A Delay Monitoring Method for Up-Link Flows in IEEE 802.11e EDCA Networks*

Filippo Cacace
Università Campus Bio-Medico di Roma
via Álvaro del Portillo 21
00128, Roma, ITALY
f.cacace@unicampus.it

Luca Vollero
Università Campus Bio-Medico di Roma
via Álvaro del Portillo 21
00128, Roma, ITALY
l.vollero@unicampus.it

ABSTRACT

We present an analytical framework to model the sending queue of an IEEE 802.11 EDCA wireless station that uses the $TXOP_{limit}$ MAC parameter to send more than one packet in each transmission opportunity. An application of this model to the estimate of packet delay from the number of packets received at the AP in each transmission is presented. This method is well suited to be deployed in an infrastructured WLAN but can be used for any wireless communication based on the EDCA access mechanism. Its main advantage is that it requires neither modifications to the MAC protocol nor cooperation of the wireless STAs. The approach suitability to predict the delay of real time VoIP calls is validated through simulation.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network Monitoring*

General Terms

Communication Networks

Keywords

Wireless LANs, Performance Monitoring, EDCA.

1. INTRODUCTION

Real time and multimedia communications with Quality of Service (QoS) support are increasingly important in wireless networks of any nature, due to the growing demand of services like VoIP, streaming and videoconferencing and to the limited capacity of the network. They are especially challenging for WLANs based on the widespread IEEE 802.11 protocol due to the distributed nature of the channel access mechanism of this standard.

*SIMUTools/QoSim 2009, March 2-6, Rome, Italy, Copyright 2009 ICST, ISBN 978-963-9799-45-5

Real-time monitoring functionalities are very useful for the QoS support in a wireless access network. They allow the dynamic tracking of the relevant QoS parameters, such as loss rate, throughput, delay and jitter of traffic flows with quality guarantees. This information can be used in a variety of ways to take decisions about flow admission [4–11], fairness management [7, 12–16] and to implement methods for the QoS support based on the adaptation of MAC parameters [12, 14, 19–22] or other measurement based methods [4, 6, 7, 11, 17, 18]. The current trend toward centralized WLANs [2] with one or more central units in charge of managing security, configuration and QoS issues is an additional reason to introduce in the WLANs monitoring functions in order to provide to the control units the information needed to manage the network.

In this paper we focus on monitoring the delay of the wireless link. Delay is in itself a relevant QoS parameter for real time traffic. Moreover, delay is related to other parameters of interest, such as the congestion state of the wireless channel, packet jitter, losses at the sending buffer as shown, for instance, in [3]. For this reason, algorithms that provide reliable estimates of packet delay are of great interest for the QoS support. Throughout this paper, the expression *sending delay* is used to define the time interval between the arrival of one packet at the sending queue and the acknowledgment of its correct reception. It therefore encompasses queuing and transmission delays.

In an infrastructered WLAN it is easy to deploy methods to monitor this delay for down-link traffic, that is, from the Access Point (AP) to the mobile stations (STAs). It is sufficent to design an AP that directly monitors this time interval, as suggested in [10]. Another approach is to monitor queue lengths at the AP to estimate, from the pattern of the incoming traffic, the sending delay [22].

The case of up-link delay is more difficult and, to our knowledge, there are no previous proposals of methods for estimating its value without modifying the MAC protocol and/or the software running on the STAs.

The approach that we present in this paper for the estimate of up-link delay is based on the Enhanced Distributed Channel Access (EDCA) method defined in the IEEE 802.11 standard [1]. EDCA is the enhanced contention channel access mechanism of IEEE 802.11 and it was designed to introduce QoS support, unavailable with the Distributed Control Function (DCF) used in the previous 802.11b/g WLANs. EDCA is based on a slotted and highly parametric CSMA/CA protocol. It has been adopted by the WiFi Alliance and it is already commercially available. EDCA defines the concept of Access Category (AC): in order to transmit data, every STA may use up to four ACs, each AC implementing a slotted CSMA/CA algorithm with its own parameter and competing to obtain transmission opportunities (TXOPs). The four ACs within a station represent four priority levels for data transmission. The standard names these levels as: background (BK), best effort (BE), video (VI) and voice (VO). In infrastructure system, APs announce the configuration of ACs in selected beacon frames. The standard assumes that the AP may use a set of parameters different than that it advertises to STAs.

To differentiate the behavior of the ACs, four parameters can be set: (i) the Arbitration Interframe Space (*AIFS*), which determines the interval during which the medium must be sensed idle before an AC is allowed to transmit, (ii) the minimum and (iii) the maximum Contention Windows (CW_{min} and the CW_{max}), which determine the average duration of the backoff process, and, finally, (iv) the Transmission Opportunity limit ($TXOP_{limit}$), which specifies the maximum channel occupancy time in the case of a successful access (see below).

Intuitively, a STA will send more than one packet in a TXOP when its sending queue contains more than one packet. The average number of packets per TXOP can be used to estimate the average queue length and sending delay of a given STA. This is the approach that we develop in this paper.

