Ensuring Quality Services on WiFi Networks for Offloaded Cellular Traffic

Gianluigi Pibiri
School of Computer Science and Statistics
Trinity College Dublin
Dublin 2, Ireland.
Email: pibirig@tcd.ie

Ciarán Mc Goldrick
School of Computer Science and Statistics
Trinity College Dublin
Dublin 2, Ireland.
Email: ciaran.mcgoldrick@tcd.ie

Meriel Huggard
School of Computer Science and Statistics
Trinity College Dublin
Dublin 2, Ireland.
Email: meriel.huggard@tcd.ie

***Abstract*—One of the more obvious ways to reduce the volume of data traffic on cellular networks is through the use of handover to fixed networks via WiFi and other radio channels. With the growing focus on emerging 5G concepts and technologies, there has been a corresponding focus on the practical mechanisms needed to achieve this handover in a timely fashion. Much less attention has been paid to the practicalities, in terms of ensuring that the end-user experiences little or no loss in the quality of their network services when the handover occurs. In this paper, a methodology for managing such handover traffic to a WiFi network is proposed. The approach integrates and leverages aspects of three quality control mechanisms to enable stable, higher-quality delivery of enhanced WiFi network services. It combines i) information adduced from a theoretical model with ii) a low complexity Quality of Experience metric that is quick and easy to estimate and iii) a queue management scheme.**

I. Introduction

Traffic on fourth generation (4G) cellular networks exceeded that on third generation (3G) systems for the first time in 2015 [1]. There has also been a concomitant growth in the volume of traffic that is offloaded from these cellular network onto fixed networks; to the extent that in 2015 over 50% of mobile data traffic was offload from cellular networks onto fixed networks [1]. This was achieved through the use of WiFi and femtocells. Many emerging 5G metaphors envisage "zero distance" connectivity between connected machines and connected people, along with many thousand-fold increases in both connected entities and mobile data; hence it is inevitable that there will be an associated increase in deployment of femtocells, macrocells and other last hop connection metaphors. Clearly this increased dynamicism in device attachment points, and massively more challenging mobility scenarios e.g. vehicular networking, will affect the end-end characteristics of the routed traffic.

The nature of the traffic on these networks has also shifted away from traditional voice calls and audio streaming towards mobile video services. The latter accounted for 55% of all mobile data traffic in 2015 [1]. One of the key challenges to be faced is to ensure that the end user does not perceive any loss in the quality of service they are receiving if their real-time cellular data traffic is handed over to a fixed network.

This paper presents an approach to minimizing the impact of these increasing dynamic network configurations on the increasingly time and delay sensitive traffic that underpins current and emerging mass-market end-user services.

II. Related Work

A. Quality of Service

The specification of Quality of Service (QoS) [2] has evolved and been extended since it was originally defined in 1994. The measurement of QoS has been divided into three layers [3]: Intrinsic QoS (IQoS), Perceived QoS (PQoS) and Assessed QoS (AQoS); however, the boundaries between these three layers are neither well defined nor clear-cut.

IQoS [3] is commonly known as just QoS. It is an evaluation of quality at the network performance level. The metrics used for its evaluation are the network parameters; however, these do not provide a direct measurement of the QoS. If the QoS parameters are good then the service is likely to be provided with high quality and vice versa. IQoS does not specify an exact percentage of packets lost that is to be considered high or low and consequently the quality level for the end user cannot be easily defined as good, acceptable, bad, etc.

PQoS is a measure of the human perception of the quality of the service provided [3]. PQoS is divided into four classes [4] [5], each of which focuses on an aspect of the QoS from the customer's or network provider's point of view: The QoS offered and achieved from the provider's point of view, and QoS required and perceived from the customer's point of view. An evaluation of components to estimate the four perceived QoS classes in future networks is proposed in [5].

The AQoS [3] is a high level measure of customer satisfaction; for example, when the user decides if they wants to continue to use a service or not. Like PQoS, it is a subjective metric as it is based on the user's opinion of the service. Quality of Experience (QoE) [6] is a practical quantification of AQoS, even if QoE quantifies aspects of the PQoS from the user's point of view [3]. PQoS and AQoS may initially appear to be similar; however, they are different as AQoS is not related to the the network parameters but only to customer satisfaction. QoE is measured at the application level; while PQoS, from the network operator's point of view, is inferred from the network's performance. The metrics used

to evaluate QoE depend only on the human, end user opinion and experience of the network quality.

B. eQoS

Two key characteristics are essential for capturing QoE: quick estimation and simple calculation. To achieve these the eQoS metric [7] is used. This metric that captures the perceived QoS through an almost instantaneous measurement of loss in network quality. eQoS is designed to be calculated per network flow and to provide a near instantaneous evaluation of service quality at the node. It can be applied at an AP or where single or multi queue systems are present and it can be used to inform the operation of queue management algorithms. eQoS is expressed as a proportion between 0 and 1 or as a percentage and is a dimensionless quantity.

eQoS provides a near instantaneous mechanism for calculating the perceived quality at a node for critical, real time services that are highly sensitive to packet loss and delay. Traditional QoE metrics require audio or video measured over an extended time period by the end user; e.g., traditional algorithms typically require more than 10 seconds of audio or video traffic in order to provide an objective evaluation [8].

