Fuzzy Redundancy Adaptation and Joint Source - Network Coding for VANET Video Streaming

Alberto J. Gonzalez, Kayhan Zrar Ghafoor, Ramon Piney, Andre Rios, Jesus Alcober, Kamalrulnizam Abu Bakar

Fuzzy Redundancy Adaptation and Joint Source - Network Coding for VANET Video Streaming

Kayhan Zrar Ghafoor[1], Alberto J. Gonzalez[2] *, Ramon Piney[2], Andre Rios[2], Jesus Alcober[2], and Kamalrulnizam Abu Bakar[1]

[1] Universiti Teknologi Malaysia,
Faculty of Computer Science and Information Systems,
81310 Skudai, Johor D. T,Malaysia,
kayhan@ieee.org
[2] Technical University of Catalonia/I2cat Foundation,
Department of Telematic Engineering,
08860 Esteve Terradas 7, Castelldefels, Spain
{alberto.jose.gonzalez, ramon.piney, andre.rios, jesus.alcober}@upc.edu

Abstract. Video communication VANETs has many applications, such as emergency video transmission or inter-vehicle entertainment. However, delivering video to high mobile vehicles faces challenges such as packet loss due to intermittent connectivity and channel variations. Thus, designing a reliable approach for video streaming over such harsh scenarios is needed. Hence, we propose a reliable approach based on random practical network coding and source coding. This approach integrates the benefits of network coding with Multiple Description Coding to achieve robust streaming over VANET. Furthermore, we propose a redundancy controller based on fuzzy inference system to adjust the amount of redundant packets based on vehicular traffic density and SNR of the channel. Simulation shows the proposed approach achieves better protection against packet loss, application layer throughput and video quality.

Keywords: VANET, video streaming, network coding, Multiple Description Coding, fuzzy inference system

1 Introduction

VANETs integrate ad hoc network and WLAN to achieve vehicle to vehicle communications. This inter-vehicle communication fosters the deployment of innovative wireless communication applications based on real time media streaming. These communications have several applications ranging from road safety to entertainment. Furthermore, IEEE 802.11 standard supports video communication over vehicular networks [1]. These networks deal with different traffic scenarios, such as during late night time, rush hour, dense and sparse traffic (e.g. in highways), which cause unstable vehicular network topology. Thus, the

---

communication link will last for a short time due to high mobility of vehicles and dynamic topology. On the other hand, the communication channel between vehicles are prone to packet loss due to high variations (low SNR) together with different forms of fading.

In video communication between vehicles, random and bursty errors may be frequent. One bit error may extend over several packets due to dependency among the frames of a compressed stream. These bursts of errors are accepted [2] as are more severe than random errors. One way of making reliable and robust video communication between vehicles is by using error resilient approaches like Network Coding (NC). In [3] showed that NC can provide robustness and high throughput in wired or wireless networks. The upshot of NC is encoding incoming packets at intermediate nodes. The receiver starts decoding once enough number of coded packets are received. However, a single packet loss can cause a generation of coded packets to be lost. Therefore, NC should be used with a proper error resilient approach for video streaming over VANET-like scenarios. In addition, NC has the potential to increase robustness against packet losses in wireless ad hoc networks [4]. The error resilient capability of NC depends on the number of encoded packets which are produced by intermediate nodes. The usefulness of this redundancy might be varied with vehicular network traffic conditions such as sparse and dense scenarios. In sparse scenarios, number of redundant packets should be increased appropriately to compensate packet losses while in dense scenarios it should be decreased to mitigate bandwidth overload. Thus, we can optimize the redundancy of NC considering vehicular traffic density. Redundancy also can be tuned with error rate of the communication channel. We propose a redundancy controller based on a fuzzy inference system [5], which can effectively adjust the amount of redundant packets to the desired target value.

Multiple Description Coding (MDC) encodes a video file into a number of different independent sub-streams.Thus, MDC can be used to reduce the effects of packet loss [6]. Unlike previous works, which not consider coded packet loss problem of NC, inefficiency of MDC and adjusting the NC redundancy with respect to vehicular traffic conditions and channel error rate variations, we propose fuzzy based NC redundancy control and combined random NC with MDC to achieve reliable video streaming over VANET.

