Accepted Manuscript

Hybrid error recovery protocol for video streaming in vehicle ad hoc networks

Sofiane Zaidi, Salim Bitam, Abdelhamid Mellouk
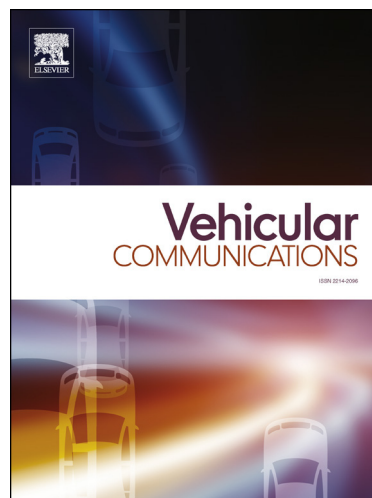
Please cite this article in press as: S. Zaidi et al., Hybrid error recovery protocol for video streaming in vehicle ad hoc networks, *Veh. Commun.* (2018), https://doi.org/10.1016/j.vehcom.2018.01.002

Hybrid Error Recovery Protocol for Video Streaming in Vehicle Ad hoc Networks

Sofiane Zaidi[a,b], Salim Bitam[a], Abdelhamid Mellouk[c,*]

[a]*University of Biskra, LESIA Laboratory, Department of Computer Science, Algeria*
[b]*University of Souk-Ahras, Department of Computer Science, Algeria*
[c]*University of Paris-Est Créteil (UPEC), IUT CV-NT Dept/ESIPEC-ItS/LISSI-TincNET, France*

Abstract

Recently, video streaming in Vehicular Ad hoc NETworks (VANETs) is considered as one of the most important challenge in vehicular communication because of the high packet loss rate and the increased transmission delay due to the highly dynamic of VANET topology. This specificity makes difficult to apply the conventional transport protocols such as UDP and TCP to video streaming over VANET. To deal with these limits and to ensure a high video quality at the end receiver, we propose in this paper a Hybrid Error Recovery Protocol (HERP) for video streaming in VANETs. The proposed protocol integrates the Sub-Packet Forward Correction (SPFEC) mechanism to recover the uniform transmission errors and the re-transmission technique to recover burst errors mainly due to the network congestion and route disconnection. In order to avoid the network overload and to reduce the transmission delay, HERP adapts dynamically the redundancy rate, retransmission limit and transmission rate according to the network condition measured by the Bit Error Rate and the network load indicated by queue length of intermediate vehicles. HERP uses the reporting technique to control the network condition and the network load. To improve the video streaming quality, HERP suggests an unequal protection of video frames type (i.e. I, P, B frames). The experimental results show that HERP achieves significant improvements of transmitted video compared to native UDP protocol and UDP based on SPFEC.

Keywords: Vehicular Ad hoc NETworks, Video Streaming, Sub-Packet Forward Error Correction, Retransmission

1. Introduction

Vehicular Ad hoc NETworks (VANET) is an emergent technology attracting currently the attention of industrial and research community in different topics such as electronics, network, security, transportation, automotive, etc. VANET researches seeking to make the vehicles more intelligent mainly in the aim to reduce road traffic accidents which are increased dramatically at the present time due to the high number of vehicles on the road [1]. The world heath organization reported based on the information from 180 countries that the number of road traffic deaths has reached 1.25 million per year [2]. In order to guarantee a road safety, traffic management and the comfort to the users, various applications are designed to be used in VANET, we mention traffic monitoring, driving assistance, sharing music and videos between passengers, etc. [3, 4].

VANET is composed of moving vehicles and fixed Road Side Units (RSUs) placed on the road edges to achieve specifics services such as sending periodic messages about the traffic conditions to the vehicles, collecting and analyzing traffic data provided by vehicles, supporting seamless communication between the

---

*Corresponding author
Email addresses: `s.zaidi@univ-soukahras.dz` (Sofiane Zaidi), `s.bitam@univ-biskra.dz` (Salim Bitam),
`mellouk@u-pec.fr` (Abdelhamid Mellouk)

vehicles, etc [5, 6]. The vehicle communicates with the others or with the RSUs in a single or multi-hop communication using a Dedicated Short Range Communication (DSRC) specified by the U.S Federal Commission Communication where 75 MHz of spectrum in the 5.9 GHz band is allocated to be used for the three vehicular communication modes; Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I) and hybrid mode (V2X). Each vehicle is equipped with On-Board Unit (OBU), Global Positioning System (GPS), Event Data Recorder (EDR) and a set of sensors in order to detect and to communicate traffic status and data [7].

VANET is a special class of Mobile Ad hoc NETwork (MANET) [8]. Like MANET, in VANET the vehicles and RSUs use the wireless channels to exchange data between them. The vehicles high mobility, the wireless link volatility and rapid change of network topology produce high number of lost packets. Also, VANET suffers from the congestion problem which forces the vehicles to drop its packets when the network is overloaded especially in high density environments like urban areas.

VANET can support several applications, which can be classified into two categories: safety oriented applications and non-safety oriented applications [9]. The first category aims to avoid the risk of accidents in the road by generating and transmitting warning messages such as in the case of intersection collision and accidents. The second one ensures the traffic control and management like information given by a RSU about road congestion. Non-safety applications allow also to passengers some conform and infotainment services such as the internet access, mobile e-commerce. In both safety and non-safety applications, development of techniques of transmitting real-time video (also known as the video streaming) have a great interest by academia and industries in reason to enhance to road safety and traffic efficiency in addition to response to drivers and passengers digital needs [10].

1.1. Motivation

The video streaming in VANET can satisfy the car driver and passenger requirements by providing a clear vision on traffic or any digital data rather than textual messages. For instance, each vehicle can use its embedded camera to capture some situation of the road traffic or any event that occurred in the road like accidents, traffic congestion, parking availability, festival event, etc. and then it warns the other vehicles by a multi-hop dissemination of this captured video. The camera can be also installed at the RSUs at road intersection to facilitate and accelerate the transmission of captured data to the concerned destinations (e.g. police cars or stations, hospital, emergency preparedness, etc.). Another example from telemedicine domain, the video captured by a vehicle or RSU about an accident can be forwarded toward the hospital or the nearest ambulance to identify and diagnostic the injuries situation by distant doctors. VANET provides also some video streaming services to enhance the passenger comfort like playing games between passengers, receiving nearest restaurant and hotel information, video conferencing between passengers, etc.

The high dynamic of VANET topology is a challenging for the video streaming, because it affects and ruptures the communication path between the sender and the receiver when the video is transmitted. More-over, the network congestion and the transmission errors are considered as other issues, which decrease the video quality [11, 12]. Several works have been recently proposed to tackle these issues. Some of these studies applied different video coding and error resiliency techniques at the application and transport layers to improve the video streaming quality. Many other works select reliable paths to disseminate video packets at the network layer to deal with the VANET challenges such as vehicles mobility. In the literature, we can find other studies that adapt some video transmission parameters like the size of contention window to enhance the video quality. In this research domain, there are some proposed video streaming solutions based on traditional transport protocols like User Datagram Protocol (UDP) or Transmission Control Protocol (TCP), which are designed originally for wired networks. However, the UDP based solutions did not adopt any error recovery mechanism then the video quality at the end user is affected [13]. Additionally, TCP based solutions are not suitable for video streaming applications because of its reliability mechanism to recover lost packets which can increase enormously the video transmission delay [14, 15]. Also, most of error resiliency works for video streaming in VANET not deal together the three causes of lost packets: transmission errors, congestion and route disconnection. It is worth noting that many of these works applied the FEC mechanism, which is based on the redundancy technique to overcome the erroneous packets. However, FEC mechanism suffers from the network overload problem due to the limited network resources.

In this paper, we propose a new error recovery protocol named Hybrid Error Recovery Protocol (HERP), to recover the lost video packets due to the transmission errors, congestion and route disconnection with a reasonable transmission delay, then HERP can guarantee a high video quality at the end receiver in terms of QoS and QoE metrics. HERP combines the SPFEC mechanism with the retransmission technique. The SPFEC mechanism considered as an error recovery mechanism based on the redundancy technique applied to recover the lost packets due to transmission errors. On the one side, HERP applies the SPFEC mechanism aiming at providing more protection compared to the FEC mechanism then the network overload is reduced [16]. On the other side, the proposed HERP uses the retransmission technique in order to recover the lost packets caused by the congestion or by the transmission route disconnection. Furthermore, HERP applies the unequal protection of video frames (I, P and B) coded according to MPEG-4 standard, in which the protection degree of the video frames is given according to the frame types to improve the video quality. HERP adapts dynamically their parameters (i.e. redundancy rate, retransmission limit, transmission rate) according to the network condition and network load. HERP is also based on the reporting technique, which represents a dynamic feedback mechanism between the receiver and sender vehicles of the video to control the network condition and the network load.

The rest of this paper is organized as follows. Section 2 outlines the related work for video streaming in VANET. Section 3 describes the design of the proposed Hybrid Error Recovery Protocol (HERP). Section 4 presents the experimental study and discussed the reached results. Finally, Section 5 concludes the paper with a summary and proposes some future research directions.

2. Related work

This section presents the most significant video coding standards as well as the error correction mechanisms used in VANET. In addition, this section reviews the different research activities proposed for VANET video streaming.

2.1. Video coding standards

2.1.1. Motion Picture Expert Group (MPEG-4)

MPEG is a video coding standard used by many mobile networks for video streaming compression. Many versions of MPEG multimedia standard are introduced such as MPEG-2, MPEG-4, MPEG-7 and MPEG-21 [17]. We have chosen in our work to use the MPEG-4 for video streaming coding in VANET, because MPEG-4 is the widely standard version of MPEG supported by the majority of multimedia applications, and it produces a good video quality in mobile networks [18].

Based on MPEG-4 standard, the video is encoded as n Groups of Pictures (GoPs), where each GoP is composed of three kinds of: Intra-coded frame (I-frame), Predictive-coded frame (P-frame) and Bi-directionally predictive-coded frame (B-frame). In the same GoP, I-frame is the most important frame compared to P-frame, which is in its turn more important than B-frame. The encoding and decoding of P-frame require previous I-frames and/or P-frames of the same GoP. Also, the encoding and decoding of B-frame require previous and follows I-frames and/or P-frames of the same GoP [19].

2.1.2. H.264/AVC

H.264/AVC (Advanced Video Codec) standard is based on Flexible Macroblock Ordering (FMO) coding. FMO divided the video frame into a set of slices, each slice consists of a set of Macro-Blocks (MBs), the Macro-Block is an elementary unit of the slice. The spatial and temporal concealment techniques allow the recovery of lost slices of any frame [20].

2.1.3. H.265/HEVC

H.265/HEVC (High Efficiency Video Coding) [21] is a new video compression standard, which reduces the bandwidth requirements by 50% compared to H.264/AVC standard with keeping the same quality of video in terms of PSNR. Like H.264/AVC, H.265/HEVC is based on the encoding of the video frame into a set of slices, and uses the spatial and temporal concealment techniques for recovery the lost slices, which are higher than those of H.264/AVC.

2.2. Error resiliency techniques

Many VANET video streaming methods use error recovery mechanisms and techniques to overcome the erroneous packets. We classify these video error recovery techniques over VANET in three classes: redundancy-based techniques, retransmission-based techniques and concealment-based techniques.

2.2.1. redundancy-based techniques

In these techniques, the sender adds duplicate data with the original data and transmits it to the receiver, when this latter receives all data, it can recover the lost data using its duplicate. There are some error resiliency techniques based on the redundancy like Forward Error Correction (FEC), interleaving, and Erasure Coding (EC). Redundancy-based techniques increases the packet delivery ratio, however it led to an increased network overload because of the high number of transmitted packets, especially when the network is dense or with high transmission rate.