The eQoS sampling time is not a fixed value; it is set according to the time needed to encode the service. It is set to ensure that the number of packets transmitted per second by each service is sufficient to obtain a reasonable eQoS estimate. For services provided with a particular protocol, like TCP, the eQoS sampling intervals may vary. This is because the number of samples per second will depend on the Round Trip Time (RTT) [9]. eQoS depends on a small number of QoS network parameters and so its calculation is very simple. It measures the perceived QoS and provides an estimate of QoE.

III. Offloaded Traffic Management System

Next generation wireless networks aim to achieve significantly higher throughput then existing systems and will make extensive use of multi queue systems; thus the wireless queue management algorithms of the future will carry out many functions beyond that of simple congestion avoidance. These new algorithms address not only congestion avoidance but also QoS assurance.

This section introduces a new mathematical model, derived using combinatorics, to describe packet transmission in a future wireless network where traffic prioritization methods like Enhanced Distributed Channel Access (EDCA) are employed. The model then forms the basis for the design of a new class of offloaded traffic management systems called Quality Queue Management (QQM) schemes.

QQM schemes measure eQoS and implement congestion avoidance mechanisms whilst simultaneously managing contention windows (CW) and queueing priorities. QQM operates on a per flow basis across all services and traffic types. It is designed to reduce delay and discard poor quality traffic.

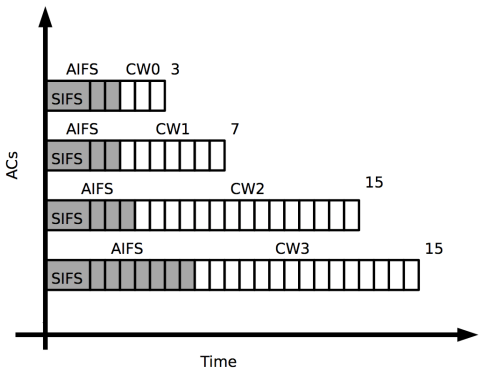

Fig. 1. EDCA: AIFS and Backoff times with CW_{min}

IV. Theoretical Analysis of a Multi Queue System

In this section the multi queue system detailed above is analysed. The analysis focuses on a theoretical description of packet behaviour in a future wireless network.

It is assumed that:

- The network uses a 160MHz channel with a Single Input and Single Output (SISO) configuration similar to that of IEEE802.11ac [10] standard is used.
- EDCA (or a similar class based procedure) is used. This gives high priority to particular traffic and message types. It is assumed that Contention Free Bursting (CFB), also known as TXOP, is used.
- CSMA/CA is not implemented on the wireless network.The high speed of future wireless networks makes CSMA/CA an obstacle for enhancing performance as CSMA/CA control packet are sent at a very low speed for backward compatibility.

As set out in Table I, each AC has a backoff time of between 0 and CW_{min} time slots with an associated $AIFS[AC]$. For example, for AC_0 the backoff time is between 0 and three timeslots and $AIFS[0]$ is a SIFS time plus two time slots.

For each new transmission attempt, each backoff time slot has an equal probability of being randomly chosen, therefore this probability follows a discrete uniform distribution across the interval $[0, CW]$.

The key probabilities of interest are the probability that a successful transmission occurs, the probability that a collision occurs and the probability that the channel is idle. The probability a transmission occurs is denoted $\mathbb{P}_t$, the probability a collision occurs is $\mathbb{P}_c$ and the probability there are no packets queued for transmission is $\mathbb{P}_e$.

A. Determining the Probability of Successfully Transmitted

The first challenge is to obtain a mathematical description of how each AC obtains access to the channel. Let each distinct access category in the system be indexed by j. The number of slots in the contention window associated with each AC_j is given by CW_j. Let the number of AC_js that are seeking to access the channel at a given instant in time be N_{AC_j}. Assuming that x is the number of ACs that are ready to transmit and that y is the index of the time slot to be used for the transmission then the total number of ways

Access Category	CW_{min}	CW_{max}	TXOP	$AIFSN$	Traffic Type
AC_0 (AC_VO)	3	7	1.504ms	2	voice and audio
AC_1 (AC_VI)	7	15	3.008ms	2	video
AC_2 (AC_BE)	15	1023	0	3	best effort
AC_3 (AC_BK)	15	1023	0	7	background

TABLE I
SUMMARY OF AC PARAMETERS [11]

$(N_{AC_j} - x)$ ACs are not ready to transmit is given by the following permutation with repetition [12]:

$$P'_{(CW_j-y),(N_{AC_j}-x)} = (CW_j - y)^{N_{AC_j}-x}.$$

The following notational simplification is used to assist the reader in the subsequent discussion:

$$P'_{(CW_j-y),(N_{AC_j}-x)} = P'_{N_{AC_j}(x,y)}.$$

This gives the total number of ways $(N_{AC_j} - x)$ ACs are not ready to transmit in slot y as:

$$P'_{N_{AC_j}(x,y)} = (CW_j - y)^{N_{AC_j}-x}. \tag{1}$$

The repetitions included in $P'_{N_{AC_j}(x,y)}$ represent future collisions.

