Voice over Internet Protocol in Wireless Mesh Networks with Opportunistic Network Coding

Erik Pertovt, Kemal Alič, Aleš Švigelj, Mihael Mohorčič
Department of Communication Systems
Jožef Stefan Institute
Ljubljana, Slovenia
erik.pertovt@ijs.si

Abstract—**In wireless networks, network coding can be used to enhance the performance of the network. By combining multiple packets into one encoded packet the network throughput is increased and the delay is lowered. In this paper, we perform the evaluation of the Voice over Internet protocol (VoIP) application in wireless mesh networks (WMNs) using network coding. For VoIP, various codecs can be used based on the Quality of Service (QoS) requirements to perform digitalization, encoding and packetization of the analog voice signal. We use various codecs and apply opportunistic network coding to observe how the network coding influences the average network delay and jitter. Moreover, we analyze the VoIP performance at the level of individual calls in order to compare the performance of calls with the International Telecommunication Union-Telecommunication (ITU-T) QoS recommendations. Network delay, End-to-End (ETE) packet delay, and packet delay variation, are investigated for each call. One of the main QoS requirements is that ETE packet delay and packet delay variation should be lower than thresholds recommended by ITU-T organization, as packets have to be transmitted through the network in real time.**

Keywords—Voice over Internet protocol; network coding; wireless mesh networks; quality of service; performance evaluation

I. INTRODUCTION

With the transition from analog to digital networks also some of traditional and by nature analog applications, most notably the telephony and related voice applications, had to adapt to new paradigms in the telecommunication networks. In this respect, the Voice over Internet protocol (VoIP) application is dealing with delivery of voice and multimedia sessions over the packet-switched broadband Internet protocol (IP) networks in real time [1]. The voice signal is split into VoIP packets based on various used codecs, which consider different Quality of Service (QoS) requirements [2]. VoIP packets are transmitted with other IP packets over the network.

Wireless mesh networks (WMNs) are typical representatives of wireless packet-switched IP networks, where nodes are connected to each other through multi-hop wireless links forming a wireless access/backbone network [3, 4]. In order to improve the WMNs performance, various mechanisms are used. Among the promising mechanisms, which experience an increasing attention, is also network coding [5]. Instead of using "classical" receive and forward mechanism for packets, network coding combines multiple received packets either from the same or from different traffic flows into one encoded packet and then forwards it in order to increase the network capacity. In wireless networks, network coding exploits the broadcast nature of the wireless medium, where nodes can overhear packets, which are not destined to them, resulting in new coding opportunities, which enable combining even more packets together [6]. A practical network coding procedure, COPE, is proposed in [7] that encodes two or more packets in a single transmission based on the nodes knowledge on what information (i.e., which packets) is available in the neighboring nodes. The procedure was tested in a real WMN deployment, which is of a particular importance [8].

VoIP application is highly exposed to QoS impairment in wireless IP networks, such as WMNs [9, 10], even when using QoS enforcement [11]. QoS performance of VoIP can be improved with various mechanisms. As VoIP is a real-time application and requires specific QoS, the benefits of using the novel mechanisms for VoIP have to be tested.

In this paper, we investigate the use of network coding for VoIP application with various codecs in WMNs. The rest of the paper is structured as follows. In Section II, the VoIP application is presented with the emphasis on various VoIP characteristics and VoIP QoS requirements. The VoIP packet structure is presented in Section III and VoIP traffic parameters for various codecs are calculated. In Section IV, VoIP QoS parameters are presented such as the one-way transmission time or End-to-End (ETE) delay, packet delay variation or jitter, and packet loss rate. We also show various codecs delay characteristics. In Section V, we explain network coding in WMNs used to decrease the network delay. In Section VI, we present the network coding simulation model used to perform extensive simulations to evaluate the performance, reported in Section VII, of various VoIP codecs in terms of an average network delay and jitter. We also analyze the QoS VoIP performance with network coding by examining a delay and jitter per packet at the level of individual calls and compare it with the recommended QoS requirements by the International Telecommunication Union-Telecommunication (ITU-T) organization. In Section VIII, we summarize the work and give the conclusions.

II. VOICE OVER INTERNET PROTOCOL APPLICATION

For enabling the VoIP application broadband internet connections for voice applications instead of, e.g., using public switched telephone network (PSTN) over analog telephone lines, various protocols and standards have been developed for the support of different VoIP functionalities. Among them are well known H.323, Session Initiation Protocol (SIP), Real-Time Protocol (RTP), Media Gateway Control Protocol (MGPC), Session Description Protocol (SDP), and Inter-Asterisk eXchange (IAX).

As the Internet is a packet-switched network, the voice signal of the VoIP telephone call has to be packetized before being sent through the network. The packetization is the process of splitting the stream data into structured blocks, called packets [12]. The packetization of the voice has to consider the fact that real time delivery of packets has to be performed [13]. For this, different types of codecs (coder/decoder) exist. Codec is a coding/decoding device which samples a voice signal and transforms it into a digitalized form with a predefined bit rate [14]. It also compresses the data of the signal to reduce the bandwidth requirements of established call. Codec selection is driven by finding a trade-off between the bandwidth efficiency and the quality (compression level) of transmitted VoIP calls [15]. Some of the most frequently used (standard) codecs for VoIP packet transmissions are G.711, G.722, G.723, G.726, G.728 and G.729 [16].

It has to be noted that IP network is not perfectly designed for real-time applications such as VoIP. In the IP network, there is no guarantee that packets are successfully delivered in sequential order to the destination, therefore, QoS is not guaranteed. Instead, best-effort transmission takes place in IP networks. If the network conditions are bad, the receiving user will have difficulties understanding the speaker's speech. In the worst case scenario, the receiving user will not be able to understand or hear the speaker at all. In these cases, the conversation through VoIP call is not possible.

There are several QoS specifications in the sense of various parameters limitations to be followed. These limitations have to be taken into account in the case of using VoIP. QoS parameters with major impact on the VoIP application are: ETE packet delay, packet delay variation or jitter, packet loss rate, bandwidth, out-of-order packet delivery and hardware capacity [17]. These parameters have to be under the required threshold values to prevent call degradation that can result in the high delay, the understanding difficulties, etc.

III. VOIP CHARACTERISTICS

Similar to other applications, VoIP application requires a specific packet structure. Moreover, it requires that a specific number of voice packets are sent per second, as it is a real time application. For various codecs, this number changes according to the speech interval, which is put into one packet.

A. VoIP Packet Structure

VoIP packet comprises payload data (digitized voice data) and an overhead of different layers of Open Systems Interconnection (OSI) model. In our case, the VoIP packet structure considers that 802.11 Wireless Local Area Network (WLAN) technologies are used at the data link layer (layer 2). At the transport layer (layer 4), User Data Protocol (UDP) is used. At the session layer (layer 5), Real-time Transport Protocol (RTP) is widely used for streaming voice data without any additional acknowledgements from the receiver side. Session Initiation Protocol (SIP) can be used for session initiation and session termination at the application layer (layer 7), and different protocols (such as Transport Layer Security, TLS) are used at the presentation layer (layer 6).

When using RTP protocol, two independent streams of data are required to establish one call connection, since RTP is a one-way protocol.

B. VoIP Traffic Parameters Calculation for Various Codecs

Each codec has different sample interval and different bit rate at which it operates. Based on this, the size of codec sample, i.e., codec sample size (CSS), can be calculated as:

$$CSS\ [bytes] = (\ CBR\ [bps] \cdot CSI\ [s]\ )\ /\ 8 \qquad (1)$$

CBR stands for codec bit rate and is the number of bits per second that has to be sent to deliver a voice call. CSI stands for codec sample interval and is the time interval of a voice signal a codec takes and handles at once. The CSS is the size of voice payload data in one VoIP packet.