Forward error correction (FEC). FEC [22] is an error resiliency mechanism, which recovers the lost packets at the receiver level based on the redundancy technique, without any interaction or feedback with the sender of these packets. FEC is based on the idea of encoding the video as a set of blocks of a fixed size n, where each block is composed of k source packets and $(n\text{-}k)$ redundant packets. The decoding of k source packets of any block needs the good reception of k packets of this block. The network overload is considered as a limit of FEC due to the redundant packets. Moreover, this mechanism can recover only the uniform errors (i.e. errors occurring with uniform distribution independently in a sequence of packets), therefore FEC cannot recover the burst errors (i.e. consecutive lost packets). Sub-Packet Forward Error Correction (SPFEC) is a special case of FEC, in which the packet is a block of original sub-packets and redundant sub-packets.

Interleaving. Interleaving [23] is a recovery technique which transforms the burst frame errors into a set of uniform frame errors. Using interleaving, the sender changes the order of original frames and adds redundant frames between them. After receiving the frames, the receiver returns the original frames in its original order.

Erasure Coding (EC). Like FEC, Erasure Coding (EC) [24] is an error resiliency mechanism based on the redundancy technique. With EC, the sender adds a set of redundant packets, representing a combination of original packets. EC applies certain coding techniques to perform this combination like XOR and linear coding. The receiver can decode the original packet successfully, by means of the redundant packets.

2.2.2. Retransmission-based techniques

The retransmission of packets is an error resiliency technique based on the following principle; when a packet is lost at the receiver level, this latter sends a negative acknowledgment to the sender requesting the retransmission of the lost packet. The retransmission reduces the bandwidth overload compared to the redundancy however, the transmission delay could be increased.

2.2.3. Concealment-based techniques

The concealment is another error resiliency technique conceived to recover lost regions of frame from other frames within the same video. This technique is applied at the decoder without any feedback with the video encoder. The concealment decreases the bandwidth overload and the delay because it recovers lost packets without any retransmission or redundancy of video packets, nevertheless the concealment produces some artefacts in the displayed video.

2.3. Video streaming schemes proposed for VANET

We classified the video streaming works in VANET into three main categories: video coding and error resiliency category, cooperative relays based category and adaptive category. All these works aim at improve the video streaming quality at the end receiver, in order to give accurate information to drivers and passengers, as shown in figure 1.
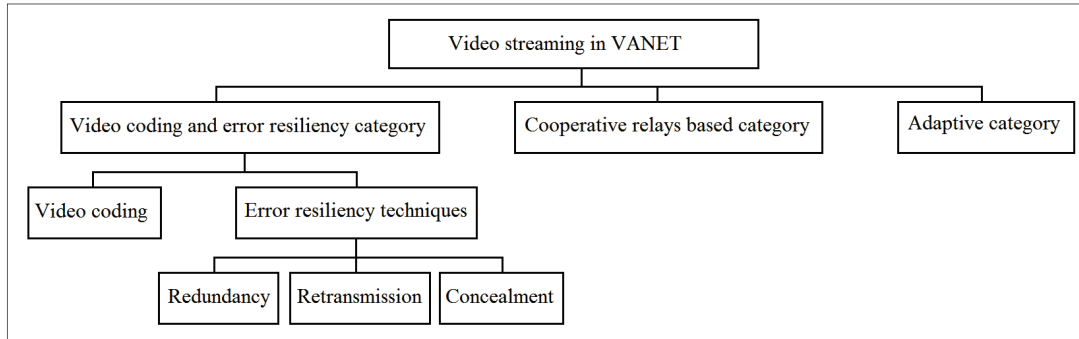
4

Figure 1: Video streaming works in VANET

2.3.1. Video coding and error resiliency category

This approach applied different mechanisms of video coding and video streaming error resiliency, mainly at the application and transport layers level.

Video coding. In the literature of VANET video streaming, many works used and evaluated different video coding standards and techniques. Torres et al. in [25] achieved the first evaluation and comparison between H.264 and H.265 video coding standards under highway VANET. This research activity proved that H.265 is more robust than H.264 when the packet loss level is high. Pinol et al. evaluated in [26] HEVC video coding standard for video streaming in VANET according to lost packet problem. In the basis of performed analysis, this study found a relationship between the quality of video and some factors like: the number of slices per frame, number of transmitted packets per second, size of packet and number of transmitted I-frames per second. We mention that [25] and [26] did not consider the transmission delay as a video evaluation metric. In [27] Vineeth and Guruprasad analyzed the transmission delay and jitter of video dissemination in VANET using the Network Coding (NC). This study proved that the transmission of video encoded based on NC is influenced by the mobility model and the vehicles density. Nevertheless, this work did not analyze other QoS parameters of video steaming like packet loss ratio.

Error resiliency techniques. In this category, the video streaming studies use error recovery mechanisms and techniques such as redundancy-based techniques, retransmission-based techniques and concealment-based techniques aiming at overcoming the erroneous packets.

A lot of video streaming works in VANET were based on redundancy-based mechanisms and techniques (FEC, interleaving and Erasure Coding). Immich et al. in [28] proposed a self-adaptive FEC-based and QoE-driven mechanism called ShiledHEVC to improve the resilience of H.265 real-time video transmission against packets loss. Using ShiledHEVC, the relay vehicles use a Hierarchical Fuzzy System (HFS) to adjust dynamically the more suitable amount of redundancy based on video characteristics and network condition. ShiledHEVC provides better video QoE for end users and reduces the network overhead. In this class, we can find an Enhanced Adaptive Sub-Packet Forward Error Correction mechanism (EASP-FEC) for video streaming in VANET proposed by Zaidi et al. in [29]. With EASP-FEC, the sender and relay vehicles calculate the number of redundant sub-packets in function of network condition, network load and the importance of video frame types (I, P, B). EASP-FEC recovers the erroneous sub-packets, avoids the congestion and provides high protection for the most important video frames. Zaidi et al. in [30] proposed an Enhanced User Datagram Protocol (EUDP) for video streaming in VANET. EUDP is based on Sub-Packet Forward Error Correction (SPFEC) and the unequal protection of video frame types (i.e. I, P, B) to improve the video streaming quality. The simulation proved that EUDP achieves higher QoS and QoE video quality in real VANET scenarios, compared with UDP and EUDP without unequal protection of video frame types (EUDP-E). ShiledHEVC, EASP-FEC and EUDP recover the uniform errors, but they cannot deal with the

5

problem of video burst errors in VANET. In this class, Bucciol et al. [31] presented a FEC and Interleaving Real Time Optimization (FIRO) algorithm to recover the uniform video errors by FEC mechanism and the burst video errors by interleaving technique. FIRO adapts dynamically the parameters of FEC and interleaving to be suitable with the channel parameters and it uses the reporting technique to estimate the loss ratio of transmission channel. FIRO provides better performance in terms of Packet Loss Rate and Peak Signal-to-Noise Ratio (PSNR) of transmitted video compared to FEC and interleaving techniques. Quadros et al. [32] integrated the interleaving technique to their proposed QOE-aware and driven REceiver-based (QORE) mechanism, in order to handle the problem of burst losses at the application layer. Rezende et al. in [33] investigated the use of EC for video streaming in VANET. The authors have conducted a comparison between EC using Random Linear Coding (RLC) and EC using XOR based coding. This work demonstrated that the use of EC solves the lost packets problem. Also, the results reached showed that EC using XOR based coding achieved higher delivery ratio and lower end-to-end delay compared to those obtained by EC based on RLC, with the same redundancy rate. Note that EC could be compared with others video error resiliency techniques in VANET to prove the effectiveness of EC. Mammeri et al. proposed in [34] an integration of Erasure Coding with Real-time Transport Protocol (EC-RTP) to address the high packet loss rate problem of VANET video streaming. The work developed two converters to adapt RTP to VANET. The first one converts the RTP packets to EC-RTP packets, which are transmitted in the network and the second one converts the EC-RTP packets to RTP packets, which are redirected to RTP player. EC-RTP decreases the packet loss and achieves higher video quality in term of PSNR compared with RTP. In this study, only a single hop was considered, a more experiments of EC-RTP must be performed for multi-hop scenario.

There are some video streaming works in VANET that applied the retransmission technique. Xie et al. in [35] proposed a multi-path solution based on node disjoint and link disjoint algorithms for video streaming in VANET. The proposed solution separates the transmission of I-frames from other frames. In this paper, the TCP is used to transmit I-frames, where the UDP is performed to transmit P-frames and B-frames. The simulation results indicated that this solution provides a higher video quality and receiving data rate comparing to FEC and UDP. The work suffers from an additional delay produced by the TCP protocol, which is not suited for the real-time video streaming. In [36], Xie et al. proposed another recovery error mechanism for real-time video streaming in VANET, named Multi-channel Error Recovery Video Streaming (MERVS). MERVS transmits I-frames through a reliable channel using TCP protocol and the other types of frames are sent through unreliable channel using UDP protocol. To decrease the transmission delay, the authors proposed an integration of MERVS with three techniques namely Priority Queue, Quick-Start and Scalable Reliable Channel (SRC). Priority Queue is adopted to ordering the video messages in the waiting queue based on their sequence ID. The Quick-Start allows sender, receiver and rooters to negotiate the transmission rate in order to maximize the throughput. Scalable Reliable Channel is proposed to schedule the video messages and balance the transmission between the two channels. The simulation results showed that MERVS provides higher transmitted video quality compared to FEC and RTP/UDP. The experiments results also showed that MERVS with Priority Queue, Quick Start and SRC provide a lower transmission delay compared to TCP.

Some works in mobile ad hoc networks were proposed to reduce the transmission delay. Khoukhi and Cherkaoui proposed in [37] an intelligent quality of service (QoS) model, named GQOS. GQOS differentiates the different services in mobile ad hoc networks based on neural networks, in order to satisfy some QoS requirement, specifically to minimize the end-to-end delay. GQOS is composed of two parts: kernel plan and intelligent learning plan. The first part uses some techniques of detecting and recovering the QoS violation, in order to guaranteeing QoS requirements. The second part is responsible of GQOS kernel training based on multilayered feedforward neural networks, to reduce the processing time at the nodes level, which decreases the end-to-end delay. The simulation results have proved that GQOS provides higher QoS performances in terms of end-to-end delay, average throughput, in mobile ad hoc network with lower and medium mobility rate. It remains to enhance GQOS performances at higher mobility rate. In [38], Khoukhi and Cherkaoui proposed a congestion control based on fuzzy logic approach in mobile ad hoc networks, named FuzzyCCG. In the aim of reducing the end-to-end delay and increasing the throughput, FuzzyCCG selects the buffer threshold at the node according to a fuzzy logic for data packet discarding. The experimental results have

been proved the QoS effectiveness of FuzzyCCG under various mobility scenarios and traffic conditions. Notice that further improvements could be devoted to adapt GQOS and FuzzyCCG for video streaming in VANET in order to reduce the transmission delay.

In the literature, there was a few video streaming works in VANET based on concealment recovery technique. The research activity published in [26] implemented a simple error concealment method for video streaming in VANET, in which when one slice of frame is missing at the decoder, this latter copies the corresponding slice of the previously decoded frame.

2.3.2. Cooperative relays based category

Considered as a proposal at level of network layer, this category aims at selecting the relay vehicles between the source (s) and destination (s) of the video, for the objective of finding the most reliable path (s) to forwarding the video streams. Zhu et al. in [39] proposed a video uploading scheme in urban VANET. In order to reduce the link failure frequency, this study proposed to select the best relay vehicles for the forwarding of the video based on the vehicle mobility prediction. The proposed forwarding protocol adopts and modifies the greedy geographic routing protocols, by adding the stability factor with the distance factor to choose the relay vehicles. It remains to consider some parameters for the relay vehicles selection like buffer management, transmission rate and encouragement factor. Zaimi et al. presented in [40] a Greedy Perimeter Stateless Routing protocol with two (2) Paths (GPSR-2P) for video transmission in urban VANET. The GPSR protocol for video routing was applied through two paths to avoid the congestion and the saturation in a same path. Note that in GPSR protocol, each sender vehicle forwards the video packets to its neighbor which is the closest geographically to the destination. The experiments results showed that GPSR-2P achieves higher packet delivery ratio and lower delay compared with GPSR. Moreover, this proposal enhances the user QoE. However, this study considered only two neighbors vehicles for each vehicle desiring sent packets.

