Multimedia Systems (2020) 26:643–654
https://doi.org/10.1007/s00530-020-00674-5

REGULAR PAPER

MQTT-SN, CoAP, and RTP in wireless IoT real-time communications

Rolando Herrero[1]

Received: 23 December 2019 / Accepted: 5 July 2020 / Published online: 14 July 2020

Abstract

A great number of Internet of things (IoT) applications rely on real-time communication (RTC) mechanisms for transmission of media. Essentially, applications analyze and process media to make decisions that typically affect actuation and control of embedded devices. IoT networks, however, are subjected to constrains that limit the computational and resource complexity of all entities involved. This is particularly critical when considering the traditional RTC protocols like real-time protocol (RTP) that was not designed to perform well in the context of low-power lossy networks (LLNs). This paper focuses on alternatives to media transport in IoT networks. Specially, constrained application protocol (CoAP) and the message queuing telemetry transport sensor network protocol (MQTT-SN) are presented as valid technologies for media propagation in LLNs. The paper models and compares CoAP, RTP, and MQTT-SN to determine the most efficient scenario for audio, speech, and video transmission.

Keywords IoT · LLN · CoAP · MQTT-SN · RTP

1 Introduction

Different IoT applications have different quality-of-service (QoS) requirements. In particular, certain IoT applications that rely on real-time communication (RTC) are highly sensitive to network latency and packet loss that can seriously affect the performance of the solution. For example, in a scenario where unmanned aerial vehicles (UAVs) are used to capture and transmit hyperspectral images to a central processor known as the planner, any delay in decision-making becomes a liability as it can lead not only to failure but also to physical and personal damage. In the context of low-power lossy networks (LLNs) and as shown in Fig. 1, long battery life is the ultimate cause of high network packet loss and low transmission rates that, in turns, are responsible of excessive network latency. Specifically, when power consumption is lowered to preserve battery life, the signal-to-noise ratio lowers, resulting in decreased Raleigh fading channel capacity [1] and higher bit error rate (BER). Limited channel capacity leads to low transmission rates, while high BER causes packets to be dropped by error detection mechanisms. Moreover, high network packet loss induces retransmissions than combined with low transmission rates which leads to increased network latency.

Retransmissions, on the other hand, can be introduced at different layers depending on the protocol. For example, IEEE 802.15.4 supports, under certain circumstances, optional timeout-based retransmission at the media access control (MAC) layer, while CoAP supports a confirmable transmission mode that by means of retransmission provides session layer reliability. When media packets are involved, however, the latency introduced by retransmissions can seriously degrade quality as, from an end-user experience, the maximum tolerable delay is around 150 ms [2]. For real-time applications where decisions are made real time, this latency tolerance threshold is even lower.

As shown in Fig. 1, delayed packets result in application packet loss, because packets arriving too late provide no relevant information and can be considered lost. Hence, a by-product of extended battery life is application packet loss that results from both media loss due to retransmissions arriving too late and from network packet loss itself. It is, therefore, expected that protocols that do not natively support retransmissions exhibit better packet loss performance. Moreover, the effect of network packet loss on application

Communicated by R. Steinmetz.

✉ Rolando Herrero
r.herrero@northeastern.edu

[1] Northeastern University, 360 Huntington Avenue, Boston, MA 02115-5000, USA

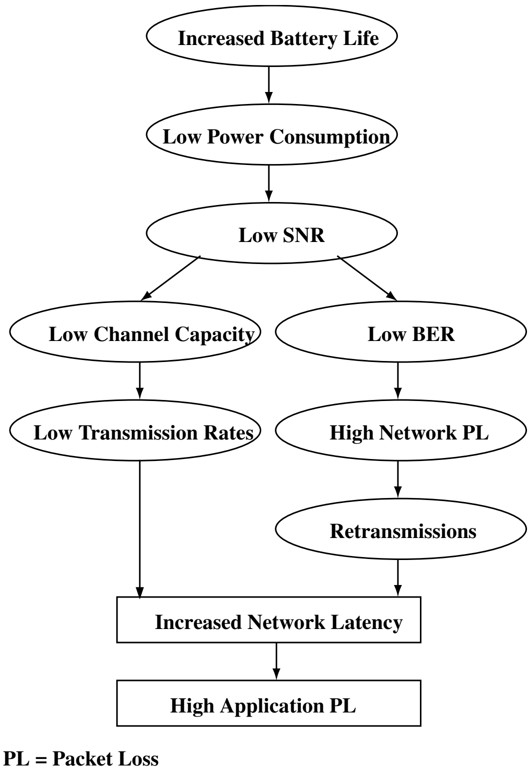


Fig. 1 Battery life and latency

packet loss can be neutralized by efficient forward error correction (FEC) techniques [3] and not by retransmissions.

Traditional RTC systems rely on real-time protocol (RTP) for media communications, and this protocol runs on top of the user datagram protocol (UDP) and, therefore, avoids packet loss due to delays. Moreover, RTP has been extended to IoT by means of the IoT-RTP adaptation proposed in [4] as an energy-aware mechanism. One of the problems with RTP, and IoT-RTP in particular, is that it is not optimized for use in small maximum transmission unit MTU size networks characteristic of IEEE 802.15.4 and other LLN wireless technologies like bluetooth low energy (BLE). Note that RTP is typically negotiated by means of the session initialization protocol (SIP) which is not only computationally complex but also throughput demanding. In addition, most embedded devices have built-in firmware support of IoT-specific session layer protocols like CoAP and MQTT but not necessarily RTP/SIP. It is, therefore, advantageous to evaluate the performance of both CoAP and MQTT when configured with no retransmissions as a mechanism to provide efficient RTC without suffering from the overhead of the RTP/SIP combo.

CoAP is a throughput efficient session layer protocol that provides representational state transfer (REST) functionality similar to that of the hypertext transfer protocol (HTTP) but with a minimalistic approach that attempts to low-power consumption [5]. In the context of RTC applications, CoAP relies on IPv6 over low-power wireless personal area networks (6LoWPAN) compressed UDP transport in a non-confirmable "fire-and-forget" fashion that prevents retransmissions [6]. Moreover, additional improvements to the protocol like those specified in RFC 7959 "Block-Wise Transfers in the constrained application protocol (CoAP)" [7] provide functionality that further enhances media applications. MQTT is a lightweight session protocol based on the subscribe/notify paradigm characterized by small size, low-power usage, and efficient distribution of information to many receivers. MQTT, however, relies on transport control protocol (TCP), so, in certain environments with high packet loss, the overall application latency becomes impractical as result of retransmissions [8]. Message queuing telemetry transport sensor network protocol (MQTT-SN) is an adaptation of MQTT for sensor networks that relying on UDP transport and through different QoS levels provides a retransmission free mechanism ideal for media transmission.