For the calculation of the bandwidth, required for establishing a VoIP call, we also need to know how many packets are sent per second (i.e., packet rate). Packet per second (PPS) represents the number of packets that has to be transmitted every second in order to deliver the codec bit rate, and is calculated with the help of CBR and CSS as follows:

$$PPS\ [pps] = CBR\ [bps]\ /\ (\ CSS\ [bytes] \cdot 8\ ) \qquad (2)$$

Then, the Bandwidth requirement per call can be calculated as:

$$Bandwidth\ [bps] = total\ packet\ size\ [bytes] \cdot PPS\ [pps] \cdot 8 \qquad (3)$$

Here, the total packet size is the size of the entire VoIP packet, which is in our case the size of a voice payload, IP header, UDP header, RTP header, 802.11 MAC header, and 802.11 Physical (PHY) header combined.

Traffic load per second is calculated similar as the *bandwidth*:

$$Load\ [bytes] = total\ packet\ size\ [bytes] \cdot PPS\ [pps] \qquad (4)$$

For example, *CSI* for codec G.711/10 is 10 ms and *CBR* is 64 Kbps. Thus, the calculated *CSS* is 80 bytes and the calculated *PPS* is 100 pps. This means that every second 100 VoIP packets of the voice payload size of 80 bytes have to be sent to achieve the requirements of this codec. Furthermore, the calculated *bandwidth* required to perform a call with G.711/10 in 802.11b WMN is then 142,4 Kbps and the traffic load (*load*) is 17800 bytes per second.

IV. VoIP QoS Requirements

In the following, the influence of ETE delay, jitter and packet loss rate on the VoIP QoS will be presented. Also, the threshold values of these parameters, beyond which network should not go if supporting a certain QoS of VoIP application, will be given. The ETE delay and jitter parameters are then investigated with network coding in Section VII.

A. ETE delay

Group TIPHON [18] classifies VoIP application into different network QoS performance classes regarding the ETE delay [19] of the voice packets. In the case of speech transmission it is a "mouth-to-ear" delay; the delay between the time a packet is sent from the "speaker" and the time a packet is received at the "listener". The classes are provided in Table I.

TABLE I. VoIP QoS classes regarding the ETE packet delay.

	3 (Wide-band)	2 (NARROWBAND) 2H (High)	2M (Medium)	2A (Acce-ptable)	1 (Best effort)
Relative Speech Quality (one way, non-interactive speech quality)	Better than G.711	Equivalent or better than ITU-T Recommendation G.726 at 32 kbps	Equivalent or better than GSM-FR	Not defined	Not defined
Delay	< 100 ms	< 100 ms	< 150 ms	< 400 ms	< 400 ms
NOTE: The delay for best effort class is a target value.					

In the ITU-T G.114 [20] recommendation, it is stated that ETE delay should never exceed 400 ms for general network planning. As long as the ETE delay is kept below 150 ms, only a few VoIP sessions may get affected. From the user point of view, delays up to 290 ms are satisfactory. Delays between 290 ms and 400 ms cause the dissatisfaction to some users. Delays above 400 ms can only be used if we suppose that the user is familiar with higher delay, e.g., as in a satellite communication.

VoIP application delay has different causes [21]: coding/ encoding, packetization, jitter buffer and network delay (or network latency). The ETE delay caused by the first three causes is described as a codec delay. It can be calculated as:

$$CodecDelay = CSI + CPP + CPP + JBS \tag{5}$$

CPP is so-called pooling period of Central Processing Unit (CPU) or CPU pooling period and is half of the *CSI*. *JBS* represents jitter buffer size. For example, *CSI* of codec G.711/10 is 10 ms. Thus, *CPP* is 5 ms and the recommended JBS for G.711/10 is 20 ms. *CodecDelay* results then in 40 ms.

The ETE VoIP packet delay (*ETEDelay*) is:

$$ETEDelay = NetworkDelay + CodecDelay \tag{6}$$

Packet network delay (*NetworkDelay*) occurs as the packet is sent through the network. However, it cannot be defined, as delays, presented above. In WMN, packet is sent through several wireless routers to be delivered to its destination. Different packets can be routed through the network with different speeds resulting in the variable delays of packets on the receiver.

B. Jitter

Jitter describes a non-constant packet delay at the receiver as the packet latency can vary when packets are sent across the IP network [15]. Jitter can occur when packets of the same stream are sent via different routes through the network. Beside this, it can occur as the traffic intensity of a network can vary over time thus delaying packets differently. The expected jitter influences the size of a jitter buffer. The higher the jitter, the greater is the size of a jitter buffer needed to compensate the difference in the delay of packets of the same stream at the receiver. This buffer essentially enables continuous speech. The jitter buffer size is the same or a multiple (i.e., 1, 2, 3) value of CSI interval. In Table II, delays are represented for various codecs when taking also into account a jitter buffer delay besides the delays in (1), assuming the jitter buffer size of two CSI (i.e., $2 \cdot$ CSI). Network delay is not considered here.

TABLE II. One-way codec delays for various codecs.

	One-way codec delay [ms]
G.711/10	40
G.711/20	80
G.711/30	120
G.723/30	120
G.723/60	240
G.729/20	80
G.729/40	160

Jitter is measured as difference in ETE delays between the two consecutive packets of the same VoIP application stream. The jitter values greater than 100 ms are causing delays which are above ITU-T organization's recommendations. Jitter values from 100 ms to 200 ms can be still handled by some jitter buffers, but already introducing some conversational problems. If the packet arrives at the VoIP device too late (i.e., out of the jitter buffer value), it is lost. In the context of a network, packet jitter is measured as the average of all jitter packets values.

C. Packet Loss Rate

Packets get lost in the network for two reasons, due to queue packet drop caused by the traffic overload or due to packet corruption when travelling through the network. In the first case, network latency (which influences the packet delay) is too high and/or packet jitter is too long. In the second case, signal degradation due to transmission over the lossy wireless medium or corruptions in the network lead to unsuccessful packet delivery to the packet destination. When a voice packet is lost, it can be replaced by (i) the silence, (ii) the former packet, or (iii) the interpolation of the former and subsequent packet. One packet contains from 10 to 60 ms of the voice data, which means that the loss of a packet is the loss of a phoneme in the speech of a call. It is considered that the targeted/tolerable packet loss is lower than 10^{-5}. The losses higher than this value influence speech intelligibility.

V. Network Coding for Wireless Mesh Networks

Network coding is the mechanism to improve the network performance. It experienced an increasing attention in the past few years in both, wired and wireless networks, mainly due to promising results from the initial research and testbed deployments [7, 8]. Network coding can be performed on different layers, i.e., on PHY (wireless network coding) or higher network and application layer. We are interested in opportunistic network layer network coding for unicast traffic.

Network coding enables encoding multiple network layer packets either from the same or from different traffic flows into one encoded packet for saving bandwidth and thus increasing the network capacity while maintaining the desired Quality of Service parameters. It can be also used to decrease the network delay, as will be demonstrated in Section VII. The basic opportunistic network coding principle can be explained on a simple example in Fig. 1, where two nodes A and B exchange packets m1 and m2 through a relay node R. Without network coding, nodes first send packets to node R and then node R forwards packets to the destinations. For this, four transmissions are required. With network coding, node R encodes both packets m1 and m2 to one encoded packet and sends it in one transmission to both nodes resulting in three transmissions required to exchange both packets between nodes A and B. As node A has its own packet m1, it can decode m2 from the encoded packet. Similarly, node B has its own packet m2, therefore, it can decode m1 from the encoded packet. With this, one transmission has been saved.

In wireless networks, network coding exploits also the broadcast nature of the wireless medium, where nodes can overhear packets which are not destined to them, resulting in new coding opportunities [6]. These packets are later on needed for the decoding process.