The probability that an AC_j transmits in time slot i is:

$$\mathbb{P}_t(AC_j)_i = N_{AC_j} \times (\mathbb{P}_{AC_j})^{N_{AC_j}} \times P'_{N_{AC_j}(1,i+1)}. \tag{2}$$

For an infrastructure wireless network that implements EDCA there will be four different types of AC that compete for access to the channel. From Figure 1, it can be inferred that at time slot 0 only two types of ACs are able to transmit: AC_0 and AC_1. Using equations 1 and 2 it is possible to calculate the probability that one AC_0 is transmitting in slot 0 as [12]:

$$\begin{aligned}\mathbb{P}_t(AC_0)_0 =& N_{AC_0}(\mathbb{P}_{AC_0})^{N_{AC_0}} P'_{N_{AC_0}(1,1)} \\ &\times (\mathbb{P}_{AC_1})^{N_{AC_1}} P'_{N_{AC_1}(0,1)}.\end{aligned} \tag{3}$$

The probability that one AC_1 is transmitting in time slot 0 is:

$$\begin{aligned}\mathbb{P}_t(AC_1)_0 =& (\mathbb{P}_{AC_0})^{N_{AC_0}} P'_{N_{AC_0}(0,1)} \\ &\times N_{AC_1}(\mathbb{P}_{AC_1})^{N_{AC_1}} P'_{N_{AC_1}(1,1)}.\end{aligned} \tag{4}$$

No packets are transmitted when slot 0 is empty for all AC_0s and all AC_1s . The probability that slot 0 is empty, $\mathbb{P}_{e_0}$, is the product of the probability that slot 0 is not randomly chosen by any of the AC_0s and the probability that slot 0 is not randomly chosen by any of the AC_1s:

$$\mathbb{P}_{e_0} = \frac{P'_{N_{AC_0}(0,1)}}{P'_{N_{AC_0}(0,0)}} \times \frac{P'_{N_{AC_1}(0,1)}}{P'_{N_{AC_1}(0,0)}}. \tag{5}$$

Collisions occur when the backoff periods for at least two ACs expire at the same time. It is assumed that collisions occur between a maximum of two stations at the same time [12]. The probability a collision occurs, $\mathbb{P}_c$, at a given slot time is the sum of the collision probabilities between two AC_0s, two AC_1s or between one AC_0 and one AC_1. The probability of a collision in time slot 0, $\mathbb{P}_{c_0}$ is:

$$\begin{aligned}\mathbb{P}_{c_0} =& \mathbb{P}_c(AC_0)_0 + \mathbb{P}_c(AC_1)_0 \\ &+ \mathbb{P}_c(AC_0, AC_1)_0.\end{aligned} \tag{6}$$

Using a similar methodology to [12], the probability two AC_0s give rise to a collision, $\mathbb{P}_c(AC_0)_0$, is given by:

$$\mathbb{P}_c(AC_0)_0 = \frac{\binom{N_{AC_0}}{2}[(CW_0-1)^{(N_{AC_0}-2)}(CW_1-1)^{(N_{AC_1})}]}{(CW_0)^{N_{AC_0}}(CW_1)^{N_{AC_1}}}. \tag{7}$$

The probability two AC_1s give rise to a collision, $\mathbb{P}_c(AC_1)_0$, is:

$$\mathbb{P}_c(AC1)_0 = \frac{\binom{N_{AC_1}}{2}[(CW_0-1)^{(N_{AC_0})}(CW_1-1)^{(N_{AC_1}-2)}]}{(CW_0)^{N_{AC_0}}(CW_1)^{N_{AC_1}}}. \tag{8}$$

The final probability to be calculated is that of a collision due to the time slot choices of one AC_0 and one AC_1. The probability that both an AC_0 and an AC_1 have chosen time slot 0 is:

$$\begin{aligned}\mathbb{P}_c(AC_0, AC_1)_0 =& (N_{AC_0} \times N_{AC_1}) \\ &\times \frac{[(CW_0-1)^{(N_{AC_0}-1)}(CW_1-1)^{(N_{AC_1}-1)}]}{(CW_0)^{N_{AC_0}}(CW_1)^{N_{AC_1}}}.\end{aligned}$$

The calculation of $\mathbb{P}_c$ is more complicated when more than two types of AC are involved. There are two possible ways that a successful transmission might occur: either a packet is successfully transmitted by AC_0 or a packet is successfully transmitted by AC_1. So the probability a collision occurs in time slot 0 is:

$$\begin{aligned}\mathbb{P}_{c_0} =& (1 - ((\mathbb{P}_t(AC_0)_0 + \mathbb{P}_t(AC_1)_0) + \mathbb{P}_{e_0})) \\ =& 1 - \mathbb{P}_t(AC_0)_0 - \mathbb{P}_t(AC_1)_0 - \mathbb{P}_{e_0}.\end{aligned} \tag{9}$$

This argument can be extended to find the probability a collision occurs when more than two ACs are competing for access in a given time slot. This model can also be extended to AC_2 and AC_3 [13]. Using this methodology it is possible to estimate the probability that a successful transmission occurs, the probability that a collision occurs and the probability that the channel is idle.

B. Contention Window Sizes and Queueing in AC_0 and AC_1

The first important consideration that merits further discussion relates to how the CW size is determined. The CW value is chosen randomly in the interval $[0, CW_{min}]$. It is possible to estimate CW sizes by considering each station separately.

One empirical method that might be suited for CW estimation is the German Tank Problem [14]. With just a few samples this method can be used to estimate the average CW size to an acceptable level of precision. An alternate, theoretical approach would be to use the statistical expectation associated with a discrete uniform distribution. Both these methods are dynamic: using the first method the CW size needs to be estimated periodically, while using the second method the CW size needs to be estimated every time the

upper CW limit is exceeded. Despite their differences, both methods provide acceptable results for use in the theoretical calculations discussed above.

A second important consideration relates to the AC_0 and AC_1 queues. Q_0 is the queue associated with AC_0 and Q_1 is the queue associated with AC_1. Both Q_0 and Q_1 are functions of the characteristics of AC_0 and AC_1 respectively.