2. TRANSMISSION OPPORTUNITIES

The introduction of transmission opportunities of multiple consecutive packets, also called packet or frame bursting, is one important feature of the EDCA method of IEEE 802.11 standard. It allows transmitting multiple consecutive frames within a limited time duration called TXOP$_{limit}$ without recurring to the contention process for each packet. Hence, the TXOP$_{limit}$ parameter does not affect the channel access mechanism, rather it differentiates the ACs with respect to the channel holding time once that the access has been granted.

Even though this mechanisms has received so far little attention in the context of the EDCA analytical performance models it is well known that it has a positive impact on the overall channel efficiency, since its use reduces contention and therefore collisions. This improves the ratio between time spent in successful transmissions and collisions. Conceptually, the TXOP parameter is also useful to mitigate fairness problems experienced by stations that receive less opportunities to access the channel, since an increase of their TXOP parameter allows longer transmissions, thus compensating their scarcity.

The TXOP mechanism dictates that a station to which has been granted a transmission opportunity on a certain AC can hold the channel to transmit packets of the appropriate AC without contention until one of the following conditions occur:

- the TXOP$_{limit}$ parameter, which is a time limit, expires;
- there is nothing more to transmit, that is, the queue of the AC becomes empty;
- the ACK of a given frame is not received.

In all three cases the channel is released. Notice that the packets sent in a transmission opportunity can have distinct receivers. After each sending within a transmission opportunity the sender waits for the ACK from the receiver, like in the usual EDCA mechanism, before continuing to send packets. Hence, in ideal channel conditions, the transmission opportunity terminates only when the TXOP$_{limit}$ expires or when the sending queue becomes empty, since collision cannot occurr after the beginning of the transmission. However, when physical errors occurr, the transmission is prematurely truncated.

3. MONITORING UP-LINK DELAY

3.1 Traffic at constant packet rate

The possibility of sending more than one packet per transmission can be used to infer, at the AP, the delay of up-link packets. Given a flow at constant packet rate λ, it is intuitive that if the STA transmits i packets in the same TXOP, with i smaller than the maximum number of packet that can be sent in the same TXOP, this conveys information about the state of the sending queue. We are thus able to build a model to measure at the AP the up-link delay of the STA starting from the number of packets contained in the same TXOP.

To this aim, we consider the queueing system of the STAs. The states of this system correspond to the number of packets in the queue. We make three assumptions to model the system:

(H1) Packets that arrive at the sending queue belong to a single flow with constant packet rate λ.

(H2) The service time is distributed exponentially with rate μ, and each service event sends $\min(N_t, N_q)$ packets, where N_t is the maximum number of packets that a transmission may contain and N_q is the number of packets in queue.

(H3) The distribution probability of the states at the arrival and departure events are geometrical with parameters q_a and q_d (resp.)

We defer to the end of this section our remarks on the relevance of hypothesis (H1), for the moment we stress that (H1) is satisfied by most VoIP flows. Notice that the value of N_t depends on the duration of the transmission (the $TXOP_{limit}$ parameter), and on the packet size (that includes MAC and physical headers). In the case of constant packet rate, N_t can be easily determined.

Using (H3) the probability of having n frames in the queue just before a new frame arrives, $P_a(n) = P\left(\left\{N_q^a = n\right\}\right)$, and the probability of having n frames in the queue just before serving a TXOP, $P_d(n) = P\left(\left\{N_q^d = n\right\}\right)$ are

$$P_a(n) = \frac{1-q_a}{1-q_a^{N+1}} q_a^n, \quad n \in \{0, 1, \ldots, N\}, \tag{1}$$

and

$$P_d(n) = \frac{1-q_d}{1-q_d^{N}} q_d^{n-1}, \quad n \in \{1, \ldots, N\}, \tag{2}$$

where N is the maximum queue length.

Eq. (2) can be used to express the average number of frames sent in a TXOP as:

$$\bar{n} = \sum_{n=1}^{N} P_d(n) \min(n,\, N_t) = \mathrm{f}_{N,\, N_t}(q_d). \tag{3}$$

It is easy to show that $f_{N,N_t}(q_d)$ is an increasing function of q_d in the range [0, 1] and, hence, it can be used to compute back q_d from $\bar{n}$. Hence, $f^{-1}_{N,N_t}(\bar{n})$ can be used to evaluate the probability distribution of queue states.

The value of $\bar{n}$ can be dynamically obtained by averaging the instantaneous value of the number of frames received in each TXOP from the station of interest. This operation can be implemented with an auto regressive filter as

$$\bar{n}(k) = \beta\,\bar{n}(k-1) + (1-\beta)\,n(k)\,, \tag{4}$$

where $\bar{n}(k)$ is the estimate of the average number of frames sent in each TXOP when the k-th TXOP is received, $n(k)$ is the number of frames in the k-th TXOP and $\beta \in [0, 1]$ is a memory factor.