Many Receiver Based Forwarding (RBF) schemes were proposed for video streaming through one path for unicasting or multi-casting forwarding, like REceiver-based solution with video transmission DECoupled from relay node selection (REDEC) [41], VIRTUS with Density-Aware relay node selection Decoupled from Video Transmission (DADVT) [42] and QOe-Driven and LInk-qualiTy rEceiver (QOALITE) [43]. Due to the use of only a single path to forward video packets, these works (i.e. [41], [42] and [43]) suffer from interference and packet collision especially for very high transmission rate and when the network is very dense. Other RBF-based schemes were proposed to deal with the collision problem through the multi-path dissemination of video packets in VANET like LocatIon-Aware multIpaTH videO streamiNg (LIAITHON) which used two and three paths, as presented in [44] and [45], respectively. We can mention that [44] and [45] use different paths that could be limited by collision and interference problems.

In the literature, the reader can find some forwarding schemes conceived for video broadcasting like Automatic Counter Distance Based (ACDB) [46] which take into consideration the vehicles density information in the flooding of the video packets. Another forwarding scheme is the video dissemination protocol (VoV) [47] that allows video broadcasting under any kind of traffic condition. In addition, Selective Rebroadcast Mechanism for Video Streaming over VANET (ReViV) was proposed in [48], which selects the minimum number of rebroadcaster vehicles in order to reduce the interferences and achieve higher video quality. Note that the limit of these broadcasting schemes is the collision and the congestion due to the high number of broadcasted packets called broadcast storm problem.

2.3.3. Adaptive category

This category adapts some video transmission parameters to improve the video quality in VANET, like adapting the size of contention window at the Medium Access Control (MAC) layer, or adapting the number of video layers at the application layer, etc. Many Adaptive studies were proposed for video streaming in VANET. Asefi et al. in [49] proposed an adaptive MAC retransmission limit scheme to improve IEEE 802.11p standard protocol [50]. The proposed scheme adapts the retransmission limit of video frame number to optimize the frequency of playback freezes and start-up delay. Compared with IEEE 802.11p, the proposed adaptation minimizes the frequency of playback freezes, but it produces a small increase in start-up delay. Xing and Cai [51] proposed a video quality adaptive algorithm for video streaming in highway VANET.

Table 1: Comparison between error resiliency techniques and mechanisms for video streaming in VANET

Error resiliency techniques and mechanisms	Network overload	Transmission delay	Artifacts in the displayed video
FEC	High	Low	Low
Erasure Coding	High	Low	Low
Interleaving	High	Low	Low
Retransmission	Low	High	Low
Concealment	Low	Low	High

Based on current download speed and the buffer level at the receiver, the receiver requests an adaptive number of video enhancement layers. The proposed solution provides good trade-off among start-up latency, interruption ratio and video quality, however this proposal does not consider real factors of channel to estimate the link throughput like: wireless shadowing, fading channel impairments and contentions of the 802.11 link.

2.3.4. Discussion

The existing video streaming studies in the VANET aim to improve the video transmission quality in terms of QoS and/or QoE metrics. We have classified these works into three categories: video coding and error resiliency category, cooperative relays based category and adaptive category. Our proposed Hybrid Error Recovery Protocol (HERP) belongs to the first category, in which its purpose is to recover all kinds of video packets lost in VANET. The similar works of this category based on error resiliency (i.e redundancy, retransmission and error concealment) were conceived to recover errors of video packets transmitted in the VANET. Table 1 summarizes a comparison between different error resiliency techniques and mechanisms in VANET in terms of network overload, transmission delay and additional artefacts in the displayed video. The redundancy-based mechanisms such as: FEC, EC and interleaving increase the network overload because of the redundant packets. The retransmission technique increases the transmission delay due to the duplicated video packets that require a receiver request. The concealment technique is applied at the receiver level, without any required additional network overload or transmission delay, however the concealment affects the displayed video quality. Our proposed HERP protocol is based on a combination between the two error resiliency techniques: the redundancy and the retransmission in order to guarantee a low network overload and a low transmission delay with high video streaming quality.

3. Hybrid Error Recovery Protocol (HERP) for video streaming in VANET

In the literature of error recovery mechanisms for video streaming in VANET, many research activities adopted the redundancy and/or the retransmission for recovering the lost video packets. On the one hand, the redundancy decreases the network load and recovers only the uniform transmission errors. On the other hand, the retransmission decreases the end-to-end delay and recovers the burst errors. We propose in this paper a new error recovery protocol for video streaming in VANET called Hybrid Error Recovery Protocol (HERP). This proposal combines redundancy and retransmission approaches to recover both uniform and burst errors. To overcome the uniform errors, to reduce network overload and transmission delay, HERP uses SPFEC mechanism. HERP adopts also the unequal protection of video packets according to its frame

types ('I', 'P', 'B') in the aim of improving the video quality at the receiver vehicle. Using the reporting technique, HERP adapts dynamically the redundancy rate, retransmission limit and transmission rate in function to network condition and network load. HERP proposes a detection mechanism of packet loss to distinguish between the lost video packets due to network condition and those due to the network overload, in order to react and cope differently with each packet loss type.

3.1. Basic concepts of HERP

3.1.1. Sub-Packet Forward Error Correction (SPFEC)

When the video sub-packets errors are uniform, HERP uses the SPFEC to recover these errors without any retransmission mechanism. Tsai et al. [16] proposed Sub-Packet FEC (SPFEC) to improve the video streaming recovery performance over wireless network. The video packet in SPFEC is composed of two parts: original sub-packets and redundant sub-packets. As shown in figure 2, SPFEC encoder adds n redundant sub-packets into k original sub-packets. When the decoder receives the video packet, it can recover the lost original sub-packets by means of redundant sub-packets. SPFEC reduces the Effective Packet Error Rate ($EPER$) and the network overload comparing with FEC, because the Sub-Packet Error Rate ($SPER$) is smaller than the Packet Error Rate (PER). SPFEC reduces also the transmission delay where the receiver decodes video packets without waiting other packets comparing to FEC.
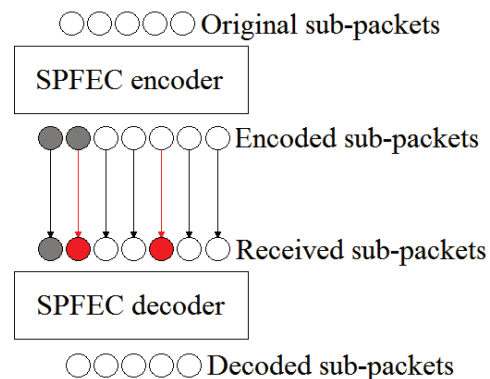

Figure 2: Sub-Packet Forward Error Correction mechanism

3.1.2. Unequal protection of video frames

As mentioned above, HERP is MPEG-4 based video frames compression. As shown in figure 3, within the GoP, if the I-frame is erroneous the other P-frames and B-frames cannot be decoded even there have been received correctly. This propagation of errors in the GoP is due to the relationship between the I-frames, P-frames and B-frames of the video stream. In order to minimize the error propagation on the quality degradation of reconstructed video, HERP proposes unequal protection of video frames in function of their types (I, P, B). According to video frames importance, HERP provides a higher redundancy rate and retransmission limit for I-frames than the other video frames, this is proposed in order to guarantee more protection of I-frames which are the most important compared to the others.

3.1.3. Reporting technique

The HERP parameters adaptation and video packets retransmission is achieved by means of periodic receiver reports. The receiver vehicle maintains a trace of received and lost video packets. When the receiver vehicle cannot recover the burst errors of video packet by SPFEC mechanism, it sends a report to the sender
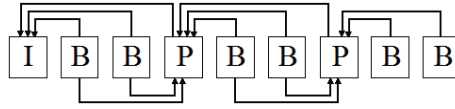
Figure 3: Relationship between GoP video frames in MPEG-4 standard

vehicle. The report represents a request of unrecovered video packets retransmission, also it imports the network condition and network load information to allow the sender vehicle for adapting the redundancy rate, retransmission limit and transmission rate. Before the report sending, the receiver vehicle applies the proposed video packet loss detection of HERP to identify and differentiate between the causes of packet loss in order to better identify the network state.

3.1.4. Video packet loss detection mechanism of HERP

In VANET, the packets may be corrupted and lost due to several reasons such as congestion, transmission errors and route disconnection. Using HERP, the receiver vehicle can detect the lost video packets and distinguish between their types. Also, HERP allows the sender vehicle to react with different types of these lost packets, by the retransmission of lost video packets and the adjustment the redundancy rates and retransmissions limits of video packets according to their types of frames (i.e. I, P, B).

Packet loss due to network congestion. In HERP, we propose to add a sub-field within the video packet header, which controls the congestion in the network, as follows. When a relay vehicle forwards the video packet, it sets this sub-field by the dropped video packets identifications. Hence, the receiver can detect the congestion in the network, and the identification of dropped packets due to the network congestion by means of this sub-field and it informs the sender vehicle for the congestion production in the network.

Packet loss due to transmission errors. HERP is based on the periodic estimation of network condition at the receiver vehicle. If the receiver cannot recover the received erroneous video packets using SPFEC mechanism, then it considers the transmission errors as a cause of these lost packets.

Packet loss due to route disconnection. When successive video packets are not received before their waiting timeout and if these packets are not dropped because the congestion and transmission errors, the receiver considers that these lost packets are caused by the network disconnection. When the receiver vehicle detects and distinguishes between different types of lost video packets, it applied the reporting technique to inform the sender vehicle for the network state. Therefore, the sender vehicle retransmits the lost video packets, adjusts the transmission rate with network load and adapts the redundancy rate as well as the retransmission limit according to network condition.

3.2. General architecture of video transmission using the proposed HERP

Figure 4 illustrates the basic architecture of HERP proposed in this study. As shown, the HERP module at the sender vehicle (with red color) consists of five components: (1) SPFEC Generator (SPG), (2) Redundancy Rate Adaptor (RRA), (3) Retransmission Limit Adaptor (RLA), (4) Transmission Rate Adaptor (TRA) and (5) Packet Retransmission Monitor (PRM). Also, figure 4 presents HERP module of receiver vehicle (with blue color) consisting of two components: (6) Network Condition Estimator (NCE) and (7) Reporting Monitor (RM). In next subsections, all these components are explained.

(1) ***SPFEC Generator (SPG):*** SPG component creates and generates video packets, each packet consists of original sub-packets and redundant sub-packets.

(2) ***Redundancy Rate Adaptor (RRA):*** RRA adjusts dynamically the redundancy rate (amount of redundant sub-packets) in function of network condition and the frame type of this packet (I, P, B). If the error rate in the network is high, the RRA increases the number of redundant sub-packets in order
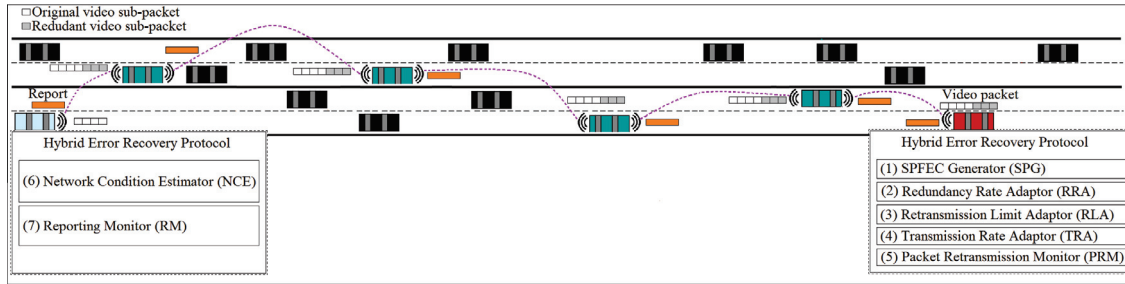
Figure 4: Video streaming using HERP protocol in VANET

to allow the receiver vehicle to recover the uniform erroneous sub-packets, otherwise (i.e. if the error rate is low, the RRA reduces the number of redundant sub-packets aiming at decreasing the network load.

