

WIRELESS NETWORKS PERFORMANCE MONITORING BASED ON PASSIVE-ACTIVE QUALITY OF SERVICE MEASUREMENTS

Yazeed A. Al-Sbou

Department of Computer Science, Applied Science University, Bahrain

ABSTRACT

Monitoring of the performance of wireless network is of vital importance for both users and the service provider which should be accurate, simple and fast enough to reflect the network performance in a timely manner. The aim of this paper is to develop an approach which can infer the performance of wireless ad hoc networks based on Quality of service (QoS) parameters assessment. The developed method considers the QoS requirements of multimedia applications transmitted over these kind of networks. This approach is based on the ideas of combination of both active and passive measurement methods. This approach uses an in-service measurement method in which the QoS parameters of the actual application (user) are estimated by means of dedicated monitoring packets (probes). Afterwards, these parameters are combined to produce and assess the application's overall QoS using the fuzzy logic assessment and based on the measured QoS parameters estimated using the probe traffic. The active scheme is used to generate monitoring probe packets which are inserted between blocks of target application packets at regular intervals. While the passive monitoring is utilized to act as a traffic meter which performs as a counter of user packets (and bytes) that belong to the application (user) traffic flow that is subjected to monitoring. After simulating the developed technique, it offered a good estimation for the delay, throughput, packet losses and the overall QoS when using different probe rates.

KEYWORDS:

Quality of Service, network monitoring, passive measurement, active measurement, fuzzy logic.

1. INTRODUCTION

As communication networks have become a very essential part of our life and due to the recent growth of real and non-real time multimedia applications as well as the high demand for the quality of these applications, many efforts were made towards improving their performance. These are in order to achieve customer satisfaction which lead to strong loyalty and therefore to more profit for the service providers [1]. Multimedia applications QoS can be guaranteed by expanding the bandwidth, but this is not always possible, costly and cannot remedy the root problem. Consequently, managing and controlling the available network resources are the points to deal with to solve these kinds of problems. These can be achieved only by measuring and monitoring the network/application QoS. One of the main motivations behind deployment of the QoS is the increasing multimedia application requirements with limited resources and limited QoS support in IP networks [2]. Therefore, QoS assessment is an essential element for satisfying different services requirements for number of applications that are sharing the same infrastructure [3]. In addition, these measurements are important for the network managers to diagnose network problems and failures, optimize the network performance, and ensure that the offered services meet service providers and end-users satisfactions.

QoS provision is a technique that generally consists of: a measure of network/application QoS state and a way to observe it and a heuristic that uses the information to deliver a QoS objective. Hence, to provide or guarantee QoS, it should be monitored firstly [4] [5]. For networks and especially in wireless ad hoc, QoS provision is not an easy issue. Therefore, many approaches have been proposed to deal with this issue [6]. These approaches include call admission protocols that first assess whether a flow should be admitted into the network based on its QoS status [7], [8], routing protocols that attempt to control the flow of traffic through sections in the network that can best afford it with acceptable QoS [9], [10], queuing schemes implemented at nodes [11], medium access schemes which give access priorities to some applications to and reserve the Radio Frequency (RF) media [12], [13]. All of these schemes must perform QoS assessment before and after applying the proposed approach to evaluate and then enhance the application/network performance.

Different multimedia applications have different QoS requirements [14]. The specific parameters which define QoS vary depending on the application and user requirements [15] [16] [17]. It is very important to determine the correct set of accurate QoS parameters for the particular media being transported; otherwise QoS guarantees cannot be obtained [18]. QoS of transmitted application through a network is characterized, in a very general way, by four key network parameters (metrics): one-way delay [19], one-way jitter (delay variation) [20], packet loss ratio [21], and bandwidth. Together, these parameters determine the QoS the traffic requires [1] [22].

Monitoring and measurement schemes usually fall into two categories: passive and active methods. The former are those based on (transparently) collecting and analyzing the traffic observed at a certain point of the network and the latter, which is based on injecting synthetic traffic flow into a network. Passive measurement is a traditional technique used to obtain measurements of QoS parameters related to a certain network element [23], [24]. This method is based on monitoring the performance of packet streams through a network by tracking the traffic passing by a measurement point without creating or perturbing it. So the packet's statistics can be gathered without adding any new traffic. This can be done by collecting traffic flow data, from routers, switches or end-point hosts. Another method, for traffic collecting, is implemented by adding a stand-alone server at the location of interest (e.g., core or edge) of the network, which acts as a traffic meter or a monitoring device by storing information about the crossing traffic. Therefore, this type of measurement methods acts as an observer inside a network and usually will not interfere with other traffics. The levels of details and accuracy of the information gathered at the measurement points depend upon how much metrics are being processed and the volume of traffic passing through the monitoring device.

Another way of measuring the network performance is the active measurement. This method is becoming increasingly important due to its simplicity, ability to attain end-to-end measurements, and freedom from the need of getting into the core of network. In this method, QoS and the performance of a network are measured by injecting of some artificial probing packet streams into the network and monitoring them from a source to a destination. Active measurements can determine the QoS experienced by the probe flow for a particular path and then measure the QoS as it is seen by applications. The purpose of these probing packets is to provide some insight into the way the user traffic is treated within the network. The QoS and performance of the probe-packet stream are monitored to infer the performance of the user's packets and the network directly. There are several tools which are based on active methods like, the Internet Control Message Protocol (ICMP) Echo Reply/Request messages (ping) which is defined in RFC 729 [25], traceroute [26], Surveyor [27]. The basic components of an end-to-end active probing structure consists of two entities: probe sender and probe receiver. In each probing experiment, the sender generates and transmits a probe stream, which traverses some route in the network and terminates at the receiver (the sink). Together with the probe sequence numbers available from

the payloads, the packet arrival and departure timestamps define the raw outcome of the experiment [28]. They are recorded by the sender monitor and the receiver monitor, respectively. By selecting particular properties at the sender (like packet size, departure time, bit rate, etc.), it is potential to compute metrics by analyzing the probe flow characteristics (e.g. arrival time) at the destination so, one can determine end-to-end metrics (from the source to the destination) [29].

Active measurements have several advantages. Among these is the flexibility to create probe flow with specific features to match measurement needs. These features include the packet sizes, types, and inter-departure times. Moreover, active measurements include reduction in the quantity of resulted measurements compared with the passive measurements. However, the main disadvantage of active measurements is their invasive nature [28]. The probe packets used for the measurements will perturb the network and the user traffic QoS metrics. Passive measurements overcome the disadvantages of active measurements with regard to the overheads and delay by monitoring (probe) streams. In addition, it can provide more precise performance evaluation of user traffic than active measurements. That is because in passive monitoring, the actual user traffic packets themselves are measured rather than depending on results of probe packets.

To overcome some of the disadvantages of both active and passive approaches, several studies were carried out. These studies were based on combination of active and passive methods. One of these methods is the Change-of-measure based active/passive monitoring (CoMPACT) [30] [31]. This is a light active measurement method transformed by using passively monitored data to correct the probe results to be closer to the actual user performance. This method was only applied to estimate the actual user delay. Another technique has been proposed which combines passive and active ways [32] [33] from a probe report. In this technique, a router sends active probe packets at regular intervals. The passive monitoring method is used to count the number of user packets passing through the router. This approach has been used to estimate the QoS parameters only (i.e. delay, packet loss... etc.) over wired networks.