In order to estimate the service rate of the queue we focus on the queue evolution on arrival events. The sending queue at the STA can be modeled considering that if no TXOP is served between two arrival events this causes a transition

$$\left\{N_q^a = i\right\} \to \left\{N_q^a = i+1\right\},$$

whereas one sending event causes the transition

$$\left\{N_q^a = i\right\} \to \left\{N_q^a = i - N_t + 1\right\},$$

when $i \geq N_t - 1$, and

$$\left\{N_q^a = i\right\} \to \left\{N_q^a = 0\right\}$$

otherwise. In particular, state $\left\{N_q^a = N\right\}$ can be reached only by states $\left\{N_q^a = N\right\}$ and $\left\{N_q^a = N-1\right\}$ when no TXOP is served. Hence we can write the following equation:

$$P_a(N) = \left[P_a(N-1) + P_a(N)\right] P_{\overline{TXOP}}, \tag{5}$$

where $P_{\overline{TXOP}}$ is the probability of not serving any TXOP during an inter-arrival time:

$$P_{\overline{TXOP}} = \int_0^{+\infty} e^{-\mu t} f_a(t)dt, \tag{6}$$

where $f_a(t)$ is the pdf of inter-arrival times. Under hypothesis (H1) packet rate is constant, that is, $f_a(t) = \delta(t - 1/\lambda)$, thus

$$P_{\overline{TXOP}} = e^{-\mu/\lambda}. \tag{7}$$

From equations (5),(7) we can derive an expression of the service rate μ as a function of q_a:

$$\mu = \lambda \log\left(1 + \frac{1}{q_a}\right). \tag{8}$$

In order to compute μ we need a law relating q_d and q_a. Although extensive simulation analysis showed us that $q_a \approx q_d$, that approximation does not work properly in the computation of the service rate when the network starts becoming congested. Indeed, the resulting μ would not consider the time spent in long transmission opportunities for sending all the frames waiting in the AC queue. To overcome that problem, we suggest the following heuristic expression of the service rate:

$$\mu = \lambda \log\left(1 + \frac{1}{\bar{n} q_d}\right). \tag{9}$$

To obtain the delay distribution, we exploit the batched nature of the queue. Let d be the delay random variable. We can express the probability of having a delay not greater than D as

$$P\left(\{d \leq D\}\right) = \sum_{k=0}^{N-1} \frac{P_a(k)}{1 - P_a(N)} P\left(\left\{T_{\left\lceil \frac{k+1}{N_t} \right\rceil} \leq D\right\}\right) \tag{10}$$

where T_k is a random variable obtained summing k independent exponentially distributed random variables with parameter μ. T_k accounts for all the TXOPs that the station has to serve in order to deliver a frame currently arriving. The probability of having T_k not greater than D is

$$P\left(\{T_k \leq D\}\right) = 1 - e^{-\mu D} \sum_{n=0}^{k-1} \frac{(\mu D)^n}{n!}. \tag{11}$$

Hence the global delay distribution follows the following expression:

$$P\left(\{d \leq D\}\right) = \\ = \sum_{k=0}^{N-1} \frac{P_a(k)}{1 - P_a(N)} \left(1 - e^{-\mu D} \sum_{n=0}^{\left\lceil \frac{k+1}{N_t} \right\rceil - 1} \frac{(\mu D)^n}{n!}\right) \tag{12}$$

As a final remark on the delay model, it is useful to highlight that the hypothesis (H1), that requires that the traffic flow is at constant packet rate, is only used in equation (6) to relate the service rate $P_{\overline{TXOP}}$ with μ and the pattern of traffic generation. The model can be extended to other traffic patterns, and this leads to different expressions in (7). However, in this paper we only report simulation results about flows at constant packet rate.

3.2 Physical errors

In real cases the wireless channel is prone to errors. A monitoring approach for QoS support must necessarily deal with situations in which some of the terminals may experience packet loss due to physical errors, and the estimates provided by the approach should be robust with respect to these situations for practical applicability

The effect of physical errors when using EDCA at the MAC layer may be different than DCF, due to the presence of transmissions with more than one packet. Specifically, whereas both collisions and physical errors affect the first packet in a transmission, only physical error can affect the transmission of subsequent packets, since the MAC protocol dictates that the other terminals cannot interfere with the one that holds the transmission opportunity. This implies that transmission errors do not only increase the number of retransmission: they also decrease the number of packets that are sent in a transmission opportunity, since after a missing ACK the transmission is aborted even if there are other packets to send. It is therefore reasonable that a delay estimate based on the number of packets in the transmission is affected by the presence of physical errors. As we show in the simulation results, the estimate obtained with the approach described so far is reliable only with a percentage of errors on packet (PER) under 5%.

Some modifications can be introduced to deal with these situations. The PER rate per terminal can be monitored at the AP by keeping statistics related to physical errors. This estimate can be used to correct the value of $\bar{n}$ in the computation of the delay. However, it must also be considered that in real scenarios the PER is not likely to be constant, since

Table 1: default MAC parameters

Class	Parameter	Value
VO	CW_{min}	16
VO	CW_{max}	32
VO	$AIFS$	2
VO	TXOP$_{limit}$	3264μs
VI	CW_{min}	32
VI	CW_{max}	64
VI	$AIFS$	2
VI	TXOP$_{limit}$	6016μs

the errors are frequently caused by transient situations, like interference or station mobility, that can make this correction less meaningful. We do not investigate further on this issue in this paper, leaving it as a subject of future work.