Future wireless networks will have a very high throughput particularly in comparison to the packet transmission frequency for VoIP traffic. The exact instant a VoIP conversation starts is unpredictable. Protocol G.729 [15] transmits and receives at a frequency of 50 packets per second. The IEEE802.11ac protocol can be considered as a spatial stream with a 160MHz channel. In this case a packet is transmitted about every $0.265ms$, not including collisions and empty slots due to backoff. At a frequency of 3.8KHz and assuming a worst case scenario where CSMA/CA control packets are used, this corresponds to more than three thousand packets per second. Even if a very large number of VoIP conversations are present in the network, the frequency at which they access the channel is low when compared to the transmission frequency used for the IEEE802.11ac packets. Since AC_0 has highest priority, Q_0 is likely to contain at most one or two packets.

AC_1 has a lower priority than AC_0, but it is still subject to similar effects and considerations. An audio or video stream, encoded using MPEG4 [16] [17] transmits one I frame per second and 29 P frames per second. I frames are, on average, composed of 12 packets with a size of 1024 bytes each. An average of 2 packets is needed for each P frame. The frequency of the video transmission, f_{Video}, is 30 frames per second but the number of packets per frame is variable. AC_1's TXOP feature is $2.008ms$ long and is sufficient for the transmission of 45 streaming packets.

From the above it can be concluded that AC_0 and AC_1 only periodically occupy the channel for packet transmissions. The frequency with which they transmit packets is related to the total number of VoIP and streaming packets passing through the AP. AC_2 manages traffic that has a low frequency of demand for access to channel. This traffic is similar to that of AC_1 and the considerations for AC_0 and AC_1 remain valid for AC_2. AC_3 will either make use of any remaining transmission time to send its data or else will transmit until it exceeds the upper limit imposed on its throughput.

V. Estimating the number of ACs simultaneously contending for channel access

A wireless network with N VoIP calls in progress has, in effect, $N+1$ classes in competition for access to the channel because the AP must be included in the calculations. It is assumed that the AP accesses the channel with the same frequency as a mobile station. However, the frequency at which an AP seeks to access the channel may well differ from that of the mobile stations because the AP manages all traffic being transmitted to the mobile stations.

It is necessary to determine the relationship between the frequency of channel access requests for a service and the number of mobile stations simultaneously accessing the channel. In the following discussion this is explored for the three traffic types of real time traffic that is expected to be offloaded from cellular networks to small cell and wifi network

Protocol G.729 is used for VoIP traffic. It needs to transmit 50 packets per second. The worst case transmission time for a single packet, including the Layer 2 ACK control packet, on future wireless networks is about $150\mu s$. If a packet is transmitted every $20ms$, then, theoretically, a maximum of 120 packets can be transmitted in $20ms$.

If it is assumed that every mobile station with a phone call in progress need to transmit a packet every $20\mu s$, then there are $N+1$ ACs seeking access to the channel in each of the 120 $slots$ that make up a $20ms$ interval. The problem is to estimate the likelihood that more than one AC accesses the channel in the same $150\mu s$ slot as this will cause a collision.

Statistically, the number of ACs competing for the channel have the same likelihood of picking each individual time slot for transmission. It is necessary to find how many possible ways there are for at least two of the $N+1$ mobile nodes to pick the same slot amongst the 120 $slots$ on offer. This can be found using permutations with repetition [13]. The required probability will be given by the ratio of the number of permutations with at least one repetition, i.e. the number of ways that at least one collision can occur, to the number all the possible permutations:

$$\mathbb{P}_{(rep)} = \frac{{P'}_{AC_{0(slots,N+1)}} - P_{AC_{0(slots,N+1)}}}{{P'}_{AC_{0(slots,N+1)}}} \tag{10}$$

Where $\mathbb{P}_{(rep)}$ is the probability that multiple ACs choose the same time slot. P and P' are the permutations without repetitions and with repetitions respectively. Here, $n = slots$ and $k = N+1$, so that:

$$P_{AC_{0(slots,N+1)}} = \frac{(slots)!}{((slots)-(N+1))!} \tag{11}$$

and

$${P'}_{AC_{0(slots,N+1)}} = slots^{(N+1)} \tag{12}$$

The probability that a repetition occurs i.e. that two ACs compete for access to the channel is given by:

$$\mathbb{P}_{(rep=2)} = \frac{slots \times \begin{pmatrix} N+1 \\ 2 \end{pmatrix} \times P_{AC_{0(slots-1,N-1)}}}{{P'}_{AC_{0(slots,N+1)}}} \tag{13}$$

Figure 2 shows the variations in $\mathbb{P}_{(rep=2)}$ as the number of phone calls in the system grows. In the graph on the left there are 120 $slots$ in each $20ms$ interval. In the graph on the right a worst case scenario of 60 $slots$ in each $20ms$ interval is shown. From the graphs it can be seen that until there are more than ten calls in progress, the probability only two AC_0s compete for access to the channel at the same time is over 90%; therefore it is reasonable to consider that only two AC_0s are ready to transmit at any given time provided the number of phone calls flowing in the wireless network does not exceed ten.

Audio streams must be considered next. These are managed by AC_0 and are similar to unidirectional phone calls and have,

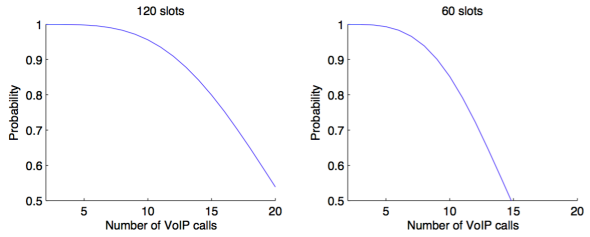

Fig. 2. Probability a maximum of two AC_0s access the channel at the same time in the same slot for VoIP calls.