(3) **Retransmission Limit Adaptor (RLA):** RLA adapts dynamically the Retransmission Limit (RL) of each video packet, according to the network condition and the frame type of this packet (I, P, B). If the error rate in the network is low, the RLA increases the RL to recover the burst erroneous sub-packets. If the error rate in the network is high, RLA decreases the RL, to avoid the additional transmission delay because of the retransmission mechanism.

(4) **Transmission Rate Adaptor (TRA):** TRA adjusts dynamically the Transmission Rate (TR) with the current network load. If the network is heavy loaded, the TRA reduces the TR to avoid the congestion, otherwise TRA increase the TR.

(5) **Packet Retransmission Monitor (PRM):** PRM retransmits the requested video packet if the number of retransmissions of this packet does not exceed its retransmission limit. Otherwise, PRM cannot send the request packet to avoid the additional retransmission delay.

(6) **Network Condition Estimator (NCE):** NCE estimates the Bit Error Rate (*BER*), *SPER* and *EPER*. In function of the estimated *EPER*, the receiver vehicle accepts or rejects the received video packet.

(7) **Reporting Monitor (RM):** RM detects the lost video packets, distinguished between their types and generates the reports which will be sent to the sender vehicle to request the retransmission of non-recovered packets and adjusts HERP parameters. The report imports the network condition estimation (*BER*), and network load state.

As shown in figure 4, the sender vehicle communicates with the receiver via a relay vehicles (with green color). The relay vehicle saves the identifications of dropped video packets at its level ignored in reason of network congestion (i.e. packets exceeding its queue size). Also, when the video packet passes through the relay vehicle, this latter adds in the packet header the identifications of the dropped packets. This exported information leads the receiver to detect lost packet of the video due to the network congestion.

3.3. HERP video packet and report

In our proposed HERP, the receiver vehicle calculates the *SPER* and *EPER* based on the essential information (cited below) imported by the received video packet header. The receiver uses this information to identify the network load state.

- *video_pkt_id:* is a sequence number identifying the video packet content.

- *video_pkt_type:* represents the type of video packet frame (I, P, B).

- *sub_pkt_size:* is the video sub-packet length measured in bits.

11

- **nb_source_sub_pkts:** is the number of source sub-packets k within the video packet.

- **nb_redundant_sub_pkts:** is the number of redundant sub-packets h within the video packet.

- **dropped_packets_id:** represents the identifications of lost video packets due to the network congestion.

The report generated by the receiver vehicle imports the following information (cited below) to identify the network condition and network load for the sender vehicle. The sender uses this information to adjust the HERP parameters (i.e. redundancy rates, retransmissions limits and transmission rate) and to retransmit the required video packets.

- **Bit Error Rate (BER):** measures the error probability of video packet bit in the network.

- **network_load_state:** is equal to 1, when the congestion is produced in the network, otherwise this variable takes the 0 value.

- **requested_video_packets_id:** the identifications of lost video packets due to the congestion, transmission errors and/or route disconnection.

3.4. HERP algorithm

3.4.1. HERP algorithm at the sender vehicle level

The general HERP algorithm at the sender vehicle level is presented in figures 5 and 6. The sender vehicle performs the pseudo-code shown in figure 5, when it wants to send a new video packet to a particular receiver vehicle. Before generating a video packet, SPG component of sender vehicle applies the SPFEC mechanism in step 1 to calculate the amount of redundant sub-packets ($nb\_redundant\_sub\_pkts$) of this packet in function of three parameters namely the type of video packet ($video\_pkt\_type$), the redundancy rate (RR_I, RR_P, RR_B) and the amount of original sub-packets ($nb\_source\_sub\_pkts$). Each video packet has a unique id according to its content. After first step, the sender generates and sends in step 2 the video packet towards the receiver vehicle via a multi-hop communication.

Input: $video\_pkt\_type$, $nb\_source\_sub\_pkts$, RR_I, RR_P, RR_B
Output: $video\_packet$
  $Initialisation$: $nb\_redundant\_sub\_pkts \leftarrow 0$
  Step 1.
  if ($video\_pkt\_type = I$) **then**
    $nb\_redundant\_sub\_pkts \leftarrow RR_I * nb\_source\_sub\_pkts$
  else if ($video\_pkt\_type = P$) **then**
    $nb\_redundant\_sub\_pkts \leftarrow RR_P * nb\_source\_sub\_pkts$
  else
    $nb\_redundant\_sub\_pkts \leftarrow RR_B * nb\_source\_sub\_pkts$
  end if
  $video\_pkt\_id \leftarrow video\_pkt\_id + 1$
  Step 2.
  Generation of video packet
  Send of generated video packet toward the receiver vehicle

Figure 5: HERP algorithm for video packet transmission at sender vehicle

Figure 6 shows the report reception algorithm at the sender. When the sender receives a report from the receiver vehicle, it extracts firstly the information exported by the report (i.e. *BER*, *network_load_state* and *requested_video_packets_id*) (step 1). Secondly (in step 2), the RRA and RLA components of this vehicle

Input: $Report$, THL, THM, THH, RR_{Ii}, RR_{Pi}, RR_{Bi}, RL_{Ii}, RL_{Pi}, RL_{Bi}, RR_i , $sent\_packets\_table$

Output: RR_I, RR_P, RR_B, RL_I, RL_P, RL_B, RR

  if ($report\ is\ received$) **then**

    Step 1.

    Extract $BitErrorRate\ (BER)$

    Extract $network\_load\_state$

    Extract $requested\_video\_packets\_id$

    Step 2. Adaptation of RR_I, RR_P, RR_B and RL_I, RL_P, RL_B with BER

    if (BER<=THL) **then**

      $RR_I \leftarrow 0$ $RL_I \leftarrow RL_{Ii}$

      $RR_P \leftarrow 0$ $RL_P \leftarrow RL_{Pi}$

      $RR_B \leftarrow 0$ $RL_B \leftarrow RL_{Bi}$

    else if (BER<=THM) **then**

      $RR_I \leftarrow RR_{Ii}/2$ $RL_I \leftarrow RL_{Ii}/2$

      $RR_P \leftarrow RR_{Pi}/2$ $RL_P \leftarrow RL_{Pi}/2$

      $RR_B \leftarrow RR_{Bi}/2$ $RL_B \leftarrow RL_{Bi}/2$

    else if (BER<=THH) **then**

      $RR_I \leftarrow RR_{Ii}$ $RL_I \leftarrow RL_{Ii}/2$

      $RR_P \leftarrow RR_{Pi}$ $RL_P \leftarrow RL_{Pi}/2$

      $RR_B \leftarrow RR_{Bi}$ $RL_B \leftarrow 0$

    else

      $RR_I \leftarrow RR_{Ii}$ $RL_I \leftarrow RL_{Ii}/2$

      $RR_P \leftarrow RR_{Pi}$ $RL_P \leftarrow 0$

      $RR_B \leftarrow RR_{Bi}$ $RL_B \leftarrow 0$

    end if

    Step 3. Adaptation of Transmission Rate with $Network\_Load\_State$

    if ($network\_load\_state$ =1) **then**

      $RR \leftarrow RR_i$ / 2

    else

      $RR \leftarrow RR_i$

    end if

    Step 4. Retransmition of each requested video packet

    Search the packet in the $sent\_packets\_table$ by its id

    if ($video\_pkt\_type$ =I) **then**

      if ($packet\_number\_retranmission <= RL_I$) **then**

        Send the requested video packet I toward reciever vehicle

        $packet\_number\_retranmission \leftarrow packet\_number\_retranmission + 1$

      end if

    else if ($video\_pkt\_type$ =P) **then**

      if ($packet\_number\_retranmission <= RL_P$) **then**

        Send the requested video packet P toward reciever vehicle

        $packet\_number\_retranmission \leftarrow packet\_number\_retranmission + 1$

      end if

    else

      if ($packet\_number\_retranmission <= RL_B$) **then**

        Send the requested video packet B toward reciever vehicle

        $packet\_number\_retranmission \leftarrow packet\_number\_retranmission + 1$

      end if

    end if

    Update the $sent\_packets\_table$

  end if

Figure 6: HERP algorithm for report reception at sender vehicle

13

adapt the redundancy rates (RR_I, RR_P and RR_B) and retransmission limits (RL_I, RL_P and RL_B) of video packets in the basis of BER and HERP thresholds (i.e. THL, THM and THH). When the BER is lower than the THreshold Low (THL), RRA prohibits the generation of redundant sub-packets to avoid the overload of the network, which not requires a high protection of video packets against error transmission. The RLA activates the retransmission mechanism with the initials retransmission limits (RL_{Ii}, RL_{Pi}, RL_{Bi}) to recover the lost video packets due to the network congestion or the route disconnection. In the case when the BER is higher than THL and lower than THreshold Medium (THM), RRA activates the redundancy mechanism but it sets the redundancy rates to the half of initial redundancy rates values (RR_{Ii}, RR_{Pi}, RR_{Bi}) in order to guarantee a high protection of video packets against uniform errors and to avoid the network overload which produce the congestion. In this error interval, RLA adapts the retransmission limits of video packets to the half of initial retransmission limits values in order to recover the burst errors and to reduce the additional transmission delay caused by the retransmission mechanism. When BER is higher than THM and lower than THreshold High (THH), RRA adapts the redundancy rates with the same value of initial redundancy rates to recover the high number of lost packets due to the transmission errors. RLA stops the retransmission of B-frames video packets to reduce the transmission delay. When BER is higher than THH due to the high number of burst errors, which affect the delay constraint, RLA stops the retransmission of P-frames video packets. However, RRA keeps the initial redundancy rates to recover a maximum number of lost packets. In the third step, the TRA component adjusts the Transmission Rate (RR) according to the network load state ($network\_load\_state$). If the congestion was produced in the network (i.e. $network\_load\_state = 1$), the TRA decreases the RR to defeat the congestion problem. If the congestion was not produced (the case when $network\_load\_state = 0$), the TRA increases the RR to improve the transmission delay. The step 4 describes the retransmission process of proposed HERP. For each requested video packet, PRM compares the retransmissions number of this packet with its retransmission limit (RL_I, RL_P, RL_B). If the retransmission number is lower then retransmission limit, the PRM retransmits the required video packet. Otherwise, PRM prohibits the retransmission of required video packet to don't increase the transmission delay. After the retransmission of the requested video packet, PRM updates the table of sent packets ($sent\_packets\_table$) by the new value of the sent packet retransmission number ($packet\_number\_retranmission$), which will be checked in the next request of this packet.