The network coding principle with overhearing enabled is presented in Fig. 2, where it is assumed that we have wireless nodes (e.g., wireless routers). Nodes S1 and S2 have to deliver packets m1 and m2 to nodes D1 and D2. Without network coding, packets are first sent to a relay node R and then forwarded to its corresponding destinations. Therefore, four transmissions are required to deliver packets. While with

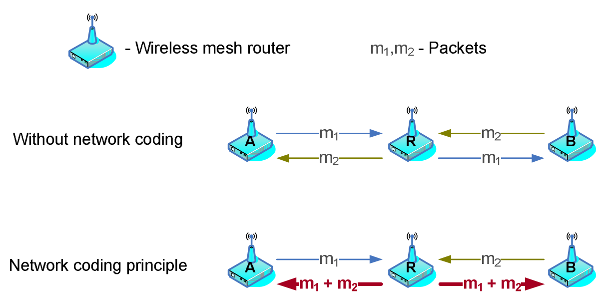

Fig. 1 Presentation of network coding principle.

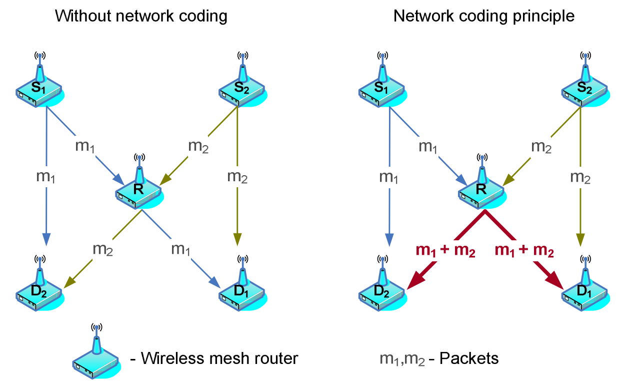

Fig. 2 Presentation of wireless network coding principle.

network coding, three transmissions are only required to deliver packets, as both packets are encoded into one packet (linear operation over the two packets) on node R, which is then broadcasted to both destinations. Therefore, only one transmission is required by node R. The coding is possible as D1 knows m2, as it hears node S2 and can decode m1 from encoded packet sent from node R. Similar, D2 knows m1, as it hears node S1 and can decode m2 from encoded packet sent from node R. Similar as in Fig. 1, one transmission has been saved.

One of the well-known network coding procedures for increasing the throughput of a WMN is COPE procedure [7], which is described in the following.

A. COPE Network Coding Procedure

COPE [7] is an intra-session network coding algorithm, which exploits the broadcast nature of the wireless medium. It codes packets for one hop, where packet decoding is done. The coding process depends on the nodes knowledge on what information (which packets) neighboring nodes have. In case the node knows which information neighbors have (through listening to neighbor's broadcasts (packets and ACKs) or receiving their updates) the coding process is straightforward and the decoding process will have a high success rate. Information arriving through particular massages and through listening to all the broadcast, is not sufficient and provides only few coding opportunities. In the case that the information on the packet presence at specific neighbor's node is not available the coding needs to guess on the situation. The node estimates the probability that the node A has packet P, by looking at the delivery probability between packet's previous hop and node A. With all the needed information the node can code together as many packets as possible, as long as none of the packets have been created on this node, all the packets have different next hops and we know that there is a strong possibility that each next hop (all the neighboring nodes that we are encoding packets in for) will be able to decode the packet. The next hop can decode the packet if it has already received all except one of the packets coded together.

VI. Network Coding Simulation Model

Network coding simulation model, presented in Fig. 3 (and also presented in [22, 23]), has been built using the OPNET

Modeler [24] simulation tool. It is used for detailed analysis of causes and consequences in network coding, as well as evaluation and comparison of the performance of various network coding and routing procedures, and for investigation of routing metrics for network coding. Moreover, we are using the simulation model to develop end evaluate new network coding and network coding aware routing procedures.

The simulation model is comprised of several modules. The supporting network topology generator module is developed in MATLAB and is able to generate random wireless topologies built around the arbitrary number of randomly positioned nodes. The nodes can communicate with an arbitrary number of neighbor nodes through wireless connections. The connections between node pairs are selected according to nodes positions and transmission "range" or other predefined node parameters. Note that the expression of node pair is used for pairs of nodes that are assumed to communicate directly, being also selected as neighbors. The network simulation description procedure, also developed in MATLAB, prepares the information on the desired topology, nodes, wireless connections and parameters for communication procedures (e.g., channel bandwidth, number of packet retransmissions, loads, etc.).

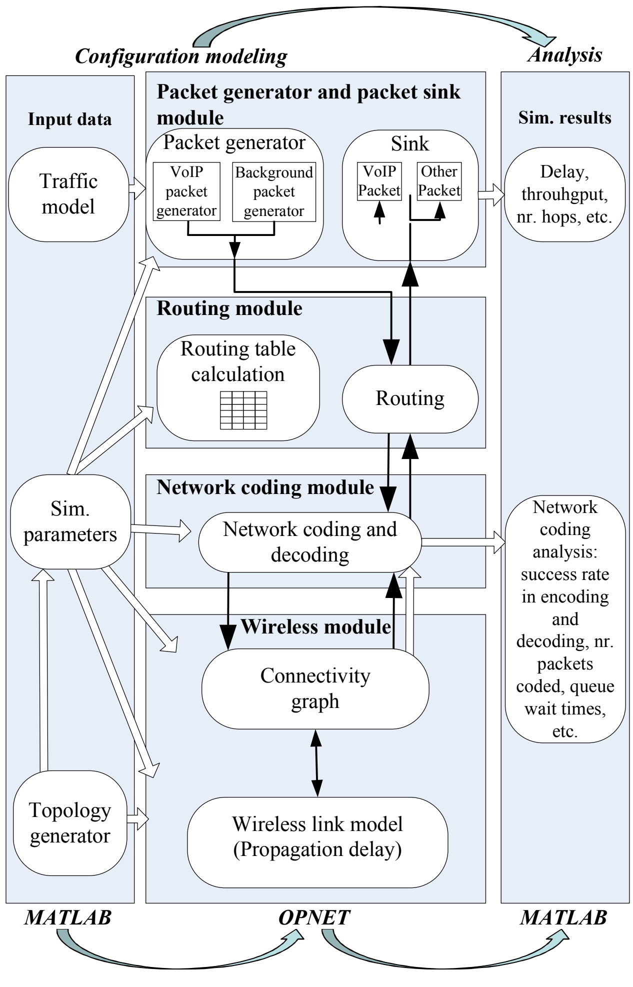

Fig. 3 Simulation model architecture.

Selected topologies and parameters are imported into the OPNET Modeler [24] simulation model, where the main simulation takes place. After the simulation, simulation results are exhaustively analyzed in MATLAB.

Simulation model is implemented in four functional layers thus facilitating separate parametric investigation of different effects on the overall network performance. Packet generator and packet sink module is responsible for creating the network traffic load and for receiving packets at their destination. Packets can be generated arbitrarily on different nodes with different intensities using various distributions (such as exponential, uniform, etc.) of inter-arrival times and packet lengths. For each packet, the destination node is selected arbitrarily among all network nodes.

Routing module takes care of routing packets through the network with the help of routing tables. Routing tables are calculated by the shortest path routing algorithm such as, e.g., Dijkstra's algorithm and Bellman-Ford algorithm. With these algorithms, various metrics are used such as hop count, distance and expected transmission count (ETX) [25] metric.

The core of the simulation model represents the network coding module, where network coding procedures are implemented. Currently, it supports COPE [7] and BON [23] network coding procedures.

The wireless module takes care of successful packet distribution through the wireless channel between neighboring nodes taking into account wireless link conditions. Implemented network coding procedures require pseudo broadcast mechanism, which was first introduced by Katti et al. [7] and is supported in the wireless module in addition to broadcast. The connectivity graph determines the overall network architecture. Besides this, it considers propagation delay representing the time needed by a signal to propagate through a wireless media from a transmitter to a receiver. In our network simulation model, signal transmission is simulated at the packet level.