The aim of this paper is to evaluate both active and passive monitoring methods to infer the performance of wireless ad hoc networks by considering the QoS requirements of multimedia applications. In addition, this study tries to overcome the disadvantages of both techniques and introduces a new approach to assess network performance by combining the advantages of both of them. This approach uses an in-service measurement method in which the QoS of the actual application (user) is estimated by means of dedicated monitoring packets (probes) [37]. Afterwards, these parameters are combined to produce and assess the application's overall QoS using the fuzzy logic assessment and based on the measured QoS parameters estimated using the probe traffic. Therefore, the contribution of this work is represented by adding the process of the overall QoS assessment to the system utilized in [33].

This paper is organized as follows: Section 2 describes the monitoring approach description and the experimental simulation set up and settings. Section 3 presents the experimental results. Section 4 provides a conclusion of the paper.

2. QoS PARAMETERS MEASUREMENT APPROACH

The aim of this work is to propose a monitoring system that can measure QoS parameters (delay, delay variation (jitter), packet loss and throughput) and the overall QoS based combining both active and passive monitoring methods. In this paper, active approach is used to produce probe packets to be injected among user application traffic at regular intervals at predetermined period of times. Using the injected packets, delay and the jitter of the user application are inferred. On the other hand, the passive approach is employed as a traffic monitor to count the application user packets (or bytes) subject to monitoring. This combination is applied also to estimate the packet

loss ratio and the throughput of the user application traffic. This means that packet loss ratio and throughput are passively measured depending on the active probes position in the monitoring block. Probing packets are periodically injected by the sending end-node every M number of user packets. M is the number of user application traffic packets between two successive probe packets.

Here, it is proposed that the sending node is used generate both: the user application traffic and the probing packets. These packets are injected within the user traffic regularly to gather useful QoS information. For every monitoring packet generated, the sending node inserts the number of user packets sent so far and the timestamp at which this monitoring packet. At the receiving end-node, the receiver needs to set a counter for the number of the received user packets. In addition, it should: detect the monitoring packets, place a timestamp in every monitoring packet which shows the current time at the receiving end, and insert the current value of the counter that keeps track of the cumulative number of the received user packets.

Based on this process, every probing packet should have, a sequence number, sending and receiving timestamps and the number (cumulative) of sent and received user traffic packets of every block. Consequently, the difference between the sent and the received packets in the same monitoring block is the number of lost packets in that block. Lost monitoring packets are detected by the missing sequence number. If a probing packet is lost, the monitoring block will be extended up to the next probing packet that succeeds to arrive at the receiving node. In addition, a sample of the packet delay between the sending and the receiving nodes is calculated by the difference between the sending and the receiving timestamps of the probing packets. Jitter is calculated from the delay results by calculating the difference between successive delays of every probing packet. After measuring these parameters (delay, jitter and packet loss ratio) they are fed to the fuzzy system to assess the user application QoS using the same procedures discussed in [35] [36].

Based on the proposed approach, it is expected to obtain the following measures: samples of the packet delay and jitter between the sending and receiving nodes, if the packet size is known, it is possible to estimate and monitor the throughput of the user application between monitoring packets, the packet loss ratio of the user application between the sending and receiving nodes for each monitoring block, the length of the loss free periods and loss periods expressed in terms of the number of consecutive monitoring blocks that does not contain lost packets and the number of monitoring blocks that contain lost packets, respectively. In addition, samples of the estimated QoS values of the user application based on the estimated QoS parameters resulted from the probe measurements of each monitoring block, and the length of the Good, Average and Poor QoS periods expressed in terms of the number of consecutive monitoring blocks that contains Good, Average and Poor QoS values.

For illustration of the effectiveness and efficiency of the proposed approach, Network Simulator (NS-2) Software was used to simulate the adopted network topologies. Wireless ad-hoc network was employed where the nodes arranged in random positions and the arrangement was made in such a way that it satisfied the single hop condition with an area of (250m x 250m). The traffic characteristics are illustrated in Table 1 with 500 second simulation time. As an example of multimedia applications videoconferencing was used. The network used in the simulation had six pairs of fixed source/destination [34] [35]. One of the pairs is used for videoconferencing application transmission and the others were used for the cross-traffic. The probing packets were constant bit rate (CBR) packets transmitted using the UDP protocol with a packet size of 64 bytes. The rate at which monitoring packets were sent is important. Too few packets result in inefficient and inaccurate results and too many results in the network traffic of being disturbed. Therefore, in order to examine the effect of probe rate on the QoS assessment, several probing

rates were used ranging from low to high probe rates. Probe packets were transmitted periodically with monitoring block sizes (M) between the probe packets. M was selected to be 375, 186, 93, 47, 31 and 25 packets (i.e. ratio between probe and traffic packets is $1/375$, $1/186$, $1/93$, $1/47$, $1/31$ and $1/25$). All simulation runs were repeated several times by using several seed values for the random number generator of the NS-2 simulator. The resulted values of the different runs of the same simulation have been averaged to get the actual values. In addition, each simulation was run twice for each seed; once with probe switched on and once with probes switched off. This allowed for testing the effect of the probe presence on the user and network behavior. Over the simulation time and in order to examine the probe measurement results with different network conditions, the network was subjected to three different situations: light load (0-170 sec), medium load (171-330 sec) and fully loaded (331-500 sec).

Table 1: Network traffic characteristics.

Traffic type	Packet Size [byte]	Generation Rate [Kbps]
Videoconferencing	512	384
Background traffic1	400	300
Background traffic2	370	360
Background traffic3	420	330
Background traffic4	350	300
Background traffic5	600	450

3. EXPERIMENTAL RESULTS AND DISCUSSIONS

In order to demonstrate the effectiveness of the proposed monitoring, various probing rates of the monitoring packets (i.e. the length of monitoring block) were tested. The network performance was evaluated in terms of one-way delay and jitter, packet loss ratio, throughput and finally the overall assessed QoS. To evaluate the effectiveness of the proposed procedure, two issues are needed to be considered. Firstly, to assess how accurate the probe results are and secondly, to know how much the application user traffic and its QoS are being perturbed by these probing traffic (biasness).

3.1. One-way Delay and Delay Variation

Figure 1 illustrates how the one-way delay varies during the measurement period for both application user and probe traffics. In addition, Table 2 summarizes the actual and the estimated values (mean, maximum, minimum and standard deviation) for the delay and jitter for two different monitoring block sizes. As examples, two monitoring block sizes (i.e. probe rates) were used to compare the results of both traffics: 25 packets and 375 packets block sizes. From Figure 1, it can be seen that the probe result of the one-way delay samples the user delay with an acceptable accuracy over the three network situations. As can be seen from the Figure 1, delay values increase when a high background traffic load is offered. That is because both probe and user traffic packets experienced the same network conditions and increasing the probe rate will produce high number of samples which will provide higher precision. So, increasing the probe rate has resulted in reducing the absolute error as can be observed in Table 1. These samples indicate that the measurements based on the monitoring packets can give fairly good estimates of the average delay and its variation.

