Measuring the perceived quality of VoIP under various wireless network conditions

Ilias Tsompanidis  Georgios Fortetsanakis  Maria Papadopouli
Department of Computer Science, University of Crete &
Institute of Computer Science, Foundation for Research and Technology - Hellas

Abstract—This paper focuses on the performance of VoIP calls under various situations, namely, during a handover, under different background traffic conditions (normal and heavy traffic load/saturation conditions) at a wireless access point, and in the presence of interference. Using empirical-based measurements, we demonstrate that various network conditions exhibit distinct statistical behavior in terms of SNR, packet losses and end-to-end delays, and thus, impact the VoIP user quality in a different manner. A novel part of this work is the comparative analysis of these conditions with respect to network and user experience benchmarks. It also proposes a Markov-chain based model for the auto-rate fallback (ARF) algorithm of the IEEE802.11 and derives analytically the packet losses and mean delays at the application layer under the presence of constant interference and a single traffic CBR source at the WLAN. The paper shows that only very low SNR values affect the user experience of a VoIP call, since the ARF algorithm conceals the packet losses at the MAC layer and only the losses at the most robust (and lowest bit-rate) modulation scheme at the MAC layer become noticeable at the application layer. Furthermore, it demonstrates situations in which common “rule-of-thumb” metrics cannot reflect the perceived user experience and finer cross-layer data are required for identifying the network conditions and performing an effective adaptation.

I. INTRODUCTION

Wireless networks often experience “periods of severe impairment” (PSIs), characterised by significant packet losses in either or both directions between the wireless Access Points (APs) and wireless hosts, increased TCP-level retransmissions, rate reduction, throughput reduction, increased jitter, and roaming/hand-off effects. A PSI can last for several seconds to the point that it can be viewed as an outage. The frequency and intensity of PSI events in modern home and enterprise wireless networks is not well understood. Very few studies analyse the impact of PSI events on user experience. For various applications, a maximum tolerable end-to-end network delay has been estimated (e.g., about 150ms for VoIP applications). However, it is unclear whether the use of this fixed threshold as a “rule-of-thumb” can result in an efficient and effective adaptation for a wireless device. Depending on the temporal statistical characteristics of the packet losses and delays during a time period, the impact on the user experience may vary. Different wireless network conditions (e.g., handover, extreme channel impairment, high traffic load) vary in their temporal statistical characteristics in terms of packet errors, losses and delays. The main objective of this research is to understand the impact that such network conditions have on the user experience in the context of a VoIP call.

Network benchmarks, such as throughput, jitter, latency, and packet loss, have been used to quantify network performance. However, what is their relation with the perceived quality of experience metrics? We shift our attention from MAC- and network-based metrics to application-based benchmarks, and quantify user satisfaction under various wireless network conditions. Specifically, we focus on the performance of VoIP calls under various situations, namely, during a handover, under different background traffic conditions (normal and heavy traffic load/saturation conditions) at an AP, and in the presence of interference. We demonstrate that various network conditions exhibit distinct statistical behaviour in terms of SNR, packet losses and end-to-end delays, and thus, impact the VoIP user quality in a different manner. Only when their value exceeds a certain threshold, these parameters affect the quality of a call as perceived by the user. For example, only very low SNR values lasting over a relatively long time period impact the user experience of a VoIP call. On the other hand, a handover process may cause a relatively long disconnection from the Internet, resulting in a dramatic degradation of the perceived quality. Also, bursts of packet losses may not result in a significant user perceived quality degradation, if they last only a second or less, whereas packet loss spread in longer time periods may degrade the overall quality in a more prominent manner. Resource demanding applications, such as BitTorrent, may result in frequent packet losses and increased overall delays, degrading the VoIP user experience. The paper makes the following contributions: It distinguishes a number of scenarios, each of them focusing on a specific network condition, namely, handover, presence of interference, high contention, and saturation conditions. Based on each scenario/network condition, we perform empirical-based measurements, analyse the results, and identify the impact of that condition on the user experience. A novel part of this work is the comparative analysis of these conditions with respect to network and user experience benchmarks. It also proposes a Markov-chain based model for the auto-rate fallback (ARF) algorithm of the IEEE802.11 and derives analytically the packet losses and mean delays at the application layer under the presence of constant interference and a single traffic CBR source at the WLAN.