This paper is organized as follows. Section 2 describes related work and details the proposed joint source-network coding. In section 3, we propose NC fuzzy controller based on fuzzy logic. The vehicles communication model is described in section 4. Video transmission end to end delay is investigated in section 5. A simulation setup and performance evaluations are highlighted in section 6. Finally, this paper ends with a conclusion in section 7.

2 Related Work

In recent years, several protocols, approaches and mechanisms are developed for vehicular networks. However, almost all of them deal with short message

communication between vehicles. In [7], investigated an architecture for video streaming over VANET whereas no real video data are used in their simulation. In [8], two routing protocols are discussed: Source Based Forwarding (SBF) and Receiver Based Forwarding (RBF). They have used real video data to simulate video streaming over VANET. They also used different traffic conditions for forwarding data between vehicles.

In addition, the challenge of video streaming over VANET can be interpreted by the high channel errors in vehicular traffic conditions. The high packet loss and limited communication range of the vehicles incur frequent link disconnection and uneven network partition. Authors in [9] have combined data mulling technique with three strategies, NC, erasure coding, and repetition coding. Specially, vehicles in the opposite direction are exploited as data mules to relay multimedia data to other vehicles to overcome intermittent connectivity in sparse vehicular scenarios. However, the analysis of the delay for relaying multimedia data is based on mathematical model. Literature on fuzzy inference system exist in variety disciplines of wireless ad hoc networks. Some authors [10] have used entropy based fuzzy logic to construct the new metric-entropy and fuzzy controllers with the help of entropy metric to reduce the number of route reconstruction so as to provide QoS guarantee in the ad hoc networks. T. C Chiang et al. in [11] have introduced fuzzy Petri net agent to be implemented in each node to learn and to adjust itself to fit the dynamic conditions in a multicast ad hoc networks. In our fuzzy based redundancy controller, two metrics has been used to adjust the value of redundant packets.

2.1 Multiple Description Coding Approach

Video coding schemes such as MDC [12] or Scalable Video Coding (SVC) are used to enhance error resiliency. SVC divides the video file into layers referred to as base layer and enhancement layers. The base layer is the most important layer while the enhancement layers are referenced to the base layer. The enhancement layers can not reproduced independently to the base layer. In contrast to SVC, MDC splits (Figure 1) the video signal into multiple sub-streams where each of the sub-streams (descriptors) can be reproduced independently. The receiver can make a reconstruction of the video file when any of the sub-streams is received. The quality of the reconstructed video file is proportional to the number of descriptors received. Thus, the more descriptors are received, the better the perceived video quality. Therefore, MDC is preferable in extreme environments like VANET because it provides increased resilience to packet losses by multiple streams that can be decoded independently. The main parts of MDC system are the splitter and the merger. The splitter is implemented in the source vehicle. The first step of MDC encoder is the pre-processing of the video stream in order to obtain the different N descriptions at the video source, which can be coded and displayed independently. Next, each encoded descriptor is ready to be transmitted.
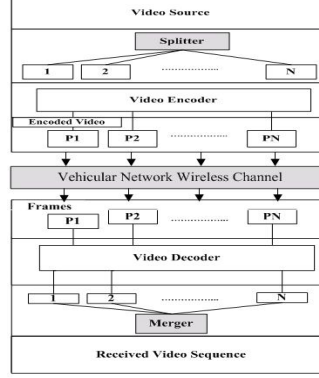
Fig. 1. MDC encoding and decoding system.

2.2 Random Linear Network Coding Approach

This work proposes to use Random Linear Network Coding (RLNC). When NC is applied in the video source, the video packets are divided into generations with the same decoding deadline [13]. This approach regards each generation as a vector and applies linear combinations before transmitting the native packets. In more detail, let $N_1, N_2, N_3, , N_K$ K native packets are in a generation. The video source applies linear combination on these K native packets, the output is j coded packets which are referred to as $Y_1, Y_2, Y_3, \cdots, Y_j$, where $Y_j = \sum_{i=1}^{K} e_{i,j} N_{i,j}$ and $j = 1, 2, 3, \cdots, h$. The set of coefficients $e_{j,i} = (e_{j,1}, e_{j,2}, e_{j,3}, \cdots, e_{j,K}) \in GF(2^q)$ denotes the encoding vector which are randomly selected at the source node to encode the native packets.