Moreover, Figure 2 below show the histogram distributions of the one-way delays for the actual traffic and for the probe traffics of $M = 25$ and $M = 375$ block sizes during the measurement period. From these histograms, it is clear that the one-way delay distribution of the $M = 25$ is

quite similar to the actual user delay distribution. This means that both delays have similar measurement results which is more accurate than the $M = 375$ results. Nevertheless, for both monitoring blocks, about 40% of the measured delays were less than 40msec which is also identical to the actual delay.

Table 2: The actual values for one-way delays and delay variations and the estimated values for block sizes using $M = 25$ and 375 packets.

Units: [msec]	Actual values	$M = 375$	$M = 25$
Mean delay	335.7	328.8	334
Absolute error	-----	6.9	1.7
Delay St. Dev.	331.7	327.3	315.6
Maximum delay	1915.7	1274.6	1586.9
Minimum delay	2.5	0.78	0.72
Mean jitter	6.4	100.2	42.3
Absolute error	-----	93.8	35.9
Jitter St. Dev.	15.6	136.5	73.5
Maximum jitter	727.9	622.5	912.3
Minimum jitter	0	0.04	0.004

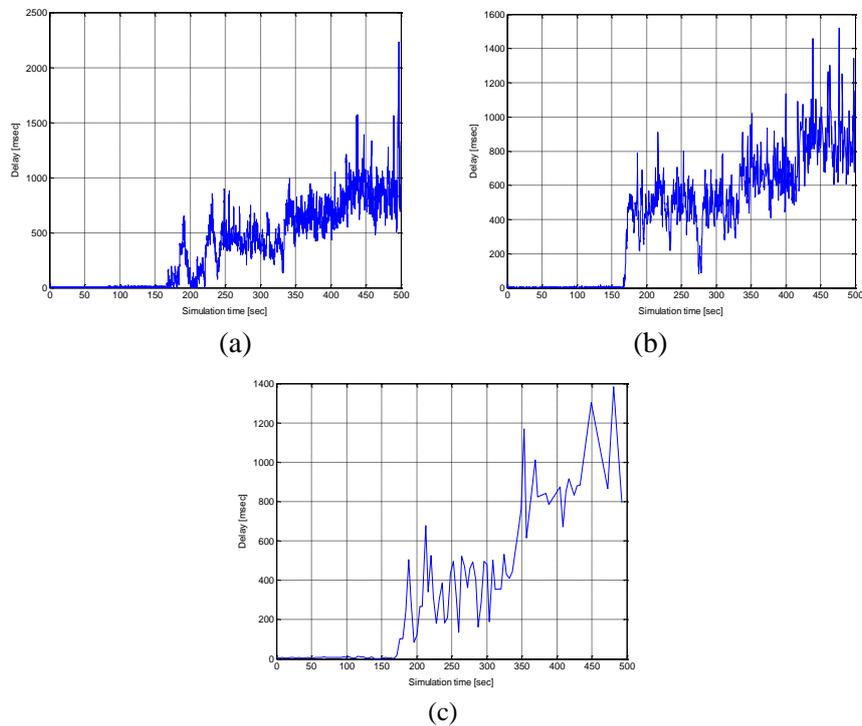


Figure 1: One-way delay of the: (a) actual traffic, (b) Monitoring traffic of $M = 25$ packets and (c) Monitoring traffic of $M = 375$ packets

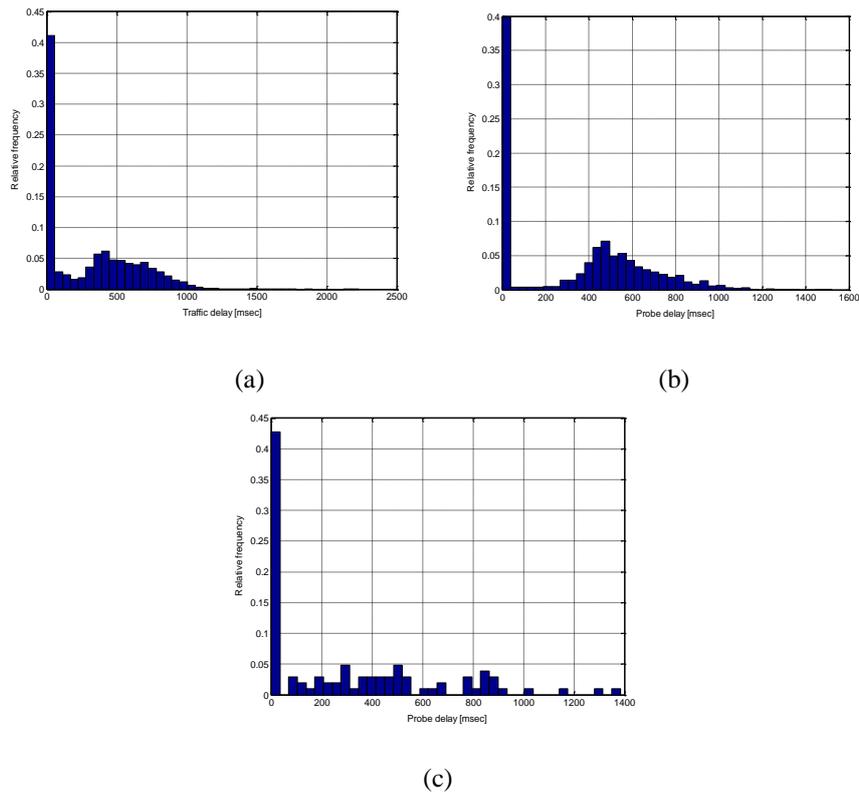
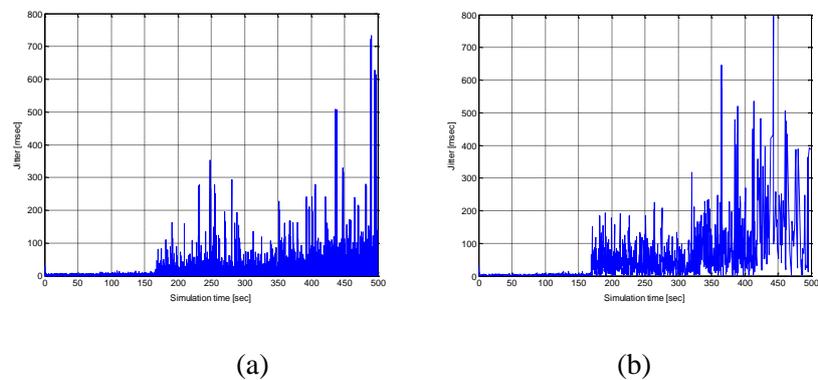


Figure 2: One-way delay distribution of the: (a) actual traffic, (b) Monitoring traffic of $M = 25$ packets and (c) Monitoring traffic of $M = 375$ packets

Figure 3 illustrates the delay variation (jitter) during the measurement period for two different block sizes. It shows that the probe jitter during the lightly loaded network situation gives a reasonable representation of the user traffic jitter. On the other hand, as the network load is increased, the probe result overestimates the user delay variation. So, the probe jitter is higher than the traffic jitter over a congested or partially congested network. That is because the more loaded the network, the higher the contention between the nodes. A partially or fully loaded network will increase the probe delay in a significant amount compared to the delay a probe experiences when it encounters an empty network situation.