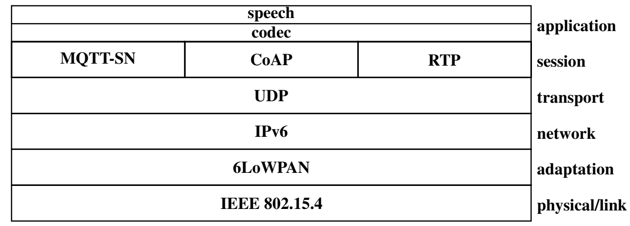


Fig. 2 MQTT-SN vs CoAP vs RTP

In this paper, we analyze the performance of these three session layer protocols, that is MQTT-SN, CoAP, and RTP, when used in a wireless IoT network where IEEE 802.15.4 provides physical as well as data link layers and IPv6 is adapted by means of 6LoWPAN, as shown in Fig. 2. Because of the throughput limitations of this scheme, the media under consideration are speech that is compressed by means of state-of-the-art codecs and transported for real-time analysis and decision-making. Note that digital signal processing (DSP) capabilities of most basic embedded processors, like the ARM Cortex M4, provide enough flexibility to efficiently support speech processing algorithms like the Durbin–Levinson recursion. With this in mind, first, we mathematically model media transmission by means of MQTT-SN, CoAP, and RTP to infer media quality out of different network conditions and assess which mechanism exhibits the best performance. Second, we optimize each instance of these protocols by fine-tuning parameters that can be used to improve application impairments. Finally, we evaluate the performance of each scheme and rely on well-known industry media assessment algorithms to obtain speech quality scores.

The remainder of the paper is organized as follows: related work is introduced in Sect. 2; the analytical model and the testing framework are presented in Sects. 3 and 4 respectively. The comparison between this model and the experimental results obtained by applying network impairments is given in Sect. 5. Conclusions are provided in Sect. 6.

2 Literature review and motivation

RTC in the context of IoT is a fairly recent research topic where several mechanisms have been proposed to accomplish efficient transmission of media. In Ref. [9], the authors analyze the experimental performance of HTTP-based media transmission over 6LowPAN transport. They specifically focus on video streaming by means of well-known codecs like International Telecommunication Union (ITU) H.264 . Similarly, video link utilization as well as power consumption are evaluated in Ref. [10]. The paper addresses throughput limitations in the context of video transmissions in LLNs. In Ref. [11], the authors use CoAP for media transmission in a surveillance system in health care scenarios. They propose a solution that integrates cloud computing with embedded sensors for transmission of video streams over CoAP by means of the ITU H.265 codec. The performance of CoAP for real-time sensor data transmission is analyzed in Ref. [12]. Specifically, a mathematical model is introduced to understand the performance of the different modes of operation of CoAP. A media streaming mechanism that relies on CoAP is presented in Ref. [13]. The mechanism is highly optimized to support synchronization between sender and receiver. In Ref. [14], the authors analyze the implications of media transmission in the context of 6LoWPAN fragmentation. The mechanism relies on FEC applied on top of 6LoWPAN to overcome the impairments associated with LLNs. In Ref. [15], the authors propose a mechanism to propagate speech by means of MQTT. The authors propose an end-to-end solution that integrates embedded devices with cloud-based Natural language processing. From a traditional media propagation perspective, the use of a constrained version of SIP is presented in Ref. [16]. SIP is upgraded to focus on power consumption to improve system performance. In Ref. [4], the authors modify the traditional RTP and develop IoT-RTP to provide media transmission capabilities in the context of IoT. The authors fails, however, to take into account the throughput limitations due to the lower layer protocols like IEEE 802.15.4 and 6LoWPAN. In all cases, these papers focus on a specific session layer protocol without comparing it against other possible alternatives. Moreover, none of them really obtain common media quality scores besides other performance metrics like application layer packet loss, throughput, and power consumption. To the best of our knowledge, no other paper in the literature addresses the comparison of MQTT-SN, CoAP, and RTP in the context of IoT media transmission.

3 Modeling session layer protocols

Obtaining mathematical models for all three session layer protocols, that is MQTT-SN, CoAP, and RTP, not only serves as a way to compare performances but also as a tool to predict behavior on different network conditions. This section explores the effect of network conditions like packet loss and latency on the quality of speech depacketized at the application layer. Since the main source of quality degradation is due to missing packets at the output of the playout buffer, analytical evaluation is by means of application layer packet loss. Before analyzing each protocol, however, it is important to understand the wireless IoT channel model.

3.1 Gilbert–Elliot two-state channel model

Bursty packet loss, the main characteristic of dynamic multipath fading in wireless environments, can be accurately represented by the Gilbert–Elliot two-state Markov channel model [17], as shown in Fig. 3. The model introduces a good and a bad state that imply low and high packet loss with loss rates given by e_{G} and e_{B}, respectively. Gibert [18] further simplifies the model with the special case of an error-free good state ($e_{\mathrm{G}} = 0$) and left

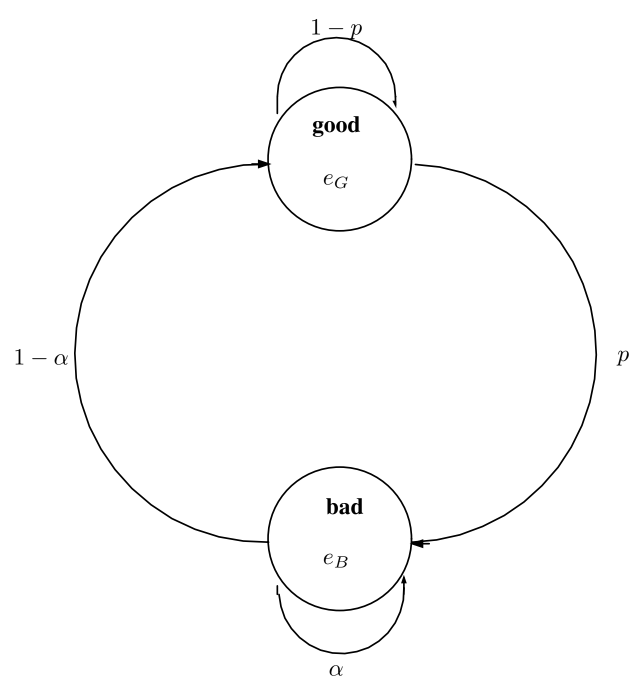


Fig. 3 Gilbert–Elliot channel model

the extension to include losses generated in both states to Elliot [19].

The model is characterized by four parameters; (1) p, the transition probability from the good state to the bad state, (2) α, the probability that the channel remains in the bad state, (3) e_G, the packet loss probability when the channel is in the good state, and (4) e_B, the packet loss probability when the channel is in the bad state. When simplifying the model by assuming $e_G = 0$ and $e_B = 1$, the steady-state probabilities that the channel is either in a "bad" or "good" state are given by:

$$P_{\text{bad}} = \frac{p}{1-\alpha+p},$$

and

$$P_{\text{good}} = \frac{1-\alpha}{1-\alpha+p},$$

respectively.

In the context of IoT, wireless communications by means of IEEE 802.15.4 result in a wireless personal area network (WPAN) that is characterized by packet loss, $P^{\text{loss}}_{\text{wpan}}$, given by:

$$P^{\text{loss}}_{\text{wpan}} = e_B P_{\text{bad}} + e_G P_{\text{good}} = P_{\text{bad}}, \tag{1}$$

where it is assumed, as previously indicated, that no loss occurs in a good state channel. Figure 4 shows how $P^{\text{loss}}_{\text{wpan}}$ changes with network layer packet loss and its burstiness. As it can be seen, this probability results from the effect of α amplifying the network packet loss probability p.