4. IMPLEMENTATION

The implementation of above monitoring techniques is based on the ability of real-time measuring the number of MAC frames belonging to the same TXOP. The major problem with implementing such a mechanism is represented by the correct time-stamping of frames, which requires a very high resolution clock. Indeed, in order to assess whether two frames belong to the same TXOP we require a time-stamp resolution of about 25μs, being 50μs the minimum delay separating two frames belonging to different TXOPs. Having such a high resolution time-stamping, we can say that two frames belong to the same TXOP, when the tail of first packet and the head of the second one have a temporal gap lower than 30μs.

This high resolution time-stamping is usually unavailable in off-the-shelf systems equipped with standard OSs and common capturing software (e.g. laptops with Linux O.S. and using Ethereal or Kismet). This is because a high resolution time-stamping is usually not required for the day-by-day use of the systems. However, Linux kernel and the MADWiFi driver, that we used to evaluate techniques implementability, can be configured in order to provide such a high precision time-stamping of frames. The high-resolution time-stampig can be enabled in the system and, in order to increase further its precision, the NIC driver can be configured in order to set directly time-stamps upon frames reception. Moreover, the same driver can be extended in order to provide real-time statistics on the average number of frames received in each TXOP for every ACs. This can be realized by the introduction of simple moving average filter aiming at filtering out channel oscillations and capturing average performance. Experiments made in our lab proved that this solution works well with off-the-shelf hardware/software systems (We used an Intel laptop with Ubuntu 7.10 - linux kernel 2.6.15.26-386 and MADWiFi v 0.9.2.1 driver for WG511T Atheros cards), providing time-stamping and statistics with the required precision.

5. SIMULATION RESULTS

This section presents simulation results to validate the queue model for the delay estimate presented in the previous sections. To this sake, we have used ns2[1], with standard extensions for 802.11e[2], and IEEE 802.11b link parameters. When no otherwise specified, the IEEE 802.11e MAC parameters (CW_{min}, CW_{max}, $AIFS$ and $TXOP_{limit}$) are set at their default values for each access category. Specifically, for the VO and VI categories we used the set of values specified in Table 1. Section 5.1 report results about VoIP traffic only, with homogeneous and heterogeneous sources. Section 5.2 considers VoIP in presence of interfering BE traffic.

We have performed simulations to assess the impact of physical errors on VoIP delay estimate, reported in Section 5.3. We conclude in Section 5.4 with a scenario in which the number of active flows, and the consequent packet delay is increased, to validate the method of dynamic delay estimate presented in Section 4.

5.1 VoIP traffic

In these simulations we consider an infrastructured wireless cell with CBR VoIP traffic at different bit-rates. We have chosen 8 and 64 kbps (that correspond to G.729 and G.761 voice codecs), and 256 kbps (a representative bit-rate for CBR videoconferencing). UDP and RTP overheads are considered in these experiments. The traffic is sent on the VO IEEE 802.11e AC and it is symmetric. In all cases, packet rate is set to 50 packets per second. We are only interested in the behavior of up-link flows. For each codec we perform simulations increasing the number of active flows until the threshold of 3% loss rate is reached for up-link flows. Losses are due to packets dropped for repeated collisions rather than to buffer overflow. This is due to the fact that the contention window for standard IEEE 802.11e VO setting is quite small, 8-16 time-slots: this reduces the delay at the expenses of a larger collision probability.

We performed 10 simulations for each traffic configuration. The results reported in Fig. 1 plot the average delay and its estimate as a function of the number of active flows. We measured also the delay of the 90% percentile of the packets, to test whether the delay distribution behaves as expected. The 95% confidence interval in the plot reflects the variance of the delay across simulations (we first averaged the delay of all the stations in each simulation and then computed the average and confidence interval across simulations).

It can be observed from the plot that the up-link delay is limited and that there is a sudden transition to the saturated region when the admission limit is reached. In this transition delay fluctuates widely from one simulation to another, and this instability is highlighted from the large confidence interval. The model tracks well the simulation results: notice that at low congestion the delay estimate is bounded to a minimum, since all the transmissions contain just 1 packet. In this case our method is not able to estimate precisely the delay, whose value is however under 5 ms.

To test the method with larger delays we changed the size of the congestion window parameter for the VO AC. In Fig. 2 we plot average delays and 90% percentile for the same traffic configurations as before when the contention window in the range 32-64. In this case the admission limit is slightly larger and the transition to the congested situation smoother, but also the delays are larger. We can notice that the estimate of the delay tracks well the transition to the saturated region.

In Fig.s 3 and 4 we consider a scenario with heterogeneous

[1][Online]. Available at `http://www.isi.edu/nsnam/ns/`.

[2][Online]. Available at `http://sourceforge.net/projects/ieee80211e-ns2`.