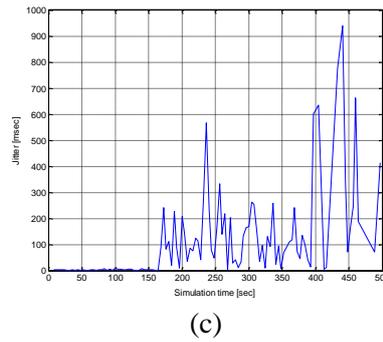


Figure 3: One-way delay variation (jitter) of the: (a) actual traffic, (b) Monitoring traffic of $M = 25$ packets and (c) Monitoring traffic of $M = 375$ packets

The variation of the jitter is due to the contention between the sending nodes for the available resources of the network. This contention will enforce the nodes to defer their transmissions for some time like Short Inter Frame Space and DCF Inter-frame Space (SIFS and DIFS) therefore, these packets will be queuing during the busy times of the network channel because it was occupied by some other nodes. The deferral of transmitting some packets will cause some variations in the delays of the consecutive probe packets. A probe packet that goes through a less busy condition may be followed by a high contention period which is met by the next probe which will experience more delay. The extreme difference in delay experienced by these probes will result in a higher jitter. The user traffic does not have this problem as the probe traffic because the next packet is more likely to be in the same burst. Therefore, the difference in delay between subsequent user packets is minimal, resulting in a lower jitter for the user traffic. In addition, from Table 1, the estimated jitter measurement values of the probe traffic are higher than the actual user values. However, increasing the probe rate reduced the difference between the two measurements. This is because increasing the probe rate increases the samples number that is in the same network condition which will provide more reasonable results for the probe traffic.

Figure 4 shows histogram distributions of the delay variation for the actual user traffic and for the monitoring packets using $M = 25$ and 375 block sizes. These diagrams reveal that there are some discrepancies between the actual and the estimated delay variation measurements. It is apparent that the actual user traffic jitter is lower than the probe jitter. These discrepancies decrease as the monitoring block decreases. For the actual user, more than 90% of the jitter values were less than 20 msec. Whereas, 58% and 46% of monitoring packets of $M = 25$ and 375 blocks had jitter less than 20msec, respectively. The reasons behind these discrepancies have been discussed earlier in the previous paragraph.

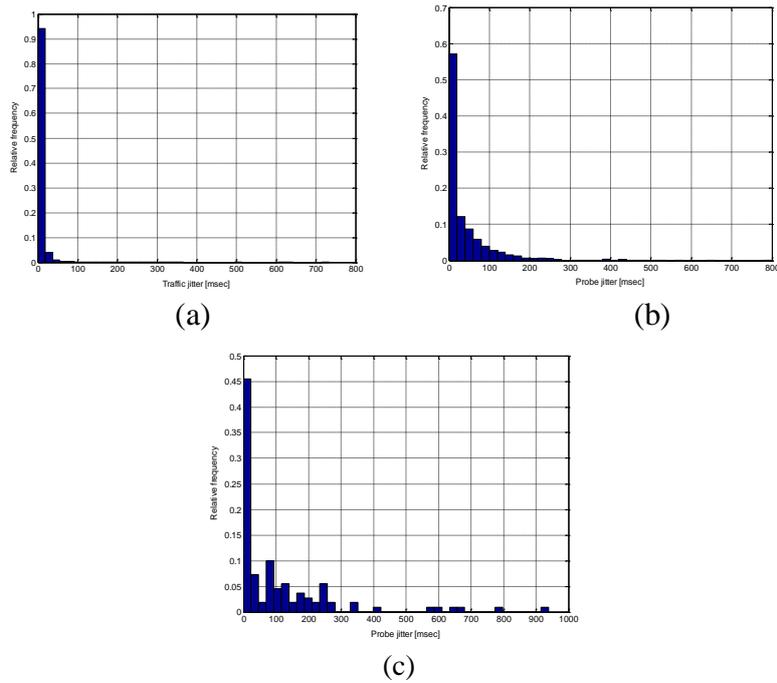


Figure 4: One-way delay variation distribution of the: (a) actual traffic, (b) Monitoring traffic of $M = 25$ packets and (c) Monitoring traffic of $M = 375$ packets

3.2. Packet Loss

Unlike one-way delay or delay variation, packet loss estimation does not rely on sampling techniques (monitoring packets) directly. Packet loss is estimated based on providing a loss ratio for each monitoring block since the number of sent and received packets are counted and sent in the monitoring packets. One advantage of using monitoring packets is that the loss process calculation can be expressed with a higher resolution rather than the long-term average for the total measurement period. The resolution of these results depends on the ratio of the monitoring packets and the user traffic packets (M). In addition, this feature can be used to define periods that contain lost packets (loss periods) and those without losses (loss-free periods) and their lengths.

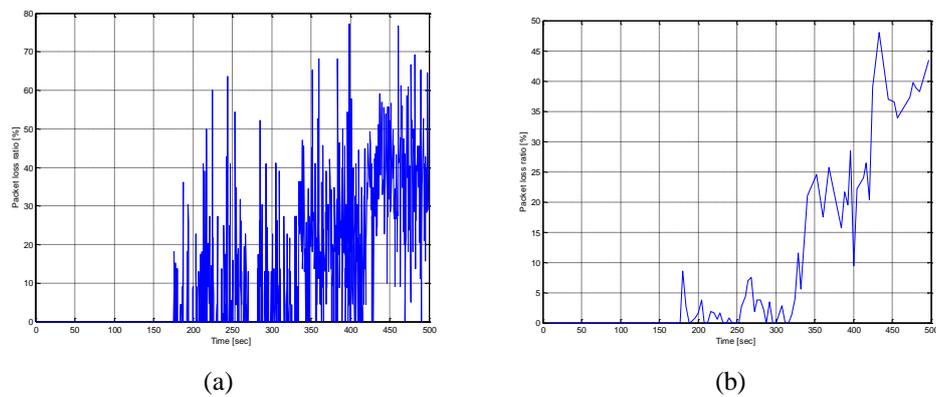


Figure 5: The packet loss ratio using: (a) $M = 25$ and (b) $M = 375$

The estimated packet loss ratios using monitoring blocks of $M = 25$ and 375 are shown in Figure 5. These figures exhibit the degree of accuracy of the achieved resolutions in losses estimation over the measurement period. The smaller the monitoring block size the higher the loss resolution. The distributions of the loss ratio in the monitoring blocks are shown in Figure 6. This figure confirms that the required resolution depends on the monitoring block size.