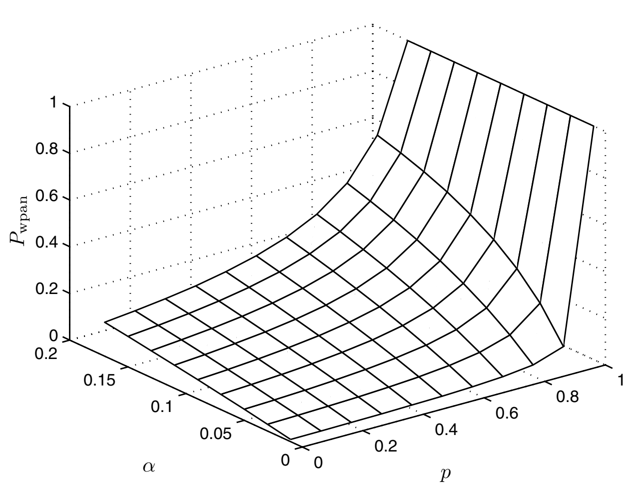


Fig. 4 $P^{\text{loss}}_{\text{wpan}}$ vs α and p

3.2 MQTT-SN

MQTT traditionally supports three reliability modes known as QoS level 0 "at most once", QoS level 1 "at least one", and QoS level 2 "exactly once" that affect the way in which client, servers, and brokers interact. MQTT-SN introduces a new QoS level −1 which provides simple functionality by allowing a sensor to send its data, without any congestion control, to a gateway. The gateway, in turns, publishes these messages to the MQTT broker that forwards the traffic to an analytics server for further processing or playback. As shown in Fig. 5, the protocol stack between sensor and gateway is WPAN IEEE 802.15.4/6LoWPAN/UDP-based, while between gateway, broker, and analytics is wide area network (WAN) IEEE 802.3/IPv6/TCP-based. To suppress MQTT-based retransmissions, the QoS level is set to 0 for communications to and from the broker. MQTT, however, is subjected to retransmissions in accordance to well-known TCP congestion control mechanisms that must be fine-tuned to minimize their effect on media. Specifically, the Nagle algorithm [20] is disabled to eliminate any additional buffering delays. In Fig. 5, sensor traffic is transmitted to the gateway by means of MQTT-SN over UDP transport. Traffic arriving at the gateway is then forwarded by means of MQTT QoS level 0 over TCP transport to the broker and from the broker by the same mechanism to the analytics application. Packet loss affects MQTT-SN by dropping datagrams (see datagram 5), while, on the contrary, MQTT QoS level 0 datagrams are not typically affected by loss as TCP retransmissions guarantee reliable delivery (see datagram 2). TCP performs traffic aggregation forwarding new and retransmitted datagrams to destination. In addition, for media IoT applications that require little to no latency, and as shown in Fig. 5, packets

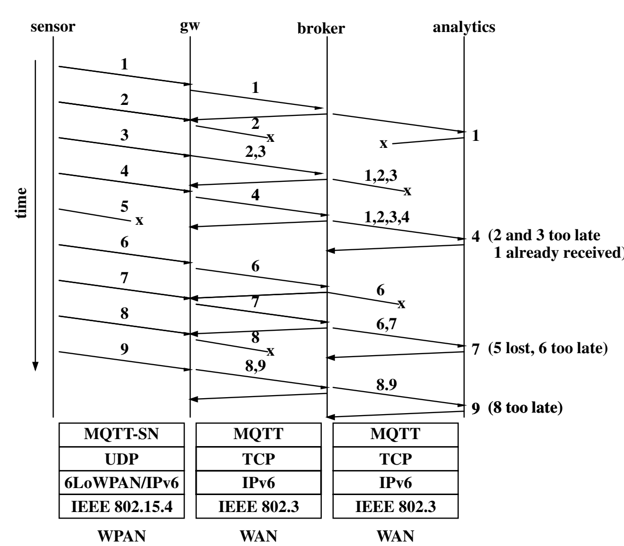


Fig. 5 MQTT-SN

that arrive too late are lost packets. In all, the scheme to estimate application packet loss has to take into account the losses due to MQTT-SN and those introduced by the delays that result from the TCP stacks transporting MQTT traffic.

In this scenario, the application packet loss probability defined as $P^{\text{loss}}_{\text{mqtt}}$ is given by:

$$P^{\text{loss}}_{\text{mqtt}} = 1 - \left(1 - P^{\text{loss}}_{\text{wpan}}\right)\left(1 - P^{\text{loss}}_{\text{wan}}\right)^2, \tag{2}$$

where $P^{\text{loss}}_{\text{wpan}}$ and $P^{\text{loss}}_{\text{wan}}$ are the WPAN and WAN packet loss probabilities, respectively. In this simplified model, the core side packet loss effect is minimized by disabling the Nagle algorithm. Note that on the WAN side, it is assumed that the speech frames are much smaller than the TCP maximum segment size (MSS), such that an array of multiple frames fits inside a single segment. This assumption is valid, since even for waveform speech codecs that exhibit very low compression rates like ITU-T Rec. G.711, a 160-byte payload is an order of magnitude smaller than the MSS associated with the IEEE 802.3 physical layers. Additionally, because MQTT relies on persistent TCP connections, the impairments overhead due to connection maintenance and establishment are not considered.