The rest of the paper is organised as follows: Section II outlines the related work. Section III describes our testbed, the different network conditions and discusses the performance analysis results. Finally, Section IV presents our main conclu-
sions and future work plans.

II. RELATED WORK

While there have been several studies discussing the network statistics under different conditions, most of them focus on the impact of these conditions on the aggregate throughput. Broustis et al. [1] provide guidelines for optimising the total throughput using an appropriate AP selection and topology control mechanism but without studying the effect of these mechanisms on user experience. Vasan et al. [2] propose two power control and frequency selection algorithms to increase throughput by reducing the impact of interference. Chen et al. [3] analysed the user satisfaction in Skype, employing the call duration as the quality benchmark. Specifically, they assumed that the more the user keeps a VoIP call active, the better the quality of the call. Hoene et al. [4] evaluated the call quality in adaptive VoIP applications and codecs. They showed that high-compression codecs (with relatively low voice quality) may behave better than top-quality codecs under packet losses and limited available bandwidth. Markopoulou et al. [5] focused on ISP network problems and showed that ISP networks suffer from PSIs (e.g., outages due to routing table changes, relatively frequent spikes in packet losses and delays) affecting the real-time applications. The IEEE 802.11 handover has been analysed and various improvements have been proposed. For example, Forte et al. [6] analysed the various delays involved in the handoff/reassociation process, namely the DHCP request, ARP query, and SIP invite, in an experimental testbed and the impact of the handoff on a SIP call. They reduced this overhead by enabling the wireless device to acquire a temporal address. SyncScan [7] aims to reduce the network unavailability during an AP handoff that causes a decrease in throughput due to the background scanning. It minimises the scanning by enabling the client to synchronise the scanning phase with the APs’ beacons.

The impact of interference on the performance of wireless network has been analysed. For example, Subramanian et al. [8] showed that the statistical properties of the interference affect the packet loss probability. For example, Bluetooth interference tends to affect low data rate packets more than high data rate packets, since their transmission lasts for a longer time period. Khan et al. [9] distinguished between the probability of a packet loss due to interference (PPER) and due to collisions (CPER) and showed via simulations that PPER is a better metric for rate adaptation than the total packet loss probability. Choi et al. [10] model the ARF algorithm of 802.11 with a Markov-chain process and compute the ARF parameters that maximise the throughput. On the other hand, our model focuses on the packet loss probability and mean end-to-end delay at the application layer, when a single wireless source produces CBR traffic in the presence of different levels of interference. Finally, Shin et al. [11] perform empirical-based measurements and simulations to estimate the capacity of an 802.11 network in terms of number of VoIP calls and analyse the impact of the preamble size, ARF algorithm, RSSI, packet loss, and scanning. They use as criterion for the quality of calls that the end-to-end delay should not exceed 150ms and the packet loss probability should be 3% or less.

III. PERFORMANCE ANALYSIS

We distinguish the following scenarios:

- handover: no background traffic, user mobility and client handover between wireless APs
- normal AP hotspot traffic: no user mobility, TCP flows corresponding to background traffic at a campus-wide wireless hotspot AP
- heavy UDP traffic: no user mobility, UDP flows saturating the wireless LAN
- heavy TCP traffic: no user mobility, TCP flows, generated by a BitTorrent client, saturating the wireless LAN
- presence of interference: no user mobility, no background traffic

We setup two control testbeds according to these scenarios and perform a number of empirical-based measurements under each scenario. Using the collected traces, we measure the network performance and the impact of each condition on the perceived user experience of the VoIP call.

H323 software and G.711 codec are used for all calls. The network adapter of the wireless VoIP client captures packets in promiscuous mode with IEEE 802.11+Radiotap pseudo-header provided by libpcap, using tcpdump with the appropriate settings. This header contains the RSSI value for each packet, the data rate, and the operating channel. User A initiates a VoIP call with user B.