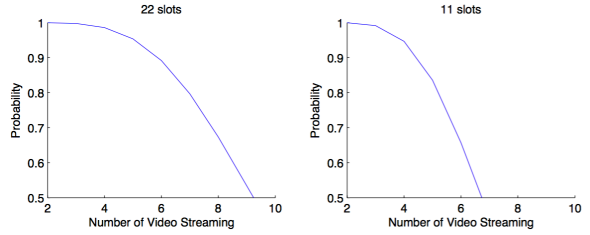

Fig. 3. Probability a maximum of two AC_1s access the channel at the same time in the same slot for video streams.

approximately, the same channel access frequency as VoIP calls.

Finally, video streams must be considered. In this case the traffic rate is variable and there are 30 frames transmitted per second, i.e. one frame is transmitted every $33ms$. In the worst case, when many flows are being streamed from the same node, the transmission time for each frame containing video can be assumed to be twice the TXOP time. Each $slot$ is assumed to be $1.5ms$ long, therefore in every $33ms$ interval there are 22 $slots$ are available for transmission, or in the worst case only 11 $slots$ are available.

Figure 3 shows the experiments carried out to explore the probability that a maximum of two AC_1 video streams access the channel at the same time in the same slot. From this it can be inferred that for less than six flows, there is a 90% chance that only two of them are competing for access to the channel.

The discussion and experiments above confirm that the number of AC_0s and AC_1s competing for access to the channel at the same time does not exceed two if less than ten phone calls and audio streams or less than six video streams are present in a wireless network.

Based on the measured eQoS it can be used to simultaneously optimise P_{e_0}, P_{c_0} and the probability to transmit a packet depending on the quality of the service provided. The sizes of CW_0 and CW_1 are set to the maximum when the qualitative score is evaluated as Excellent, to the minimum size when the qualitative score is Fair, Poor or Bad and to the median value when the score is Good.

The differences in the CW_0 and CW_1 sizes reflects the different priorities assigned by the EDCA method. When a collision occurs the CWs are doubled. The optimal CW_0 and CW_1 values summarised in Table II are used to design a fuzzy CW controller within the QQM algorithm.

VI. The Quality Queue Management algorithm

The QQM algorithm manages traffic at a wireless AP to provide services over the network with the best possible QoE. It is designed to operate on future wireless and small cell networks to manage the ACs parameters, and their interactions, according to the measured eQoS. QQM not only manages the traffic crossing the AP but it can also be used to make handover decisions and give feedback to applications to reduce the traffic generated by some services.

QQM algorithms incorporate three key features. The first feature is quality assurance. This is achieved by implementing eQoS flow preservation; flows are dropped if the minimum eQoS cannot be guaranteed. The minimum eQoS is a QoE value; by definition once the eQoS drops below this level the end-user makes the decision to drop the service. The second feature is the management of the transmission, collision and idle channel probabilities through modulation of the contention window (CW). The third feature of QQM is that it seeks to manage the queue length and avoid congestion.

The novelty of QQM is that these three features are combined in a single system. The interaction and information exchange between the features contributes to provision of the service with the best quality possible over the network. The implementation of these features is now considered in detail.

eQoS flow preservation is achieved through continuous checking of the eQoS for each flow and comparison of the values obtained with historical eQoS data. Flows for which the eQoS falls below an acceptable threshold are dropped. This feature guarantees that only flows that achieve a minimum and acceptable eQoS are transmitted.

Packets are lost in the wireless network when one of three types of dropping event occurs. The first such dropping event is a collision. In this case a packet retransmission is carried out by the QQM algorithm and packets may be dropped if the delay exceeds the maximum acceptable delay for the service. Quality thresholds and levels have been inferred from literature [18]. The second type of dropping event occurs when the queue length exceeds the congestion threshold or if the queue is such that the delay will exceed the maximum delay allowed for the service. The third type of dropping event occurs due to eQoS quality preservation. This occurs when a flow does not achieve the minimum eQoS required to provide a satisfactory service to the end user and so all of the flow's packets are dropped.

QQM also includes a queue management algorithm. The packet dropping decision process differs from that of traditional AQM algorithms: It uses eQoS to guarantee, in so far as possible, the provision of services that meet at least the minimum quality standards.

QQM uses fuzzy logic and a fuzzy control system to manage traffic on future wireless networks and small cell networks. Figure 4 shows a flow chart that captures the key operational elements of QQM. The algorithm is composed of three fuzzy controller blocks and it aims to manage the queues and the contention windows under the control of the eQoS block.

CW_0 \ CW_1	Excell.	Good	Fair	Poor	Bad	N/A
Excellent	8 16	8 12	8 8	8 8	8 8	4 8
Good	6 16	6 12	6 8	6 8	6 8	4 8
Fair	4 16	4 12	4 8	4 8	4 8	4 8
Poor	4 16	4 12	4 8	4 8	4 8	4 8
Bad	4 16	4 12	4 8	4 8	4 8	4 8
N/A	4 8	4 8	4 8	4 8	4 8	4 8

TABLE II
VIDEO STREAMING DETAILS

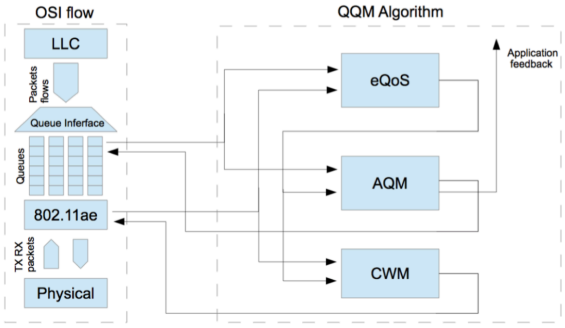

Fig. 4. QQM algorithm flow chart.