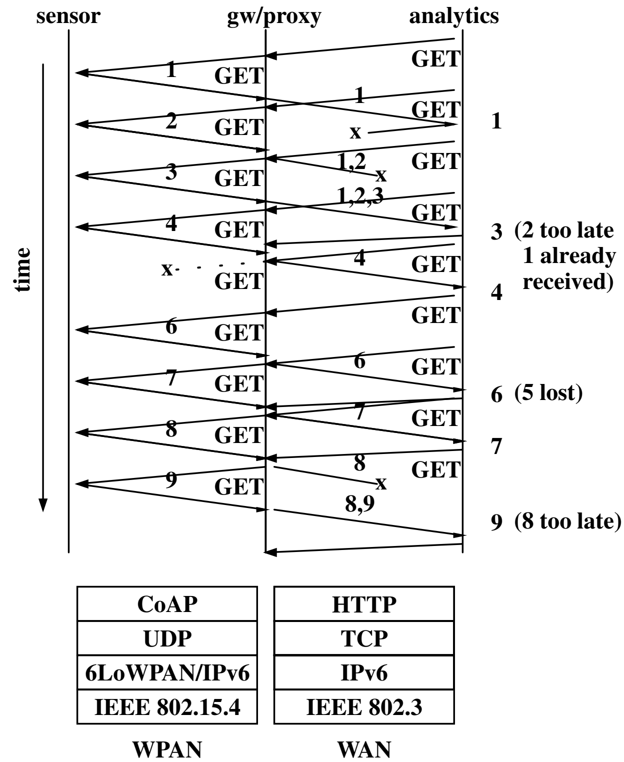


Fig. 6 CoAP/HTTP

3.3 CoAP

In this scenario, CoAP is configured in non-confirmable mode to prevent session layer retransmissions. A standard CoAP/HTTP proxy runs alongside the gateway to transform back-and-forth CoAP and HTTP traffic. As shown in Fig. 6, access traffic between sensor and gateway is CoAP-based, while core traffic between gateway and analytics server is HTTP-based. HTTP introduces retransmissions due to the nature of the TCP transport layer protocol, but its effect can be minimized by making sure, as in the MQTT-SN case in Sect. 3.2, that the Nagle algorithm is disabled. In Fig. 6, the analytics application requests sensor readouts by transmitting GET requests through HTTP over TCP. Requests are converted to CoAP over UDP, forwarded to the sensor, and affected by loss. The sensor traffic is transmitted to the gateway by means of CoAP over UDP transport. Traffic arriving at the gateway is then forwarded by means of HTTP over TCP transport to the analytics application. Packet loss affects CoAP by dropping datagrams, while, on the contrary, HTTP datagrams are not typically affected by loss as TCP retransmissions guarantee reliable delivery (see datagram 8). TCP performs traffic aggregation forwarding new and retransmitted datagrams to destination. Note that his scenario is based on the assumption that most core IoT infrastructure is HTTP based and, therefore, CoAP is left for access only. Alternatively, as shown in Fig. 7, when the core IoT infrastructure supports CoAP, it is possible to have end-to-end CoAP over UDP traffic that can take advantage of observation,

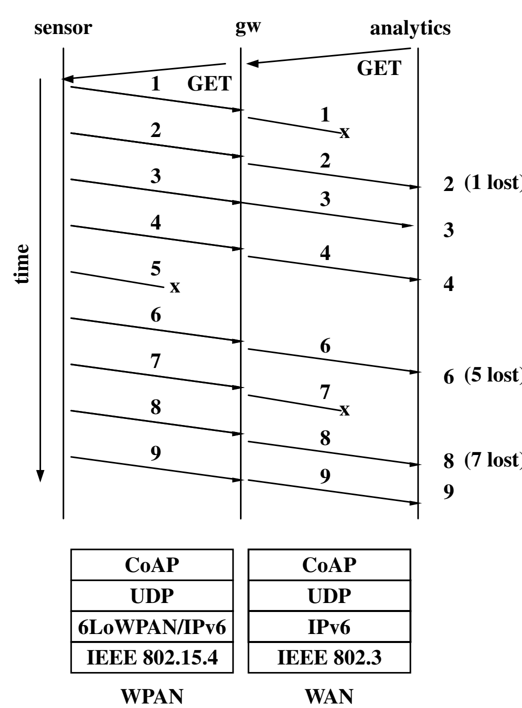


Fig. 7 End-to-end CoAP with observation

specified in IETF RFC 7641 "Observing Resources in the Constrained Application Protocol" [21]. In this case, a single CoAP GET message is used to enable a sensor to transmit data as it becomes available. In Fig. 7, the analytics applications request sensor readouts by transmitting a single CoAP GET request through CoAP over UDP. The requests are forwarded to the sensor and affected by loss. The sensor traffic is transmitted to the gateway by means of CoAP over UDP transport. Traffic arriving at the gateway is then forwarded to the analytics application. Packet loss affects CoAP by dropping datagrams (see datagrams 5 and 1).

The application packet loss probability for the CoAP/HTTP case, defined as $P^{\text{loss}}_{\text{coap/http}}$, is given by:

$$P^{\text{loss}}_{\text{coap/http}} = 1 - \left(1 - P^{\text{loss}}_{\text{wpan}}\right)^2 \left(1 - P^{\text{loss}}_{\text{wan}}\right)^2, \qquad (3)$$

where $P^{\text{loss}}_{\text{wpan}}$ and $P^{\text{loss}}_{\text{wan}}$ are the WPAN and WAN packet loss probabilities, respectively. As in the MQTT case, the HTTP protocol on the WAN side relies on TCP transport. As before, the Nagle algorithm is disabled and it is assumed that the MSS is an order of magnitude larger than the maximum speech frame size. In addition, HTTP is configured in persistent mode to rely on a persistent TCP connection that removes the latency introduced by reconnections. On the other hand, the application packet loss probability for the end-to-end CoAP case, defined as $P^{\text{loss}}_{\text{coap}}$, is given by:

$$P^{\text{loss}}_{\text{coap}} = 1 - \left(1 - P^{\text{loss}}_{\text{wpan}}\right)\left(1 - P^{\text{loss}}_{\text{wan}}\right), \qquad (4)$$

where, again, $P^{\text{loss}}_{\text{wpan}}$ and $P^{\text{loss}}_{\text{wan}}$ are as previously defined.

3.4 RTP

RTP does not introduce any retransmissions as it is an end-to-end UDP based protocol. As shown in Fig. 8, RTP only provides traffic ordering by means of sequence numbers and timestamps, but does not introduce of any congestion control mechanism. Since no TCP transport retransmissions are involved, all loses are due to network impairments and media packets typically arrive on time for processing. In Fig. 8, sensor traffic is transmitted to the gateway by means of RTP over UDP transport. Traffic arriving at the gateway is then forwarded by means of RTP over UDP transport to the analytics application. Packet loss affects RTP by dropping datagrams (see datagrams 5 and 2).

In this scenario, the application packet loss probability defined as $P^{\text{loss}}_{\text{rtp}}$ is given by:

$$P^{\text{loss}}_{\text{rtp}} = 1 - \left(1 - P^{\text{loss}}_{\text{wpan}}\right)\left(1 - P^{\text{loss}}_{\text{wan}}\right), \qquad (5)$$

where, as before, $P^{\text{loss}}_{\text{wpan}}$ and $P^{\text{loss}}_{\text{wan}}$ are the WPAN and WAN packet loss probabilities, respectively. Since transport is

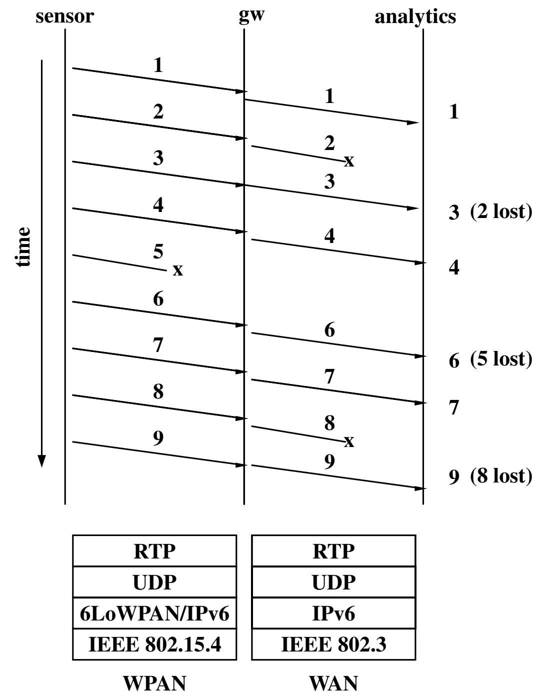


Fig. 8 RTP

UDP based on an end-to-end basis, there are no retransmissions and, therefore, application layer packet loss is entirely due to the effect of the different network packet losses.

3.5 Analysis

Both CoAP and MQTT-SN session scenarios rely on TCP for WAN traffic transport. The effect of retransmissions is minimized by disabling the Nagle algorithm on the TCP stacks. Moreover, because the playout buffer drops packets that arrive too late, it can be seen that even when replacing TCP with more efficient quick UDP Internet connections (QUIC) transport, the application layer packet loss probabilities remain the same as those presented in Sects. 3.2 and 3.3.