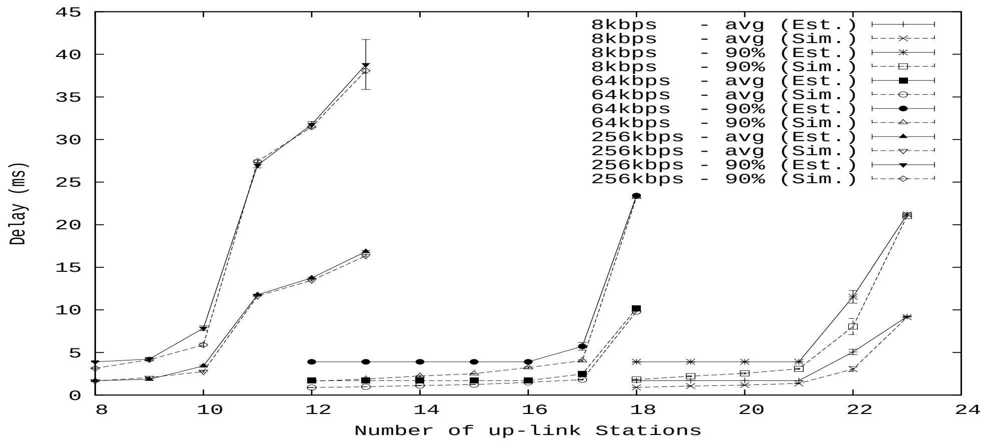

Figure 1: Average delay and 90% perc. delay for homogeneous VoIP traffic at different bit-rates.

VoIP sources. We have 5 symmetric flows at 256 kbps and a variable number of additional 64 kbps flows. The delays in the plots refer only to 256 kbps flows. Again, the transition is smoother and the delay larger when $CW_{min} = 32$ is used.

5.2 VoIP and interfering BE traffic

This set of simulations refer to a scenario with mixed voice and best-effort traffic. VoIP traffic is 8, 64 and 256 kbps, at 50 packets per second as before. Best-effort traffic is a fixed number of UDP up-link flows, 8 stations transmitting 1 Mbps at 100 packets per second. We have used standard MAC parameters for both the VO and BE access categories.

The plots in Fig. 5 highlight that, as expected, BE traffic has a limited effect: the number of VoIP flows that can be admitted is approximately the same of that measured without interfering BE flows, even though the delay is slightly larger and the transition to the saturated situation less abrupt.

These simulations confirm that the delay estimate is precise independently from the traffic generated in the wireless channel by sources belonging to other access categories: the same result can be expected in presence of legacy stations.

5.3 Delay estimate with physical errors

For the simulations reported in Fig. 6 we have introduced a small modification in the ns-2 code to introduce the specification of the packet error rate (PER). The simulation set-up is the same as in Fig. 4, with extended contention windows and heterogeneous VoIP flows. The PER applied to all the up-link flows and the value used in the simulations reported in the plot is 5%. The plots are very similar to those of Fig. 4 (notice however that the maximum number of admissible station is lower). The estimate obtained through our approach is now slightly below the actual average delay: the reason is the fact that the number of packets correctly transmitted is less than the number of queued packets, as predicted in Sec. 3.2. At this PER the estimate is still reasonably precise, thus we can assume that the approach works well with a PER less or equal to 5%.

5.4 Delay estimate with varying traffic conditions

The last set of simulation aims at validating the method for estimating at real-time the delay variations described in Section 3. The plot in Fig. 7 refers to the up-link delay experienced by one of 10 64 kbps VoIP flows (at 50 packets per second), that use the default VO access category settings. The simulation delay is instantaneous and the total duration of the simulation is 200 s. Starting at $t = 200s$, a 256 kbps VoIP flow (at 50 packets per second) starts every 20 s, until $t = 340s$. A total of 8 256 kbps flows joins the wireless cell. There is no notable delay increase until the 5th flow is added at $t = 280s$. The plot reports the delay estimate with the memory parameter β set at 0.90. The estimated delay is averaged on samples of 50 ms, too. Also in this case, when the up-link delay exceeds the threshold of 5 ms the estimator starts tracking the delay. The time shift between the actual and the estimated delay is under 1s.

6. CONCLUSIONS

In this paper we presented an analytical framework to model the sending queue of an IEEE 802.11e EDCA wireless station that uses the $TXOP_{limit}$ MAC parameter to send more than one packet in each transmission opportunity. An application of this model to the estimate of packet delay from the number of packets sent in each transmission is presented. The model has been validated through simulations analysis proving the ability of capturing the delay in a wide range of configurations. Several extensions to this model are not contained in this paper and are the objective of on-going effort: the validation of the approach in the case of traffic at variable packet rate, espcially in the relevant case of traffic generated by video codecs, the extension to the estimate of packet loss due to queue overflow, the generalization to the case of multiple flows, with heterogeneous features, on the same sending queue. We are also currently working on the implementation of presented mechanisms in off-the-shelf AP (based on openWRT OS) and, following the same ideas of [22], in the definition, analysis and experimentation of reactive admission control mechanisms.