These three blocks measure and control the traffic flows to guarantee an acceptable level of QoE for real time services, and a good level of QoS for all other services.

The eQoS block estimates the quality of service per flow. It is also an active block which interacts with the traffic through the AQM and contention window manager blocks, dropping packets that do not conform to the minimum quality level established for provision of the service.

The AQM block is at the heart of the QQM system. It provides queue management and queue length control. It can also provide application layer feedback to the traffic sources. The contention window management block is based on the model described above and performs priority management of the contention windows. Its main goal is to optimise priorities across all ACs in order to reduce queue lengths.

VII. EVALUATION

The effectiveness of the theoretical model and the QQM algorithm are explored via simulation. This evaluation of QQM not only validates the mathematical formulae derived and their underlying assumptions, but also serves as a demonstrator of the QQM algorithm and the significant role it could play in (i) making decisions on the handover of traffic from cellular networks and (ii) the management of traffic once it has been handed over to small cell and future wireless networks.

The methods used to evaluate the efficiency of QQM concentrate on true-to-life, real time traffic scenarios in an infrastructure wireless network. QQM is systematically compared with the original EDCA [11] method via simulation using ns-2 [19] and, when necessary, Matlab [20].

The VoIP traffic model used in the simulations below is that of a constant presence of simultaneous VoIP calls on the network. The Erlang [21] calculation supposes 300 minutes of VoIP traffic per hour, generated by 5 VoiP calls of 5 minutes duration. Therefore, each AP manages at least 5 flows of VoIP traffic, and these are then increased up to a maximum of 10 simultaneous calls per Access Point.

The audio streaming traffic is represented in the simulations by UDP-like packets using Evalvid (http://www.tkn.tu-berlin.de/menue/research/evalvid) [22] [23] [24] [25] [26] in ns-2. It was necessary to extend Evalvid to provide and monitor streaming services in the simulator.

Evalvid was adapted to the last release of ns-2 to include EDCA [27], VoIP [28] and IEEE802.11ac. A few modifications were made: information on UDP packet creation times was added to headers and TCP acknowledgements were moved from AC_0 to the same AC as the TCP data packets

A. Application of the Theoretical Model

One practical application of the theoretical model presented above is in the calculation of the average time needed to transmit a single packet in a future wireless network. This is done numerically by comparing the average number of packets transmitted in one second during a simulation with the expected time needed to transmit the same number of packets as inferred from the theoretical model.

The number of packets transmitted per second and the number of packets per second estimated by the theoretical model are shown to be in close agreement. That means that the theoretical model can estimate the average time to transmit a packet in future wireless networks when the probability of events not captured by the model is low e.g. collisions between more than 2 ACs, channel interupts etc.

The ns-2 simulation results and the average values predicted by the theoretical model are shown in figure 5. The x-axis shows the simulation time, while the y- axis shows the packets per second exchanged between the AP and the wireless nodes. The blue lines are the simulation results and the red dashed lines are the averages.

The lines marked with AC_VO represent the traffic managed by AC_0. The lines representing AC_0 traffic overlap because VoIP and audio traffic is transmitted using a constant number of packets per second. The lines marked AC_VI represent the sum of the traffic managed by AC_1 with the AC_VO traffic.

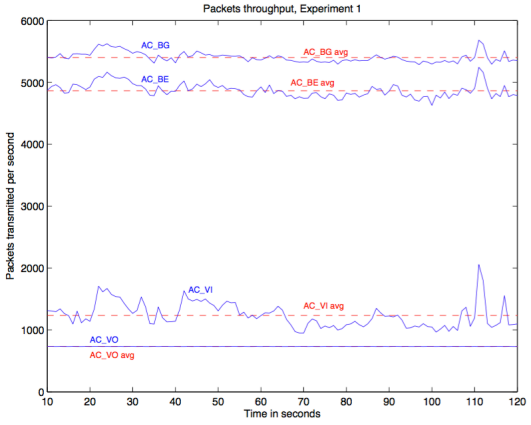

Fig. 5. Comparison of Simulation Results with those obtained using the Theoretical Model for 5 VoIP, 5 Audio and 5 Video streams, 5 TCP flows and a mix of 3 TCP and 2 UDP flows

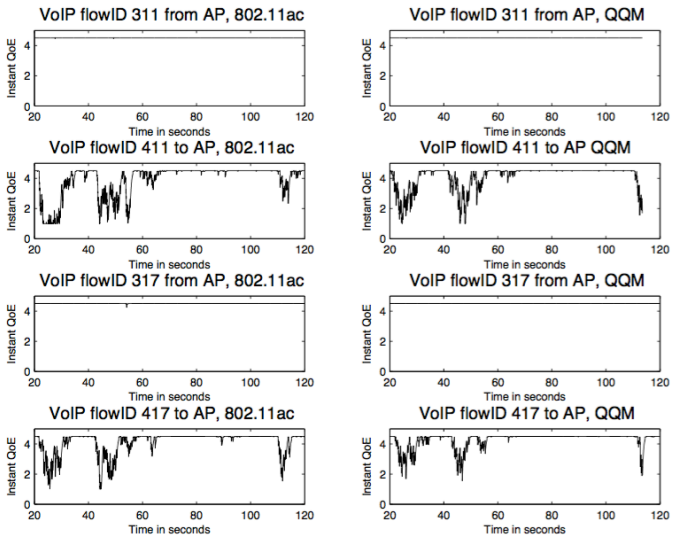

Fig. 6. Comparison of the QoE of VoIP Flows at Wireless Nodes 11 and 17 for two systems: (i) IEEE802.11ac with EDCA and (ii) the same system with the addition of QQM.