Based on the application layer packet loss probabilities of MQTT-SN, CoAP, and RTP given by Eqs. 2–5, respectively, it can be seen that both CoAP/HTTP and MQTT-SN exhibit performances that are inferior to those of end-to-end CoAP and RTP. Moreover, the analytical models presented in this paper show that both, end-to-end CoAP and RTP, behave similarly in regards to application layer packet loss under the same circumstances. Both RTP and end-to-end CoAP require neither session nor transport layer transformations and only rely on gateways to provide adaptation between the IEEE 802.15.4 and IEEE 802.3. Of all, however, CoAP/HTTP shows the worst performance as it relies on mapping CoAP messages into HTTP traffic for transport over the core infrastructure. Because HTTP does not support observation, each sensor read-out must be requested by the application; this increases the likelihood of missing packets at the application layer. In addition, latency is also increased due to the additional processing performed at the HTTP proxy in accordance with the IETF (constrained RESTful environments (CoRE) workgroup. Note that this poor performance of the CoAP/HTTP combo is slightly inferior to that of MQTT-SN. In either case, the theoretical analysis leads to preferring RTP and end-to-end CoAP over MQTT-SN and CoAP/HTTP in that order.

Next section presents an experimental framework to validate the theoretical analysis; specifically, values of application layer packet loss, throughput, power consumption, and speech quality scores are explored.

4 Experimental framework

Figure 9 shows the experimental framework to estimate application layer packet loss, throughput, power consumption, and speech quality scores as a function of network layer impairments, α and p for WPAN and $P_{\text{wan}}^{\text{loss}}$ for WAN. K sensors transmit 20-ms frames of speech at a 50 packet-per-second rate. Under MQTT-SN and RTP, this is done automatically, while under end-to-end CoAP, this is triggered by an individual CoAP observation GET request. The CoAP/HTTP case requires the application to continuously issue HTTP GET requests in order for sensors to transmit data.

VPS+ is used to emulate this scenario [22] configured to introduce network packet loss in accordance with the channel model presented in Sect. 3.1. For a fixed set of impairments and each of the scenarios shown in Figs. 5, 6, 7, and 8, a single test is performed by averaging for all $K = 20$ sensors the values of packet loss, throughput, power consumption, and quality scores obtained for the transmission of a well-known sequence of speech. Specifically, the speech reference consists of a 60-s sequence obtained from concatenating multiple instances of ITU P.501 Annex B sample speech files [23].

In this experimental framework, the following speech codecs are considered (1) ITU-T Rec. G.711 μ-law, (2) ITU-T Rec. G.729A, (3) AMR-NB, and (4) EVS. G.711 is a narrowband (8 KHz sampling rate) high bitrate codec (HBC) that preserves the speech waveform through non-linear compansion at the cost of increased transmission rate [24]. G.729A is a narrowband parametric low bitrate codec (LBR) that operates at 8 Kbps and relies on linear prediction and prediction error encoding [25]. AMR-NB is also a narrowband LBR codec that provides a wide range of compression rates at different quality levels [26]. EVS provides speech compression at different rates and sampling rates but configured as narrowband in this case [27]. These codecs are configured negotiated to operate at the following rates: G.711 at 64 Kbps, G.729A at 8 Kbps, AMR-NB at 7.95 Kbps, and EVS at 7.2 Kbps.

Speech quality scores are obtained by means of ITU-T Rec. P.863 [28] that defines a speech quality estimation algorithm known as perceptual objective listening quality assessment (POLQA). This mechanism relies on a perceptual model that it is used to extract key parameters that represent both, the reference and the speech sequence under analysis, and that are compared to obtain a quality score between 1 (bad) and 5 (excellent). Mean opinion score listening quality objective (MOS-LQO) is the name of the quality scores obtained by subjecting POLQA to speech samples under test.

Packet loss is measured as the percentage of transmitted packets that are actually processed by analytics. Throughput, in units of Kbps, is the rate of encoded speech that arrives to analytics. Power is measured in units of decibel-milliwatts (dBm) to reflect the efficiency of the IEEE 802.15.4 physical layer in regards to battery consumption in the different scenarios under consideration. Note, that in a similar fashion to quality scores, both throughput and power depend on the

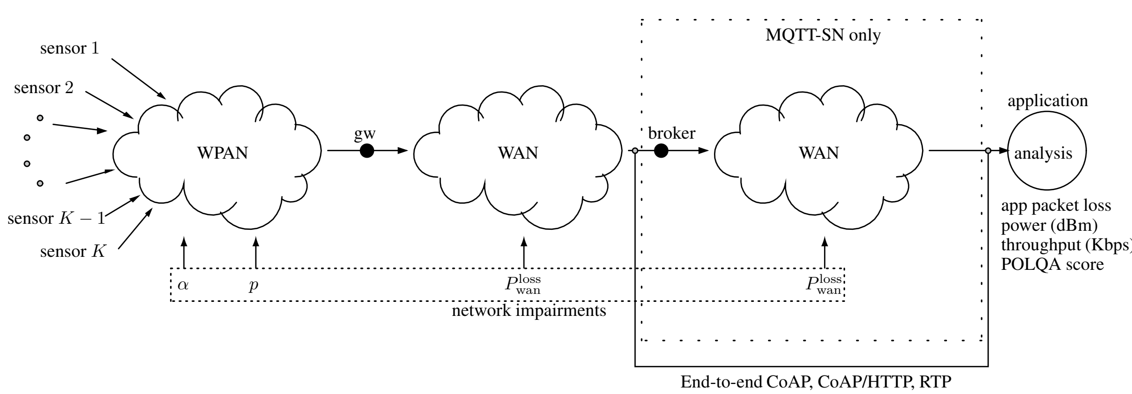


Fig. 9 Experimental framework

codec under analysis, because each codec encodes speech into frames of different fixed sizes.

5 Comparative results and analysis

Figure 10 shows experimental and theoretical application layer packet loss probabilities of the aforementioned session layer protocols as a function of network layer packet loss for two representative values of $P_{\text{wan}}^{\text{loss}}$. Note that analytical values are obtained from Eqs. 5 through 8, while experimental values result from the emulation scenario presented in Sect. 4. From both plots, it can be seen that experimental values are, on average, within 11.87% of those values obtained from the theoretical analysis. Moreover, the plots show that, as expected, both RTP and CoAP exhibit the lowest application layer packet loss due to their low latency transport. On the other hand, MQTT and, to a minor degree, CoAP/HTTP show high packet loss introduced by the nature of the stream-based transport.

Figures 11, 12, and 13 show the experimental values associated with the MOS-LQO coefficients, the throughput, and the power consumption, respectively, as a function of network layer impairments when $P_{\text{wan}}^{\text{loss}} = 0.01$. Each case takes into account the performance of all four speech codecs under consideration; that is; ITU-T Rec. G.711 μ-law, ITU-T Rec. G.729A, AMR-NB, and EVS. As shown in Fig. 11, POLQA scores are quite dependent on the codec under consideration where high-throughput codecs like ITU-T Rec. G.711 μ-law exhibit better performance than low-throughput codecs like EVS. Average scores are 4.16, 3.61, 3.46, and 3.66 for ITU-T Rec. G.711 μ-law, ITU-T Rec. G.729A, AMR-NB, and EVS, respectively. The effect of the session layer protocol is not as important mostly due to the fact that these codecs include packet loss concealment that tend to minimize the effect of bursts of lost packets.

