvement in MPTCP performance in vehicular networks. By implementing dynamic and proactive subflow management techniques, our PM can provide a reliable and uninterrupted connectivity, while also optimizing the use of available network links. The result is a more effective exploitation of network resources, which can greatly enhance the overall performance of the system. In the future, we plan to enhance our PM algorithm to support multiple parameters in the decision process, including signal strength, application requirements, vehicle location information, and speed.

ACKNOWLEDGEMENT

The authors wish to acknowledge the Portuguese funding institution "FCT - Fundação para a Ciência e a Tecnologia" for supporting their research. Programa doutoral AESI, bolsa de investigação PD/BDE/150506/2019, cofinanciada pelo FSE através do Programa Operacional Regional Norte.

REFERENCES

[1] J. Lianghai, A. Weinand, B. Han, and H. D. Schotten, "Multi-RATs support to improve V2X communication," in *IEEE Wireless Communications and Networking Conference, WCNC*, vol. 2018-April, 2018, pp. 1–6.

[2] E. Ndashimye, S. K. Ray, N. I. Sarkar, and J. A. Gutiérrez, "Vehicle-to-infrastructure communication over multi-tier heterogeneous networks: A survey," *Computer Networks*, vol. 112, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2016.11.008

[3] M. T. Suer, C. Thein, H. Tchouankem, and L. Wolf, "Multi-Connectivity as an Enabler for Reliable Low Latency Communications - An Overview," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 1, pp. 156–169, 2020.

[4] A. Zekri and W. Jia, "Heterogeneous vehicular communications: A comprehensive study," pp. 52–79, jun 2018.

[5] S. R. Pokhrel, N. Kumar, and A. Walid, "Towards Ultra Reliable Low Latency Multipath TCP for Connected Autonomous Vehicles," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 8175–8185, 2021.

[6] L. Chao, C. Wu, T. Yoshinaga, W. Bao, and Y. Ji, "A brief review of multipath tcp for vehicular networks," *Sensors*, vol. 21, no. 8, pp. 1–34, 2021.

[7] N. Williams, P. Abeysekera, N. Dyer, H. Vu, and G. Armitage, "Multipath TCP in Vehicular to Infrastructure Communications," *Grenville Armitage Centre for Advanced Internet Architectures Tech. Reports A. 140828*, no. August, p. 15, 2014.

[8] I. Khan and K. Chen, "EBA: Efficient Bandwidth Aggregation for Connected Vehicles with MPTCP," *IEEE Internet of Things Journal*, 2021.

[9] Q. De Coninck and O. Bonaventure, "Observing network handovers with multipath TCP," *SIGCOMM 2018 - Proceedings of the 2018 Posters and Demos, Part of SIGCOMM 2018*, pp. 54–56, 2018.

[10] A. Ford, C. Raiciu, M. J. Handley, and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses," RFC 6824, Jan. 2013. [Online]. Available: https://www.rfc-editor.org/info/rfc6824

[11] A. Ford, C. Raiciu, M. J. Handley, O. Bonaventure, and C. Paasch, "TCP Extensions for Multipath Operation with Multiple Addresses," RFC 8684, Mar. 2020. [Online]. Available: https://www.rfc-editor.org/info/rfc8684

[12] V. S. Hapanchak and A. D. Costa, "Emulation of multipath solutions in heterogeneous wireless networks over ns-3 platform," in *Simulation Tools and Techniques*, D. Jiang and H. Song, Eds. Cham: Springer International Publishing, 2022, pp. 3–18.

[13] ——, "Design and evaluation of a cross-layer MPTCP path manager for vehicular networks," in *30th International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2022, Split, Croatia, September 22-24, 2022*. IEEE, 2022, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9911227

[14] "Ieee standard for local and metropolitan area networks–part 21: Media independent services framework," *IEEE Std 802.21-2017 (Revision of IEEE Std 802.21-2008 as amended by IEEE Std 802.21a-2012, IEEE Std 802.21b-2012, IEEE Std 802.21c-2014, and IEEE Std 802.21d-2015)*, pp. 1–314, 2017.

[15] 3GPP, "System architecture for the 5g system, document ts 23.501, v16.4."