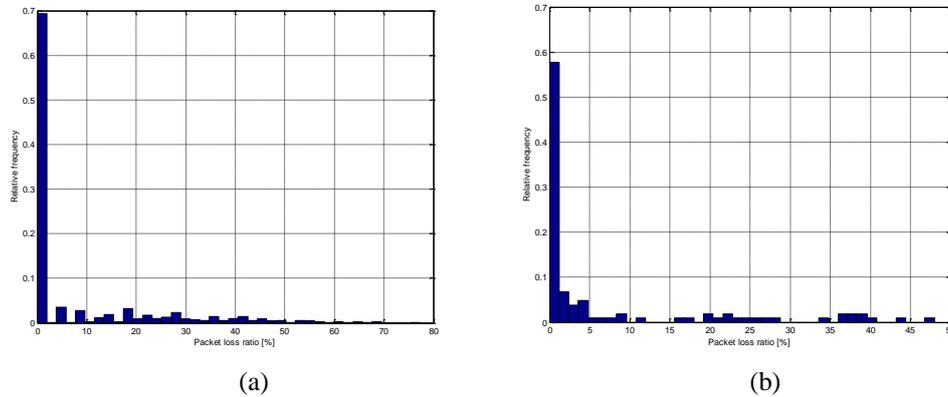


Figure 6: The packet loss ratio distributions using: (a) $M = 25$ and (b) 375

Table 3 summarizes the mean, minimum and maximum lengths of the loss and loss-free periods expressed in time units. The loss rate may not be sufficient enough to signify bursty losses. This type of representation provides information about the length of consecutive packet loss period distribution and about the bursty nature of the packet losses. This length is determined by the difference between the arrival timestamps of the monitoring packets. Loss-free period is computed in terms of the number of successive monitoring blocks that do not contain lost packets. This period is the time difference between the first monitoring packet and the last monitoring packet. The same principle is applied to calculate the loss periods. From Table 3, it is noticeable that as the block size increases, the mean, maximum and minimum of the loss and loss-free periods increases. Generally, it can be noticed that during the measurement period and over all the network situations, the network was lossless because the loss periods were very short compared to the loss-free periods for the whole monitoring blocks.

Table 3: Loss and loss-free period's length measurements based on two different monitoring blocks

Monitoring Block	Loss-free period length [msec]			Loss period length [msec]		
	Mean	Minimum	Maximum	Mean	Minimum	Maximum
$M = 375$	94.3	7.8	283.4	10.7	5	16.3
$M = 186$	46.8	4.3	259.9	3.8	2.4	6
$M = 93$	30.6	1.2	251.7	2	0.8	4
$M = 47$	12.3	0.5	200.8	1.3	0.4	2.9
$M = 25$	6.3	0.2	182.4	0.72	0.13	3

Another powerful representation can be obtained using the loss periods. This is illustrated in Figure 7, which characterizes the length of the loss and loss-free periods (in seconds) for $M = 25$ and 375 . It is clear that the $M = 25$ monitoring block provides more details of the loss and loss-free periods variations than the $M = 375$ monitoring block. In addition, this representation is capable of showing how many loss and loss-free periods have taken place during the measurement period. Monitoring block ($M = 25$) shows that there were 76 loss-free periods and

18 loss periods over the monitoring period. While the monitoring block ($M = 375$) shows that there were 6 loss-free periods and 2 loss periods over the same measurement period. The ratio between the loss-free time and the total measurement period is 83.3% for the $M = 25$ and 80.8% for the $M = 375$ monitoring block. Whilst the ratio between the loss time and the total measurement period is 2.8% for the $M = 25$ and 7.3% for the $M = 375$ monitoring block.

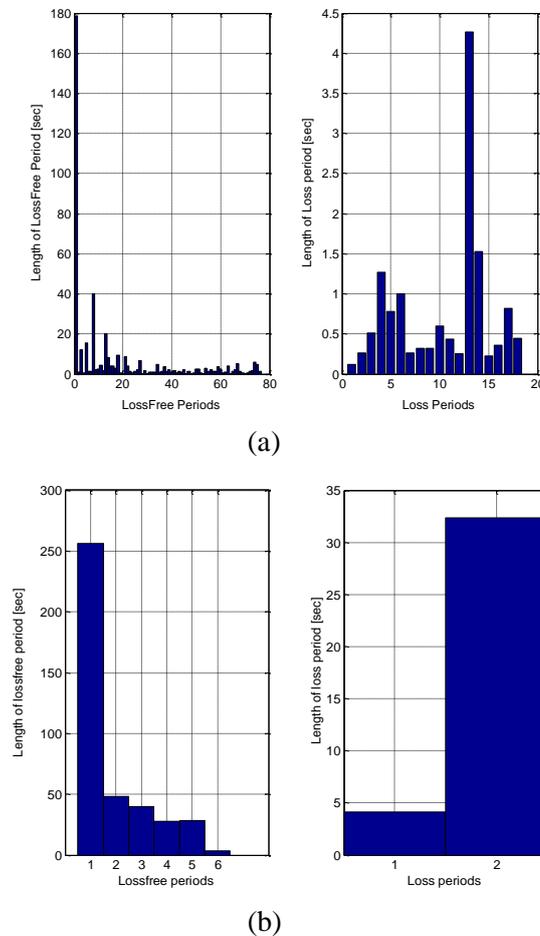


Figure 7: the length of loss and loss-free periods versus time during the measurement period: (a) $M = 25$ and (b) $M = 375$

3.3. Throughput

Using the monitoring block concept and in addition to the long-term average of the utilized capacity (throughput) for an application, it is often useful to obtain the maximum and the minimum values as well as the variation during the measurement period. This can be calculated since the packet size, the number of the sent and received packets along with the timestamps are available for each monitoring block. This throughput is calculated between two monitoring packets in Kbps using the following equation:

$$Throughput = \frac{8 * PS * N}{1024 * (Timestamp(i) - Timestamp(i - 1))} \quad \text{Eq 1}$$

Where PS is the actual traffic packet size in byte, N is the number of packets between two monitoring packets and i is the current monitoring packet.

In Table 4, the average, maximum, minimum and standard deviation of the throughput per monitoring block are presented for several monitoring block sizes. The estimated average throughput is in the range of 345-355 Kbps for all values of monitoring block sizes. In this case, the estimated maximum throughput increases when the monitoring block size decreases. On the other hand, the minimum throughput decreases as the monitoring block size decreases. Moreover, the standard deviation increases when the block size decreases. The reason for this is that reducing the block size increases the number of samples. This in turn increases the throughput within the different network load situations over the monitoring period. The estimated throughput values vary between large and small values resulting in an increase in the standard deviation.

Table 4: The actual throughput estimations based on different monitoring blocks

Units: [Kbps]	$M = 375$	$M = 186$	$M = 93$	$M = 47$	$M = 25$
Average throughput	355.6	355.5	355.6	354	345.9
Maximum throughput	423	460.6	518.1	656	705.5
Minimum throughput	207.4	177.4	140	124	62.3
Throughput St. Dev	53.32	54.5	56.3	61.6	70.7

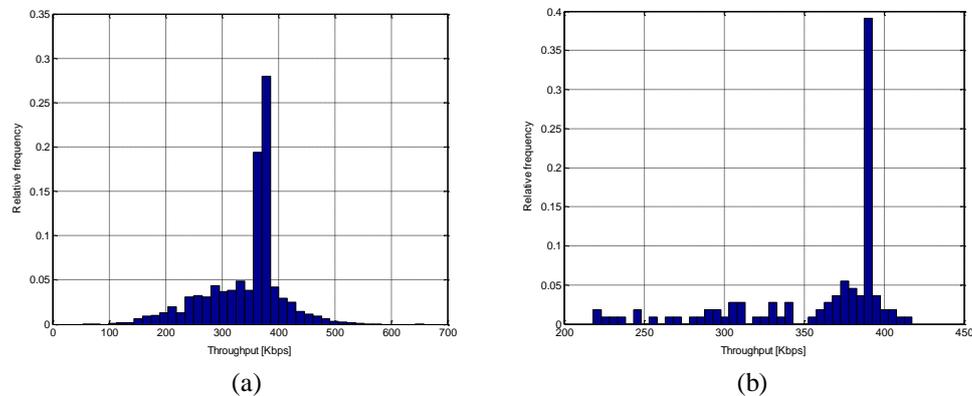


Figure 8: The throughput distributions based on monitoring block of: (a) $M = 25$ and (b) $M = 375$

Figure 8 depicts the distribution of the throughput, per monitoring block for $M = 25$ and 375 . The distributions provide an accurate estimate of the actual throughput (384Kbps) as most of the estimated throughput values are distributed around this value. It is clear that the resolution produced by the $M = 25$ block size is more than that of $M = 375$. So, the desired estimated throughput resolution will be dependent on the required accuracy.