Throughput, as indicated in Fig. 12, lowers with increased loss as fewer packets make it to destination. Average throughput values are 50.96 Kbps, 6.41 Kbps, 6.39 Kbps, and 5.81 Kbps for ITU-T Rec. G.711 μ-law, ITU-T Rec. G.729A, AMR-NB, and EVS respectively. In this case, however, the effect of the session layer protocol is more noticeable. Traffic transported via MQTT and CoAP/HTTP exhibits slightly lower throughput than traffic transported by means of CoAP and RTP.

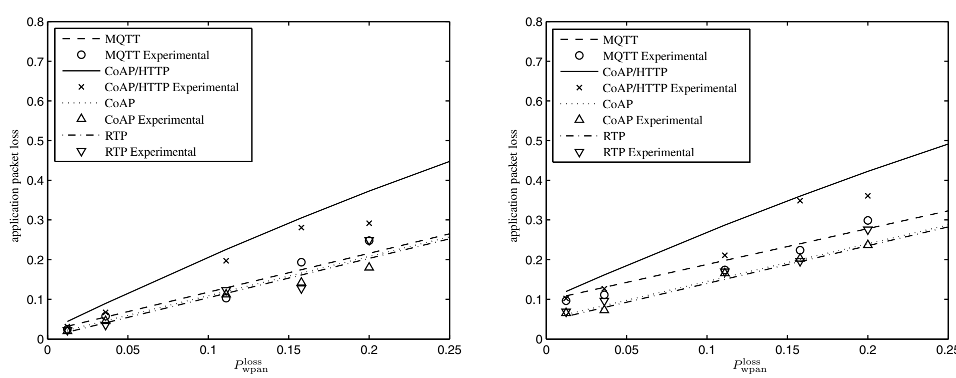
Power consumption, shown in Fig. 13, is pretty constant for same codec type regardless of packet loss and session layer protocol. This has to do with the fact that all scenarios rely on power measured at an IEEE 802.15.4 radio transmitter generating traffic that is always 6LoWPAN/UDP-based. Average power consumption is 7.67 dBm, 1.91 dBm, 1.96 dBm, and 1.81 dBm for ITU-T Rec. G.711 μ-law, ITU-T Rec. G.729A, AMR-NB, and EVS, respectively.

6 Conclusions

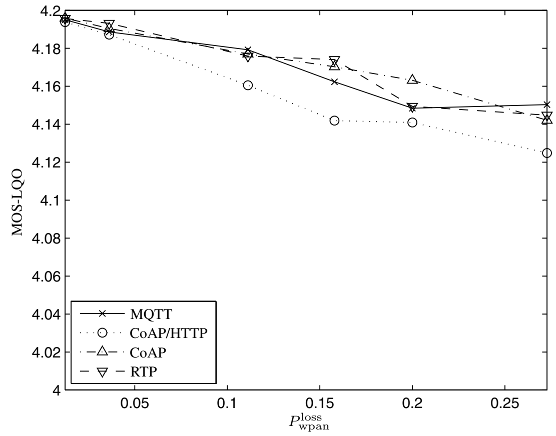
A few conclusions can be drawn from the observations presented in the previous sections: (1) RTP and end-to-end CoAP exhibit the lowest application layer packet loss of all analyzed session layer protocols, (2) RTP relies on other more complex session signaling protocols like SIP that are not always available or even implemented in light weight embedded platforms, (3) packet loss concealment provided by modern speech codecs minimizes the effects of the packet loss introduced by the different session layer protocols, (4) both throughput and power consumption are highly dependent on the speech codec under consideration, and (5) speech quality, as measured by POLQA scores, decreases with network packet loss.

From these facts and from the elements presented in this analysis, it can be seen that end-to-end CoAP transporting EVS encoded media present the best solution for efficient speech transmission. In this context, the loss of speech quality due to network packet loss can serve as topic of future study, where through CoAP-specific FEC, POLQA scores can be further improved by addressing those cases not covered by individual codec packet concealment mechanisms.

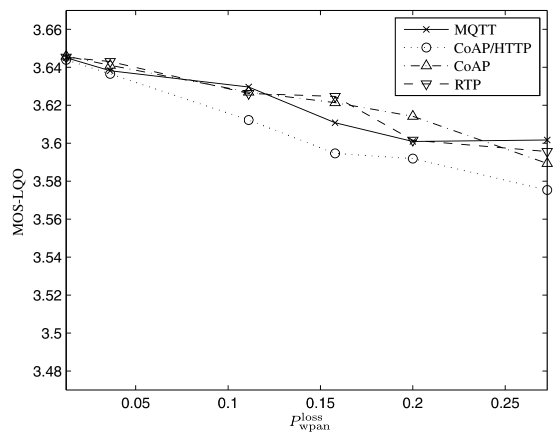
Fig. 10 Application packet loss probability



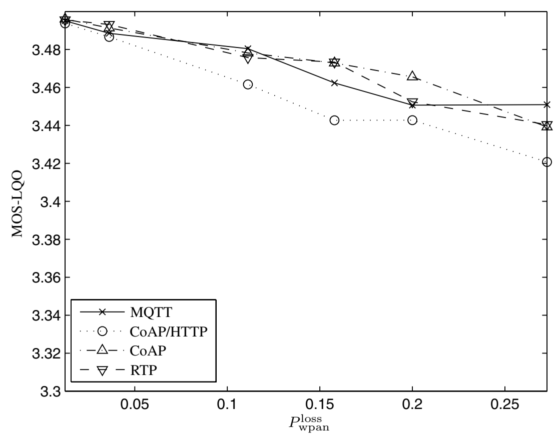
(a) $P_{\text{wan}}^{\text{loss}} = 0.01$ **(b)** $P_{\text{wan}}^{\text{loss}} = 0.05$