A. VoIP Implementation

The VoIP application has been implemented in Packet generator and packet sink module (see Fig. 3) in the OPNET Modeler [24] simulation environment, as an additional application possible to be used optionally. In this module, one additional packet stream has been introduced on a node, representing VoIP traffic. At the receiving side, VoIP traffic handling has been implemented to treat VoIP packets differently to other packets.

VoIP application traffic is introduced in the simulation as VoIP calls established by two wireless nodes for a predefined period of time. In other words, VoIP call is simulated with two packet streams, being sent simultaneously between the two wireless nodes in the opposite directions (not necessarily over the same route), which are representing the two speakers in a VoIP call. The signalization traffic is neglected in the simulation and we do not investigate this traffic in the paper. This is reasonable, as it presents about 5 % of all VoIP traffic. We perform the comparison between using network coding and without network coding, in both cases, without the signalization.

At the receiving part of the call, packet delay, introduced by the WMN network, is computed. Then, jitter is calculated as the difference between the network delays of two consecutive packets of the same stream.

Other delays (i.e., coding/encoding, packetization, jitter buffer) are added to the network delay to evaluate the performance of the established VoIP call.

We have implemented seven various VoIP codecs: G.711/10, G.711/20, G.723/30, G.723/60, G.729/20, G.729/40.

VII. Performance Evaluation of VoIP Using Network Coding

We have performed the evaluation of VoIP with network coding using network coding simulation model, presented in Section VI. The simulation model has been built using OPNET Modeler [24] simulation tool. In this section, we present and analyze the results obtained by simulation runs in the simulation model. We compare simulation results when using VoIP without network coding to the simulation results when using VoIP with COPE network coding procedure. First, we observe VoIP average network delay and jitter. Then, we investigate the use of network coding in terms of VoIP QoS performance requirements.

The performance of VoIP with network coding was tested in different network topologies and simulation results were collected for each of them. After analyzing the results, one network topology was chosen for the representation as an example, although similar results were obtained by different topologies. The results from the presented network topology were chosen to present the VoIP performance using various codecs in a typical WMN with or without network coding.

A. Simulation Parameters and assumptions

In this section, we describe the main parameters and assumptions that are used in different simulation runs. First, we assume that all wireless network nodes are of the same type and have identical configuration, representing homogeneous network. Networks with different number of nodes and topologies are investigated, where each node is given a random location within a given area. The wireless nodes have been randomly positioned within the square area of 2000 meters by 2000 meters, which is the size of the simulated wireless environment.

A typical network topology for WMNs with 10 wireless nodes and 3 neighbors per each node, depicted in Fig. 4, has been selected and is further analyzed in this paper.

Each node has 1 Mbit/s of channel bandwidth. Wireless connections established between neighbors are graphically presented in Fig. 4 with dashed lines between nodes. For the simulation purposes, all the links are symmetrical and are lossless, meaning that no packets get lost during transmissions. Lossless links mean that network conditions have to be perfect or that there is a mechanism implemented that transparently guarantees lossless transmission to higher layers. Moreover, packets on wireless links are delayed due to propagation through wireless medium.

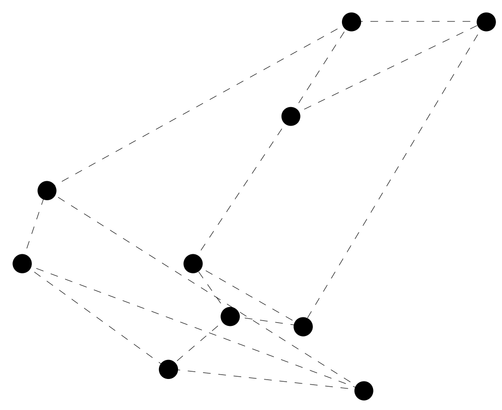

Fig. 4 Representative network topology with 10 nodes and 3 neighbors per node.

In the simulation, VoIP application is simulated establishing VoIP calls between node pairs. VoIP call is simulated with two packet streams being sent between the two wireless nodes, which are representing the two speakers of a VoIP call. VoIP calls are established between each node pair in the network. For a network topology with 10 wireless nodes in Fig. 4, it results in 45 individual calls. Only one VoIP call is established at the same time in the network. Each simulated VoIP call lasts for 30 seconds. In Table III, the parameters for various used codecs are presented. The total size of VoIP packet and the number of VoIP packets sent each second (i.e., packet per second, PPS), are calculated for various codecs used in simulation, considering the fact that VoIP application is implemented in 802.11b WMN. The VoIP packet total size represents the size of voice payload data (i.e., the codec sample size, CSS) and 802.11b overhead in one VoIP packet. The CSS depends on the codec bit rate (CBR), which determines the number of bits per second that have to be sent to deliver a voice call, and, the codec sample interval (CSI), which is the interval of a speech a codec takes and handles at once. PPS represents the number of packets that have to be transmitted every second in order to deliver the codec bit rate. It depends on the CSI. Bandwidth required for a call is also provided. In addition, traffic load per second, produced by VoIP call every second, is calculated by multiplying PPS value and VoIP packet total size for each codec in Table III.

Background traffic is simulated all the time during VoIP calls and allows the evaluation of its impact on the performance of VoIP calls. It is generated as packet streams between all nodes with the same intensity using exponential distribution of inter-arrival times and constant packet lengths (i.e., 10 kbit). The background traffic load is increased through simulation runs until the VoIP packet delay in the network is increased due to this traffic and VoIP traffic cannot be handled anymore. All network nodes are source nodes for generating background traffic with the same probabilities and they select destination nodes using uniform probability distribution among all network nodes. Results are presented for six different intensities of traffic background loads (i.e., for six different total amounts of background traffic sent into the network) denoted by L1, L2, L3, L4, L5, and L6. L1 represents the lowest network traffic load used in the presented results, while L6 represents the highest load (i.e., when the network is congested). The network diameter is 3. The network average hop count is 2.

TABLE III. CODECS PARAMETERS.

Codec	CBR [kbit/s]	CSI [ms]	CSS [bytes]	PPS [pps]	Bandwidth [Kbps]	VoIP packet total size [bytes]	Traffic load per second [bytes]
G.711/10	64	10	80	100	142,4	178	17800
G.711/20	64	20	160	50	206,4	258	12900
G.711/30	64	30	240	33	270,4	338	11154
G.723/30	6.4	30	24	33	97,6	122	4026
G.723/60	6.4	60	48	16	116,8	146	2336
G.729/20	8	20	20	50	94,4	118	5900
G.729/40	8	40	40	25	110,4	138	3450

COPE network coding procedure [7] has been used for encoding packets to increase the network throughput. The simulation cases without network coding are compared with the cases when COPE is used in the network to evaluate the impact of network coding on the performance of VoIP application in WMN.

Important modification has been made to the COPE procedure to increase packet delivery reliability at the network coding module. Instead of using cumulative ACKs as described in the original paper, each coded packet is immediately confirmed with the individual ACK packet. This allows us to shorten the round time and schedule possible retransmissions sooner. This is an important modification as it lowers the jitter and decreases the possibility of receiving packets with the delay higher than allowed by QoS requirements. The individual ACKs, however, increase the overhead in the network and thus lower the network goodput.

Routing of packets through the network was done using static tables, which were calculated ahead of simulation runs. Routing tables are calculated using Dijkstra's algorithm taking into account hop count distances between nodes.

The timeline of simulation was as follows. Every simulation run lasted for 1400 seconds. The background traffic was generated during the entire simulation run. The time of 5 seconds (warm up time) is required at the beginning of the simulation to have steady state conditions. Only one VoIP call between two wireless nodes was established at the same time in the network using one of the codecs in Table III. VoIP calls were generated consecutively with the 1s delay between them. In one simulation run, 45 individual calls were simulated and each VoIP call lasted 30 seconds resulting in 1350 seconds of VoIP calls simulation. At the end, 5 seconds are used for simulation control purposes as, e.g., the intensity of network congestion in the case of high background traffic, which is detected by receiving VoIP packets at the receivers also after the 1355th second of simulation, up to the 1400th second. In each simulation run, the background traffic was increased.