The handover testbed includes one VoIP client connected via FastEthernet and one VoIP client connected via IEEE 802.11 to the ICS-FORTH infrastructure network. A user holding a wireless laptop (User A) roams in the premises of ICS-FORTH. While moving, the wireless client slowly fades out

\[1\] For a crude validation of the RSSI values, we used our spectrum analyser and compared the recorded values with its output. The fluctuation of the RSSI values was within 3 dBm.
of range of the AP and a handover is performed.  

In general, handoff between APs in wireless LANs can consume from one to multiple seconds, as associations and bindings at various layers need to be re-established. Such delays include the acquisition of a new IP address, duplicate address detection, the reestablishment of secure association, discovery of available APs. The overhead of scanning for nearby APs can be of 250ms, far longer than what can be tolerated by VoIP applications. The active probing (i.e., sending probe requests for discovering the available APs in different channels) in the handoff process of the IEEE 802.11 MAC is the primary contributor to the overall handoff latency and can affect the quality of service for many applications.

The background traffic testbed includes a VoIP client connected via 802.11, a VoIP client connected via FastEthernet, four wireless nodes connected via 802.11 and one node connected via FastEthernet. The four wireless nodes produce the background traffic according to predefined scenarios. All wireless nodes are connected to a single AP.

1) Normal hotspot AP traffic: We employed traces from the UNC/FORTH data repository [12], [13], [14], [15] that correspond to a real-life hotspot AP under normal traffic conditions in a large campus-wide wireless infrastructure. Each flow of the trace is "replayed" as a TCP flow with its sender and receiver corresponding to a wired and a wireless node, respectively. 5-minute traces are replayed as background traffic, while a VoIP call of the same duration is established between the two VoIP clients. The AP operates in 802.11b, in accordance with the APs of the UNC/FORTH traces. This type of traffic cannot overload the AP or cause any bottleneck at the AP, and thus, the network provides a near-excellent VoIP quality.

2) Heavy UDP traffic: This scenario focuses on the quality of VoIP under congestion caused by a large amount of traffic load generated by a small number of flows, overloading the AP. Each of the four wireless nodes sends packets of 1500 bytes of UDP traffic to a wired node at a 2 Mb/s data rate. The aggregate traffic exceeds the theoretical maximum throughput of an IEEE802.11 network (approximately 6 Mb/s [16]). The two VoIP clients initiate a call under this conditions. The AP operates in 802.11b mode. Saturation traffic can overload the AP. The main issues are the congestion of the wireless channel and the continuous contention of the wireless nodes.

3) Heavy TCP traffic: In this scenario, the background traffic is generated by one wireless node running a BitTorrent client, downloading three highly seeded files (while the VoIP call takes place). The BitTorrent protocol splits the files into small chunks and simultaneously downloads and uploads the shared chunks. In general, the number of generated flows in BitTorrent is high, often causing low-end routers to run out of memory and CPU. As in the previous scenarios, the AP operates in 802.11b mode. The BitTorrent protocol introduces a high number of small TCP flows in both uplink and downlink
directions, contending for the medium. This behaviour puts stress on the AP's queue, CPU and memory. In this scenario, the expected behaviour is that the TCP congestion control mechanism will repeatedly increase the throughput when the network is not congested, up to the point that packet losses occur. TCP will lower then the traffic demand, only for the next flow to increase its throughput.

We performed measurements of the aforementioned scenarios in controlled testbeds. We measured the end-to-end delay and packet loss of the VoIP flow under the different network conditions, namely, handover, normal AP hotspot, heavy UDP traffic, heavy TCP traffic at the application layer. To measure the performance of a VoIP call, we employed the E-model[17] and the Mean Opinion Score (MOS). The E-model depends on various factors, such as voice loudness, background noise, equipment impairment, packetisation distortion, codec robustness under various packet loss and end-to-end delays and impairments introduced by the packet loss and end-to-end delays [17].