3.4.2. HERP algorithm at the receiver vehicle level

Figure 7 shows the pseudo-code of HERP module at the receiver vehicle. When this latter receives a video packet, it applies firstly step 1 of the algorithm to extract the information exported by the packet header: $video\_pkt\_id$, $sub\_pkt\_size(n)$, $nb\_redundant\_sub\_pkts(k)$, $nb\_source\_sub\_pkts(h)$ and $dropped\_packets\_id$. Secondly, the NCE component of the receiver estimates BER, $SPER$ and $EPER$ (step 2). In the step 3, RM component generates a uniform random variable r to check the recovery probability of received video packet. If RM can recover the errors of video sub-packets using SPFEC mechanism ($EPER$ is lower than r), the RM accepts the received video packet and calculates the number of dropped video packets ($number(dropped\_packets\_id)$), which are packets lost during the network congestion. If some video packets were dropped, RM adds the identifiers of dropped packets ($dropped\_packets\_id$) into the requested video packets identifiers ($requested\_video\_packets\_id$), updates the $network\_load\_state$ variable, creates a report, adds the currents information into the report (BER, $network\_load\_state$ and $requested\_video\_packets\_id$) and sends this report to the sender vehicle. If $EPER$ is higher than r, which means that RM cannot recover the burst errors (i.e. packet lost due to transmission errors) of video sub-packets using SPFEC mechanism, the RM adds the unrecovered video packet identifier ($packet\_id$) into $requested\_video\_packets\_id$, rejects the unrecovered video packet, creates a video report, adds the currents information into the report (BER, $network\_load\_state$ and $requested\_video\_packets\_id$) and sends the report to the sender. The receiver vehicle sends the report towards the sender vehicle, in order to require the retransmission of the burst lost video packets and to adjust the HERP parameters at the sender vehicle. If the received packet sequence number ($video\_pkt\_id$) is higher than the expected packet sequence number ($expected\_pkt\_id$), that means there are some video packets not received (their id is between $expected\_pkt\_id$ and $video\_pkt\_id$). For each $packet_i$ of these packets, RM starts a waiting time, to make sure that these packets are lost because the route disconnection. After the expiration of video $packet_i$ waiting time (step 4), and if this $packet_i$ is not received, the RM considers this

14

Input: *Video packet, expected_pkt_id*
Output: *Report*
  Initialisation : $EPER \leftarrow 0$, $SPER \leftarrow 0$, $network\_load\_state \leftarrow 0$
  if *(Video packet is received)* **then**
    Step 1.
    Extract *video_pkt_id*
    Extract *sub_pkt_size(n)*
    Extract *nb_redundant_sub_pkts(k)*
    Extract *nb_source_sub_pkts(h)*
    Extract *dropped_packets_id*
    Step 2.
    Estimate *BER* by equation 1
    Calculate *SPER* by equation 2
    Calculate *EPER* by equation 3
    Step 3.
    Generate uniform random number r in the interval [0, 1]
    if $(r > EPER)$ **then**
      Accept video packet (recovered packet)
      if (number(dropped_packets_id) > 0) **then**
        /** Detect the packet loss due to network congestion **/
        $network\_load\_state \leftarrow 1$
        Add *dropped_packets_id* to *requested_video_packets_id*
        Create and send a report toward sender vehicle
      else
        /** Detect the packet loss due to to wireless errors **/
        Add *video_pkt_id* to *requested_video_packets_id*
        Reject video packet (no recovered packet)
        Create and send a report toward sender vehicle
      end if
    end if
    if $(video\_pkt\_id > expected\_pkt\_id)$ **then**
      For each $video\_packet_i$ / $i >= expected\_pkt\_id$ and $i < video\_pkt\_id$
      Start waiting time of video $video\_packet_i$
    end if
  end if
  Step 4.
  if (waiting time of any *video_packet_i* is expired) **then**
    if (*video_packet_i* not received) **then**
      /** Detect the packet loss due to route disconnexion **/
      Add *id* of *video_packet_i* to *requested_video_packets_id*
      Create and send a report toward sender vehicle
    end if
  end if

Figure 7: HERP algorithm at receiver vehicle

15

loss is due to the route disconnection, then RM requires the retransmission of this packet by sending a new report to the sender vehicle.

Estimation of Bit Error Rate (BER). NCE uses the following equation to estimate *BER* in each interval time δ_t.

$$BER(\delta_t) = 1 - (1 - \frac{success(\delta_t)}{Total(\delta_t)})^{\frac{1}{Total(\delta_t)}} \tag{1}$$

Where, success(δ_t) is the number of corrected received video packets during the interval time δ_t without using the SPFEC mechanism. Total (δ_t) is the total number of transmitted video packets from the sender vehicle to the receiver vehicle during the interval time δ_t.

Sub-Packet Error Rate (SPER). NCE calculates *SPER* based on the estimated *BER*. *SPER* is the probability that a video sub-packet cannot be recovered at the receiver vehicle, it is given by the equation:

$$SPER = 1 - (1 - BER)^{sub\_pkt\_size} \tag{2}$$

Effective Packet Error Rate (EPER). NCE uses the following equation to calculate *EPER* based on calculated *SPER*. *EPER* is the probability that a video packet cannot be recovered at the receiver vehicle.

$$EPER = 1 - (\sum_{i=k}^{k+h} C_i^{k+h} * (1 - SPER)^i * SPER^{k+h-i}) \tag{3}$$

Where k is the number of source sub-packets and h is the number of redundant sub-packets within a video packet.

3.4.3. HERP algorithm at the relay vehicle level

Figure 8 shows the HERP algorithm at the relay vehicle. When this latter receives the video packet (case 1), it checks its queue length (*queue_length*), which is compared with maximum size of queue length (*max_queue_length*). When *queue_length* equals to *max_queue_length* (it means that the vehicle buffer is full), the relay vehicle adds *video_packet_id* of received packet to its local table, which saves the *video_packet_id* of dropped video packets (*dropped_packets_id_table*), then the relay vehicle drops the received video packet due to the network congestion. If *queue_length* is lower than *max_queue_length* (it means that the vehicle buffer is not full), the relay vehicle inserts the video packets in its queue and increases the *queue_length*. The relay vehicle applies the step 1.2 when it wants to forward the received video packet. If its *dropped_packets_id_table* contains at least one *video_packet_id*, the relay vehicle adds the elements of *dropped_packets_id_table* to *dropped_packets_id* field of the received video packet. Then, it sends this received packet toward the receiver vehicle and decreases the length of its queue. In the case of the reception of a report (case 2) and if the relay vehicle buffer is full (*queue_length* equals to *max_queue_length*), the relay vehicle drops the received report. Otherwise, it extracts the *requested_video_packets_id*, removes it from its *dropped_packets_id_table*, because the receiver vehicle has been informed that these video packets were dropped, next, it inserts the report in its queue and increases the queue length. The relay vehicle applies the step 2.2 when it desires to send the report toward the sender vehicle, consequently it decrements the length of its queue.

4. Performance evaluation and results

In order to evaluate HERP behavior, we have performed an experimental analysis of its performance. We have divided these experiments into two groups: primary evaluation and performance comparison. The first group aims to fix the HERP parameters (*THL*, *THM* and *THH*) with varying the network condition. The second group represents a series of comparisons between HERP based on chosen thresholds values with two other protocols: UDP and UDP with SPFEC under real scenarios and through different levels of network condition and network load.

16

Input: *video_packet*, *report*
Output: *video_packet*, *report*
  Case 1. Reception of a video packet
  if (*video_packet_is_received*) **then**
    Step 1.1.
    if (*queue_length = max_queue_length*) **then**
      Add video_packet_id to dropped_packets_id_table
      Drop the video_packet
    else
      Add video_packet_id to queue
      queue_length ← queue_length + 1
      Step 1.2.
      if (*length*(*dropped_packets_id_table*) > 0) **then**
        Add (*dropped_packets_id_table*) to (*dropped_packets_id* of received video packet)
        Send a received video packet toward receiver vehicle
        queue_length ← queue_length − 1
      end if
    end if
  end if
  Case 2. Reception of a report
  if (*report_is_received*) **then**
    Step 2.1.
    if (*queue_length = max_queue_length*) **then**
      Drop the report
    else
      Extract *requested_video_packets_id*
      Remove all *requested_video_packets_id* from *dropped_packets_id_table*
      Add report to queue
      queue_length ← queue_length + 1
    end if
    Step 2.2.
    Send a received report toward sender vehicle
    queue_length ← queue_length − 1
  end if

Figure 8: HERP algorithm at relay vehicle

Table 2: General parameters of simulation

Parameter	Value	Parameter	Value
Number of vehicles	100	Scenario	V2V
Routing protocol	AODV	Communication range	300 m
Propagation model	TowRayGround	Bit Error Rate	$\{0,\ldots, 0.005\}$
Video file	Foreman.yuv	Frame rate (fps)	30
Video packet size	1024 bits	Sub-packet size	100 bits
RR_{Ii}	75%	RL_{Ii}	7
RR_{Pi}	50%	RL_{Pi}	5
RR_{Bi}	25%	RL_{Bi}	3
Evaluation metrics	PDR, average delay, DFR, PSNR, MOS	Number of video frames	400

4.1. Simulation and parameter settings

In order to measure the performance of HERP in VANET, we have conducted several simulations performed on network simulator ns-2 [52] version 2.35. We compared the following protocols for video streaming:

- **HERP:** is the proposed protocol, which integrates SPFEC with the retransmission and the unequal protection of video frame types.

- **UDP-SPFEC:** is the traditional UDP protocol with SPFEC mechanism.

- **UDP:** is the traditional UDP protocol without SPFEC.

Figure 9: Studied urban area for video streaming in VANET

The simulation parameters are presented in table 2. We have used EvalVid framework [53] to generate the video streaming trace at the sender and receiver vehicles. We have also used SUMO [54] for road traffic

18

simulation based on downtown area of Oum El Bouaghi city (Algeria), imported from Open Street Map [55] and showed in figure 9. SUMO takes into consideration several VANET particularities like street capacity, traffic light and vehicles movement, in order to generate the urban mobility model required by ns-2. All the results are represented at a confidence interval of 95%. The AODV routing protocol is chosen in our simulation under V2V scenario. The metrics used for the primary evaluation and for the comparison between the studied protocols are Packet Delivery Ratio (PDR), average transmission delay, Decodable Frame Rate (DFR), Peak Signal-to-Noise Ratio ($PSNR$) and Mean Opinion Score (MOS). We have used packets with a maximum size of 1024 bits. The video transmitted in our simulations is the well-known video benchmark named the foreman.yuv. It is composed of 400 frames that are encoded with MPEG-4, using GoP structure of IBBPBBPBB and temporal resolution of 30 frames per second. For the proposed HERP, we have assumed that the initial values of RR_I, RR_P and RR_B are 75%, 50% and 25%, respectively, and the initial values of RL_I, RL_P and RL_B are 7, 5, 3, respectively. We assume that (RR_I, RL_I) are greater than (RR_P, RL_P) and these latter are greater than (RR_B, RL_B), because the I-frames are more important than P-frames, and P-frames are more important than B-frames. In addition, it is assumed that the UDP-SPFEC is submitted to the same redundancy rates of video frame types, as for HERP.

4.2. Evaluation metrics

To evaluate the effectiveness of HERP, we analyze the (PDR), average transmission delay, Decodable Frame Rate (DFR) and Peak Signal to Noise Ratio ($PSNR$) as QoS metrics. We also take into account the Mean Opinion Score (MOS) as a QoE metric. The next subsections explain the analyzed QoS and QoE metrics.

4.2.1. Packet Delivery Ratio (PDR)

PDR represents the total number of received video packets over the total number of sent video packets. It is calculated as follows:

$$PDR = \frac{\sum ReceivedPackets}{\sum SendPackets} \tag{4}$$

4.2.2. Average transmission delay

The transmission delay of a packet is the time interval between the sending moment of this packet at the sender and the complete reception time of this packet at the receiver level. The average transmission delay is the sum of all received packets delay divided by the number of the total number of the received packets. The average transmission delay is computed by the following formula:

$$Average\_transmission\_delay = \frac{\sum_{i=0}^{n}(RTimeOfPkt_i - STimeOfPkt_i)}{\sum ReceivedPackets} \tag{5}$$

Where, $RTimeOfPkt_i$ is the reception time of the $packet_i$ and $STimeOfPkt_i$ is the sending time of the $packet_i$.