3.4. QoS

The most important QoS parameters that affect the videoconferencing performance are the delay, delay variation and the packet loss. These parameters can be estimated (as described earlier) by probing the network. Delay and delay variation can be taken (estimated) directly from the probe traffic and packet loss is estimated using the monitoring block concept. After measuring these parameters, they were fed to the fuzzy system to produce the estimated overall QoS of the videoconferencing application based on the results obtained from the monitoring packets. In addition, and in order to check the accuracy of the estimated overall QoS result, these parameters were measured for the actual user with the probe traffic switched off. The actual traffic

parameters were averaged using the blocking technique for $M = 25$ and $M = 375$ packets. Fuzzy system outputs of the estimated QoS using the probe and the actual user overall QoS are shown in Figure 9 (a)-(c).

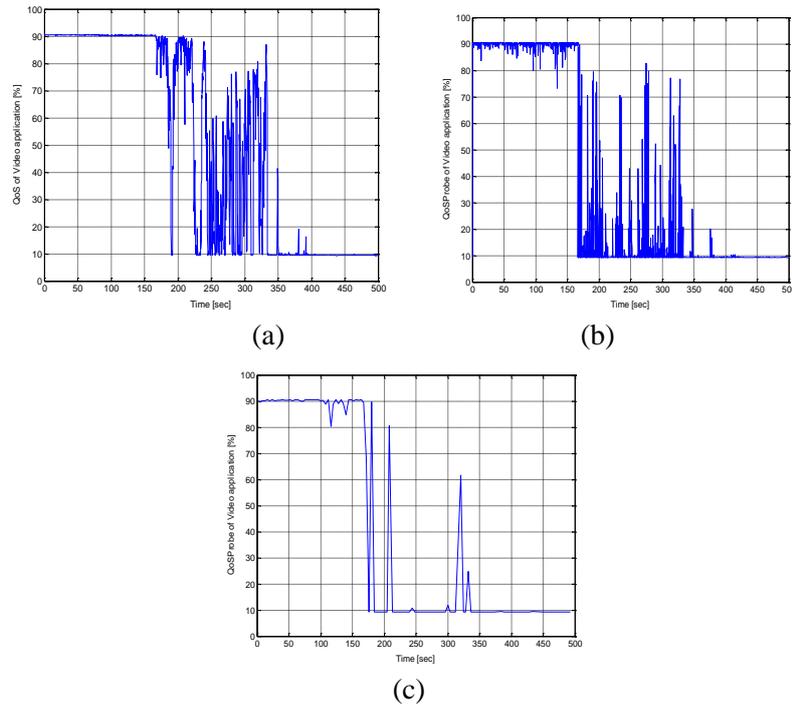


Figure 9: Measured overall QoS of the: (a) actual traffic, (b) Monitoring traffic of $M = 25$ packets and (c) Monitoring traffic of $M = 375$ packets

It can be observed from Figure 9 that the QoS of the monitoring probe packets can infer the actual user overall QoS during the periods of light and heavy loaded network situations. On the other hand, during the partially loaded state, the probe QoS could not estimate the actual user QoS especially when using the $M = 375$ monitoring block size. However, the probe gave a better estimation of the actual overall QoS using the $M = 25$. This means that the QoS estimation was affected by the probe rate (i.e. number of samples). In addition to that, the poor QoS estimation was, mainly, due to jitter. As the network load is increased, the jitter will increase and in particular the probe jitter as explained earlier. The probe jitter will be higher than the actual traffic jitter. Occasionally the probe jitter will exceed the limits of the required QoS while the actual traffic jitter may stay within these limits. Due to this, the probe QoS will underestimate the actual traffic QoS and especially during the partially loaded situations because during the heavy loaded state periods both the probe and the traffic parameters will go beyond the required values and so the overall measured QoS will be poor.

Table 5 illustrates the long-term statistics (mean, standard deviation, maximum and minimum) that characterize the overall QoS values for the actual user traffic and the estimated values using different monitoring block sizes. This table reveals that as the monitoring block size increases the estimated QoS is enhanced compared with the actual QoS value. Increasing the block size will provide more samples to be evaluated using the fuzzy system which will monitor the network more accurately. The estimated overall QoS standard deviation, maximum and minimum are mostly the same as the actual values. This means that the long-term average QoS estimation using monitoring packets is a good approximation of the actual QoS.

To compare the levels when the overall QoS was poor, average and good, for both the actual and the probe traffics ($M = 25$ and 375), a bar chart distribution was used. The length of the bar was representative of the percentage of each QoS case. Figure 10 shows the bar charts of both application's overall QoS. Monitoring traffic using $M = 25$ was closer to the actual overall QoS regions. That was due to the fact that the network was subjected to more assessments over the measurement period using this rate which will result in a higher precision in the QoS estimation than the $M = 375$ probe rate.

Table 5: The actual and the estimated values for overall QoS using different block sizes M

units: [%]	Actual values	$M = 375$	$M = 186$	$M = 93$	$M = 47$	$M = 25$
Evaluated QoS	52.74	40.94	40.51	42.14	43.24	44.51
Absolute error		11.8	12.25	10.6	9.5	8.25
QoS Std. Dev	37.09	38.96	38.49	38.79	38.84	38.66
Maximum QoS	90.48	90.52	90.52	90.52	90.52	90.52
Minimum QoS	9.30	9.27	9.27	9.27	9.27	9.27

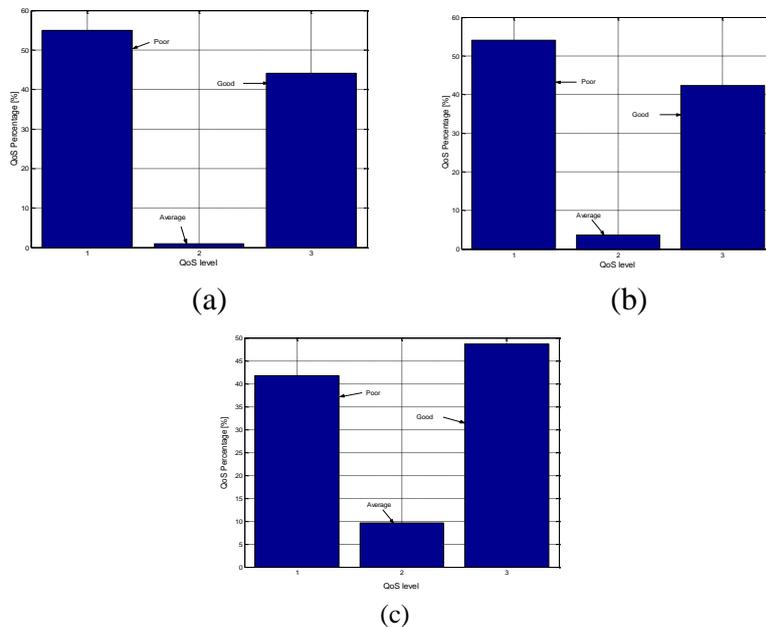


Figure 10: The overall QoS bar chart for: (a) Monitoring packets using $M = 375$ packets, (b) Monitoring packets using $M = 25$ packets, (c) actual traffic