Intermediate nodes collect the encoded packets of the specific generation, and then re-encode the stored packets. When an intermediate node has received u encoded packets of a given generation, then it will select a new brand of coefficients $\zeta_f = (\zeta_{f,1}, \zeta_{f,2}, \zeta_{f,3}, \cdots, \zeta_{f,u})$ from $GF(2q)$. Thus, the outgoing packets of intermediate nodes are $X_1, X_2, X_3, \cdots, X_k$, where $X_k = \sum_{i=1}^{u} zeta_{f,i} Y_i$. It is worth to observe that the new brand of coefficients with respect to the native packets is $\gamma_k = (\gamma_{k,1}, \gamma_{k,2}, \gamma_{k,3}, \cdots, \gamma_{k,j})$ where $\gamma_{k,f} = \sum_{i=1}^{u} e_{j,i} zeta_{f,i}$,then the intermediate nodes embed it in the header of outgoing re-encoded packets.

When the destination has received K coded packets, it first checks the linear independence of the coefficients in the header of the received coded packets. Next, it builds $K \times K$ matrix using the coded packets. The rows of the formed matrix are represents the coefficients of coded packets. Afterwards, it first computes the inverse of the coefficients matrix and then multiplies it with the received coded packets.

2.3 Proposed Joint Source Coding and Network Coding Approach

The wireless nodes are communicating based on TCP/IP including 802.11 MAC layer. In the protocol stack each protocol layer adds all necessary information to the incoming data. In application layer, the specific header is added whenever the data is passed to the next layer. The source and destination IP addresses and port numbers are added by network and transport layers respectively. These addresses ensure end to end delivery of the packet. In case of link disconnection the network layer drops the packet. On the other hand, MAC and physical layers deals with nearby host communications. Fast channel variation at physical layer will produce packet loss at other layers. Thus, in order to increase robustness of video streaming over VANET, the protocol stack should be modified by embedding coding approaches so that it can tackle packet losses. This can be achieved by joint source coding and NC for video streaming.

The overall block diagram of the proposed joint source-network coding is depicted in Figure 2. In our video streaming environment, a single video source transmits descriptions to a group of receivers over VANET. The MDC is applied for robust transmission over error- prone VANET. The MDC generates several descriptions which are independently decodable. These descriptions are passed down to the next protocol layers for end to end delivery. On the other hand, we implemented the RLNC between network and MAC layer running over an IEEE 802.11 MAC protocol. The reason of this embedding are: 1) processing at lower layers (physical and MAC layers) are faster than upper ones, 2) provide transparent NC operations to upper protocol layers. At the NC shim layer, each description is divided into numbers of generations, this division depends upon the size of generations and MDC descriptions. The outgoing coded packets are recognized by adding generation-ID and description-ID into its header. This information is useful for the receiver to merge generations and descriptions at the NC and application layers respectively. If receivers collects enough linearly independent coded packets, the decoder part of NC can obtain original data belonging to that generation.

It is well known fact that there is a level of dependency among video frames; loss of one frame may cause other frames to be useless and undecodable. In other words, in certain network conditions a receiver may not decode the original packets which are belonging to the same generation, because it does not receive enough innovative packets based on its maximum flow capacity. In these network conditions, MDC can compensate the effect of coded packet loss by decoding the video, albeit with lower data rate. This is because one description is divided to multiple generations while in NC loss of one generation severely affect the whole video file due to tight correlation between generations to reproduce original video file. Therefore, the reliable approach is proposed to improve the video streaming service by reducing the multiple packet losses over VANET.
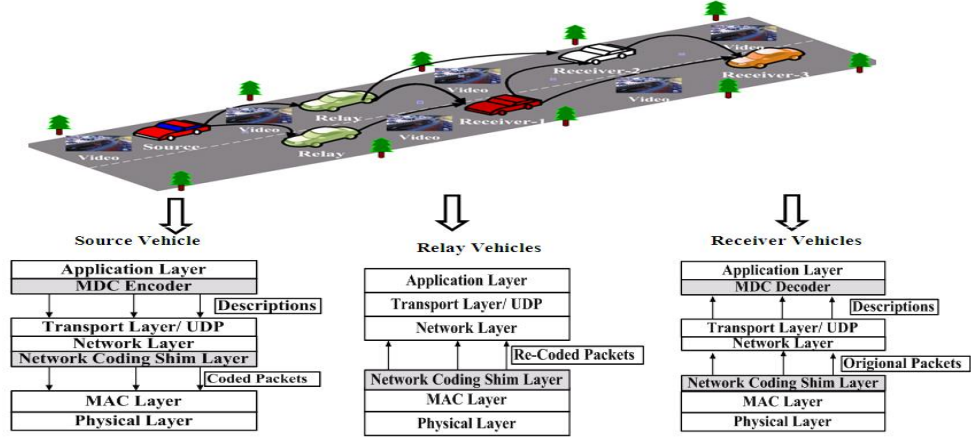
Fig. 2. Proposed joint source-network coding (dark blocks).