The E-model is a psycho-acoustic computational model, that takes into consideration transmission parameters to estimate the quality of the user’s experience during a VoIP call. Various factors as voice loudness, background noise, equipment impairment, packetisation distortion, codec robustness under packet losses, end-to-end delay and round trip time are some of the parameters producing a R-factor, a rating that estimates the voice quality. ITU-T provides a R-to-MOS conversion formula.

The $R$ is produced by adding the perceived impairments of different nature into one value while the MOS is the average of scores of each call from 1 to 5, given by individuals in a controlled environment.

$$R = R_o - I_s - I_d - I_{e-eff} + A$$

The $R_o$ expresses the basic signal-to-noise ratio the user receives. It includes the loudness of the voice, the noise introduced by the circuit and by background sources. $I_s$ represents voice specific impairments, such as: too loud speech level (non-optimum OLR), non-optimum sidetone (STMR) and quantization noise (qdu). The term $I_d$ represents the impairments introduced by delay and echo effects. $I_{e-eff}$ is the equipment impairment factor, which represents impairments caused by low bit-rate codecs and the effect due to packet losses. $A$ is an "advantage factor". Its values depend on the knowledge of the users have for trading voice quality over convenience. All factors are extensively analysed in ITU-T’s G.107 recommendation (E-model), but we will briefly describe the ones affected by different network conditions.

The delay impairment factor $I_d$, representing all impairments due to delay is composed of:

$$I_d = I_{dte} + I_{dle} + I_{dd}$$

where $I_{dte}$ accounts for Talker Echo and $I_{dle}$ for Listener Echo. $I_{dd}$ represents the impairment when absolute delay $T_a$ is too long. This can happen even with perfect echo cancellation.

For $T_a \leq 100$ ms

$$I_{dd} = 0$$

For $T_a > 100$ ms

$$I_{dd} = 25 \left\{ \left( 1 + X^6 \right)^{\frac{1}{2}} - 3 \left( 1 + \left[ \frac{X}{3} \right]^6 \right)^{\frac{1}{2}} + 2 \right\}$$

where:

$$X = \frac{\log \left( \frac{T_a}{100} \right)}{\log 2}$$

The equipment impairment factor $I_{e-eff}$ represents the impairment introduced by faulty or lossy equipment ($I_e$). It accounts for the Packet-loss Probability ($Ppl$), the Packet-loss Robustness ($Bpl$) which is codec-specific and the Burst Ratio ($BurstR$). $I_{e-eff}$ is computed using this formula:

$$I_{e-eff} = I_e + (95 - I_e) \cdot \frac{Ppl}{Ppl_{burstR}} + Bpl$$

$BurstR$ is an index of the packet loss burstiness; it indicates whether packet losses occur closely together.

$$BurstR = \frac{\text{Average length of observed bursts in an arrival sequence}}{\text{Average length of bursts expected for the network under "random" loss}}$$

During call 1 in the handover scenario (Figures 6a and 6c), there is a fast handover (at the 57th sec). The overall quality is excellent (MOS>4) with minor changes when minor packet losses or delays are experienced. During the handover, all packets are lost (for about 300ms) and the MOS reaches 2, but only for one second. The client initiates active AP discovery at the 42nd sec of the VoIP call, sending probe-requests on all channels, receiving probe-response messages from available APs. The signal strength of the probe-requests is rather low (< -67 dBm) and the client does not try to authenticate/associate with a new AP. The client initiates a new round of AP discovery at the 51st sec. It receives probes from three APs, with RSSI as high as -52 dBm (while the
RSSI from the current AP is −80 dBm). While the client scans for APs, packets are queued up. As soon as the queue becomes empty (at the 57th sec), the client authenticates and reassociates with the new AP, hence the packet loss. The disconnection from the wireless LAN lasted for 319 sec. During call 2 (Figures 6b and 6d), four handovers are performed. The first three handovers (at the 9th, 21st and 47th sec, respectively) are performed in a similar manner as the handover in call 1. The fourth handover lasts longer than the previous three handovers as the AP deauthenticates the client (by sending “Previous authentication no longer valid” messages). According to the AP manufacturer [18], this is triggered by high error rates and too many unacknowledged retransmissions. After the 66th sec of the session, the client performs an active scan, where it receives probe response messages, including messages from one AP with high RSSI (greater than −60 dBm). The client remains tuned in the channel 5 until the 73rd sec, at which it is authenticated and associated with a new AP (channel 13) that was discovered during the previous phase of active scanning. The dramatic decrease of the MOS (in the Handover call 2) is due to the long disconnection from the wireless Internet caused by the overhead of the re-association process.