In order to quantify how much the overall QoS of each application was; poor, average or good, the variation of these values, mean and standard deviation were calculated. Table 6 illustrates these statistics that characterize each region of each the traffic overall QoS values for the actual user traffic and the estimated values of $M = 25$ and 375 monitoring block sizes. Table 6 exhibits that the probe rate of $M = 25$ had better QoS approximation of the actual overall QoS because all of its estimated statistics are closer to the actual values.

Table 6: Statistics of actual and estimated overall QoS region for $M = 25$ and 375

Units: [%]	Actual values			$M = 375$			$M = 25$		
QoS	Poor	Average	Good	Poor	Average	Good	Poor	Average	Good
Mean	11.8	51.9	88.1	9.8	38.8	89.7	10.5	46.2	89.5
Std. Dev.	5.4	9.9	5.2	2.6	0	2.2	4	7.8	2.5

So as to obtain a more specific picture about the actual and the estimated overall QoS for each application without classification of the QoS values into good, average and poor regions, probability distribution functions have been generated of each QoS. These distributions are shown in Figure 11. This figure illustrates the cumulative distributions, $\Pr\{X < a\}$, where the random variable X denotes the end-to-end QoS. The usefulness of this method stems from the fact that it gives the percentage that the QoS is less than any preselected threshold value (a). Using these types of distributions, for example, it is very easy to assess the probability of the QoS. In addition to that, it can be observed that the minimum and maximum values of the QoS can be found from these figures. It is apparent that the monitoring packets could, to some extent, estimate the actual QoS cumulative distribution. For example, it can be seen from the figures that it is very easy to assess the probability that the QoS was less than 40%. It is from the actual traffic 0.47, 0.57 and 0.55 using the monitoring traffic of $M = 25$ and 375 respectively. In addition to that, it can be observed that the minimum and maximum values of the QoS can be found from these figures. The minimum value for both traffics (actual and monitoring) was 9.3%. The maximum value for the actual traffic was 90.5% and 90.5% for $M = 25$ and 375 probe traffic.

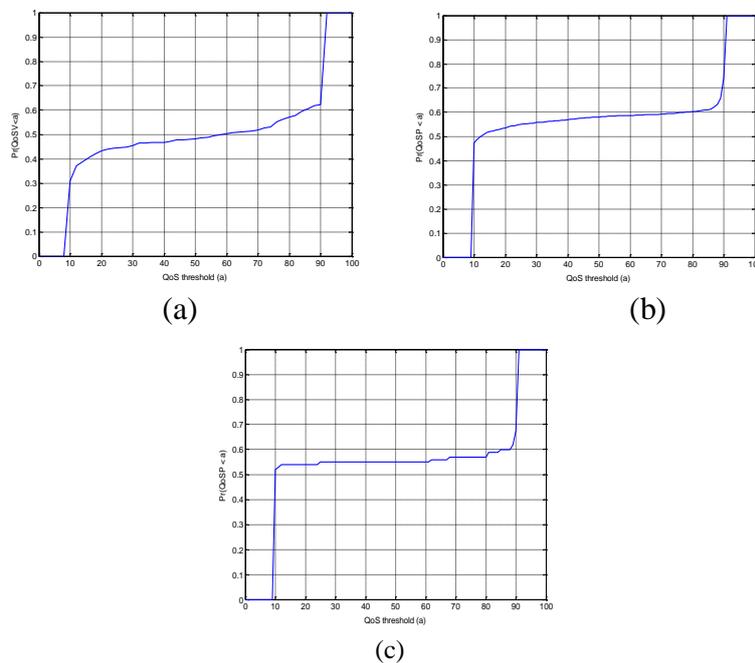


Figure 11: The overall QoS distribution for: (a) actual traffic, (b) Monitoring packets using $M = 25$ packets, (c) Monitoring packets using $M = 375$ packets

4. CONCLUSIONS

The aim of this paper is to devise a new approach for monitoring the performance of wireless networks based on measurements of QoS parameters and the overall QoS obtained from the

combination of passive and active measurement approaches. Delay and jitter parameters were taken directly from the injected probe packets (active measurement). Packet loss and throughput were taken from combination of both active and passive measurements. The overall QoS was estimated based on the measured delay, jitter and packet loss which were combined and fed to a fuzzy logic system to get a unified value which summarised these parameters in a one single value. The simulation results showed that this approach is very effective to be used to monitor the actual traffic transmitted over wireless networks using different probe traffic rates. Furthermore, these measurements were also tested and examined in terms of its accuracy and biasness to be representative of the actual traffic results. The proposed approach provided good accuracy in estimation of the overall QoS and the QoS parameters. On the other hand, it showed drawbacks in jitter estimations which affected the precision of the estimated overall QoS especially during moderate and heavy traffic periods. Further work may be conducted to devise a new estimation method to overcome some of the shortcomings of the proposed approach in jitter measurements. In addition, there were some discrepancies between the estimated QoS using the proposed approach and the measured one without the use of the probe traffic. Therefore, some modifications may be incorporated to the proposed approach to minimize and correct these discrepancies. Moreover, the proposed system was tested using a simple wireless network topology. Further experiments may be conducted on larger and more sophisticated network topologies and on may be implemented on real networks and for different multimedia applications.

CONFLICT OF INTEREST

The author declares no conflict of interest.

REFERENCES

- [1] A. Alkahtani, M. Woodward, and K. Al-Begain, (2003) "An Overview of Quality of Service (QoS) and QoS Routing in Communication Networks," In Proceedings of 4th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet). Liverpool, UK.
- [2] T. Braun, (2004) "End-to-End Mechanisms for QoS Support in Wireless Networks," COST 279 Mid-Term Workshop. Rome, Italy.
- [3] W. Jiang, (2003) "QoS Measurement and Management for Internet Real-time Multimedia Services." PhD thesis, Columbia University.
- [4] A. Binsahaq, T. R. Sheltami , and Kh. Salah, (2019) "A Survey on Autonomic Provisioning and Management of QoS in SDN Networks, IEEE Access, Vol. 7, pp. 73384- 73435.
- [5] S. Narayana, A. Sivaraman, V. Nathan, M. Alizadeh, D. Walker, J. Rexford, V. Jeyakumar, and C. Kim, (2016) "Hardware-Software Co-Design for Network Performance Measurement," In Proceedings of the 15th ACM Workshop on Hot Topics in Networks (HotNets '16). Association for Computing Machinery, New York, NY, USA, 190–196.
- [6] F. Righetti, C. Vallati, D. Comola and G. Anastasi, (2019) "Performance Measurements of IEEE 802.15.4g Wireless Networks," In Proceedings of the IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), June 10-12, 2019 - Washington Dc, United States
- [7] M. Chiang and G. Carlsson,(2001) "Admission control, power control and QoS analyses for ad hoc wireless networks," In proceedings of IEEE International Conference on Communications.
- [8] Y. Dong, D. Makrakis and T. Sullivan, (2003) "Effective Admission Control in Multihop Mobile Ad Hoc Networks," In Proceedings of International Conference on Communications Technology (ICCT 2003), pp. 1291-1294. Ottawa University, Canada.
- [9] Q. Xue and A. Ganz, (2003) "Ad Hoc QoS On-demand Routing (AQOR) in Mobile Ad Hoc Networks," Journal of Parallel and Distributed Computing, vol. 63, pp. 154-165.