4.2.3. Decodable Frame Rate (DFR)

DFR is defined as the number of decodable video frames over the total number of sent video frames in a given $EPER$ (Effective Packet Error Rate), it is calculated as follows:

$$DFR = \frac{NDF(I) + NDF(P) + NDF(B)}{\sum SendFrames} \tag{6}$$

Where, $NDF(I)$ is the Number of Decodable Frames I, $NDF(P)$ is the Number of Decodable Frames P and $NDF(B)$ is the Number of Decodable Frames B. The $NDF(I)$ is calculated by the following formula:

$$NDF(I) = (1 - EPER)^{aI} * NGoP \tag{7}$$

Where '$NGoP$' is the total number of GoPs in the video stream, 'aI' is the average packets number in frame 'I'.

19

The *NDF(P)* is given by the following formula:

$$NDF(P) = (1 - EPER)^{aI} * \sum_{i=1}^{nP} (1 - EPER)^{i*aP} * NGoP \tag{8}$$

Where *'nP'* is the total number of frames *'P'* in a GoP and *'aP'* is the average packets number in frame *'P'*.

The *NDF(B)* is calculated as follows:

$$NDF(B) = [(1 - EPER)^{aI*nP*aP} + \sum_{j=1}^{nP} (1 - EPER)^{j*aP} * (1 - EPER)^{aB}] * (M-1) * (1 - EPER)^{aI+aB} * NGoP \tag{9}$$

Where *'aB'* is the average packets number in frame *'B'* and *'M'* is the distance between frames *'I'* and frames *'P'* in a GoP.

4.2.4. Peak Signal to Noise Ratio (PSNR)

PSNR measures the quality of reconstructed video file comparing with original video file. In mathematical way, *PSNR* is the logarithmic ratio between the maximum value of a signal and the Mean Squared Error (*MSE*) [56]. Usually, the pixels are represented using 8 bits, consequently the maximum value of signal is equal to 255.

$$PSNR = 10 * \log \frac{255^2}{MSE} \tag{10}$$

For the original frame *o* and distortion frame *d*, Mean Squared Error (*MSE*) represents the cumulative square error between them, it is calculated as follows:

$$MSE = \frac{1}{M * N} \sum_{m=1}^{M} \sum_{n=1}^{N} |o(m,n) - d(m,n)|^2 \tag{11}$$

Where, *M * N* is the frame size in pixel, *o(m, n)* and *d(m, n)* are the luminance pixels in the same position *(m, n)* within the frames *o* and *d*.

4.2.5. Mean Opinion Score (MOS)

MOS allows the quantification of subjective tests realized by a human evaluator, during the subjective tests several users are invited to judge the quality of video and give a specific measured value for the video quality. At the end of these subjective tests, *MOS* is calculated by averaging all video quality values. In our simulations, a mapping of *PSNR* values to *MOS* values is performed to estimate the human quality perception for video streaming.

4.3. Preliminary evaluation

The HERP performance is dictated by *THL*, *THM* and *THH* parameters. *THL* is used to start the redundancy to recover the packet loss due to the transmission errors. *THM* is used to stop the retransmission of B-frame video packets and *THH* is called to stop the retransmission of P-frame video packets. HERP employs *THM* and *THH* in order to reduce the retransmission effect on the transmission delay of video packets. Our goal by this preliminary evaluation is to analyze the behavior of HERP to choose values of *THL*, *THM* and *THH*. For this initial evaluation, we have chosen to observe the performance of HERP under the Decodable Frame Rate (*DFR*) metric.

To choose the *THL* value, we have set the *THM* and *THH* primary values at the maximum *BER* value considered in our simulation (fixed at 0.005) and we have performed many simulations of two HERP variants:

- **HERP with *THL* = 0:** in this version, the HERP starts the redundancy with the retransmission when the *BER* is higher than 0.
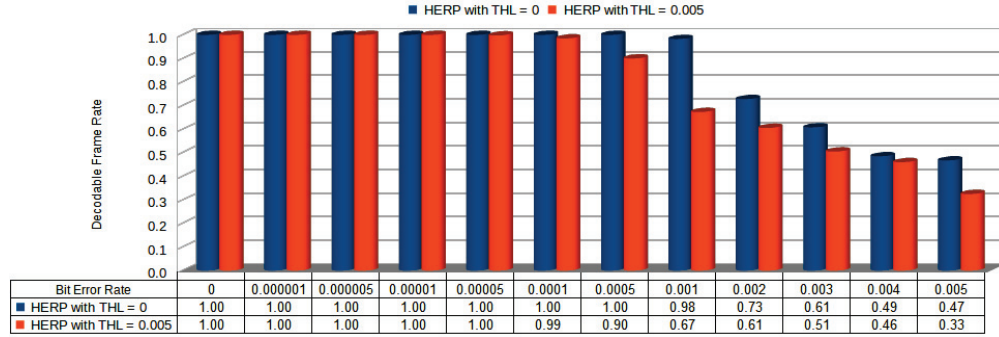
Bit Error Rate	0	0.000001	0.000005	0.00001	0.00005	0.0001	0.0005	0.001	0.002	0.003	0.004	0.005
■ HERP with THL = 0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.73	0.61	0.49	0.47
■ HERP with THL = 0.005	1.00	1.00	1.00	1.00	1.00	0.99	0.90	0.67	0.61	0.51	0.46	0.33

Figure 10: Variation of Decodable Frame Rate with BER between HERP with THL = 0 and HERP with THL = 0.005

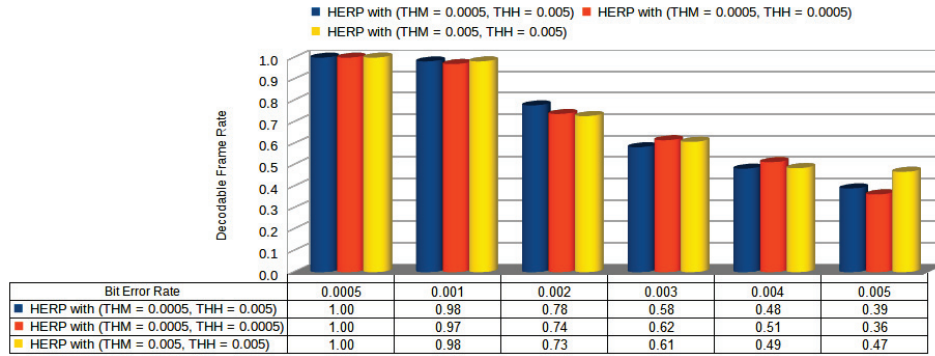| Bit Error Rate	0.0005	0.001	0.002	0.003	0.004	0.005
■ HERP with (THM = 0.0005, THH = 0.005)	1.00	0.98	0.78	0.58	0.48	0.39
■ HERP with (THM = 0.0005, THH = 0.0005)	1.00	0.97	0.74	0.62	0.51	0.36
■ HERP with (THM = 0.005, THH = 0.005)	1.00	0.98	0.73	0.61	0.49	0.47

Figure 11: Variation of Decodable Frame Rate with BER between HERP with ($THM = 0.0005$, $THH = 0.005$), HERP with ($THM = 0.0005$, $THH = 0.0005$) and HERP with ($THM = 0.005$, $THH = 0.005$)

- **HERP with $THL = 0.005$:** in this version, the HERP starts the redundancy with the retransmission when the BER is higher than 0.005, but when BER is lower than 0.005, HERP uses only the retransmission without the redundancy.

Figure 10 shows the DFR of the two HERP versions, while varying the BER. We see in this figure that when BER is lower than 0.00001, the two HERP variants provide the same DFR, which means that the redundancy has not any utility on the HERP protection performance at this error level. On the other hand, When BER is higher than 0.00001, HERP with $THL = 0$ provides better DFR than HERP with $THL = 0.005$, which means that the use of the redundancy with the retransmission at this error level guarantees more protection of video frames than the use of the retransmission only. Based on these results and in order to reduce the network overload, we fix the THL value of the proposed HERP at 0.00001. When BER is lower than 0.00001, HERP applied only the retransmission, and when BER is higher than 0.00001 HERP applied both retransmission and redundancy. We remark in figure 10 that when BER is higher than 0.0005, the DFR of HERP with $THL = 0$ begins to decreasing, because at this error level the retransmission starts to avoid the transmission delay which effects the DFR of video stream. Based on this remark, the THH and THM values must be higher than 0.0005, in order to reduce the retransmission of video packets in function of their frame types (I, P, B). We have analyzed the following HERP variants to choose the THM and THH values. As mentioned above, the THL value is fixed at 0.00001.

- **HERP with ($THM = 0.0005$, $THH = 0.005$):** in this scenario, HERP stops to retransmt B-frame video packets when BER is higher than 0.0005 and it stops retransmitting P-frame video packets
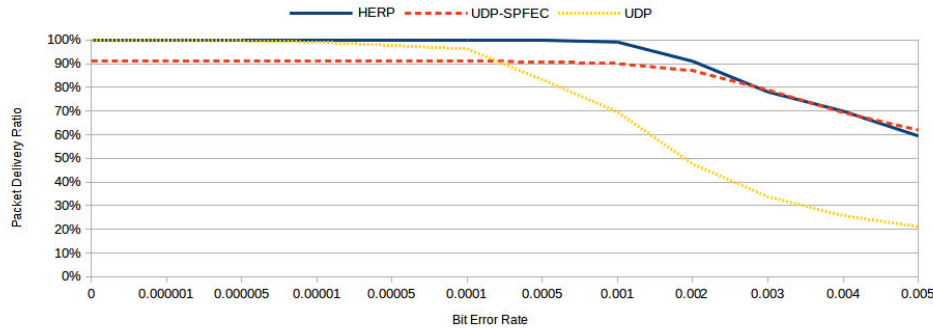
21

Figure 12: Variation of Packet Delivery Ratio with BER

when *BER* is higher than 0.005.

- **HERP with ($THM = 0.0005$, $THH = 0.0005$):** in this case, HERP stops the retransmission of B-frame and P-frame video packets when *BER* is higher than 0.0005.

- **HERP with ($THM = 0.005$, $THH = 0.005$):** HERP stops retransmitting both B-frame and P-frame video packets if *BER* is higher than 0.005.

Figure 11 depicts *DFR* of the three HERP variants according to *BER*. As shown in this figure, when *BER* is between 0.0005 and 0.002, HERP with ($THM = 0.0005$, $THH = 0.005$) provides a higher *DFR* than the other HERP variants. At this error interval, the HERP must avoid only the retransmission of video packets of B-frame to reduce the transmission delay and at the same time it allows the retransmission of the other I and P video frame packets to guarantee a high protection of video stream. When the *BER* is higher than 0.002, HERP with ($THM = 0.005$ and $THH = 0.005$) provides a best *DFR* value because it allows only the retransmission of I-frame video packets and it stops the retransmission of P-frame and B-frame video packets which improves the transmission delay and keep the HERP protection performance. According to these results, we have fixed *THM* value at 0.0005 and *THH* value at 0.002. When the *BER* is between 0.0005 and 0.002, HERP stops the retransmission of B-frame video packets, and when the *BER* is higher than 0.002, HERP ends the retransmission of P-frame video packets. We have also remarked that when *BER* = 0.005, the HERP with ($THM = 0.005$ and $THH = 0.005$) gives a higher *DFR* than the other HERP variants. In this case HERP must stop the retransmission of all video packet types.