B. *Average Network Delay and Jitter*

We have averaged the network delay of all calls established in one simulation scenario. In every scenario, a particular codec has been used for transmitting VoIP calls. To simulate different traffic densities in the network, we have created different amounts of background traffic. Then, we have evaluated how background traffic affects the VoIP application performance with various codecs. The results are presented in Fig. 5.

From Fig. 5, we can see that VoIP delays are increasing with the increased background traffic in the network for all codecs, as expected. Codec G.711/10 has the highest average network delay, while G.723/30, G.723/60, G.729/20 and G.729/40 have lower delays. This is because of the specific traffic load per second a particular codec has, which is presented in Table III. For example, G.711/10 has the highest traffic load per second; therefore, it has also the highest network delay. Note that we do not present the scenarios, where the network gets congested (i.e., delays go towards

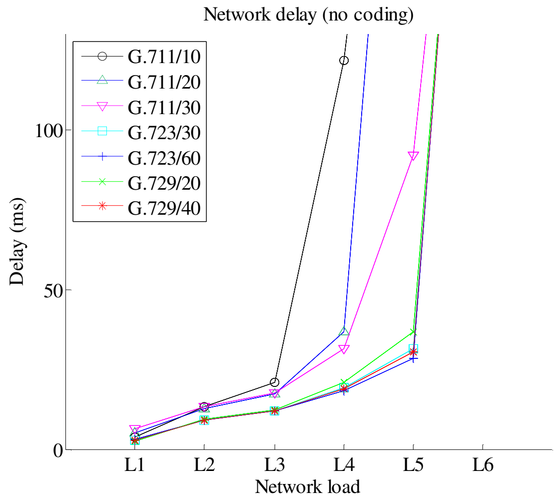

Fig. 5 Network delay when network coding is not used.

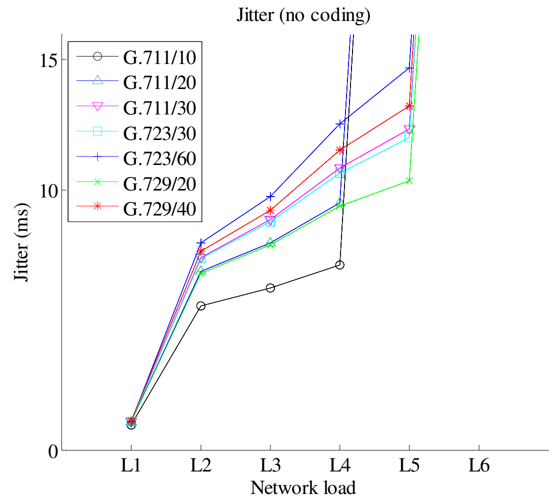

Fig. 6 Jitter when network coding is not used.

infinity). Therefore, there is no mark for these scenarios on the graph in Fig. 5 (curves going out from the figure). Similar is also done in the figures, which are presented in the following. Background traffic loads (e.g., L4, L5, L6), when delays are very high, cause (in some cases) network congestion, when using a particular VoIP codec. It means that we are presenting the results, when network is highly loaded or is already congested, with the exception of L1.

We have done the same for jitter measurements. In Fig. 6, jitter is presented for various used codecs in dependency of background traffic load. We can see that average jitter is increasing with background traffic, but not as rapidly as network delay in Fig. 5.

After analyzing the VoIP performance without network coding, we have also performed simulations, when COPE network coding procedure has been used in the WMN network to increase the throughput of the network. The scope of that was to investigate the impact of network coding on the VoIP

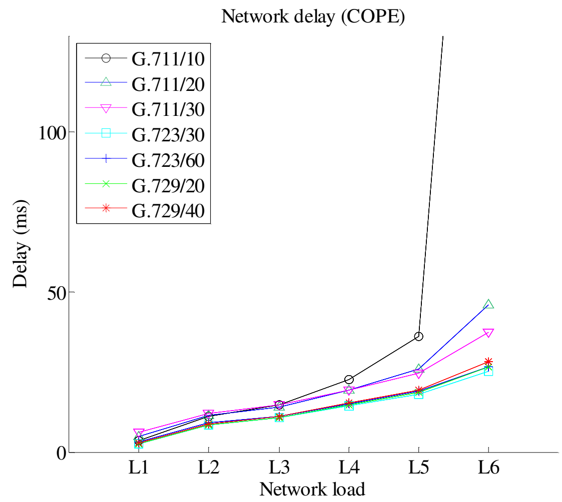

Fig. 7 Network delay when COPE is used.

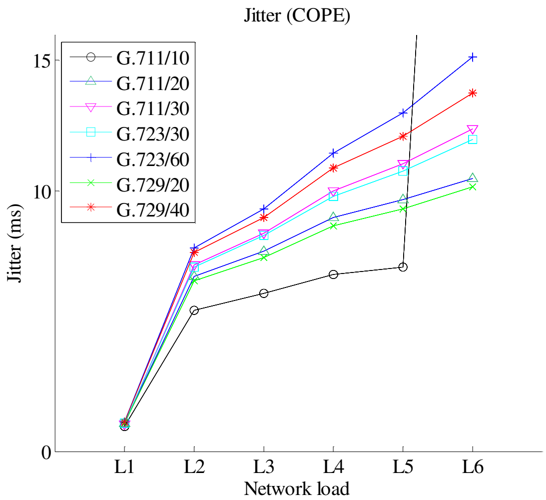

Fig. 8 Jitter when COPE is used.

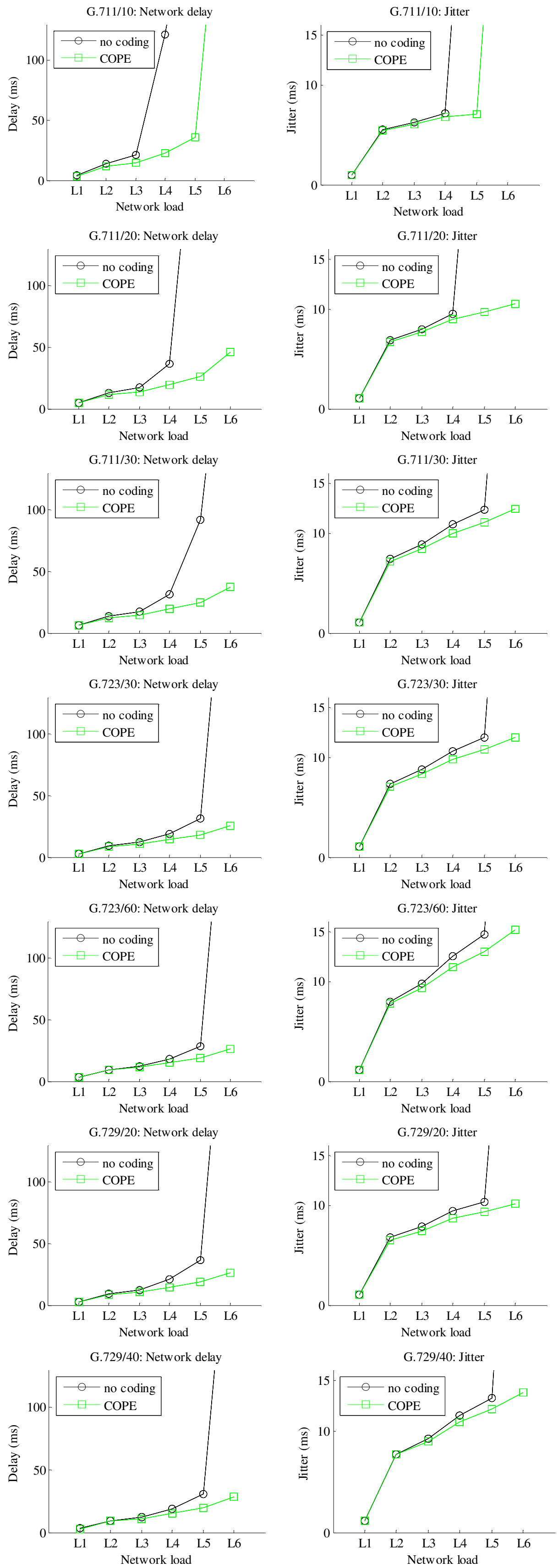

Fig. 9 Network delay and jitter for various codecs with and without COPE.

performance. In Fig. 7 and Fig. 8, network delays and jitters are presented respectively for the cases, when network coding (i.e., COPE procedure) is used on wireless nodes in the network. The results are presented for the same scenarios as in Fig. 5 and Fig. 6.