3   Proposed Fuzzy Logic Redundancy Control

We know that fuzzy logic simulates the interpretation of uncertain sensory information as sensed by human brain.The appropriate decision making of fuzzy inference system depends on the precise design of membership functions and fuzzy inference rules. However, in VANET the right decision making is difficult due to uncertain movement of vehicles and SNR value of the communication channel.Therefore, fuzzy inference system has been applied to control the amount of redundant packets for improving network load.Concretely, the first step of designing a fuzzy controller is to determine the impact of each input/output parameters and the range of values that each variable can have. The next step is to determine membership functions to the input and output parameters based on the defined range and to design a rule base for the fuzzy inference system. One main characteristic of the fuzzy controller is that it is modifiable. It is easy to tune rules, membership functions, value range or even change the system parameters to enhance its performance. Selection of the most relevant input variables is important.Furthermore, the fuzzy inference system consists of fuzzification , knowledge rule base and defuzzification. A group of rules can be used to represent knowledge base for articulating the control action in linguistic form. The inputs to the fuzzy controller to be designed for adjusting the amount of redundancy are: i) SNR of the communication channel and ii)vehicular traffic density. These two variables make the node's to optimize and adjust the amount of redundancy and then improves the network performance (bandwidth utilization). The overall process involved in calculation of redundant packets is described next.

3.1   Fuzzification of Inputs and Outputs

In order to control the amount of redundant packets,which are produced by using NC operations, the source and intermediate nodes encode m received packets

out of all K packets it has received so far [14]. The ratio m/K is refereed to as the coding density (output parameter) and it is used to adjust the degree of redundancy as a function of two input parameters (traffic density and SNR).

The redundancy controller is based on the fuzzy logic. This is because it simplifies the design of stable controller which employed two input variables and also the alterations are easy to perform to the rule inference system in order to enhance the performance of the controller. It is worth to mention the developed controller is implemented in source node and intermediate nodes in a distributed way. We are using traffic density and communication channel SNR in our redundancy controller (see Figure 3 a,b,c). The reason of this selection is that higher traffic density variation in vehicular environment yields high communication channel fluctuation, hence change of SNR. All nodes in a distributed way estimate the channel condition in the form of channel SNR. Furthermore, each node calculates the traffic density based on the proposed method in [15] (see section V). The specific node counts its neighbouring nodes based on the handshaking between them and then adjusts the value of redundancy using fuzzy controller. Moreover, the triangular and trapezoidal functions are chosen as the membership function since they have been extensively used in real-time applications due to their simple formulas and computational efficiency.

3.2 Fuzzy Inference Engine

The fuzzy controller is designed based on 9 rules which are presented in Table I. Channel SNR is presented on a horizontal axes and traffic density is presented on a vertical axes.In order to demonstrate the designed fuzzy controller, One rule is used to show how the algorithm works and the outputs of each rule are combined for generating the controller decision [5].Consider a rule "If SNR is Low and Traffic Density is very sparse, the change in coding density is very high" as an example of calculating output of the specified rule. Our fuzzy inference system considers a case where SNR is 2.22 dB and traffic density is 12.9 car/km(this value is increased to 13). The output is 0.891(0.9).This value (0.9) represents the coding density, i.e the intermediate nodes encode 9 packets (in our simulation 10 is total number of packets in each generation) out of all received packets. This high redundancy is because low value of SNR and sparse traffic density.This output is achieved by using Mamdani's fuzzy inference method [5]. Figure 4 depicts the tuning behavior of the redundancy controller.