Both VoIP calls under the normal hotspot AP traffic (Figure 7) enjoy near packet-loss-free transmissions ($P_{pl} < 0.5\%$) with low delay (14 ms or less). It is apparent that this scenario does not saturate the network, which can be explained by the nature of the background traffic trace: Even though we used the AP in which the largest amount of traffic was recorded, the generated flows are of small size and rarely simultaneous. Their small size results in low throughput, since the TCP slow start algorithm does not reach a large window size. On the other hand, unlike in the BitTorrent (heavy TCP traffic scenario), there is a relatively small number of flows that transmit simultaneously, leaving the network underutilised. Even if a small number of flows were fully utilising the AP bandwidth, the TCP congestion control mechanism would limit its throughput, and the small UDP VoIP flows would ‘sneak through’.

In the VoIP under heavy UDP traffic scenario (Figure 8), the MOS deteriorates due to the high packet delays. In this scenario, the very large delays (as shown in Table I) are due to the presence of heavy background traffic resulting in an arrival rate higher than the ‘service’ rate at the AP [19]. Indeed, a saturated network with full buffers will increase the mean
delay values, trying to deliver all packets and occasionally dropping packets from the queue when a timeout occurs. For the VoIP user, the quality offered is mediocre (MOS < 2.9). This scenario emphasises the need for a prioritisation scheme for different traffic classes.

In the VoIP under heavy TCP traffic scenario (Figures 9a and 9c), the first call suffers from packet losses (6.5%) and delays. The overall voice quality is acceptable (3.5 MOS), while packet delays exceed the 150ms psychological threshold. The nature of the BitTorrent protocol can explain this behaviour: a BitTorrent client initiates many flows, with small payload sizes. Each flow tries to expand its TCP window, up to the point that packet losses occur. This flow drops its throughput, only to be followed by the next one. Since the number of flows at any given time is large, this behaviour is repeated frequently, causing severe performance degradation. The second call under heavy TCP traffic (Figures 9b and 9d) exhibits lower packet delays than the first one but almost a double packet loss percentage, resulting in a deteriorated MOS of 2.86. This is caused by the extensive packet losses (that reach a 50% during the 19th-25th sec of the call). Once again, a large number of flows initiated by the BitTorrent client saturates the wireless LAN.

The following scenario illustrates the impact of the interference on the quality of a VoIP call. A packet loss occurs when the MAC layer fails to forward a packet to its destination. The MAC layer retransmits a packet (up to a certain number, typically 7, including the first attempt). If all these transmissions fail, the corresponding packet is discarded. When ARF is enabled and two consecutive transmissions fail, the data rate is decreased. The physical layer will use a more robust modulation scheme to increase the probability of a successful packet transmission. After ten successful consecutive transmissions, the data rate is increased. We assume that during the VoIP call, the interference remains constant and appears as noise. Furthermore, we assume that there is no other AP in a different channel available (scanning is disabled). We develop a Markov-chain-based model to describe the ARF algorithm and compute the packet loss probability and mean end-to-end delay at the application layer for various SNR values. Unlike the previous scenarios that were explored using empirical measurements, this scenario was studied analytically.