- [10] M. Curado and E. Monteiro, (2003) "An Overview of Quality of Service Routing Issues," In Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001), Orland, USA.
- [11] V. Kanodia, C.Li, A. Sabharwal, B. Sadegi, and E. Knightly, (2002) "Ordered Packet Scheduling in Wireless Ad Hoc Networks: Mechanisms and Performance Analysis," In Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, pp. 58-70. Lausanne, Switzerland.
- [12] J. Sheu, C. Liu, S. Wu, and Y. Tseng, (2004) "A Priority MAC Protocol to Support Real-time Traffic in Ad hoc Networks. *Wireless Networks*, vol. 10, no. 1, pp. 61-69.
- [13] G. Holland, N. Vaidya, and P. Bahl, (2001) "A Rate-adaptive MAC Protocol for Multi-hop Wireless Networks," In Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, pp. 236 – 250. Rome, Italy.
- [14] Kim Sang Hyong, Kwan-Jong Yoo, Yoojae Won, (2017) "Transmission Algorithm with QoS Considerations for a Sustainable MPEG Streaming Service," *Sustainability*, Vol. 9, no. (3), pp. 1-13.
- [15] J. Kasigwa, V. Baryamureeba, and D. Williams, (2004) "A Framework for End-to-End Quality of Service Provisioning Models in IP based Networks," *Transactions on Engineering, Computing and Technology*, vol. 3, pp. 223-228. Istanbul, Turkey.
- [16] U. Umoh and D. Asuquo, (2018) "Fuzzy Logic Based Quality of Service Evaluation in Multimedia Transmission Over Ad Hoc Wireless Network," *International Journal of Network and Communication Research* Vol.5, no.21, pp.12-30.
- [17] I. Alsukayti, (2019) "An Adaptive Neuro-Fuzzy Model for Quality Estimation in Wireless 2D/3D Video Streaming Systems," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol. 9, no. 2
- [18] F. Cheong and R. Lai, (1999) "QoS Specification and Mapping for Distributed Multimedia Systems: A Survey of Issues," *The Journal of Systems and Software*, vol. 45, no. 2, pp. 127-139.
- [19] G. Almes, S. Kalidindi, and M. Zekauskas, (1999) "A One-Way Delay Metric for IPPM," RFC 2679.
- [20] C. Demichelis, and P. Chimento, (2002) "IP Packet Delay Variation Metric for IPPM," RFC 3393.
- [21] G. Almes, S. Kalidindi, and M. Zekauskas, (1999) "A One-Way Packet Loss Metric for IPPM," RFC 2680.
- [22] K. Bouraqia, S. Essaid, S. Mohamed and L. Latif, (2019) "Quality of Experience for Streaming Services: Measurements, Challenges and Insights." *IEEE Access* 8, pp. 13341-13361.] [X. Tao, J. Liu, S. Mao, M. Debbah and Ch. Jian, (2018) "Measurements and applications of QoE for multimedia communications," *IEEE China Communications*, Vol. 15 , no. 10.].
- [23] V. Paxson, "End-to-end Internet Packet Dynamics, (1999) " *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 277–292.
- [24] A. Johnsson, (2005) "Bandwidth Measurements in Wired and Wireless Networks," Licentiate thesis, Mälardalen University Press.
- [25] J. Postel, (1981) "Internet Control Message Protocol," RFC 792.
- [26] Traceroute, (2002), available at URL: <ftp://ftp.ee.lbl.gov/tracroute.tar.gz>.
- [27] Surveyor Measurement Tool, (2004), Available at: <http://www.advanced.org/surveyor>.
- [28] A. Pasztor, and D. Veitch, (2001) "High Precision Active Probing for Internet Measurement," In Proceedings of INET'2001, Stockholm, Sweden.
- [29] F. Michaut and F. Lepage, (2005) "Application-oriented Network Metrology: Metrics and Active Measurement Tools," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 2.
- [30] M. Aida, N. Miyoshi, and K. Ishibashi, (2003) "A Scalable and Lightweight QoS Monitoring Technique Combining Passive and Active Approaches: On the Mathematical Formulation of CoMPACT Monitor," *INFOCOM 2003*, San Francisco, CA, USA.
- [31] K. Ishibanishi, T. Kanzawa, M. Aida, and H. Ishii, (2004) "Active/Passive Combination-type Performance Measurement Method Using Change-of-Measure Framework". *ELSEVIER Computer Communication*, vol. 27, no. 9, pp. 868-879.
- [32] T. Lindh, (2001) "A Framework for Embedded Monitoring of QoS Parameters in IP-Based Virtual Private Networks," In Proceedings of Passive and Active Measurement Workshop (PAM2001). Amsterdam, The Netherlands. available at URL: <http://www.nlanr.net/Data-users/cityear.html>.
- [33] T. Lindh, (2002) "A New Approach to Performance Monitoring in IP Networks - Combining Active and Passive Methods," In Proceedings of Passive and Active Measurements (PAM 2002), pp. 128-137. Colorado, USA.

- [34] Y.-H Choi and I. Hwang, (2005) "In-service QoS Monitoring of Real-time Applications Using SM MIB," International Journal of Network Management, vol. 15, no. 1, pp. 31-42.
- [35] Yazeed Al-Sbou, (2006) "Quality-of-Service Assessment and Analysis of Wireless Multimedia Networks," Ph.D. dissertation, Sheffield Hallam University.
- [36] Yazeed Al-Sbou, (2017) "Quality of Service Estimation of Multimedia Transmission Using Nonlinear Autoregressive Exogenous Model," Jordan Journal of Electrical Engineering (JEE), vol. 3, no. 1, pp. 49-63.

AUTHOR

YazeedA. Al-Sbou is Professor at the Computer Science Department at the Applied Science University, Bahrain. Professor Al-Sbou received his PhD in Computer Engineering from Sheffield Hallam University (UK), BSc and MSc in Electrical Engineering /Telecommunications from Mutah University and University of Jordan in 1997 and 2001, respectively. His research interests include QoS, Computer Networks performance evaluation, Image Processing, Cognitive Networking, and Cloud computing. Professor Al-Sbou has more than 30 research articles and two book chapters published in refereed journals, books and conferences. He is serving as a TPC and reviewer for several international refereed journals and conferences.