3.3 Defuzzification

Defuzzification refers to the way a crisp value is extracted from a fuzzy set as a representation value. Our fuzzy controller takes the centroid of area strategy for defuzzification.
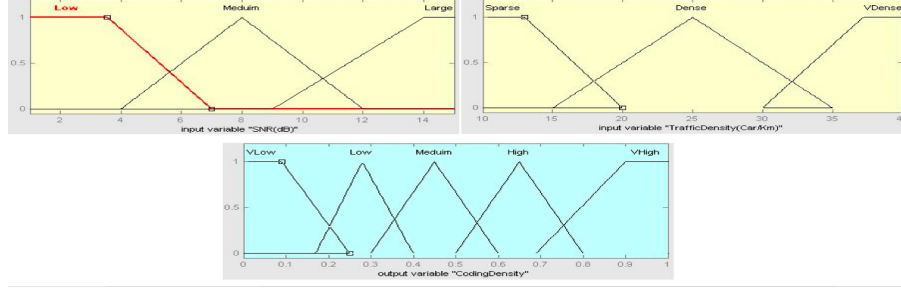
Fig. 3. Membership function for a) SNR ; b) Traffic Density; c) Coding Density

Table 1. Rule-based redundancy control

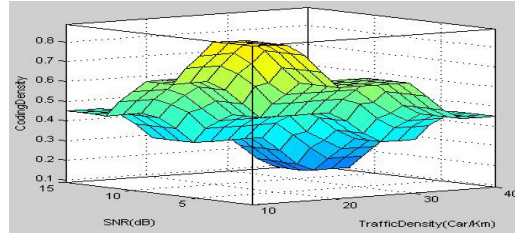| TrafficDensity / SNR	Low	Medium	Large
Sparse	VHigh	High	Medium
Dense	High	Medium	Low
VDense	Medium	Low	VLow

Fig. 4. Correlation between input and output parameters.

4 Delay Constraint of Video Transmission

Consider maximum number of video packets,M, that can be transmitted in a communication session (source to destination) and satisfy maximum allowable delay T. The overall video packet delay (T) consists of three components: 1) the time that it takes for encoding and decoding of generations in NC operations T_{NC}, 2) the time needed to split the video file at the source and merge it at the destination T_{MDC}; and 3) the delay introduced due to video packet transmission from sender to receivers, i.e., the sum of packetization delay, propagation delays over intermediate links and queuing delays in intermediate vehicles T_t.   In order to fulfil a video decoding deadline constraint, the worst case scenarios should be considered when calculating maximum allowable video packet delay. Thus, it is necessary to calculate each delay during a communication session.

   The delay caused by NC operations is incurred at the encoder and decoder $T_{NC} = T_{Encoding} + T_{Decoding}$. This encoding and decoding delay depends on the number of native packets in each generation. The descriptions are divided

Parameters	Description
n	total number of packets for a given generation
m	encoded packets out of n native packets
f	frame rate (number of frames per second)
$1/(f \times n)$	inter-arrival time between successive native packets
m/n	coding density
z	time needed to inverse coefficient matrix
$T_{Waiting}$	time needed to collect enough coded packets

to a number generations of the same size, referred to as the native generations $(g_1, g_2, g_3, \cdots, g_r)$. The time needed to encode the native packets of each generation is denoted by $T_{Encoding}$ while $T_{Decoding}$ is the time to decode all collected coded packets at the receiver. The decoding delay is shown in (2) under the assumption that the packets are uniformly and periodically received over a frame.

$$T_{Decoding} = T_{Waiting} + T_{Retrieval}. \tag{1}$$

$$T_{Decoding} = (\frac{m}{n} \times \frac{1}{f \times n}) + z \tag{2}$$

For encoding packets at the source and intermediate nodes, it is important to collect enough number of packets belonging to the same generation to produce coded packets. Therefore, the encoding time $T_{Encoding}$ depends on the number of packets of a given generation. Table II shows all parameters related to delay.