4.4. Performance comparison

In this part, we compare HERP performance with UDP-SPFEC and UDP protocols and we discuss the obtained results. Figure 12 displays the result of Packet Delivery Ratio (*PDR*) on the y-axis, while the x-axis represents the *BER* varying from 0 to 0.005. As depicted in this figure, when the *BER* increases, the *PDR* decreases due to the lost video packets produced in the network due to transmission errors. When *BER* is lower than 0.002, HERP achieves higher *PDR* than UDP-SPFEC and UDP, because by means of the hybrid error recovery between the redundancy and retransmission, HERP can recover all types of lost packets in reason of network congestion, transmission errors and route disconnection, contrary to UDP-SPFEC which can only recover the uniform packet errors due to the transmission errors, and UDP that cannot recover any kind of lost packets. When *BER* is higher than 0.002, HERP and UDP-SPFEC provide the same *PDR*, because at this interval, HERP deactivates the retransmission of P-frame and B-frame video packets and it uses the same redundancy rate like UDP-SPFEC. Also, as shown in figure 12, UDP have not any error recovery mechanism but it achieves higher *PDR* than UDP-SPFEC when *BER* is lower than 0.0005, because at this interval error, UDP-SPFEC suffers form the congestion problem due to the transmitted redundant video sub-packets which increase the number of dropped packets.
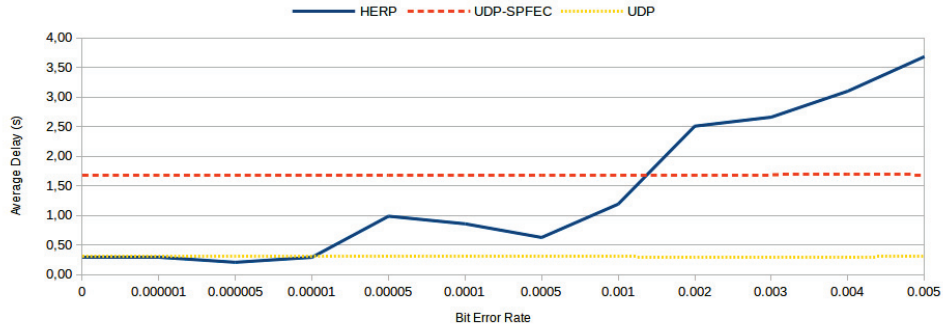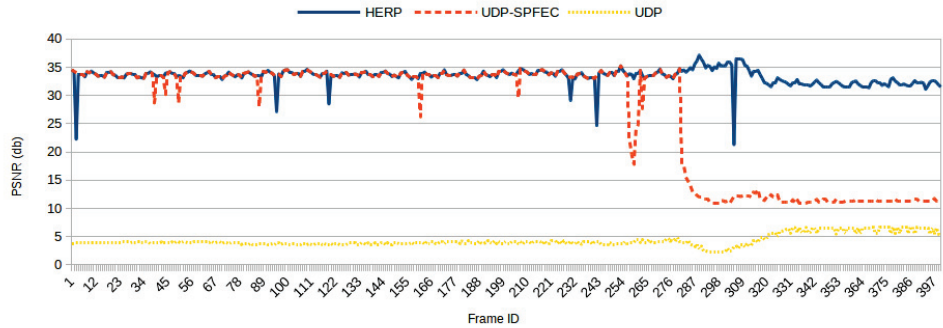
22

Figure 13: Variation of Average delay with BER

Figure 14: PSNR of video frames in the case of BER = 0.001

Figure 13 shows the average transmission delay achieved by each solution. When *BER* is lower than 0.002, HERP achieves lower average delay compared to UDP-SPFEC. The reason of the UDP-SPFEC limited performance at this interval error is that it suffers from the network overload which affects the transmission delay. Contrary, HERP achieves lower transmission delay because it does not use the maximum rate of the redundancy in order to avoid the network overload and does not reduce the transmission rate at the same interval error. In the other hand, when the *BER* is higher than 0.002, UDP-SPFEC provides lower average delay than HERP, due to the adaptive mechanism of HERP which decreases the transmission rate to avoid the congestion problem and due to the high number of I-frame video packets retransmissions. We note that HERP average delay does not exceed the time requirements defined by CISCO for video streaming [57], in which the delay should not be higher than 4 to 5 seconds. The figure 13 shows also that UDP achieves lower average delay than HERP and UDP-SPFEC while varying *BER*, because UDP does not suffer from the congestion problem like UDP-SPFEC and does not reduce the transmission rate like HERP. The *PSNR* of video frames achieved by each protocol is shown in the figure 14 when the *BER* is equal to 0.001. We can see that for all video frames, HERP provides higher *PSNR* against the other protocols due to its strength error protection. UDP provides lower *PSNR*, it does not adopt any error recovery technique. We remark that UDP-SPFEC provides lower *PSNR* values for the last video frames (from 287 to 400), because many video packets of these frames were losses due to the congestion or the route disconnection which are not tackled by UDP-SPFEC. Figure 15 illustrates the average *PSNR* of all video frames for the simulated protocols. When the *BER* is lower than 0.002, HERP achieves higher *PSNR* than UDP-SPFEC and UDP, because the *PDR* of HERP is higher than the other protocols, which provides higher *DFR*. When *BER* is higher than 0.002,
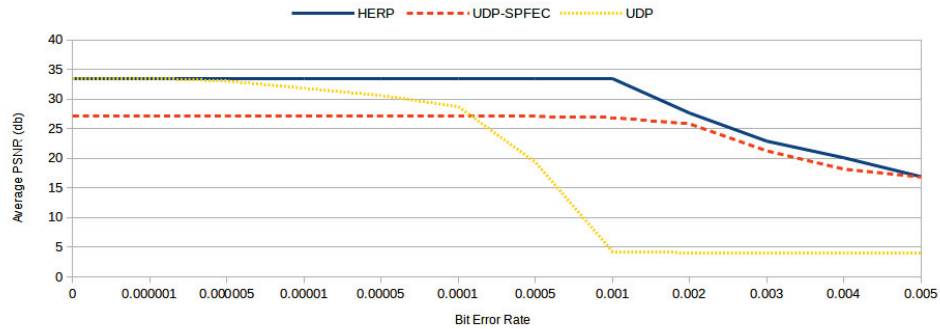
23

Figure 15: Variation of average PSNR of video frames with BER

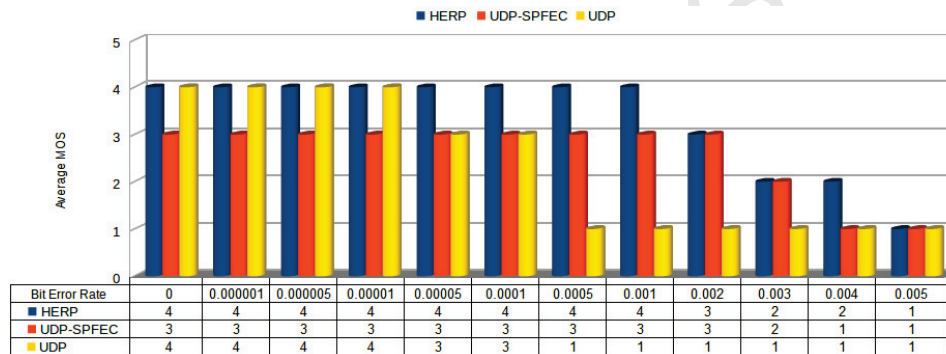| Bit Error Rate	0	0.000001	0.000005	0.00001	0.00005	0.0001	0.0005	0.001	0.002	0.003	0.004	0.005
HERP	4	4	4	4	4	4	4	4	3	2	2	1
UDP-SPFEC	3	3	3	3	3	3	3	3	3	2	1	1
UDP	4	4	4	4	3	3	1	1	1	1	1	1

Figure 16: Variation of average MOS of video frames with BER

HERP and UDP-SPFEC achieve almost the same *PSNR* because these two protocols provide the same *PDR* at this interval time which make the *DFR* almost the same. UDP provides the highest *PSNR* when *BER* is low, because it does not suffer from the network congestion like UDP-SPFEC, but it achieves lower *PSNR* when *BER* is high because it cannot recover the lost video packets like HERP and UDP-SPFEC.

The *MOS* QoE metric is presented in figure 16. In the case of *BER* lower than 0.0005, HERP and UDP provide a good video quality in terms of *MOS*, contrary to UDP-SPFEC which achieves fair *MOS* video quality, due to the weakness of UDP-SPFEC to deal the dropped packets problem. When *BER* value is chosen between 0.0005 and 0.002, HERP achieves a good *MOS* video quality than the other protocols, because it can recover uniform and burst video packets errors, which affect the video quality of experience. When *BER* value is between 0.002 and 0.005, HERP achieves a poor *MOS* video quality, but better than the other two protocols (i.e. UDP and UDP-SPFEC). In the case of *BER* higher than 0.005, all protocols achieve bad *MOS* video quality due to the frequent loss of video packets.

In figure 17, we have selected the transmitted video frame #281 when the *BER* equal to 0.001, aiming to give an idea of the user's point-of-view when he evaluates the video. Due to the robust protection mechanism of HERP that conceived to reach a higher protection for the video I-frames against the other frames, the transmitted video frame #281 is exposed to a lower distortion under a bad network condition. In contrast, the same frame is highly distorted with UDP-SPFEC and UDP protocols, which cannot recover all kind of lost packets like HERP under the same network condition.
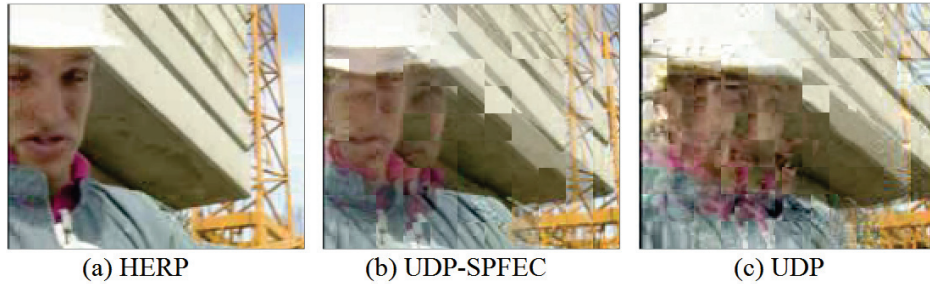
(a) HERP (b) UDP-SPFEC (c) UDP

Figure 17: Comparison between different simulated protocols at the frame #281

5. Conclusion

In this paper, we have proposed a Hybrid Error Recovery Protocol (HERP) to achieve a high quality and real-time of video streaming over VANET. HERP performs SPFEC mechanism to overcome the packet loss caused by the uniform transmission errors. This SPFEC is combined with the retransmission technique to recover the lost packets mainly due to the network congestion, route disconnection or due to successive packet transmission errors. HERP limits the number of packet retransmissions to respect the transmission delay constraint and it adapts dynamically at the sender level the redundancy rates and the limited number of retransmitted frames (I, P, B) based on the reports received periodically from the receiver vehicle. HERP offers high protection to the most important frames using the unequal protection of video packets according to the frame types (I, P and B). Simulations results showed that HERP provides better video streaming quality in terms of QoS and QoE metrics than UDP with SPFEC (UDP-SPFEC) and the conventional UDP protocol. We specify that HERP can recover all lost packet types unlike UDP-SPFEC, which can only recover lost packets due to the uniform transmission errors and unlike UDP, which does not apply any error recovery mechanism.

As a future research direction, we intend to improve the adaptive mechanism of HERP by using the metaheuristic methods to calculate the optimum values of the redundancy rate and the retransmission limit in order to enhance the video quality over VANET.

References

[1] L. Zhu, Z. Zhang, C. Xu, Security and Privacy Preservation in VANET, in Secure and Privacy-Preserving Data Communication in Internet of Things, Springer Singapore, 2017, pp. 53-76.

[2] Global status report on road safety 2015, World Health Organization.
http://www.who.int/violence_injury_prevention/road_safety_status/2015/en/, 2017 (accessed 9 August 2017).

[3] Z. Liu, M. Dong, B. Zhang, Y. Ji, Y. Tanaka, RMV: Real-Time Multi-View Video Streaming in Highway Vehicle Ad-Hoc Networks (VANETs), 2016 IEEE Global Communications Conference (GLOBECOM), 2016, pp. 1-6.