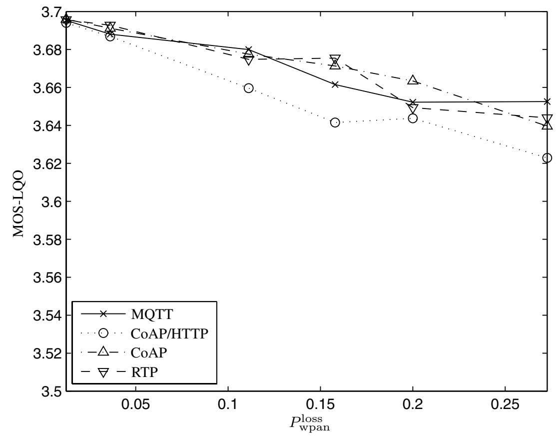
(a) ITU-T Rec. G.711 μ-law



(b) ITU-T Rec. G.729

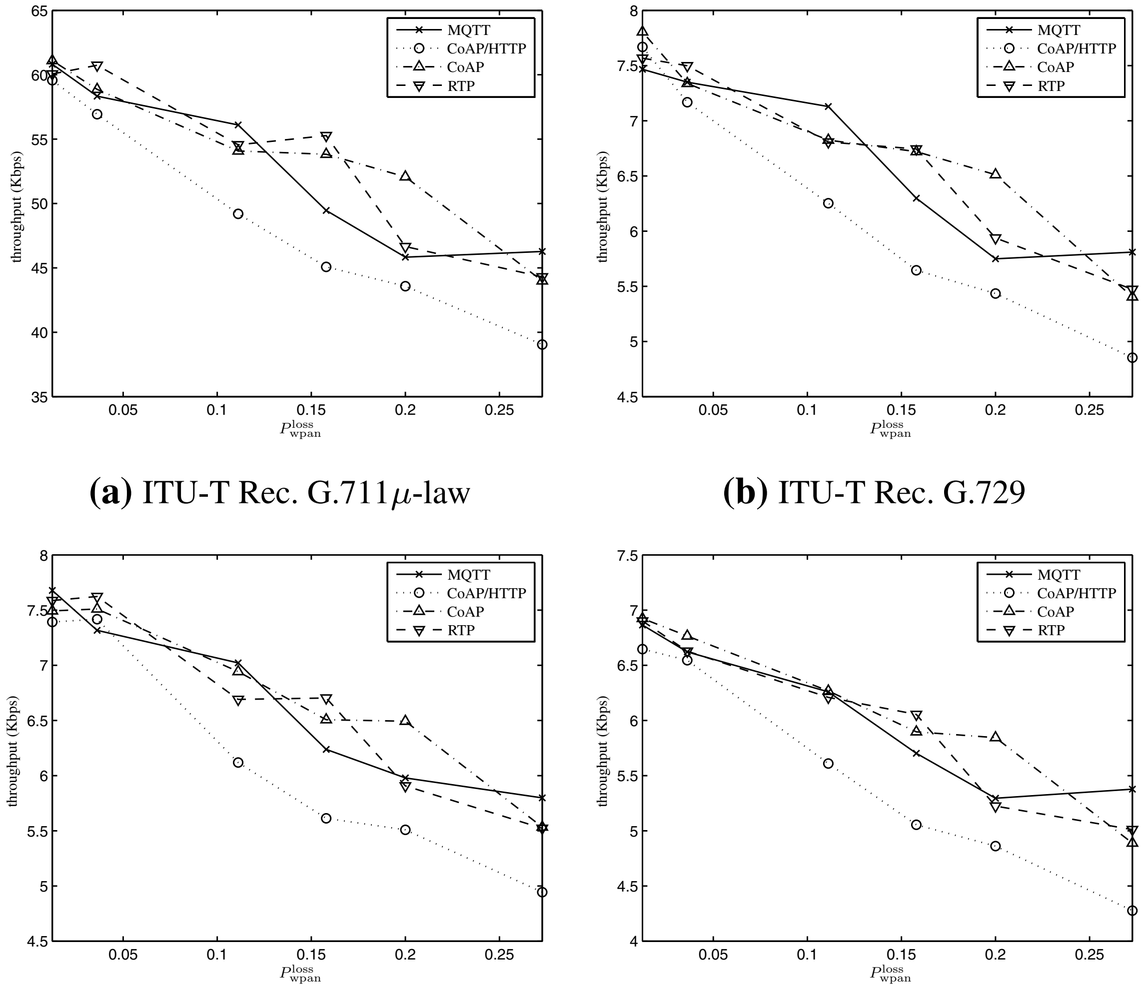


(c) AMR-NB



(d) EVS

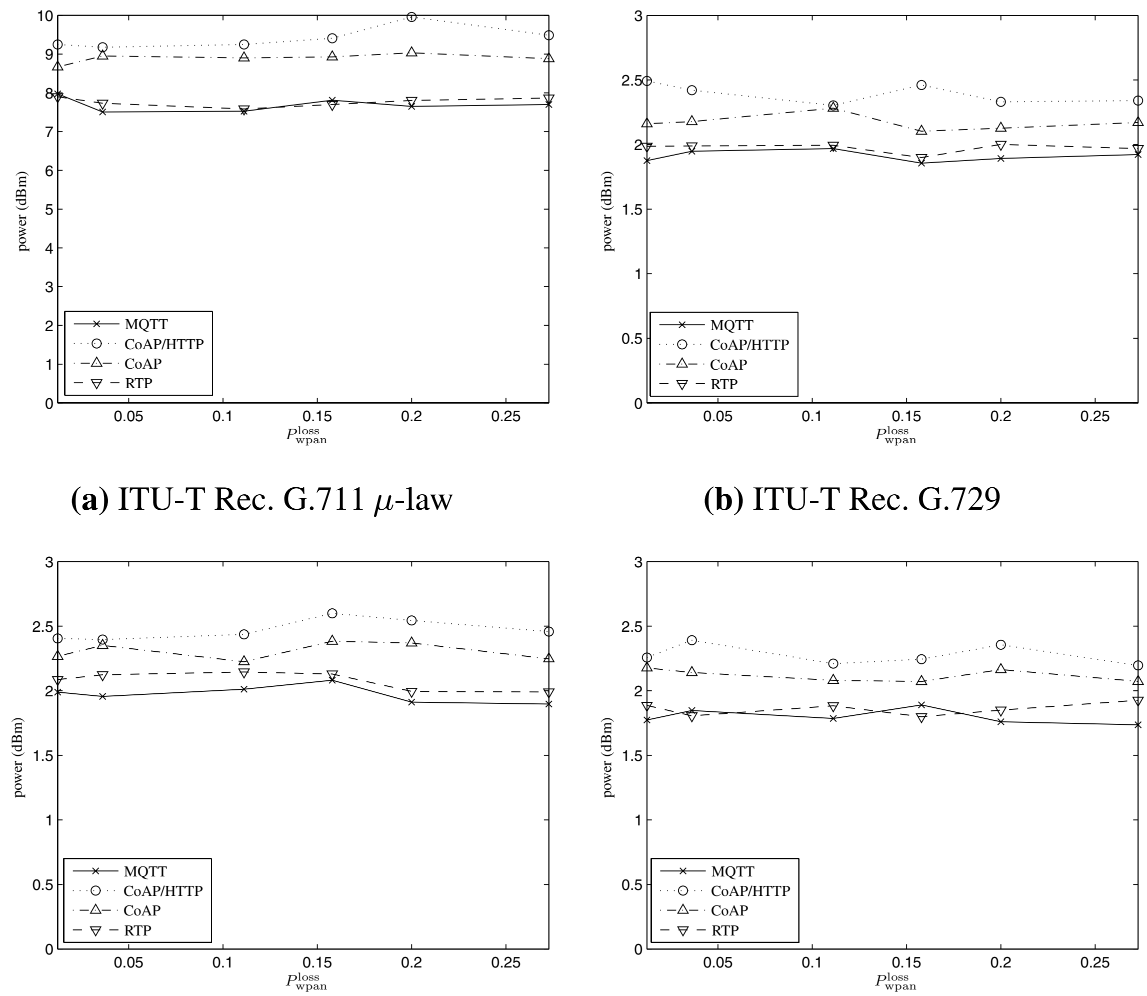
Fig. 11 MOS-LQO vs $P_{\mathrm{wpan}}^{\mathrm{loss}}$ ($P_{\mathrm{wan}}^{\mathrm{loss}} = 0.01$)