Acknowledgment

This work is partly funded by the 6th Framework Programme, Information Society Technologies, under Contracts no. FP6-IST-038423 (CONTENT), no. FP6-IST-034819 (ONELAB)

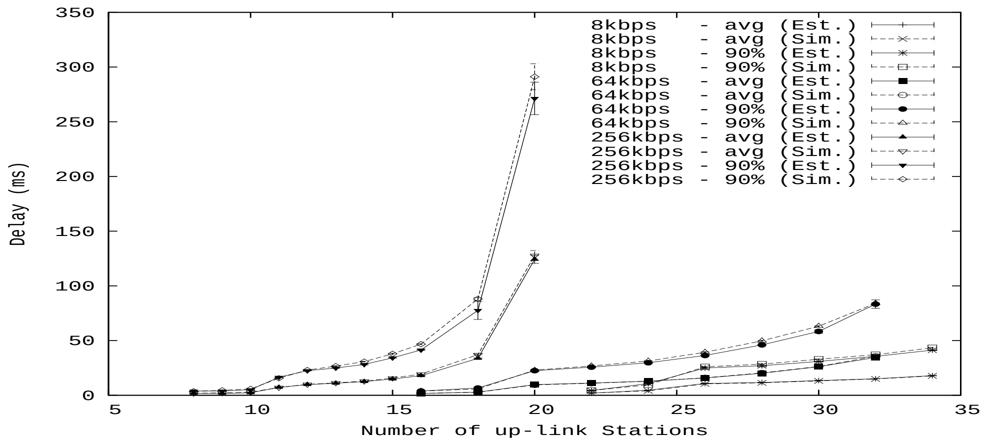

Figure 2: Average delay and 90% perc. delay for homogeneous VoIP traffic at different bit-rates, $CW_{min} = 32$.

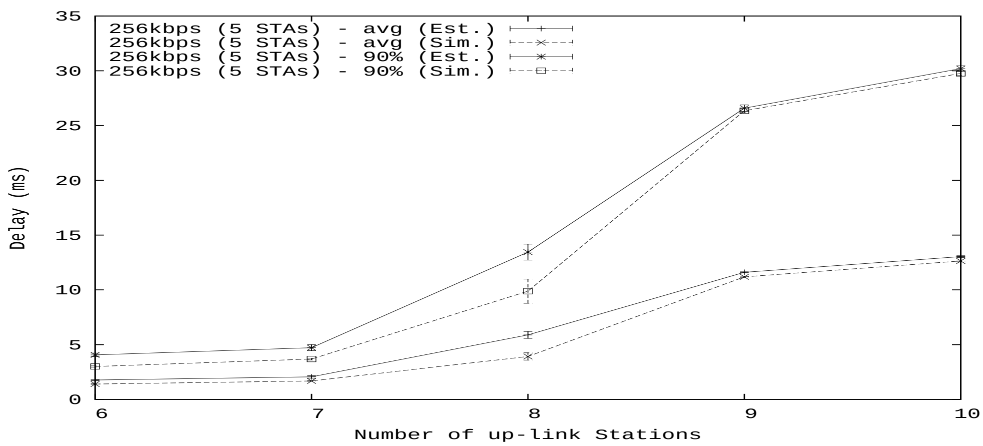

Figure 3: Average delay and 90% perc. delay for 256 kbps VoIP traffic in presence of 64 kbps VoIP traffic.

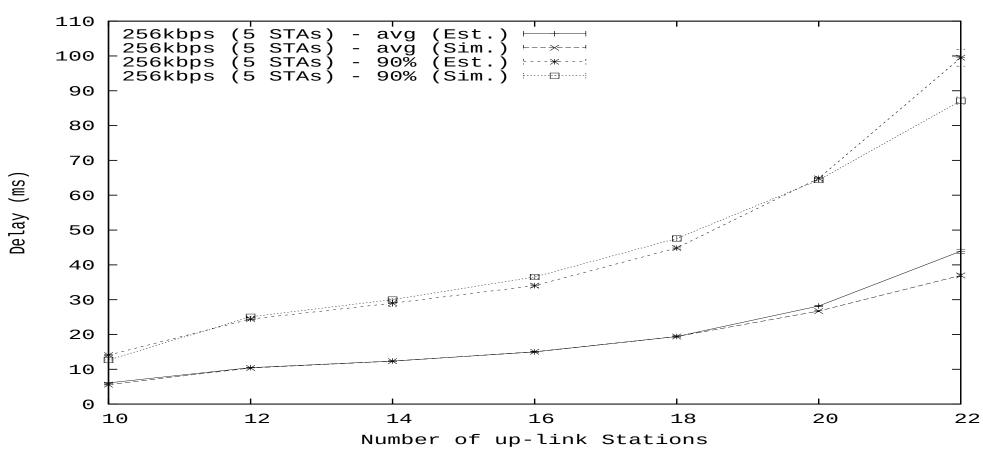

Figure 4: Average delay and 90% perc. delay for 256 kbps VoIP traffic in presence of 64 kbps VoIP traffic, $CW_{min} = 32$.