[16] IETF, "Path aware networking research group," https://irtf.org/panrg , 2022, [Online; accessed December-2022].

[17] H. Sinky, B. Hamdaoui, and M. Guizani, "Seamless Handoffs in Wireless HetNets: Transport-Layer Challenges and Multi-Path TCP Solutions with Cross-Layer Awareness," *IEEE Network*, vol. 33, no. 2, pp. 195–201, 2019.

[18] H. R. Q. Vxs, L. P. Xqj, K. H. Q. Ulfk, R. Q. Dqg, D. Q. J. Rq, W. Olplwdwlrq, E. Q. Lqj, W. K. H. Rwkhu, H. Q. G. Xvlqj, E. X. S. W. R. Shufhqw, and A. M.-p. Tcp, "Cross-layer path management in Multi-path transport protocol for mobile devices," pp. 1815–1823, 2014.

[19] B. Liao, G. Zhang, Q. Wu, Z. Li, and G. Xie, "Cross-layer path selection in multi-path transport protocol for mobile devices," *arXiv*, 2020.

[20] D. Zhu, C. Xu, J. Qin, Z. Zhou, and J. Guan, "Mobility-aware multimedia data transfer using Multipath TCP in Vehicular Network," *2017 13th International Wireless Communications and Mobile Computing Conference, IWCMC 2017*, pp. 1041–1046, 2017.

[21] H. Wu, G. Caso, S. Ferlin, O. Alay, and A. Brunstrom, "Multipath Scheduling for 5G Networks: Evaluation and Outlook," *IEEE Communications Magazine*, vol. 59, no. 4, pp. 44–50, 2021.

[22] L. Ding, Y. Tian, T. Liu, Z. Wei, and X. Zhang, "Understanding commercial 5G and its implications to (Multipath) TCP," *Computer Networks*, vol. 198, 2021.

[23] M. Karimzadeh, L. Valtulina, H. Van Den Berg, A. Pras, P. G. Ortiz, and R. Sadre, "MultiPath TCP to support user's mobility in future LTE network," *Proceedings - WMNC 2017: 10th Wireless and Mobile Networking Conference*, vol. 2018-Janua, pp. 1–8, 2017.

[24] B. Y. Kimura, D. C. Lima, and A. A. Loureiro, "Packet Scheduling in Multipath TCP: Fundamentals, Lessons, and Opportunities," *IEEE Systems Journal*, vol. 15, no. 1, pp. 1445–1457, 2021.

[25] https://github.com/multipath-tcp/mptcp_net-next, 2020, [Online; accessed December-2022].

[26] S. Ferlin, O. Alay, O. Mehani, and R. Boreli, "BLEST: Blocking estimation-based MPTCP scheduler for heterogeneous networks," *2016 IFIP Networking Conference (IFIP Networking) and Workshops, IFIP Networking 2016*, pp. 431–439, 2016.

[27] B. Hesmans, G. Detal, S. Barre, R. Bauduin, and O. Bonaventure, "SMAPP: Towards smart multipath TCP-enabled applications," *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies, CoNEXT 2015*, 2015.

[28] O. Othman, "Using Upstream MPTCP in Linux Systems," Intel, Tech. Rep., 2020.

[29] V. S. Hapanchak and A. D. Costa, "Emulation of Multi-Connectivity in Hybrid Vehicular Networks," *International Conference on Electrical, Computer, and Energy Technologies, ICECET 2022*, no. July, 2022.

Vadym S. Hapanchak is a Ph.D. student in Advanced Engineering Systems for Industry (AESI), promoted by the University of Minho in cooperation with Bosch Car Multimedia Portugal. His received the M.S. degree in Telecommunications and Computer Science from the University of Minho (Portugal) in 2018. His current research interests include V2X communication, Intelligent Transportation Systems and multipath communications.

António D. Costa is an Assistant Professor at Department of Informatics, University of Minho, Portugal, where he develops teaching and research activities in the fields of Computer Networks and Computer Communications since 1992. As a researcher, he currently integrates the Computer Communications and Networks (CCN) research group, at Centro Algoritmi, School of Engineering, University of Minho.