When using COPE, average network delays are notably lower also when background traffic load is high. This difference between the COPE and no-COPE is increasing with the increased background traffic, as expected. More packets are in the network, more coding opportunities arise and more packets can be encoded, thus saving more bandwidth at the transmission. Moreover, it can be seen from Fig. 7 that VoIP application using COPE, in most cases (except for codec G.711/10 with L6), still performs well having L4, L5 and L6 background traffic in the network, while without network coding the VoIP application is degraded due to high delays of VoIP packets (represented with no marks on the graph). Here, we can conclude that network coding improves the performance of VoIP application in WMN, especially when the network is highly loaded or overloaded to a certain point. For jitter values, the difference between COPE and no-COPE case is very small, so the improvement is, in most cases, negligible. It is worth noting that using COPE does not increase the value of jitter.

Next, we have investigated the difference in the impact of network coding with different VoIP codecs. We have compared the scenario of using COPE and the scenario when network coding is not used for various codecs in Fig. 9. The comparison of network delays and jitters, in dependency of different background traffic loads, using COPE procedure and without network coding, is presented for various VoIP codecs, separately.

It can be seen that codecs, which require higher traffic load per second, benefit from network coding more than codecs with lower traffic load per second. Once more, this is due to the fact that more VoIP packets are encoded with other packets (because of the increased overall traffic load in the network), thus increasing the capacity gain achieved by the network coding.

C. *VoIP QoS Requirements Investigation*

In this section, we have evaluated VoIP from the QoS point of view. The same scenarios as presented in Section VI.B have been used.

First, we have investigated the VoIP network delay of the packet which had the highest delay from all the calls in an individual scenario (or simulation run). This has been done for all the scenarios, i.e., for various codecs with COPE and without network coding, and for different background traffic loads. The results are presented in Table IV, denoted by "*COPE*" for network coding scenarios and by "*No NC*" for the cases when network coding is not used. The same has been done for the jitter values and is presented in Table V. The values in both tables are presented in milliseconds (ms). Similar as in Section VI.B, we do not present the scenarios, where the network gets congested and delays go towards infinity. For those scenarios, there is no value in the table. Instead, the cross signs in both tables mark that delays for the scenarios were very high; therefore, they were not interesting to investigate. Similar is also done in the tables, which are presented in the following.

TABLE IV. THE HIGHEST VOIP PACKET NETWORK DELAY.

		The highest VoIP packet network delay [ms]						
		G.711/ 10	**G.711/ 20**	**G.711/ 30**	**G.723/ 30**	**G.723/ 40**	**G.729/ 20**	**G.729/ 40**
L1	***No NC***	54	45	48	45	38	43	39
	COPE	46	41	49	46	39	40	40
L2	***No NC***	156	138	124	106	106	115	104
	COPE	132	107	112	98	112	92	89
L3	***No NC***	253	200	198	182	189	187	184
	COPE	137	118	124	123	118	127	113
L4	***No NC***	1172	479	344	215	182	230	200
	COPE	234	147	144	131	120	132	122
L5	***No NC***	x	x	1114	393	349	419	371
	COPE	404	253	240	197	189	209	202
L6	***No NC***	x	x	x	x	x	x	x
	COPE	x	519	265	199	216	206	226

TABLE V. THE HIGHEST VOIP PACKET JITTER.

		The highest VoIP packet jitter [ms]						
		G.711/ 10	**G.711/ 20**	**G.711/ 30**	**G.723/ 30**	**G.723/ 40**	**G.729/ 20**	**G.729/ 40**
L1	***No NC***	50	39	40	42	35	40	37
	COPE	39	36	40	43	35	38	37
L2	***No NC***	76	84	91	100	96	79	88
	COPE	66	76	77	74	102	68	69
L3	***No NC***	108	129	165	153	163	137	170
	COPE	84	77	93	93	92	85	99
L4	***No NC***	1148	125	113	107	118	118	106
	COPE	94	91	99	91	104	102	108
L5	***No NC***	x	x	133	142	169	127	137
	COPE	98	115	125	118	130	95	123
L6	***No NC***	x	x	x	x	x	x	x
	COPE	x	118	148	114	129	118	139

The highest packet network delay and jitter gives the information of how bad the delays can be within a call. In this respect, we are here particularly interested in the performance of COPE in comparison with no network coding scenario.

The results show that the worst packet network delays with COPE are lower than the worst packet network delays without network coding for up to 80%, for the presented scenarios. For jitter, the benefits are even bigger resulting in up to 90% decrease of the worst packet jitter within individual scenarios. For the scenarios, marked with the cross sign, which we did not investigated, the benefits would be even higher.

For further research on the QoS, the ITU-T recommendation threshold values for VoIP packet delays and jitters, presented in Section IV, has been used. Used values for packet delays are 100 ms, 150 ms, 290 ms, and 400 ms. For jitters, we use value 100 ms and, instead of the recommended value 200 ms in Section IV.C, we use the value 150 ms, since in our case there is no scenario with jitters above this value. We say that packet delays of all individual calls in a scenario are below a specific threshold value if delays for less than ten packets from this scenario are above the corresponding threshold value. We then mark this with, e.g., "<100 ms", for delays lower than 100 ms. For the delays of scenarios, where values go above the maximum thresholds, we use a mark ">400 ms", for, e.g., delays above 400 ms. We do the same also for jitter values. The VoIP packet network delay and jitter threshold values of individual scenario are presented in Table VI and Table VII, for all the scenarios. The values in the tables are presented in ms.

TABLE VI. QoS CLASSIFICATION FOR VoIP PACKET NETWORK DELAY.

		QoS classification for VoIP packet network delay						
		G.711/ 10	G.711/ 20	G.711/ 30	G.723/ 30	G.723/ 40	G.729/ 20	G.729/ 40
L1	No NC	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L2	No NC	<150 ms	<150 ms	<150 ms	<100 ms	<100 ms	<100 ms	<100 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L3	No NC	<290 ms	<290 ms	<150 ms	<150 ms	<150 ms	<150 ms	<150 ms
	COPE	<150 ms	<150 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L4	No NC	>400 ms	>400 ms	<400 ms	<290 ms	<290 ms	<290 ms	<290 ms
	COPE	<290 ms	<150 ms	<150 ms	<150 ms	<100 ms	<150 ms	<150 ms
L5	No NC	x	x	>400 ms	<400 ms	<290 ms	<400 ms	<400 ms
	COPE	<400 ms	<290 ms	<290 ms	<150 ms	<150 ms	<290 ms	<290 ms
L6	No NC	x	x	x	x	x	x	x
	COPE	x	>400 ms	<290 ms	<290 ms	<290 ms	<290 ms	<290 ms

TABLE VII. QoS CLASSIFICATION FOR VoIP PACKET JITTER.