All the video traffic flows start at the same instant in time. This represents a worst case scenario for video traffic because all the transmission peaks overlap and so they are amplified. The lines marked with AC_BE represent the sum of the traffic managed by AC_2 and the AC_0 and AC_1 traffic.The lines marked with AC_BG represent the sum of the traffic managed by AC_3 and the AC_0, AC_1 and AC_2 traffic.

The calculations performed using the theoretical model are shown on the graph. All flows managed by AC_0 and AC_1 are transmitted first. After this traffic has been transmitted, the traffic managed by AC_2 is transmitted. This is TCP/FTP traffic. The remaining time is then reserved for the transmission of the traffic managed by AC_3. In general, AC_2s and AC_3s transmit UDP/CBR traffic before any TCP/FTP packets are sent in their respective transmission time slots. In both cases the TCP/FTP traffic consists of data packets in one direction and acknowledgement packets in the opposite direction.

It is assumed that the RTT for TCP/FTP flows exceeds the typical delay accumulated on the wired network; that is, it exceeds 40 milliseconds [29]. This means there are less than 25 RTTs per second. In each RTT a flow can transmit a maximum of 32 packets if the congestion window is large [29]. In the case of Experiment 1, less than 4000 packets are transmitted by the AC_2s. This includes both data and acknowledgement packets. In theory, one of the 5 AC_2s in the simulation can transmit 400 packets. This corresponds to an average congestion window size of 16 packets. It can be assumed that, in theory, each AC_2 is accessing the channel less than 16 times every 40 milliseconds. About 13 packets can be transmitted every 2.5 milliseconds.

A large number of flows are from the wired to the wireless network; so the AP is in saturation and always has a packet ready to transmit. This means that a maximum of 2 AC_2s are competing to access the channel at any instant in time. AC_3 traffic sources all originate on the wired network and converge at the AP; therefore, the only traffic flowing from the wireless to the wired network is TCP/FTP acknowledgement packets. It is assumed that a maximum of 2 AC_3s are competing to access the channel at the same time.

In the experiment, an average of about 5400 packets are transmitted each second. This is shown in figure 5. The theoretical model estimates that it will take 1.0036 seconds to transmit this number of packets.

This means that the average transmission time for each packet estimated by the theoretical model differs from the average time observed in the simulation by 0.36%. The difference is 0.0036 seconds and this corresponds to the time needed to transmit 15 more packets using the theoretical model.

B. QQM in Operation

QQM's efficiency is demonstrated through simulations that compare the performance of future wireless networks where EDCA is implemented with that of the same networks where QQM is implemented. QQM is implemented at the queue inputs and it operates and interacts with the controllers through the eQoS metric. To assist the reader in their understanding of the graphs presented below, the eQoS estimates are translated onto a scale that is comparable with that used for the MOS [30] score. This is indicated on the graphs as QoE.

Figure 6 shows a simulation for the original system, i.e. for IEEE802.11ac with EDCA only, and for the enhanced system where QQM is deployed. Results for the former are on the right hand side of each figure; while those for the latter are on the left side. It can be seen that, in general, the quality of flows from the wireless to the wired network are most affected by the introduction of QQM. Drops in QoE are smoothed and reduced in amplitude by the QQM algorithm through the actions and interactions of the three individual controllers and the dropping of a few flows from the system. After 113 seconds the flow with ID number 411 is dropped along with the related flow with ID 311 and VoIP call 3 is terminated.

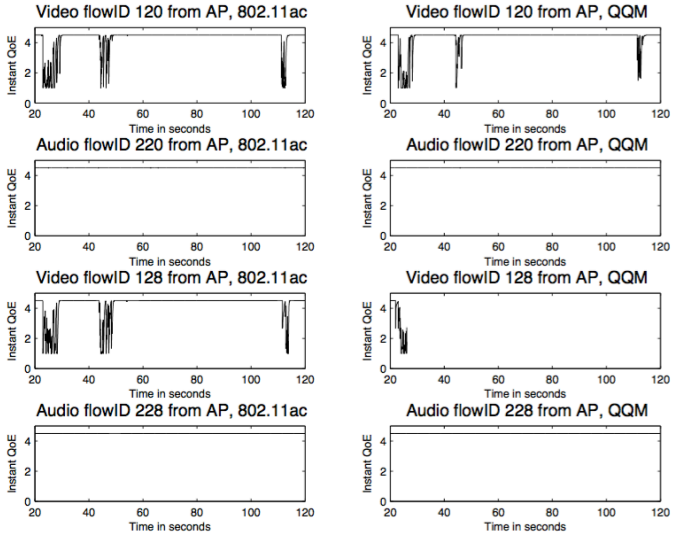

Fig. 7. Comparison of QoE of audio and video unidirectional flows at Wireless Nodes 20 and 28 for two systems: (i) IEEE802.11ac with EDCA and (ii) the same system with the addition of QQM.