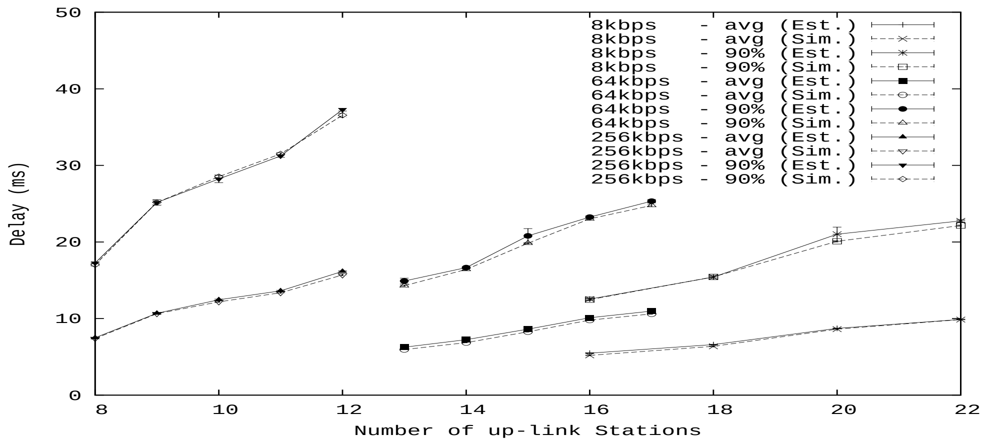

Figure 5: Average delay and 90% perc. delay for VoIP traffic in presence of BE traffic.

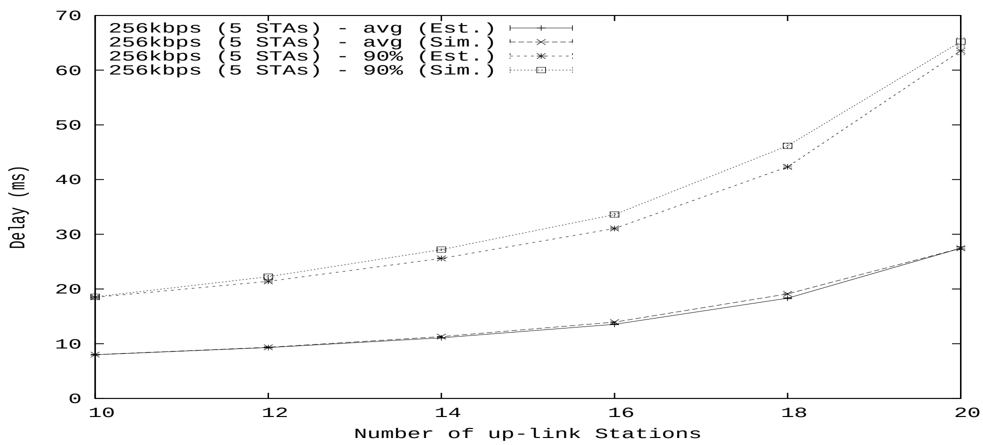

Figure 6: Average delay and 90% perc. delay for 64 kbps VoIP traffic in presence of 256 kbps VoIP traffic, $CW_{min} = 32$**, PER=0.05.**

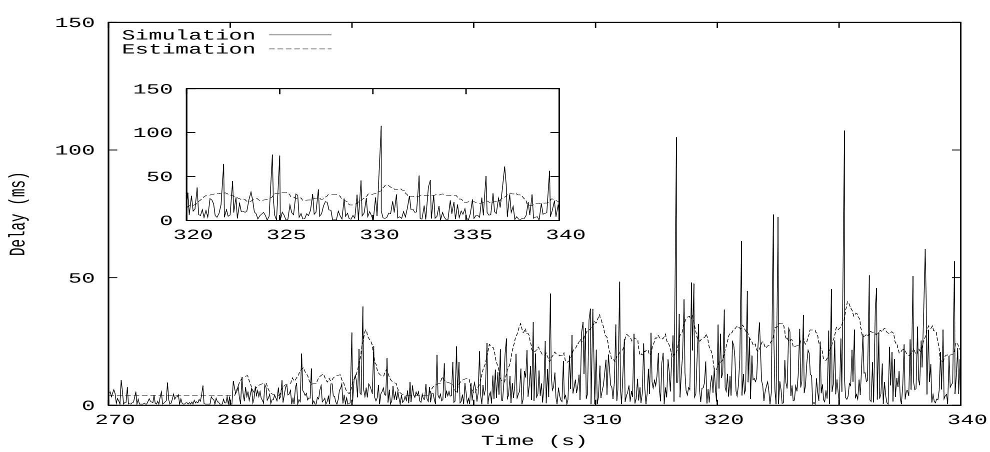

Figure 7: Dynamic up-link delay estimate of a 64 kbps VoIP flow for an increasing number (1-8) of competing 256 kbps VoIP flows.

and no. FP6-IST-033516 (NETQOS).

7. REFERENCES

[1] I. S. 802.11 2007. *IEEE Standard for Information Technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE802.11-REVma/D9.0, Revision of Std. 802.11-1999, August 2007.

[2] L. Yang, P. Zerfos, and E. Sadot. "Architecture Taxonomy for Control and Provisioning of Wireless Access Points (CAPWAP)". RFC 4118 (Informational), June 2005.