[4] M. Dixit, R. Kumar, A. Sagar, VANET: Architectures, research issues, routing protocols, and its applications, 2016 International Conference On Computing, Communication And Automation (ICCCA), 2016, pp. 555-561.

[5] S. Dharmaraja, R. Vinayak, K. Trivedi, Reliability and survivability of vehicular ad hoc networks: An analytical approach, Reliability Engineering & System Safety. 153 (2016) 28-38.

[6] J. Chi, S. Do, S. Park, Traffic flow-based roadside unit allocation strategy for VANET, 2016 International Conference On Big Data And Smart Computing (Bigcomp), 2016, pp. 245-250.

[7] H. Hasrouny, A. Samhat, C. Bassil, A. Laouiti, VANet security challenges and solutions: A survey, Vehicular Communications. 7 (2017) 7-20.

[8] S. Bitam, A. Mellouk, S. Zeadally, Bio-Inspired Routing Algorithms Survey for Vehicular Ad Hoc Networks, IEEE Communications Surveys & Tutorials. 17 (2015) 843-867.

[9] A. Rasheed, S. Gillani, S. Ajmal, A. Qayyum, Vehicular Ad Hoc Network (VANET): A Survey, Challenges, and Applications, Advances In Intelligent Systems And Computing. (2017) 39-51.

[10] M. Ben Brahim, Z. Hameed Mir, W. Znaidi, F. Filali, N. Hamdi, QoS-Aware Video Transmission Over Hybrid Wireless Network for Connected Vehicles, IEEE Access. 5 (2017) 8313-8323.

[11] N. Taherkhani, S. Pierre, Centralized and Localized Data Congestion Control Strategy for Vehicular Ad Hoc Networks Using a Machine Learning Clustering Algorithm, IEEE Transactions On Intelligent Transportation Systems. 17 (2016) 3275-3285.

[12] A. Huang, J. Bao, Transmission Performance Analysis for VANET Based on 802.11p, 2013 International Conference On Computational And Information Sciences, 2013, pp. 1178-1181.

[13] Ju-Hyung Son, Ji-Hoon Yun, Seung-Woo Seo, Chan-Seok Park, Performance evaluation of a telematics system using the ns (network simulator), 9Th Asia-Pacific Conference On Communications (IEEE Cat. No.03EX732), 2003, pp. 270-274.

[14] A. Argyriou, Real-time and rate–distortion optimized video streaming with TCP, Signal Processing: Image Communication. 22 (2007) 374-388.

[15] H. Luo, S. Ci, D. Wu, H. Tang, End-to-end optimized TCP-friendly rate control for real-time video streaming over wireless multi-hop networks, Journal Of Visual Communication And Image Representation. 21 (2010) 98-106.

[16] M. Tsai, C. Shieh, C. Ke, D. Deng, Sub-packet forward error correction mechanism for video streaming over wireless networks, Multimedia Tools And Applications. 47 (2009) 49-69.

[17] L. Beloqui Yuste, F. Boronat, M. Montagud, H. Melvin, Understanding Timelines Within MPEG Standards, IEEE Communications Surveys & Tutorials. 18 (2016) 368-400.

[18] A. Lo, G. Heijenk, I. Niemegeers, Evaluation of MPEG-4 Video Streaming over UMTS/WCDMA Dedicated Channels, First International Conference On Wireless Internet (WICON'05), 2005, pp. 182-189.

[19] I.E.G. Richardson, J. E. Richardson, H. 264 and MPEG-4 Video Compression, 2003.

[20] T. Wiegand, G. Sullivan, G. Bjontegaard, A. Luthra, Overview of the H.264/AVC video coding standard, IEEE Transactions On Circuits And Systems For Video Technology. 13 (2003) 560-576.

[21] G. Sullivan, J. Ohm, W. Han, T. Wiegand, Overview of the High Efficiency Video Coding (HEVC) Standard, IEEE Transactions On Circuits And Systems For Video Technology. 22 (2012) 1649-1668.

[22] A. Nafaa, T. Taleb, L. Murphy, Forward error correction strategies for media streaming over wireless networks, IEEE Communications Magazine. 46 (2008) 72-79.

[23] M. Claypool, Y. Zhu, Using interleaving to ameliorate the effects of packet loss in a video stream, 23rd International Conference on Distributed Computing Systems Workshops, 2003, pp. 508-513.

[24] M. S. Srouji, Z. Wang, J. Henkel, RDTS: A reliable erasure-coding based data transfer scheme for wireless sensor networks, 2011 IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS), 2011, pp. 481-488.

[25] A. Torres, P. Pinol, C. Calafate, J. Cano, P. Manzoni, Evaluating H.265 real-time video flooding quality in highway V2V environments, 2014 IEEE Wireless Communications And Networking Conference (WCNC), 2014, pp. 2716-2721.

[26] P. Pinol, A. Torres, O. Lopez, M. Martinez, M. Malumbres, Evaluating HEVC video delivery in VANET scenarios, 2013 IFIP Wireless Days (WD), 2013, pp. 1-6.

[27] N. Vineeth, H. Guruprasad, Performance analysis of network coded video streams in VANETs based on mobility models, 2015 IEEE International Advance Computing Conference (IACC), 2015, pp. 170-175.

[28] R. Immich, E. Cerqueira, M. Curado, Towards a QoE-driven mechanism for improved H.265 video delivery, 2016 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), 2016, pp. 1-8.

[29] S. Zaidi, S. Bitam, A. Mellouk, Enhanced Adaptive Sub-Packet Forward Error Correction Mechanism for Video Streaming in VANET, 2016 IEEE Global Communications Conference (GLOBECOM), 2016, pp. 1-6.

[30] S. Zaidi, S. Bitam, A. Mellouk, Enhanced user datagram protocol for video streaming in VANET, 2017 IEEE International Conference On Communications (ICC), 2017, pp. 1-6.

[31] P. Bucciol, J. Zechinelli-Martini, G. Vargas-Solar, Optimized Transmission of Loss Tolerant Information Streams for Real-Time Vehicle-to-Vehicle Communications, 2009 Mexican International Conference On Computer Science, 2009, pp. 142-145.

[32] C. Quadros, A. Santos, M. Gerla, E. Cerqueira, QoE-driven dissemination of real-time videos over vehicular networks, Computer Communications. 91 (2016) 133-147.

[33] C. Rezende, M. Almulla, A. Boukerche, The use of Erasure Coding for video streaming unicast over Vehicular Ad Hoc Networks, 38Th Annual IEEE Conference On Local Computer Networks, 2013, pp. 715-718.

[34] A. Mammeri, A. Boukerche, Z. Fang, Video Streaming Over Vehicular Ad Hoc Networks Using Erasure Coding, IEEE Systems Journal. 10 (2016) 785-796.

[35] H. Xie, A. Boukerche, A. Loureiro, A Multipath Video Streaming Solution for Vehicular Networks with Link Disjoint and Node-disjoint, IEEE Transactions On Parallel And Distributed Systems. 26 (2015) 3223-3235.

[36] H. Xie, A. Boukerche, A. Loureiro, MERVS: A Novel Multichannel Error Recovery Video Streaming Protocol for Vehicle Ad Hoc Networks, IEEE Transactions On Vehicular Technology. 65 (2016) 923-935.

[37] L. Khoukhi, S. Cherkaoui, A quality of service approach based on neural networks for mobile ad hoc networks, Second IFIP International Conference on Wireless and Optical Communications Networks, WOCN 2005, 2005, pp. 295-301.

[38] L. Khoukhi, S. Cherkaoui, FuzzyCCG: A fuzzy logic QoS approach for congestiosn control in wireless ad hoc networks, ACM international workshop on Quality of service & security in wireless and mobile networks, 2005, pp. 105-111.

[39] X. Zhu, Jingping Bi, Mingfu Li, Huanyu Liu, Adaptive video streaming uploading with moving prediction in VANETs scenarios, 2015 IEEE International Conference On Networking, Architecture And Storage (NAS), 2015, pp. 39-44.

[40] I. Zaimi, Z. Houssaini, A. Boushaba, M. Oumsis, An improved GPSR protocol to enhance the video quality transmission over vehicular ad hoc networks, 2016 International Conference On Wireless Networks And Mobile Communications (WINCOM), 2016, pp. 146-153.

[41] C. Rezende, A. Mammeri, A. Boukerche, A. Loureiro, A receiver-based video dissemination solution for vehicular networks with content transmissions decoupled from relay node selection, Ad Hoc Networks. 17 (2014) 1-17.

[42] C. Rezende, A. Boukerche, H. Ramos, A. Loureiro, A Reactive and Scalable Unicast Solution for Video Streaming over

VANETs, IEEE Transactions On Computers. 64 (2015) 614-626.

[43] C. Quadros, E. Cerqueira, A. Santos, J. Lim, M. Gerla, Beacon-less video streaming management for VANETs based on QoE and link-quality, 2015 IFIP/IEEE International Symposium On Integrated Network Management (IM), 2015, pp. 191-198.

[44] R. Wang, C. Rezende, H. Ramos, R. Pazzi, A. Boukerche, A. Loureiro, LIAITHON: A location-aware multipath video streaming scheme for urban vehicular networks, 2012 IEEE Symposium On Computers And Communications (ISCC), 2012, pp. 000436-000441.

[45] R. Wang, M. Almulla, C. Rezende, A. Boukerche, Video streaming over vehicular networks by a multiple path solution with error correction, 2014 IEEE International Conference On Communications (ICC), 2014, pp. 580-585.

[46] A. Torres, C. Calafate, J. Cano, P. Manzoni, Y. Ji, Evaluation of flooding schemes for real-time video transmission in VANETs, Ad Hoc Networks. 24 (2015) 3-20.

[47] G. Maia, C. Rezende, L. Villas, A. Boukerche, A. Viana, A. Aquino, A. Loureiro, Traffic aware video dissemination over vehicular ad hoc networks, Proceedings Of The 16Th ACM International Conference On Modeling, Analysis & Simulation Of Wireless And Mobile Systems - Mswim '13, 2013, pp. 419-426.

[48] A. Bradai, T. Ahmed, ReViV: Selective Rebroadcast Mechanism for Video Streaming over VANET, 2014 IEEE 79Th Vehicular Technology Conference (VTC Spring), 2014, pp. 1-6.

[49] M. Asefi, J. Mark, X. Shen, A Mobility-Aware and Quality-Driven Retransmission Limit Adaptation Scheme for Video Streaming over VANETs, IEEE Transactions On Wireless Communications. 11 (2012) 1817-1827.

[50] D. Jiang, L. Delgrossi, IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments, VTC Spring 2008 - IEEE Vehicular Technology Conference, 2008, pp. 2036-2040.

[51] M. Xing, L. Cai, Adaptive video streaming with inter-vehicle relay for highway VANET scenario, 2012 IEEE International Conference On Communications (ICC), 2012, pp. 5168-5172.

[52] The Network Simulator - ns-2, Isi.Edu. http://www.isi.edu/nsnam/ns/, 2017 (accessed 9 August 2017).

[53] J. Klaue, B. Rathke, A. Wolisz, Evalvid – a framework for video transmission and quality evaluation, the 13th International Conference on Modeling Techniques and Tools for Computer Performance Evaluation, 2003, pp. 255–272.

[54] M. Behrisch, L. Bieker, J. Erdmann, D. Krajzewicz, SUMO - simulation of urban mobility: An overview, Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation, 2011, pp. 63–68.

[55] OpenStreetMap. http://www.openstreetmap.org/, 2017 (accessed 9 August 2017).

[56] K. Piamrat, C. Viho, J. M. Bonnin, A. Ksentini, Quality of experience measurements for video streaming over wireless networks, Sixth International Conference Information Technology: New Generations, ITNG'09, 2009, pp. 1184-1189.

[57] T. Szigeti, C. Hattingh, End-to-End QoS Network Design: Quality of Service in LANs, WANs, and VPNs, Cisco Press, 2004.

27