Figure 7 shows the comparison between the QoE of the original system, i.e. for IEEE802.11ac with EDCA only, and the same system with the addition of QQM. In the figure the flow with ID 128 is dropped by the QQM algorithm after 26 seconds. Audio quality is not affected in the unidirectional flows from the wired to the wireless network.

VIII. Conclusion

The simulation results show that the theoretical model presented can be used to predict traffic and to estimate the probabilities associated with transmission, collision and idle events for the handover traffic on wireless and small cell networks. They also provided compelling evidence of the effectiveness of QQM in the management of this traffic. The QQM algorithm can be used in handover decision making to ensure that real-time traffic for on-demand services is only passed to small cell and wireless networks when they can provide quality guarantees that meet service requirements.

References

[1] C. V. Forecast, "Cisco visual networking index: Global mobile data traffic forecast update, 20152020 w," *Cisco Public Information, February*, 2016.

[2] "ITU-T recommendation E.800: Definitions of terms related to quality of service," Sep. 2008.

[3] R. Stankiewicz, P. Cholda, and A. Jajszczyk, "QoX: What is it really?" *IEEE Communications Magazine*, vol. 49, no. 4, pp. 148–158, April 2011.

[4] "ITU-T recommendation G.1000: Communications Quality of Service: A framework and definitions," Nov. 2001.

[5] E. Ibarrola, J. Xiao, F. Liberal, and A. Ferro, "Internet QoS regulation in future networks: a user-centric approach," *IEEE Communications Magazine*, vol. 49, no. 10, pp. 148–155, Oct 2011.

[6] "ITU-T recommendation P.10/G.100: Vocabulary for performance and quality of service," Jul. 2006.

[7] G. Pibiri, C. M. Goldrick, and M. Huggard, "Enhancing AQM performance on wireless networks," in *Wireless Days (WD), 2012 IFIP*, Nov 2012, pp. 1–3.

[8] "ITU-T recommendation BS.1387: Method for objective measurements of perceived audio quality," 11 2001.

[9] V. Misra, W. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," *ACM SIGCOMM Computer Communication Review*, vol. 30, no. 4, p. 160, 2000.

[10] "IEEE Standard for Information technology– Telecommunications and information exchange between systemsLocal and metropolitan area networks– Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz." *IEEE Std 802.11ac-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012)*, pp. 1–425, Dec 2013.

[11] "IEEE Standard for Information technology–Local and metropolitan area networks–Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," *IEEE Std 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003)*, pp. 1–212, Nov 2005.

[12] F. Bouabdallah and N. Bouabdallah, "The tradeoff between maximizing the sensor network lifetime and the fastest way to report reliably an event using reporting nodes' selection," *Computer Communications*, vol. 31, no. 9, pp. 1763–1776, 2008.

[13] G. Pibiri, "Quality Queue Management for Future Wireless Networks," Ph.D. dissertation, Trinity College Dublin, 2016.

[14] R. Ruggles and H. Brodie, "An empirical approach to economic intelligence in World War II," *Journal of the American Statistical Association*, vol. 42, no. 237, pp. 72–91, 1947.

[15] "ITU-T recommendation G.729 : Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)," Jan 2007.

[16] S. kak Kwon, A. Tamhankar, and K. Rao, "Overview of H.264/MPEG-4 part 10," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 186 – 216, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1047320305000696

[17] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable H.264/MPEG4-AVC Extension," in *2006 International Conference on Image Processing*, Oct 2006, pp. 161–164.

[18] T. Szigeti and C. Hattingh, "Quality of service design overview," *Cisco, San Jose, CA, Dec*, 2004.

[19] S. McCanne, S. Floyd, K. Fall, K. Varadhan *et al.*, "Network simulator ns-2," 1997.

[20] MATLAB, *version 7.10.0.499 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010.

[21] M. Schwartz, *Mobile wireless communications*. Cambridge University Press, 2005.

[22] J. Klaue, B. Rathke, and A. Wolisz, "Evalvid-A framework for video transmission and quality evaluation," *Computer Performance Evaluation. Modelling Techniques and Tools*, pp. 255–272, 2003.

[23] ——, "EvalVid-A Video Quality Evaluation Tool-set," *Telecommunication Networks*, 2011.

[24] C. Ke, C. Shieh, W. Hwang, and A. Ziviani, "An evaluation framework for more realistic simulations of mpeg video transmission," *Journal of information science and engineering*, vol. 24, no. 2, pp. 425–440, 2008.

[25] J. Gross, J. Klaue, H. Karl, and A. Wolisz, "Cross-layer optimization of OFDM transmission systems for MPEG-4 video streaming," *Computer Communications*, vol. 27, no. 11, pp. 1044–1055, 2004.

[26] A. Lie and J. Klaue, "Evalvid-RA: trace driven simulation of rate adaptive MPEG-4 VBR video," *Multimedia Systems*, vol. 14, no. 1, pp. 33–50, 2008.

[27] S. Wiethölter and C. Hoene, "Design and verification of an IEEE 802.11e EDCF simulation model in ns-2.26," in *Technische Universität Berlin, Tech. Rep. TKN-03-019*, November 2003.

[28] C. Hoene, "Simulating playout schedulers for VoIP-software package," 2004.

[29] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validations," in *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4. ACM, 1998, pp. 303–314.

[30] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, March 2010.