		QoS classification for VoIP packet jitter						
		G.711/ 10	G.711/ 20	G.711/ 30	G.723/ 30	G.723/ 40	G.729/ 20	G.729/ 40
L1	No NC	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L2	No NC	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L3	No NC	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L4	No NC	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L5	No NC	x	x	<150 ms	<150 ms	<150 ms	<100 ms	<150 ms
	COPE	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms
L6	No NC	x	x	x	x	x	x	x
	COPE	x	<100 ms	<100 ms	<100 ms	<100 ms	<100 ms	<150 ms

In the following, we have classified all the presented scenarios based on the packet network delay and packet jitter threshold values in Table VI and Table VII. We have made several classes based on the ITU-T recommendation threshold values. The modified classes are presented in Table VIII. It is considered that a scenario is of a specific class, if packet delays and jitters in this scenario meet the requirements of that class. Based on the packet network delay and packet jitter threshold values in Table VI and Table VII, where the delay and jitter performance of all the scenarios was tested, we can arrange individual scenario in the corresponding class. The scenarios are classified in Table IX. Lower the class, better is the VoIP performance. For example, packet network delay threshold value of the scenario G.711/10 without network coding under the background traffic load L1 is "< 100 ms" and packet jitter threshold value is also "<100 ms". Therefore, the scenario fits in "Class 1", which is the best class according to our modified classification. We can see that scenarios with COPE are of the same or better class as those without network coding. Higher the background traffic load, higher is the difference in classes. However, the benefit of network coding changes with the used codec, similar as in Section VII.B. The classification with packet network delay gives us the insight of the network influence with various codecs on the VoIP performance. But such classification is still not adequate, as also other influences, such as jitter buffer size, has to be considered in the evaluation. This will be performed in the following.

TABLE VIII. MODIFIED CLASSES OF VoIP (QoS) PERFORMANCE.

(QoS) Class	Packet delay [ms]	Packet jitter [ms]
1	< 100	< 100
2	< 150	< 100
3	< 290	< 100
4	< 290	< 150
5	< 400	< 100
6	< 400	< 150
7	> 400	< 100
8	> 400	< 150

TABLE IX. MODIFIED VoIP QoS CLASSES BASED ON PACKET NETWORK DELAY AND JITTER.

		Modified VoIP QoS classes based on packet network delay and jitter						
		G.711/ 10	G.711/ 20	G.711/ 30	G.723/ 30	G.723/ 40	G.729/ 20	G.729/ 40
L1	No NC	1	1	1	1	1	1	1
	COPE	1	1	1	1	1	1	1
L2	No NC	2	2	2	1	1	1	1
	COPE	1	1	1	1	1	1	1
L3	No NC	3	3	2	2	2	2	2
	COPE	2	2	1	1	1	1	1
L4	No NC	7	7	5	3	3	3	3
	COPE	3	2	2	2	1	2	2
L5	No NC	X	X	8	6	4	5	6
	COPE	5	3	3	2	2	3	3
L6	No NC	X	X	X	X	X	X	X
	COPE	X	7	3	3	3	3	4

Similar as above, we have classified all the presented scenarios based on the packet ETE delay and packet jitter threshold values in the modified VoIP QoS performance classes, presented in Table VIII. First, we had to determine the required jitter buffer sizes considering that the size of a jitter buffer for various codecs has to be (i) according to the specifications in Section IV.B, and (ii) large enough for the presented load with the highest jitters. The required jitter buffer sizes for various codecs are presented in Table X.

TABLE X. (REQUIRED/USED) JITTER BUFFER SIZES (JBS) FOR VARIOUS CODECS.

	Jitter Buffer Size (JBS) [ms]
G.711/10	90
G.711/20	100
G.711/30	120
G.723/30	120
G.723/60	120
G.729/20	100
G.729/40	120

Based on the determined jitter buffer sizes, we have calculated codec delays in Table XI. The highest packet ETE delay can be then calculated for all the presented scenarios based on Table IV and Table XI properly summing the values from both tables for various used codecs, as presented in (6) in Section IV.A. Similar as the highest packet network delays in Table IV, the highest packet ETE delays are presented in Table XII for all the scenarios; the values are presented in ms.

TABLE XI. CODEC DELAY $= 2 \cdot \mathrm{CSI} + \mathrm{JBS}$.

	Codec delay [ms]
G.711/10	110
G.711/20	140
G.711/30	180
G.723/30	180
G.723/60	240
G.729/20	140
G.729/40	200

TABLE XII. THE HIGHEST VOIP PACKET ETE DELAY.

		The highest VoIP packet ETE delay [ms]						
		G.711/ 10	G.711/ 20	G.711/ 30	G.723/ 30	G.723/ 40	G.729/ 20	G.729/ 40
L1	*No NC*	164	185	228	225	278	183	239
	COPE	156	181	229	226	279	180	240
L2	*No NC*	266	278	304	286	346	255	304
	COPE	242	247	292	278	352	232	289
L3	*No NC*	363	340	378	362	429	327	384
	COPE	247	258	304	303	358	267	313
L4	*No NC*	1282	619	524	395	422	370	400
	COPE	344	287	324	311	360	272	322
L5	*No NC*	x	x	1294	573	589	559	571
	COPE	514	393	420	377	429	349	402
L6	*No NC*	x	x	x	x	x	x	x
	COPE	x	659	445	379	456	346	426

The results show that the worst packet ETE delays with COPE are better than the worst packet ETE delays without network coding for up to 70%, for the presented scenarios. However, the delays are now higher than in Table IV and consider all the influences on the VoIP packet delay performance. For example, the packet highest ETE delay of the scenario G.711/10 with COPE under the L1 background traffic load is by 218% higher than the highest packet network delay, while the highest packet ETE delay of the scenario G.711/20 with COPE under the L1 background traffic load is by 341% higher than the highest packet network delay.

The same as we have done for the packet network delays in Table VI, we have done also for packet ETE delay threshold values in Table XIII for all the presented scenarios. These delays are now equivalent to the ITU-T recommendation threshold values for ETE delays in Section IV.A. Based on the packet ETE delay threshold values of individual scenario in Table XIII and packet jitter threshold values in Table VII, we can now perform the QoS classification for all the presented scenarios according to the recommendations in Section IV based on the modified VoIP QoS performance classes in Table VIII. With this, we also perform the QoS performance evaluation of various codecs with COPE and without network coding, which is also the main purpose of this section. The VoIP QoS performance classification of the presented scenarios, based on packet ETE delay and packet jitter threshold values, is presented in Table XIV.

TABLE XIII. QOS CLASSIFICATION FOR VOIP PACKET ETE DELAY.

		QoS classification for VoIP packet ETE delay						
		G.711/ 10	G.711/ 20	G.711/ 30	G.723/ 30	G.723/ 40	G.729/ 20	G.729/ 40
L1	*No NC*	<150 ms	<290 ms	<290 ms	<290 ms	<290 ms	<290 ms	<290 ms
	COPE	<150 ms	<290 ms	<290 ms	<290 ms	<290 ms	<290 ms	<290 ms
L2	*No NC*	<290 ms	<290 ms	<290 ms	<290 ms	<400 ms	<290 ms	<290 ms
	COPE	<290 ms	<290 ms	<290 ms	<290 ms	<400 ms	<290 ms	<290 ms
L3	*No NC*	<400 ms	<400 ms	<400 ms	<400 ms	<400 ms	<290 ms	<400 ms
	COPE	<290 ms	<290 ms	<290 ms	<290 ms	<400 ms	<290 ms	<290 ms
L4	*No NC*	>400 ms	>400 ms	>400 ms	<400 ms	<400 ms	<400 ms	<400 ms
	COPE	<400 ms	<290 ms	<400 ms	<290 ms	<400 ms	<290 ms	<400 ms
L5	*No NC*	x	x	>400 ms	>400 ms	>400 ms	>400 ms	>400 ms
	COPE	>400 ms	<400 ms	<400 ms	<400 ms	<400 ms	<400 ms	<400 ms
L6	*No NC*	x	x	x	x	x	x	x
	COPE	x	>400 ms	>400 ms	<400 ms	>400 ms	<400 ms	<400 ms

TABLE XIV. MODIFIED VOIP QOS CLASSES BASED ON ETE PACKET DELAY AND JITTER.