A simple model to describe themechanisms of the physical and MAC layer is a Markov chain (Figure 10). If we define the set of available modulation schemes as $M$, then for each $m_i \in M$ we can produce the following set of states:

$$S_i = \{(m_i, n)|n = -1, 0, \ldots , 9\} \quad (8)$$
We say that a wireless device is at the state \((m_i, n)\), for \(n = 0, 1, \ldots, 9\) when it currently uses the modulation scheme \(m_i\) and \(n\) consecutive transmit attempts have already been successful using this modulation scheme. On the other hand, a device is at state \((m_i, -1)\) if the previous transmit attempt happened using the modulation scheme \(m_i\) and failed. From the state \((m_i, 9)\) if a packet is transmitted correctly, a transition occurs to the modulation scheme with the next higher data rate \((m_{i+1}, 0)\). Similarly if a packet is lost while being at the state \((m_i, -1)\), a transition occurs to the modulation scheme with the next lower data rate \((m_{i-1}, 0)\). It is also important to notice that for the modulation scheme with the highest data rate \((m_9)\) only two states are required, namely \((m_9, -1)\) and \((m_9, 0)\). This is due to the fact that a transition to an even higher data rate is impossible so states \((m_9, 1)\) through \((m_9, 9)\) are meaningless. On the other hand, for the modulation scheme with the lowest data rate \((m_0)\) there is no need for a state \((m_0, -1)\). This is because there is no case that a transition could occur to a more robust modulation scheme.

Let us now denote as \(R(\gamma) = [R_1(\gamma) \ R_2(\gamma) \ \ldots \ R_8(\gamma)]^T\) the vector which contains the packet error probabilities for the 8 available modulation schemes of 802.11g, where \(\gamma\) is the value of SNR at the receiver. From every state of the

<table>
<thead>
<tr>
<th>Scenario / Network conditions</th>
<th>Mean delay (ms)</th>
<th>Std delay (ms)</th>
<th>Packet loss (%)</th>
<th>Std packet loss (%)</th>
<th>BurstR</th>
<th>MOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handover 1</td>
<td>27.34</td>
<td>51.14</td>
<td>0.39</td>
<td>2.24</td>
<td>8</td>
<td>4.35</td>
</tr>
<tr>
<td>Handover 2</td>
<td>57.51</td>
<td>109.76</td>
<td>11.84</td>
<td>28.93</td>
<td>108.75</td>
<td>2.41</td>
</tr>
<tr>
<td>Normal hotspot AP traffic 1</td>
<td>11.23</td>
<td>0.61</td>
<td>0.37</td>
<td>0.86</td>
<td>1.17</td>
<td>4.37</td>
</tr>
<tr>
<td>Normal hotspot AP traffic 2</td>
<td>15.71</td>
<td>0.59</td>
<td>0.05</td>
<td>0.41</td>
<td>1.32</td>
<td>4.39</td>
</tr>
<tr>
<td>Heavy UDP traffic 1</td>
<td>411.78</td>
<td>8.80</td>
<td>1.69</td>
<td>1.66</td>
<td>1.69</td>
<td>2.84</td>
</tr>
<tr>
<td>Heavy UDP traffic 2</td>
<td>499.23</td>
<td>11.4</td>
<td>2.57</td>
<td>2.09</td>
<td>1.78</td>
<td>2.57</td>
</tr>
<tr>
<td>Heavy TCP traffic 1</td>
<td>154.78</td>
<td>33.35</td>
<td>6.50</td>
<td>10.64</td>
<td>1.58</td>
<td>3.50</td>
</tr>
<tr>
<td>Heavy TCP traffic 2</td>
<td>96.28</td>
<td>104.10</td>
<td>11.24</td>
<td>15.42</td>
<td>2.11</td>
<td>2.86</td>
</tr>
</tbody>
</table>
Markov chain only two transitions occur. If the i-th modulation scheme is currently used, the first transition corresponds to a successful packet transmission with probability \(1 - R_i(\gamma)\). The second transition corresponds to a transmission failure with probability \(R_i(\gamma)\). If we now define an enumeration of states, we could construct the transition matrix of the Markov chain. This matrix is denoted as \(T\). The steady state probability distribution (ps) which corresponds to matrix \(T\) can be computed by the following set of equations:

\[
\begin{align*}
T \cdot p_s &= p_s \\
1^T \cdot p_s &= 1
\end{align*}
\]