The main parts of the MDC source coding are the splitter and merger which provide particular processing of the video sequences at the source and destinations respectively. Thus, the time needed for MDC operations is as follow:

$$T_{MDC} = T_{Splitter} + T_{Merger} \tag{3}$$

Therefore, delay parameters should be chosen to satisfy the delay constraint:

$$T \geq T_{MDC} + T_{NC} + T_t \tag{4}$$

5 Performance Evaluation

5.1 Simulation Setup

In this section, we present simulation setup to evaluate the performance of the proposed reliable joint coding approach. The simulation is performed using NS-2 simulator (with default values for MAC and physical layer configuration parameters) integrated with Evalvid [16]. Furthermore, the MDC simulation has been experimented to generate video descriptions [17]. Two different video streaming

approaches have been experimented: one was proposed joint coding approach (JSNC) and the other was the NC approach.The SUMO tool [18] has been employed to create the vehicular scenario, and to generate mobility traces of the simulated vehicles. SUMO interposes vehicles in each lanes at a given traffic rate. This mobility model allows us to simulate vehicular scenarios such as speed deceleration/accelerations and different vehicular traffic conditions. The vehicle which is assigned as video source is generated first in one route (each route has its own parameters such as speed, acceleration).

All vehicles in our simulations have a transmission range of 200 meter and 6 Mbps channel capacity. The roadway used is a two-lane road of 5 km length. Vehicles enter the road according to a Poisson distribution and travel at a speed between (10-23) m/second. The simulation is run for 100 seconds, resulting in a total of 125 vehicles generated (Figure 2). In the simulation, the CIF bit stream has been used with resolution 352x288 pixels and 300 frames (Foreman). The CIF Foreman sequence is encoded with a quality 35.68 dB. The encoder includes three types of frames: Intra frames (I-frames) that are independent on other frames, and inter frames that use either P-frames or B-frames temporal prediction from other frames. A number of mutually dependent frames form a group of pictures (GOP). In our experiment, the GOP size is set to 12 frames.

5.2 Simulation Results and Discussions

We consider three different metrics in our evaluation: average Peak Signal to Noise Ratio (PSNR); protection efficiency: average packet loss rate, which represents the robustness of the protocol against packet losses; and application layer throughput: this throughput depends upon packet losses at the lower layers.

Figure 5 a) shows that the overall video quality is higher for proposed approach than that of NC approach. When vehicles are travelling with low speed, the value of PSNR is slight different for both approaches. This is because at low velocity the wireless channel is stable and intermittent connectivity occurs less frequently. As velocity of vehicles increases, the disconnectivity rate between source vehicle and receiver increases frequently as well as the high channel variation leads to coded packet loss. A coded packet loss in one generation causes multiple packet loss of it. As result, the average video quality of the proposed approach is higher than that of the NC approach.

To compare the packet loss rate of the proposed approach with other approach, the video source transmits video packets in a given period of video communication session. The source vehicle transmits the video coded packets from source to receiver by applying NC operations. In the next communication session, we have repeated the video transmission by applying the proposed approach. Subsequently, during off line analysis, we have computed the average packet loss rate through a comparison of receiver and source trace files. We observed that the instantaneous packet loss rate is increasing in both approaches and the overall packet loss rate is lower for proposed approach than that of NC approach as depicted in Figure 5 b). Initially, there is a small difference of
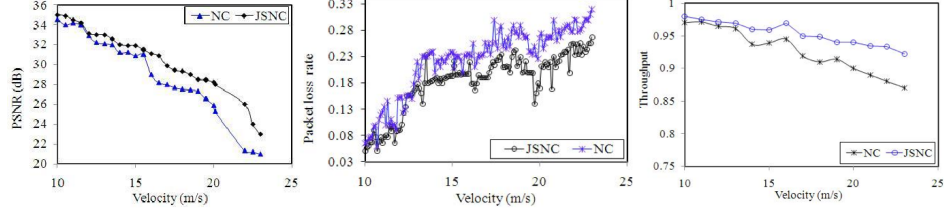
Fig. 5. a)Video quality for different vehicle speeds; b)Instantaneous packet loss rate for different vehicle speeds; c)Application layer throughput for different vehicle speeds

instantaneous packet loss values for the proposed approach and networks coding approach due to less channel variation (less bit error rate and fading of the channel) as well as less frequent link disruption and hence less number of packet losses. As mobility increases the channel instability (high bit error) and intermittent connectivity increases as well. As a result packet loss increases in both approaches. However, the NC has more coded packet loss than proposed approach at high speed. This is because one coded packet losses has a potential to generate multiple coded packet losses. On the other hand, the MDC increases error resilient of NC by generating multiple descriptions.

We observed similar trends in application layer throughput as we vary the velocity from low to high in Figure 5 c). The NC approach shows significant decrease in throughput due to high speed of vehicles which leads to high coded packet losses, while application layer throughput in our approach does not change as much at high velocity of vehicles. This is because high coded packet losses (due to high speed) at the lower layers reduces effective application layer throughput as well as increases the variation of bandwidth.