		Modified VoIP QoS classes based on ETE packet delay and jitter						
		G.711/ 10	G.711/ 20	G.711/ 30	G.723/ 30	G.723/ 40	G.729/ 20	G.729/ 40
L1	*No NC*	2	3	3	3	3	3	3
	COPE	2	3	3	3	3	3	3
L2	*No NC*	3	3	3	3	5	3	3
	COPE	3	3	3	3	5	3	3
L3	*No NC*	5	5	5	5	5	3	5
	COPE	3	3	3	3	5	3	3
L4	*No NC*	7	7	7	5	5	5	5
	COPE	5	3	5	3	5	3	5
L5	*No NC*	X	X	8	8	8	7	8
	COPE	7	5	5	5	5	5	5
L6	*No NC*	X	X	X	X	X	X	X
	COPE	X	7	7	5	7	5	6

From the QoS requirements point of view, we can conclude that COPE significantly improves the VoIP QoS performance in the sense of the speech quality at the receiver. For example, in case of the background traffic load L5 (high network traffic), VoIP using COPE is performing still satisfactorily, while VoIP performance without network coding is already degraded or is not working at all. When the network is already congested (L6), VoIP without network coding cannot work, while with COPE we can still establish VoIP calls (except for G.711/10). Once more, we can also notice that the benefit of using COPE changes with the used codec. For example, COPE for G.723/30 in the case of background traffic load L4 improves the QoS performance by two classes, while for G.723/40 the performance stays in the same class as without COPE.

We can conclude that VoIP application benefits from the use of network coding in WMN, as the network delay is decreased and the VoIP performance from the QoS requirements point of view is improved when the network traffic is high or the network is already congested to a certain point. Moreover, network coding does not degrade jitter.

VIII. Conclusion

This paper evaluates the use of network coding for real time VoIP application, using various codecs to transmit voice signal, in WMNs. We describe the use of network coding in wireless mesh networks and present the well-known COPE procedure for network coding. In addition, we present developed network coding simulation model, in which we implemented VoIP application, and evaluate the use of VoIP application with and without using the network coding in WMNs. We classify all the presented scenarios based on the highest VoIP packet delay and the highest VoIP packet jitter from all the calls in an individual scenario. The classification is made on modified classes based on the ITU-T recommendation threshold values. The simulation results show that network coding can improve the VoIP performance in WMNs especially when the network is highly loaded or congested, as more packets are in the network, which creates more coding opportunities and more packets can be encoded saving more bandwidth at the transmission. Network coding decreases the network delay while the jitter is not degraded due to the packets coding on forwarding nodes. It also improves the VoIP QoS performance in the sense of the improvement in modified classes based on the QoS parameter values recommended by the ITU-T. In some of the presented scenarios with higher traffic load, the VoIP application without network coding cannot be used because of the too high delays, while with network coding the VoIP application still performs well. The benefit provided by network coding depends on the used VoIP codec. Codecs which require higher traffic loads per second benefit from network coding more than codecs with lower traffic loads per second, as more packets can be encoded at higher loads.

References

[1] M. E. Fernandes, C. R. C. Lima, J. Schimiguel, “Strategic management using VoIP technology: a case study in a call center company,” WSEAS Trans. Comm., vol. 10, no. 1, pp. 34-43, 2011.

[2] M. Voznak, “Speech bandwidth requirements in IPsec and TLS environment,” Proc. 13th WSEAS Int. Conf. Comp., pp. 217-220.

[3] D. Benyamina, N. Hallam, A. Hafid, “On optimizing the planning of multi-hop wireless networks using a multi objective evolutionary approach,” Int. J. Commun., vol. 4, no. 4, pp. 213-221, 2008.

[4] I. F. Akyildiz, X. Wang, “A survey on wireless mesh networks,” IEEE Comm. Mag., vol. 43, no. 9, pp. S23-S30, September 2005.

[5] R. Ahlswede, N. Cai, S.-y. Robert Li, R. W. Yeung. “Network information flow,” IEEE Trans. Inf. Theory, vol. 46, no. 4, pp. 1204-1216, July 2000.

[6] C. Fragouli, D. Katabi, S. Katti, A. Markopoulou, M. Medard, H. Rahul, “Wireless network coding Opportunities and challanges,” Proc. Milit. Comm. Conf., 2007.

[7] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, “XORs in the air: Practical wireless network coding,” IEEE/ACM Trans. Networking, vol. 16, no.3, pp. 497-510, June 2008.

[8] M. Conti, J. Crowcroft, A. Passarella, “Multi-hop ad hoc networks from theory to reality,” Nova Science Publisher, Inc. Commack, NY, USA, 2008.

[9] R. Libnik, G. Kandus, A. Svigelj, “Simulation model for performance evaluation of advancde SIP based mobility management techniques,” Int. J. Commun. , vol. 5, no. 1, pp. 26-35, 2011.

[10] R. Libnik, A. Svigelj, G. Kandus, “Performance evaluation of SIP based handover in heterogeneous access networks,” WSEAS Transactions on Communications, vol. 7, no. 5, pp. 448-458, 2008.

[11] S. Munir, I. Ahmad, “VoIP on wireless LAN: a comprehensive review,” 7th WSEAS Int. Conf. Elect., Hard., Opt. Comm., Cambridge, pp. 225-230, UK, February 2008.

[12] B. Ngamwongwattana, “Effects of packetization on VoIP performance” 5th Int. Conf. ECTI-CON, vol. 1, pp. 373-376, May 2008.

[13] B. Goode, “Voice over Internet protocol (VoIP),” Proc. IEEE, vol. 90, no. 9, pp. 1495-1517, September 2002.

[14] R. S. Naoum, M. Maswady, “Performance evaluation for VoIP over IP MPLS,” World Comp. Sc. and Inf. Tech. J. (WCSIT), vol. 2, no. 3, pp. 110-114, 2012.

[15] V. Skorpil, D. Novak, “Service for advanced communication networks,” WSEAS Trans. Comm., vol. 9, no. 10, pp. 636-645, October 2010.

[16] R. S. Ramakrishnan, V. Kumar, “Pefromance analysis of different codecs in VoIP using SIP,” Nat. Conf. Mobile Perv. Comput. (CoMPC), India, 2008.

[17] Y. Ogunjimi, “Practical analysis of voice quality problems of Voice over Internet protocol (VoIP),” Bachelor’s Thesis, USA, 2012.

[18] ETSI TIPHON TR 101 329-2 (2002), Quality of Service (QoS) Classes.

[19] ITU-T recommendation Y.1541, Internet protocol aspects – Quality of service and network performance, December 2011.

[20] ITU-T G.114 ITU-T recommendation G.114, One-way transmission time, May 2003.

[21] Cisco Inc., While Paper. Document ID: 5125, Understanding Delay in Packet Voice Networks, 2008.

[22] K. Alič, E. Pertovt, A. Švigelj, “Simulation environment for network coding,” Proc. Mosharaka Int. Conf. Comm. Net. Inf. Tech. (MICCNIT 2011), pp. 1-6, Dubai, UAE, December 2011.

[23] K. Alič, E. Pertovt, A. Švigelj, “Network coding simulation model in OPNET Modeler,” OPNETWORK 2012, pp. 1-7, Washington, USA, August 2012.

[24] OPNET web page. URL http://www.opnet.com/.

[25] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” 9th ACM International Conference on Mobile Computing and Networking (MobiCom ‘03), New York, NY, USA, pp. 134-146, 2003.