(a) ITU-T Rec. G.711μ-law **(b)** ITU-T Rec. G.729

(c) AMR-NB **(d)** EVS

Fig. 12 Throughput vs $P_{\mathrm{wpan}}^{\mathrm{loss}}$ ($P_{\mathrm{wan}}^{\mathrm{loss}} = 0.01$)



(a) ITU-T Rec. G.711 μ-law

(b) ITU-T Rec. G.729

(c) AMR-NB

(d) EVS

Fig. 13 Power vs $P_{\text{wpan}}^{\text{loss}}$ ($P_{\text{wan}}^{\text{loss}} = 0.01$)

References

1. Lee, W.C.Y.: Estimate of channel capacity in Raleigh fading environment. In: 38th IEEE Vehicular Technology Conference, pp. 582–584 (1988). https://doi.org/10.1109/VETEC.1988.195421
2. Herrero, R.: Integrating HEC with circuit breakers and multipath RTP to improve RTC media quality. Telecommun. Syst. **64**(1), 211–221 (2017). https://doi.org/10.1007/s11235-016-0169-z
3. Herrero, R., St-Pierre, C.: Dynamic forward error correction in wireless real-time Internet of things networks. IET Netw. **6**(6), 218–223 (2017). https://doi.org/10.1049/iet-net.2017.0110
4. Said, O., Albagory, Y., Nofal, M., Raddady, F.A.: IoT-RTP and IoT-RTCP: adaptive protocols for multimedia transmission over Internet of things environments. IEEE Access **5**, 16757–16773 (2017). https://doi.org/10.1109/ACCESS.2017.2726902
5. Bormann, C., Hartke, K., Shelby, Z.: The constrained application protocol (CoAP). RFC 7252 (2015). https://doi.org/10.17487/rfc7252. https://www.rfc-editor.org/rfc/rfc7252.txt. Accessed June 2014
6. Montenegro, G., Hui, J., Culler, D., Kushalnagar, N.: Transmission of IPv6 Packets over IEEE 802.15.4 Networks. RFC 4944 (2007). https://doi.org/10.17487/RFC4944. https://www.rfc-editor.org/rfc/rfc4944.txt. Accessed Sept 2007
7. Bormann, C., Shelby, Z.: Block-wise transfers in the constrained application protocol (CoAP). RFC 7959 (2016). https://doi.org/10.17487/RFC7959. https://www.rfc-editor.org/rfc/rfc7959.txt
8. Asghar, M.H., Mohammadzadeh, N.: Design and simulation of energy efficiency in node based on MQTT protocol in Internet of things. In: 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), pp. 1413–1417 (2015). https://doi.org/10.1109/ICGCIoT.2015.7380689
9. Pereira, E.G., Pereira, R.: Video encoding and streaming mechanisms in IoT low power networks. In: 2015 3rd International Conference on Future Internet of Things and Cloud, pp. 357–362 (2015). https://doi.org/10.1109/FiCloud.2015.88
10. Kim, H.: Low power routing and channel allocation method of wireless video sensor networks for Internet of things (IoT). In: 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 446–451 (2014). https://doi.org/10.1109/WF-IoT.2014.6803208

11. Plageras, A.P., Psannis, K.E., Ishibashi, Y., Kim, B.G.: IoT-based surveillance system for ubiquitous healthcare. In: IECON 2016—42nd Annual Conference of the IEEE Industrial Electronics Society, pp. 6226–6230 (2016). https://doi.org/10.1109/IECON.2016.7793281
12. Herrero, R.: Dynamic CoAP mode control in real time wireless IoT networks. IEEE Internet Things J. (2018). https://doi.org/10.1109/JIOT.2018.2857701
13. Choi, G., Kim, D., Yeom, I.: Efficient streaming over COAP. In: 2016 International Conference on Information Networking (ICOIN), pp. 476–478 (2016). https://doi.org/10.1109/ICOIN.2016.7427163
14. Herrero, R.: 6Lowpan fragmentation in the context of IoT based media real time communication. Internet Technol. Lett. **3**(2), e144 (2020). https://doi.org/10.1002/itl2.144
15. Lai, C., Hwang, Y.: The voice controlled Internet of things system. In: 2018 7th International Symposium on Next Generation Electronics (ISNE), pp. 1–2 (2018). https://doi.org/10.1109/ISNE.2018.8394640
16. Belli, L., Cirani, S., Davoli, L., Ferrari, G., Melegari, L., Montón, M., Picone, M.: A scalable big stream cloud architecture for the Internet of things. Int. J. Syst. Serv. Orient. Eng. **5**(4), 26–53 (2015). https://doi.org/10.4018/IJSSOE.2015100102
17. Hohlfeld, O., Geib, R., Hasslinger, G.: Packet loss in real-time services: Markovian models generating QOE impairments. In: 2008 16th Interntional Workshop on Quality of Service, pp. 239–248 (2008). https://doi.org/10.1109/IWQOS.2008.33
18. Gilbert, E.N.: Capacity of a burst-noise channel. Bell Syst. Tech. J. **39**(5), 1253–1265 (1960). https://doi.org/10.1002/j.1538-7305.1960.tb03959.x
19. Elliott, E.O.: Estimates of error rates for codes on burst-noise channels. Bell Syst. Tech. J. **42**(5), 1977–1997 (1963). https://doi.org/10.1002/j.1538-7305.1963.tb00955.x
20. Nagle, J.: RFC 896: Congestion control in IP/TCP internetworks (1984). ftp://www.ftp.internic.net/rfc/rfc896.txt,ftp://ftp.math.utah.edu/pub/rfc/rfc896.txt. Accessed Jan 1984
21. Hartke, K.: Observing resources in the constrained application protocol (CoAP). RFC 7641 (2015). https://doi.org/10.17487/rfc7641. https://www.rfc-editor.org/rfc/rfc7641.txt. Accessed Sept 2015
22. VPS+: Vps+ protocol emulator. https://www.l7tr.com. Accessed July 2020
23. ITU-T Recommendation P.501: Test Signals for Use in Telephonometry. Telephonometry, International Telecommunication Union, Geneva (2000)
24. ITU-T: G.711 : Pulse code modulation (PCM) of voice frequencies. Technical report G.711, International Telecommunication Union, Geneva (2006)
25. Salami, R., Laflamme, C., Bessette, B., Adoul, J.: Description of ITU-T recommendation g.729 annex a: reduced complexity 8 kbit/s cs-acelp codec. In: Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97), vol. 2, ICASSP '97, p. 775. IEEE Computer Society, Washington, DC, USA (1997)
26. 3GPP: Ts 26.071: Mandatory speech codec speech processing functions; AMR speech codec; general description. Technical report TS 26.071, 3rd Generation Partnership Project (2008)
27. Ramo, A., Toukomaa, H.: Subjective quality evaluation of the 3gpp evs codec. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5157–5161 (2015). https://doi.org/10.1109/ICASSP.2015.7178954
28. ITU-T Recommendation P.863: Technical report, International Telecommunication Union, Geneva, Switzerland