6 Conclusions

In this paper, we have proposed joint source coding and NC for reliable video streaming over vehicular networks. In this approach, the NC operations are implemented above MAC layer while MDC operations are embedded in application layer. The MDC descriptions are adapted with NC generations. This two stage of coding increases the robustness of video streaming in vehicular networks. Moreover, the NC redundancy are adjusted using fuzzy inference system controller, the tuning was based on vehicular traffic density and SNR of the communication channel .The mathematical analysis for latency is derived to constrain end to end delay. NS-2 extensive simulations were successfully carried out for significant gains in terms of average video quality, application layer throughput and average packet loss rate. The video streaming with NC has been compared to the proposed joint coding approach and the proposed scheme gives better performance.

References

1. Singh, J. P., Bambos, N., Srinivasan, B., Clawin, D.: Wireless LAN performance under Varied Stress Conditions in Vehicular Traffic Scenarios, IEEE International Vehicular Technology Conference, vol.2, pp. 743-747 (2002).
2. Lee, C. W., Yang, C. S., Su, Y. C.: Adaptive UEP and packet size assignment for scalable video transmission over burst error channels, EURASIP Journal on Applied Signal Processing, vol 2006, pp.1-9 (2006).
3. Ahlswede, R., Cai, N., Li, S., Yeung, R.: Network information flow, IEEE Transactions on Information Theory, pp. 1204-1215 (2000).
4. Guo, Z., Wang, B., Xie, P., Zeng, W., Cui, J.H.: Efficient Error Recovery with Netowrk Coding in Under Water Sensor Networks, Journal on Ad Hoc Networks, Vol. 7 , pp. 791-802 (2008).
5. Mamdani, E.H.: Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis, IEEE Trans. on Computers, vol. C-26, pp. 1182-1191 (1977).
6. Goyal, V. K.: Multiple description coding: compression meets the network, IEEE Signal Processing Magazine, pp. 74-93 (2001).
7. Gao, M., Ammar, M.H., Zegura, E.W.: V3: a Vehicle-to-Vehicle Live Video Streaming Architecture, IEEE International Conference on Pervasive Computing and Communications, Kauai, HI, pp. 404-424 (2005).
8. Xie, F., Hua, K. A., Wang, W., Ho, Y. H.,: Performance study of live video streaming over highway vehicular Ad hoc networks, International IEEE Vehicular Technology Conference, USA, pp. 2121-2125 (2007).
9. Park, J. S., Lee, U., Oh, S., Gerla, M.: Delay analysis of car-to-car reliable data delivery strategies based on data mulling with network coding, IEICE Transaction on Information and Systems, pp. 2524-2527 (2008).
10. Chen, H., Sun, B., Zeng, Y., He, X.: An Entropy-Based Fuzzy Controllers QoS Routing Algorithm in MANET, International Conference on Hybrid Intelligent Systems, pp. 235-239 (2009).
11. Chiang, T. C., Tai, C. F., Hou, T. W.: A knowledge-based Inference Multicast Protocol using Adaptive Fuzzy Petri nets, Journal of Expert Systems with Applications, vol. 36, pp. 8115-8123 (2009).
12. Wang, Y., Reibman, A., Lin, S.: Multiple Description Coding for Video Delivery, Proceedings of the IEEE, vol. 93, no. 1, pp. 57-70 (2005).
13. Chou, P. A., Wu, Y., Jain, K.: Practical Network Coding, Annual Alerton Conference on Communication, Control and Computing, pp. 40-49 (2003).
14. Wang, M., Li, B.: How Practical is Network Coding, IEEE International Workshop on Quality of Service, pp. 274-278 (2006).
15. Balon, N., Guo, J.: Increasing Broadcast Reliability in Vehicular Ad Hoc Networks, VANET'06, Los Angeles,California, pp. 104-105 (2006).
16. Evalvid, `htltp://www.tkn.tu-berlin.de/research/evalvid/,accessed: 3March,2011.`
17. Multiple Description Coding (MDC), `http://140.116.72.80/~smallko/ns2/MDC. htm,accessed:7~March,2011.`
18. SUMO Mobility Model,`http://sumo.sourceforge.net/,accessed:5March,2011.`