[3] P. Salvo Rossi. G. Romano, F. Palmieri and G. Iannello; Joint end-to-end loss-delay hidden Markov model for periodic UDP traffic over the Internet IEEE Transactions on Signal Processing, Volume: 54, Issue: 2, page(s): 530- 541, Feb. 2006

[4] D. Gao, J. Cai, and K. N. Ngan, Admission Control in IEEE 802.11e Wireless LANs, IEEE Network, vol. 19, 4 (2005), pp. 6-13.

[5] L. X. Cai, X. Shen, J. W. Mark, L. Cai, and Y. Xiao, Voice Capacity Analysis of WLAN With Unbalanced Traffic, in IEEE Trans. on Vehicular Technology, 55,3 (2006).

[6] D. Pong, and T. Moors, Call Admission Control for IEEE 802.11 Contention Access Mechanism, in *Proceedings Globecom 2003*, San Francisco, December 2003.

[7] D. Gu, and J. Zhang, A New Measurement-Based Admission Control Method for IEEE 802.11 Wireless Local Area Networks, in *Proceedings IEEE Int. Symp. on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, vol. 3, pp. 2009-2013, September 2003.

[8] S. Garg, and M. Kappes, Can I add a VoIP call?, in *Proceedings IEEE ICC*, May 2003, vol. 2, pp. 779-783.

[9] B. Bellalta, M. Meo, and M. Oliver, A BEB-Based Admission Control for VoIP Calls in WLAN with Coexisting Elastic TCP flows, in *Proceedings of NEW2AN*, 2006, pp. 130-141.

[10] L. Zhang and S. Zeadally, HARMONICA: Enhanced QoS Support with Admission Control for IEEE 802.11 Contention-based Access, in *Proceedings IEEE RTAS âĂŹ04*, Toronto, Canada, May 2004, pp. 64âĂŞ71.

[11] H. Zhai, X. Chen, Y. Fang, A call admission and rate control scheme for multimedia supoprt over IEEE 802.11e wireless LANs, ACM Wireless Networks, vol. 12,4 (2006), pp. 451-463.

[12] I. Dangerfield, D. Malone, and D. J. Leith, Experimental Evaluation of 802.11e EDCA for Enhanced Voice over WLAN Performance, in *Proceedings of the Second Int. Work. On Wireless Network Measurement (WiNMee 2006)*, Boston, April 2006.

[13] S. Harsha, A. Kumar, and V. Sharma, An Analytical Model for the Capacity Estimation of Combined VoIP and TCP File Transfers over EDCA in an IEEE 802.11e WLAN, in *Proceedings 14th IEEE International Workshop on Quality of Service (IWQoS 2006)*, New Haven, June 2006, pp. 178-187.

[14] Y. Tanigawa, J. Kim, H. Tode, K. Murakami, Proportional Control and Determinstic Protection of QoS in IEEE 802.11e Wireless LAN, in *Proceedings Int. Wireless Communications and Mobile Computing Conf. (IWCMC 06)*, Vancouver, July 2006.

[15] P. Clifford, K. Duffy, J. Foy, D. Leith, and D. Malone, Modeling 802.11e for data traffic parameter design, in *Proceedings 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt 2006)*, April 2006, pp. 1-10.

[16] C. Casetti, and C.-F. Chiasserini, Improving Fairness and Throughput for Voice Traffic in 802.11e EDCA, in *Proceedings 15th IEEE Int. Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC 04)*, Barcelona, 5-8 September 2004.

[17] Y. Xiao and H. Li, Voice and Video Transmissions with Global Data Parameter Control for the IEEE 802.11e Enhanced Distributed Channel Access, IEEE Transactions Parallel Distributed Systems, vol. 15,11, (2004), pp. 1041âĂŞ53.

[18] Y. Xiao, F. H. Li, and B. Li, Bandwidth Sharing Schemes for Multimedia Traffic in the IEEE 802.11e Contention-Based WLANs, IEEE Transactions on Mobile Computing, vol. 6,7 (2007), pp. 815-831.

[19] S. El Housseini, and H. Alnuweiri, Adaptive Contention-Window MAC Algorithms for QoS-Enabled Wireless LANs, in *Proceedings of 2005 International Conference on Wireless Networks, Communications and Mobile Computing*, Vol.1,Âă pp. 368- 374, June 2005.

[20] S. Acanfora, Filippo Cacace, Giulio Iannello, and Luca Vollero, Dynamic Configuration of MAC QoS Mechanisms in 802.11 Access Networks, in *Proceedings 6th International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking, NEW2AN 2006*, St. Petersburg, Russia, May 2006, pp. 542-553.

[21] J. Freitag, N.L.S. da Fonseca, J.F. de Rezende, J.F., Tuning of 802.11e network parameters, IEEE Communications Letters, vol. 10,8 (2006).

[22] F. Cacace, G. Iannello, M. Vellucci, and L. Vollero, A Reactive Approach to QoS Provisioning in IEEE 802.11e WLANs, in *Proceedings 4th EURO-NGI Conference on Next Generation Internet Networks*, 28-30 April 2008, Krakow, Poland.