(9)

The first of them is used to express that vector \(p_s\) is an equilibrium point of the transformation that is performed by matrix \(T\). On the other hand the second equation is a constraint which is used in order to ensure that vector \(p_s\) is a probability density function. Finally symbol \(1^T\) denotes the vector \([1 \ 1 \ \ldots \ 1]^T\). Equation (9) can be rewritten as follows:

\[
T \cdot p_s = p_s \\
1^T \cdot p_s = 1
\]

(9)

The steady state probability distribution can be computed as the solution of the following set of equations:

\[
\begin{bmatrix}
T_1^T \\
T_2^T \\
\ldots \\
T_N^T
\end{bmatrix} \cdot \begin{bmatrix}
p_{s1} \\
p_{s2} \\
\ldots \\
p_{sN}
\end{bmatrix} = \begin{bmatrix}
p_{s1} \\
p_{s2} \\
\ldots \\
p_{sN}
\end{bmatrix}
\]

(10)

Vectors \(T_1^T, T_2^T, \ldots, T_N^T\) are the rows of matrix \(T\) and are linearly dependent because:

\[
T_1^T + T_2^T + \ldots + T_N^T = 1^T
\]

(11)

Equation (11) is an attribute of the transition matrix \(T\) and can be used in order to eliminate the Nth equation from system (10). The steady state probability distribution can be computed as the solution of the following set of equations:

\[
\begin{bmatrix}
T_1^T - e_1 \\
T_2^T - e_2 \\
\ldots \\
T_{N-1}^T - e_{N-1}
\end{bmatrix} \cdot \begin{bmatrix}
p_{s1} \\
p_{s2} \\
\ldots \\
p_{sN}
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
\ldots \\
1
\end{bmatrix}
\]

(12)

where: \(e_1 = [1 \ 0 \ldots \ 0]^T\), \(e_2 = [0 \ 1 \ldots \ 0]^T\), \ldots, \(e_{N-1} = [0 \ 0 \ldots \ 1 \ 0]^T\).
Let us now define the binary random variable $X_i$ which indicates whether or not the i-th packet at the application layer and takes the value 1 if the packet is transmitted correctly. Also the random variable $Y_i$ indicates the state of the system at the beginning of the transmission of the i-th packet. We estimate the average packet loss probability ($P_{pl}$) by measuring the $P(X_i = 0)$ assuming that the system operates at the steady state:

$$P(X_i = 0) = \sum_{k=1}^{N} P(X_i = 0|Y_i = k)P(Y_i = k)$$  \hspace{1cm} (13)

In order to measure the burstiness of packet losses it would be useful to estimate also the $P(X_i = 0|X_{i-1} = 0)$. This is the probability that a packet is lost given that the previous packet was also lost and can be computed by the following equation:

$$P(X_i = 0|X_{i-1} = 0) = \frac{P(X_i = 0,X_{i-1} = 0)}{P(X_{i-1} = 0)} = \frac{\sum_{k=1}^{N} P(X_i = 0,X_{i-1} = 0|Y_{i-1} = k)P(Y_{i-1} = k)}{\sum_{k=1}^{N} P(X_{i-1} = 0|Y_{i-1} = k)}$$  \hspace{1cm} (14)

Probabilities $P(X_i = 0|Y_i = k)$ and $P(X_i = 0, X_{i-1} = 0|Y_{i-1} = k)$ for $k=1,2,\ldots,N$ can be easily computed from the Markov chain.

Another parameter that should also be estimated in order to compute the value of MOS is the mean end-to-end delay. Let us now assume that an arbitrary packet that is produced by the wireless client is received successfully from the AP after $k$ transmit attempts. The mean end-to-end delay $D(k)$ can be estimated as the sum of the wired transmission and propagation delay, the aggregate wireless transmission delay (for all the copies of the packet), the wireless propagation delay, the aggregate timeout and the mean delay due to contention.

Let us now define a new random variable $A_i$. Its value corresponds to the number of transmission attempts that are performed, for the i-th packet to be delivered to its destination. Also if the i-th packet is lost the value of $A_i$ is 0. The mean end-to-end delay can now be computed as follows:

$$E\{D\} = \sum_{k=1}^{7} D(k)P(A_i = k)$$  \hspace{1cm} (15)

The value of $P(A_i = k)$ in the above equation can be estimated by the law of total probability:

$$P(A_i = k) = \sum_{j=1}^{N} P(A_i = k|Y_i = j)p_{kj}$$  \hspace{1cm} (16)

Finally the probabilities $P(A_i = k|Y_i = j)$ for $j = 1,2,\ldots,N$ can be computed from the Markov chain.

Figure 11a shows the packet loss probability at the application layer and the conditional packet loss probability when the previous packet (at the application layer) is also lost. Interestingly, these probabilities are almost the same for the entire range of SNR values. In other words, the probability of a packet loss does not depend on whether the previous packet was successfully received or not. Thus, under strong interference the packet loss follows a Bernoulli process. We compute the mean end-to-end delay for various SNR values (in the range of 0 - 11 dB), which is 5ms or less. Based on the packet loss probability ($P_{pl}$) and mean end-to-end delay ($I_{dd}$) for various SNR values, we can determine the R-factor and the MOS (as shown in Figure 11b). As in the previous scenarios, the G.711 codec parameters were also used here. The user perceived quality of VoIP is affected only in the case of high interference: the MOS drops only when the SNR value is so low that even the most robust modulation scheme results in a high BER. A similar observation was also made in [11].

Delay can be caused by heavy traffic generated by one or more clients in a wireless LAN or by very low SNR values that trigger the scanning mechanism. Packet loss can be caused by low SNR or collisions, or extremely high delays due to heavily
utilised buffers at the AP. Extremely delayed VoIP packets are dropped at the application buffer (the user perceives them as lost). To improve the handover process in the case of traffic demand with a relatively high interarrival time (e.g., about 20 ms, as in G.711 codec), the client could scan and listen for available APs during the idle time between two packet arrivals. 20 ms is a sufficient period for a client to tune in a different frequency, transmit or listen for a few packets and retune in the previous channel, before a new voice packet arrives.

IV. CONCLUSION AND FUTURE WORK

The paper shows that only very low SNR values affect the user experience of a VoIP call, since the ARF algorithm ‘hides’ the packet losses at the MAC layer and only the losses at the most robust (and lowest bitrate) modulation scheme at the MAC layer become noticeable from the application layer. Furthermore, it demonstrates situations in which common “rule-of-thumb” metrics (e.g., the “150ms threshold of the mean end-to-end delay”) cannot reflect the perceived user experience and finer cross-layer data are required for identifying the network conditions and performing an effective adaptation. Examples of such situations are handovers with long bursts of packet losses or heavy TCP background traffic. Also, network conditions with low quality exhibit different statistical characteristics in terms of packet loss and delays.

We plan to extend the evaluation by employing additional benchmarks, such as PESQ and subjective listening tests with a number of users, and by experimenting with other codecs, such as AMR-WB, Speex and SILK, that include packet loss concealment to make lost frames inaudible. Furthermore, it would be interesting to develop a MOS prediction model that takes into consideration a number of cross-layer measurements, such as statistical characteristics of the MAC retransmissions and delays, type of control packets, and SNR values, to predict when the MOS will be under a certain threshold and trigger an adaptation process. For example, we speculate that a continuous degradation of RSSI indicates the need for a handover (potentially to a different AP or interface) to avoid a low quality in the VoIP call. Similarly, a continuous increase of the average delay in the presence of UDP background traffic will likely result in low MOS. A smoother degradation in the MOS is more likely in the presence of a few TCP flows as background traffic. To conclude, not all network conditions result in the same type of application-layer delays and packet losses, and thus, do not impact the quality of VoIP applications in the same manner. Understanding which conditions cause severe impairment in the VoIP application, and which cross-layer measurements can be used to predict such impairment is important in the design of adaptation mechanisms.

REFERENCES


